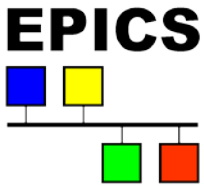


SLAC NATIONAL ACCELERATOR LABORATORY

Introduction to EPICS

EPICS Spring 2012
Collaboration Meeting

April 23rd, 2012

Matt Boyes

Overview

- The Collaboration
- What is EPICS
- Major Components of the EPICS
- Channel Access, IOCs, Clients
- Further Training and Resources
  - Collaboration Website
  - Examples
  - More

Acknowledgements

- Much of this material has been *blatantly* plagiarized from the following EPICS training material
  - Getting Started with EPICS Lecture Series- “Introductory Session I” - *Ned Arnold*
  - Getting Started with EPICS Lecture Series – “Introductory Session II” - *John Maclean*
  - Based On Getting Started with EPICS Lecture Series - “Introduction to Channel Access Clients” - *Kenneth Evans, Jr.*

What is EPICS

- A Collaboration
- A Control System Architecture
- A Software Toolkit

What is EPICS?

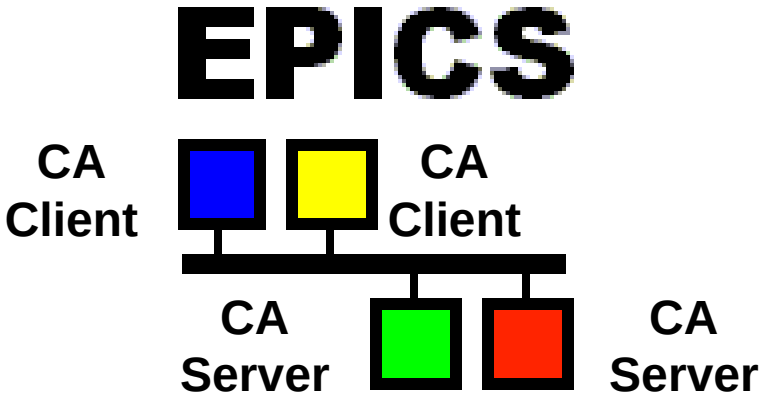
- A Collaboration
  - Began in 1989 between LANL/GTA & ANL/APS
    - (Bob Dalesio, Marty Kraimer, & Jeff Hill)
  - Over 150 license agreements were signed *before* EPICS became “open source”
  - List server; *tech-talk*: the collaboration in action
  - Collaborative efforts vary
    - Assist in finding bugs
    - Share tools, schemes, and advice

Major Sites

- **Major Collaborators**
  - The Advanced Photon Source at Argonne National Laboratory
  - Australian Synchrotron
  - Berlin Electron Synchrotron (BESSY II)
  - Deutches Elektronen Synchrotron (DESY)
  - Diamond Light Source
  - Fermilab (FNAL)
  - Jefferson Laboratory (JLAB)
  - Keck Observatory
  - KEK B-Factory
  - Laboratori Nazionali di Legnaro (INFN-LNL)
  - Lawrence Berkeley National Laboratory (LBL)
  - Los Alamos National Laboratory (LANL)
  - Swiss Light Source (SLS/PSI)
  - Spallation Neutron Source (SNS)
  - Stanford Linear Accellerator Center (SLAC)
- **EPICS Commercially**
  - BiRa Systems Inc (New Mexico)
  - Cosylab Limited (Slovenia)
  - Hytec Electronics Ltd (UK)
  - Instrument Design Technology (UK)
  - Observatory Sciences Ltd (UK)
  - ZTEC Instruments, Inc. (New Mexico)

What is EPICS?

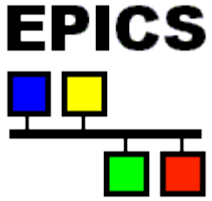
- A Collaboration
- A Control System Architecture
  - **EPICS** is **E**xperimental **P**hysics and **I**ndustrial **C**ontrol **S**ystem
  - Network-based "client/server" model (hence the EPICS logo)

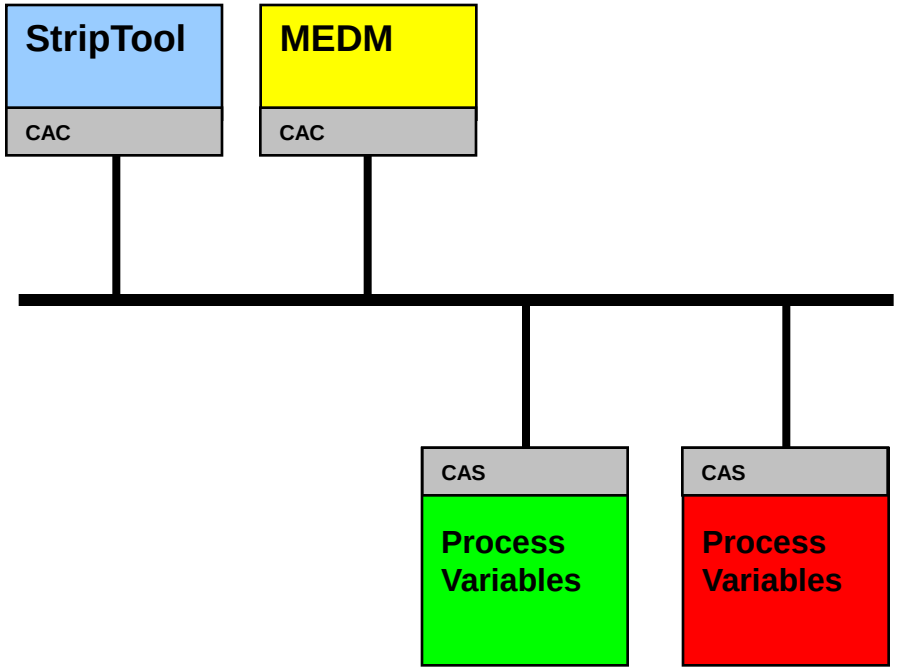


- For EPICS, *client* and *server* speak of their Channel Access role
  - *i.e. Channel Access Client & Channel Access Server*

What is EPICS?

- Channel Access *clients* are programs that require access to *Process Variables* to carry out their purpose





- **The "service" that a Channel Access *server* provides is access to a *Process Variable**

*** A *Process Variable* (PV) is a named piece of data.**

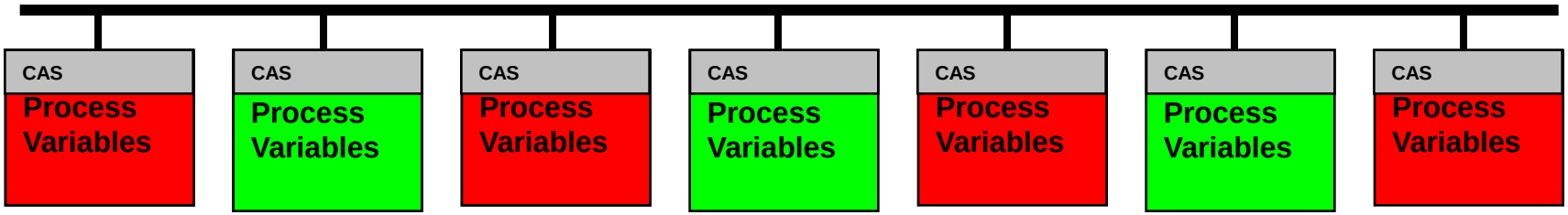
What is EPICS?

- Process Variable
  - A ***Process Variable*** (PV) is a named piece of data associated with the machine (e.g. status, readback, setpoint, parameter)
  - Examples of PV names and values:
    - S1:VAC:reading 3.2e-08 torr
    - LINAC:BPM4:xPosition -0.323 mm
    - BOOSTER:gateValvePosition 'OPEN'
    - S3:DIPOLE:PS:setPoint 123.4 Amps
    - APS:Mode 'Stored Beam'
    - BL3:HISTOGRAM {3, 8, 1, 2, 56, 44, 32, 43, 3, 5, 1}

- Process Variable
  - A ***Process Variable*** is a named piece of data with a set of Controllable properties fields
  - Examples of Fields:
    - Alarm Severity (e.g. NO_ALARM, MINOR, MAJOR, INVALID)
    - Alarm Status (e.g. LOW, HI, LOLO, HIHI, READ_error)
    - Timestamp
    - Number of elements (array)
    - Normal Operating Range
    - Control Limits
    - Engineering Unit Designation (e.g. degrees, mm, MW)

What is EPICS?

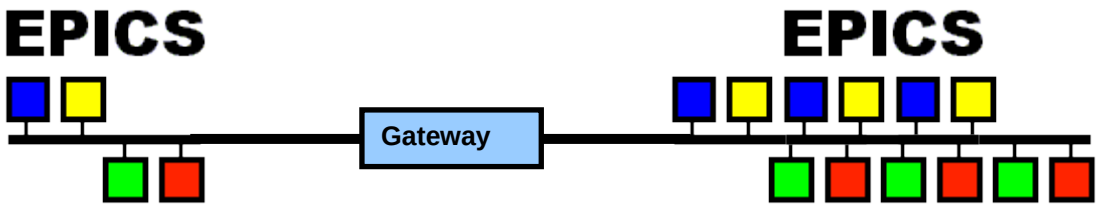
- A Control System Architecture
  - Network-based “client/server” model where the basic data element is a Process Variable
  - The Channel Access Protocol defines how Process Variable data is transferred between a server and client
  - The entire set of Process Variables establishs a *Distributed Real-time Database* of machine status, information and control parameters



What is EPICS?

- By default, Channel Access traffic is constrained to a single subnet, but configuration options can direct traffic elsewhere

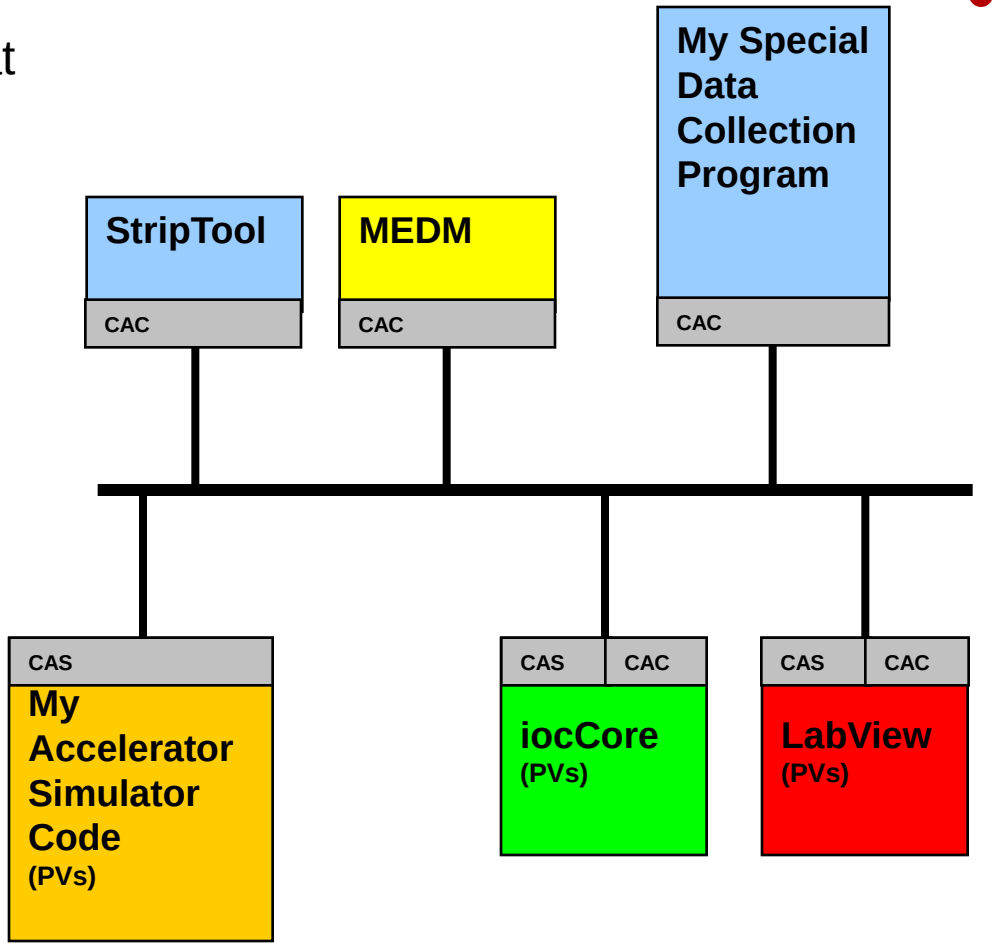
- Physical hierarchies can be implemented using switches, routers, and gateways



What is EPICS?

- Any tool/program/application that abides by the Channel Access protocol could be described as “EPICS Compliant”.

- **EPICS can be viewed as a “toolkit” of EPICS compliant programs. One can select the appropriate tool for their need or develop their own.**



EPICS is

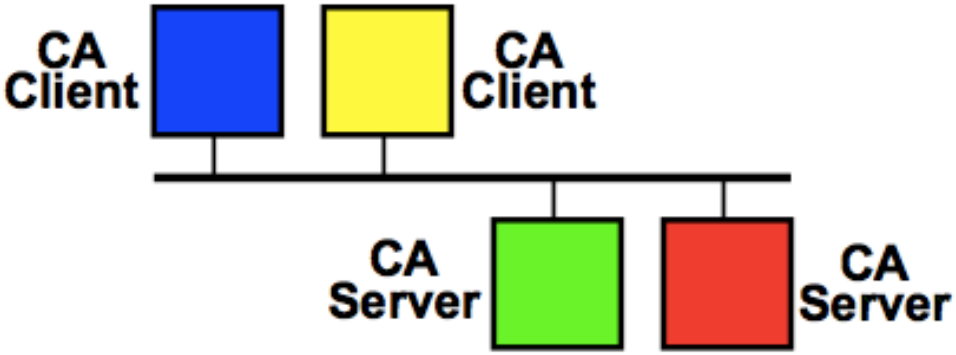
- A Collaboration
  - A world wide collaboration that shares designs, software tools, and expertise for implementing large-scale control systems
- A Control System Architecture
  - A client/server model with an efficient communication protocol (Channel Access) for passing data
  - A distributed real-time database of machine values
- A Software Toolkit
  - A collection of software tools collaboratively developed which can be integrated to provide a comprehensive and scalable control system

So What Does it Do?

- EPICS tools are available to accomplish almost any typical Distributed Control System (DCS) functionality, such as:
  - Remote Control & Monitoring of Technical Equipment
  - Data Conversion/Filtering
  - Closed Loop Control
  - Access Security
  - Equipment Operation Constraints
  - Alarm Detection/Reporting/Logging
  - Data Trending/Archiving/Retrieval/Plotting
  - Automatic Sequencing
  - Mode & Facility Configuration Control (save/restore)
  - Modeling/Simulation
  - Data Acquisition
  - Data Analysis

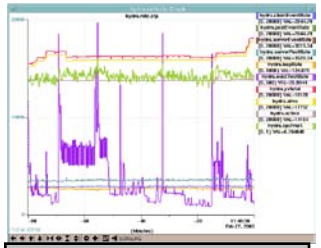
How does it do it?

- Network-based “client/server” model
  - Also known as “publish/subscribe”

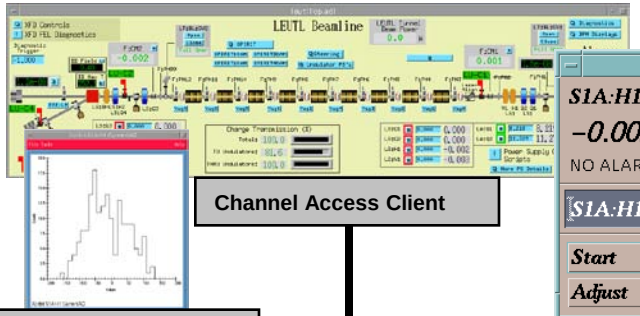


- Channel Access (CA) is the protocol (middleware) that connects the clients and servers—it is a virtual bus
  - It is very efficient (high throughput, low latency, no polling)
  - Robust (partial degradation, incremental recovery)
  - Self-configuring (uses a “discovery” protocol)
  - Scalable (100s of elements; 100,000s of connections)
  - Simple API: “set”, “get”, “monitor”

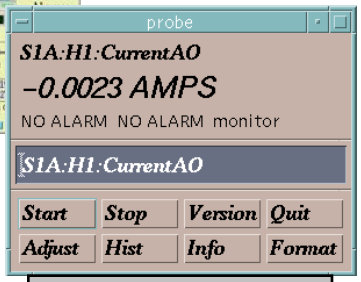
How does it do it?



Channel Access Client



Channel Access Client



Channel Access Client

Channel Access Client

Channel Access Server

Process Variables:
S1A:H1:CurrentAO

S1:P1:x
S1:P1:y

S1:G1:vacuum

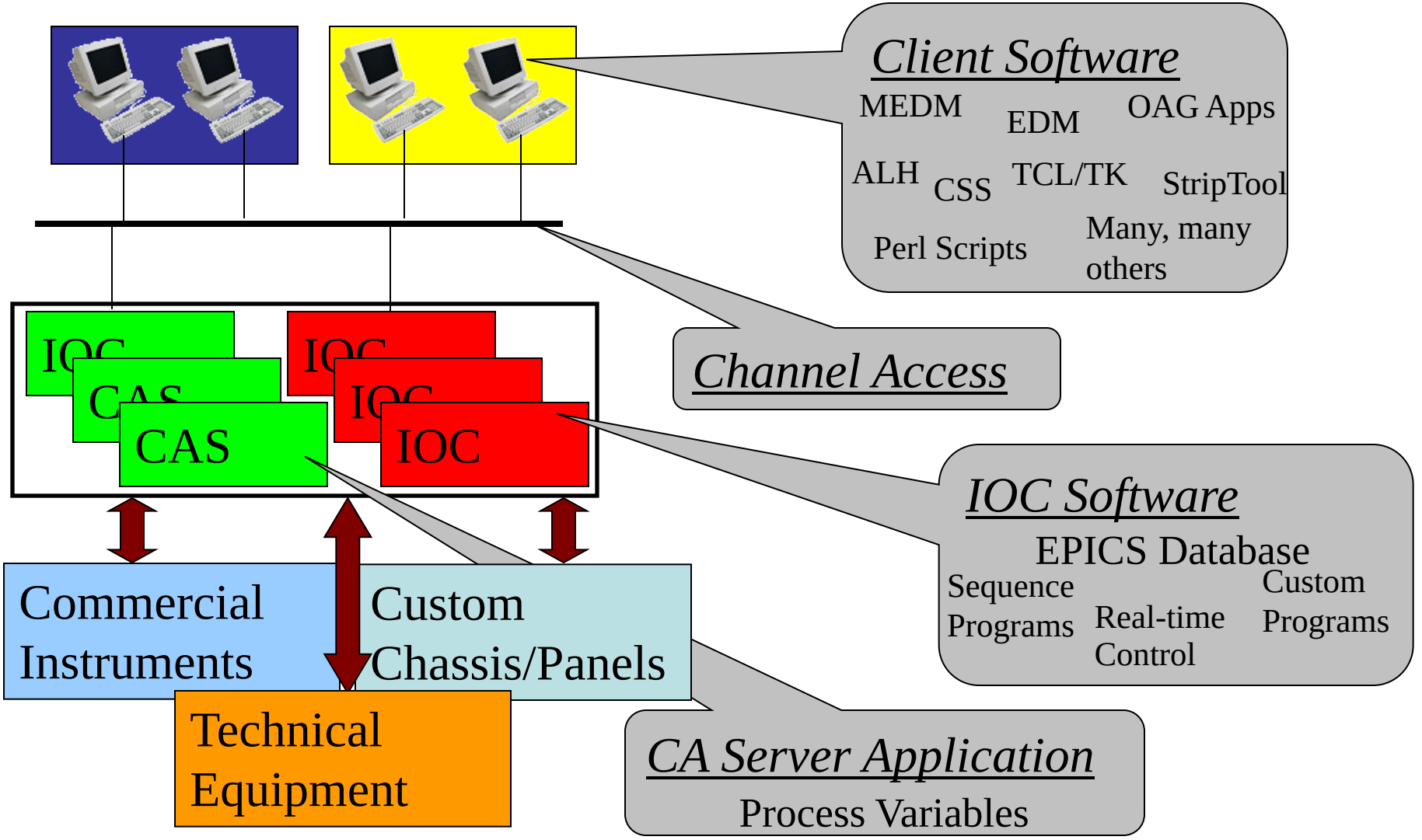
Computer Interface → Power Supply

Computer Interface ← Beam Position Monitor

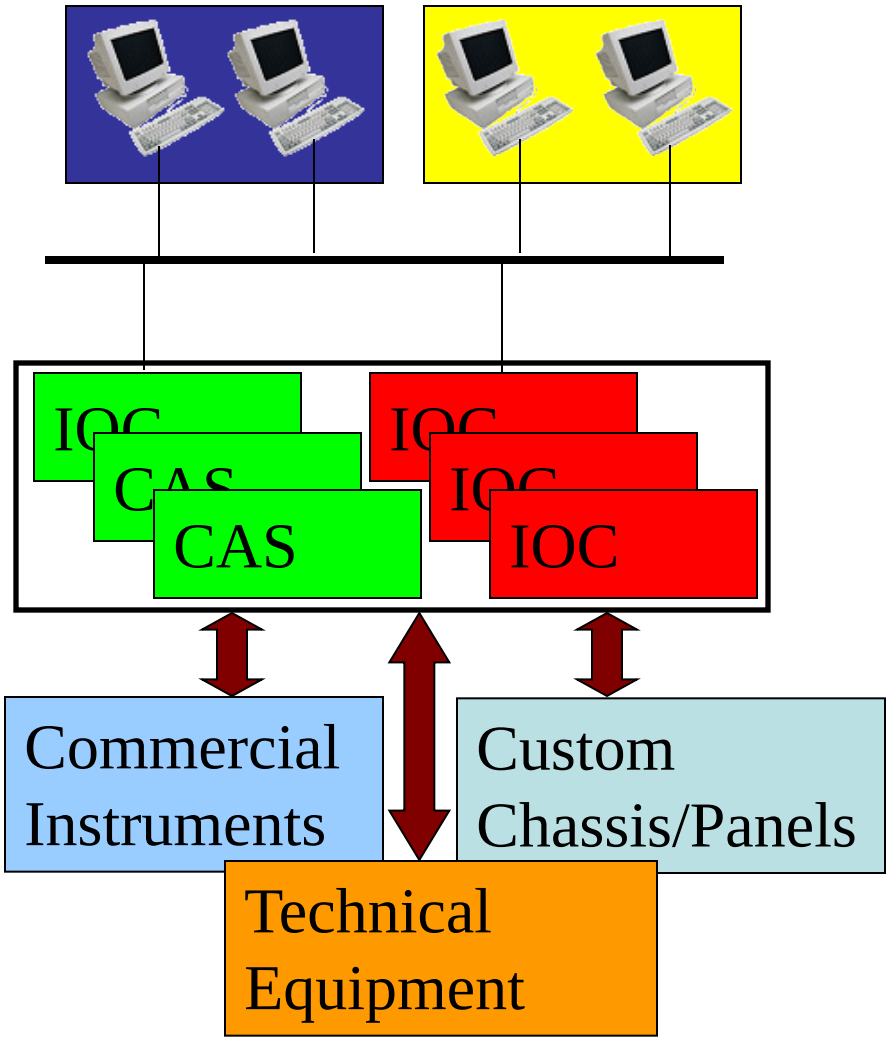
Computer Interface ← Vacuum Gauge

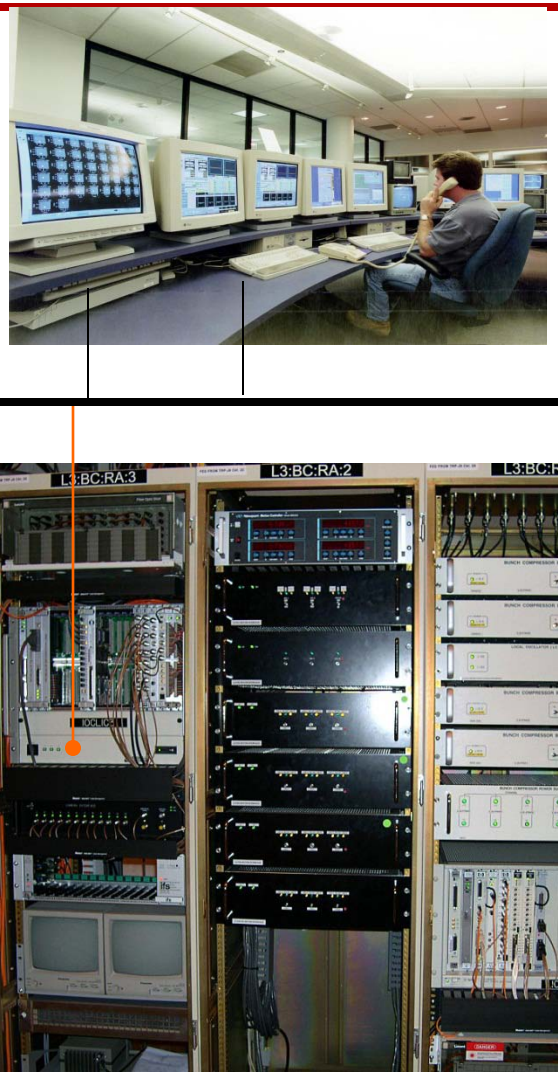
Where does it do it?

Canonical Form of an EPICS Control System

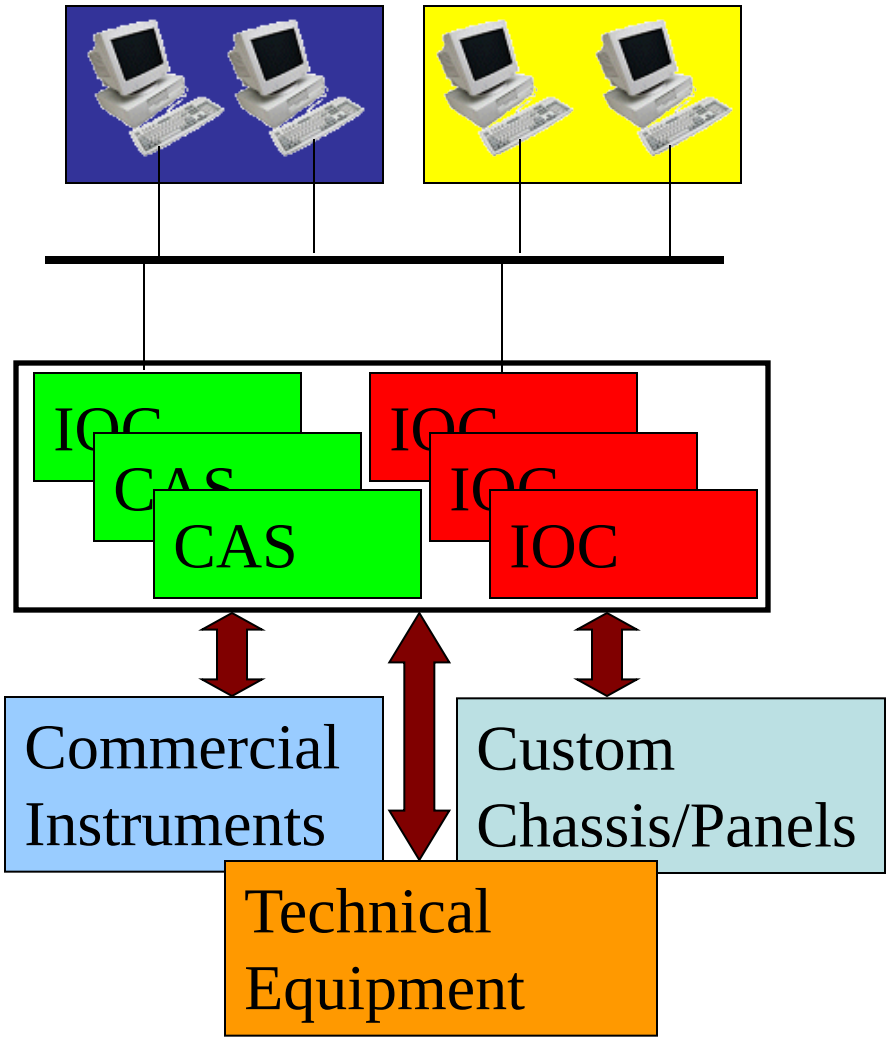


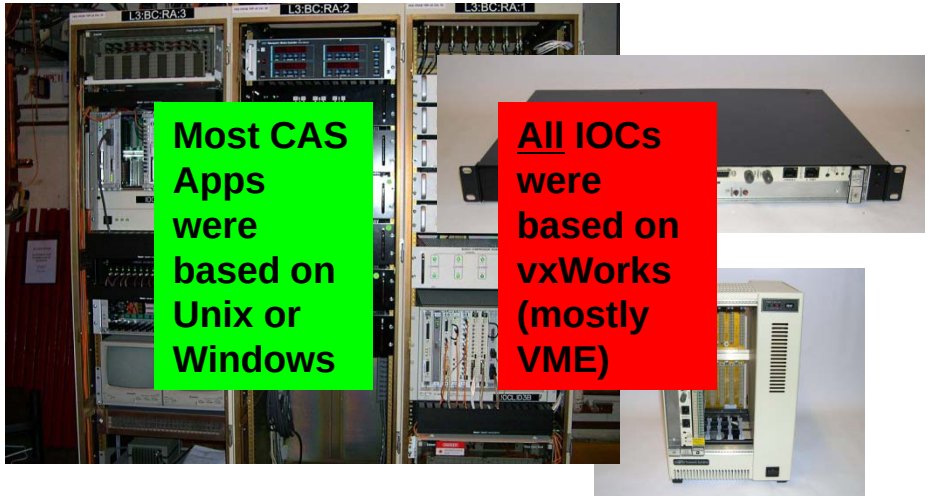
Typical Realizations of an EPICS System



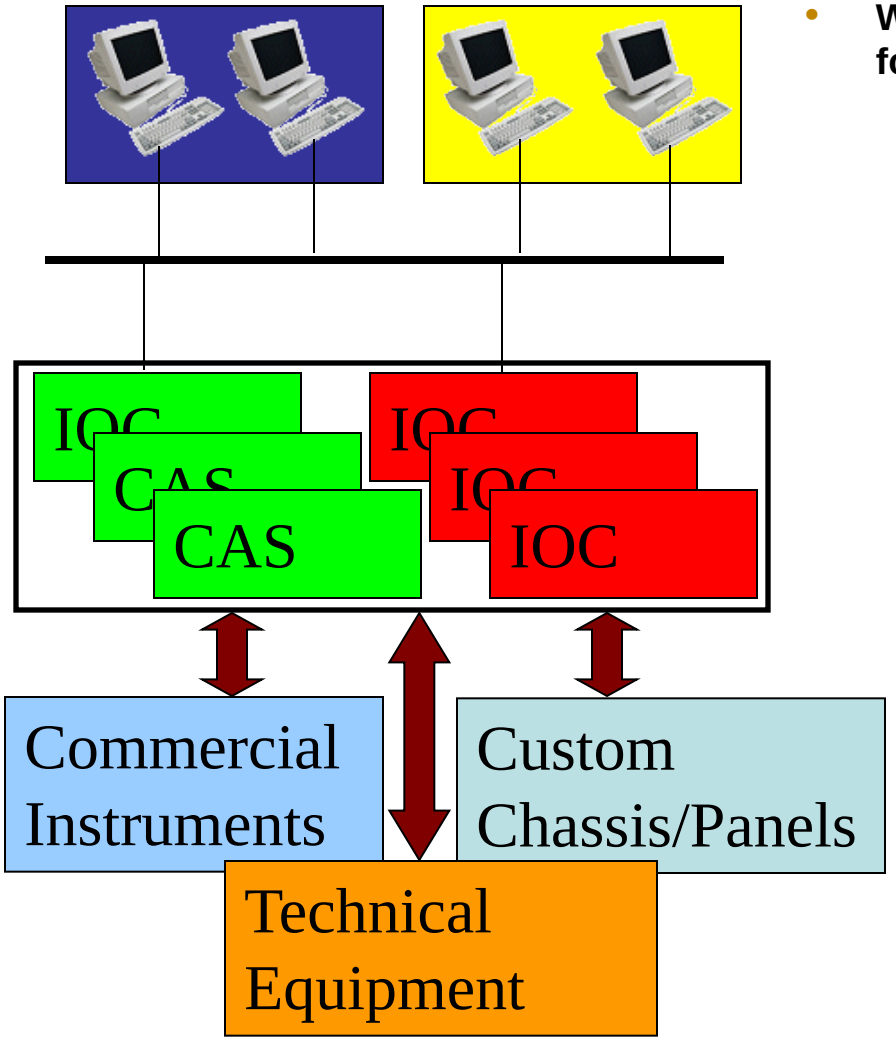


Typical Realizations of an EPICS System



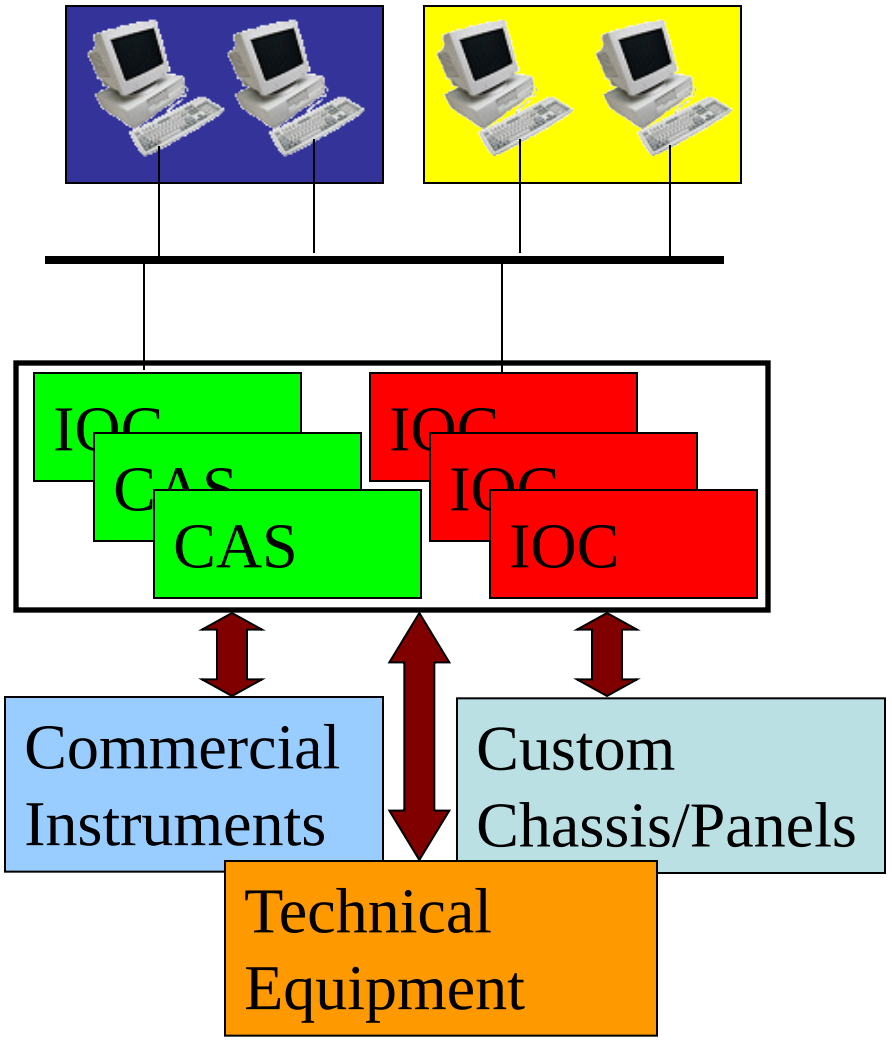


Typical Realizations of an EPICS System



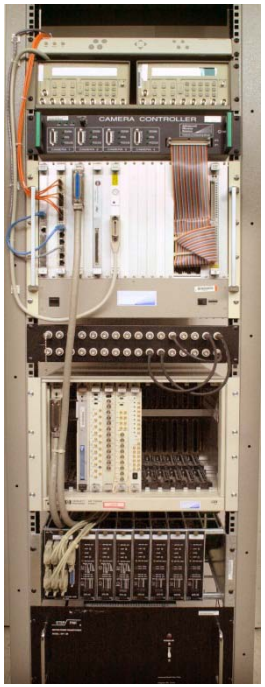
- **With Release 3.14, the operating system limitations for iocCore have been removed.**

Typical Realizations of an EPICS System



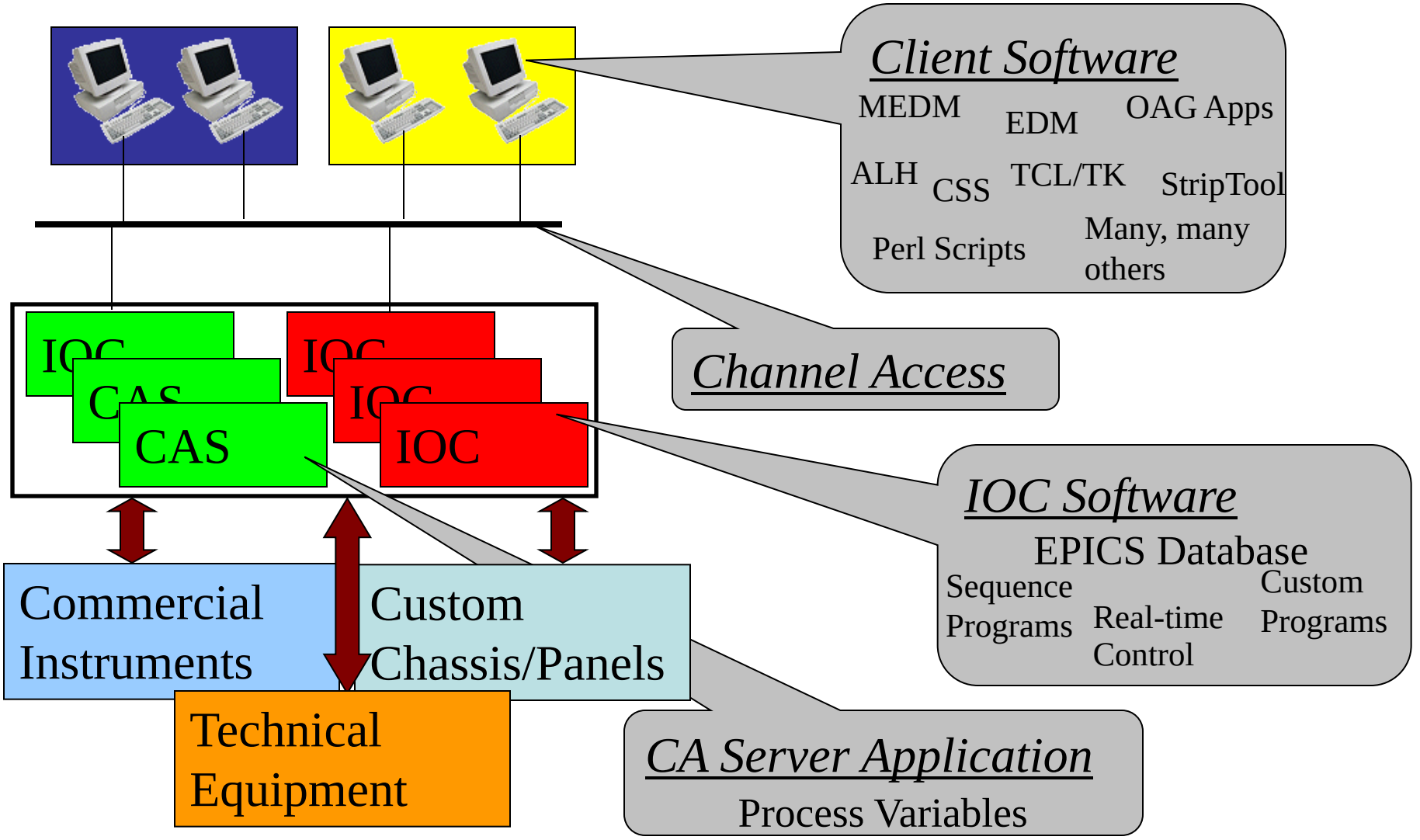
Driving a motor with EPICS

circa 1995



circa 2002

Canonical Form of an EPICS Control System



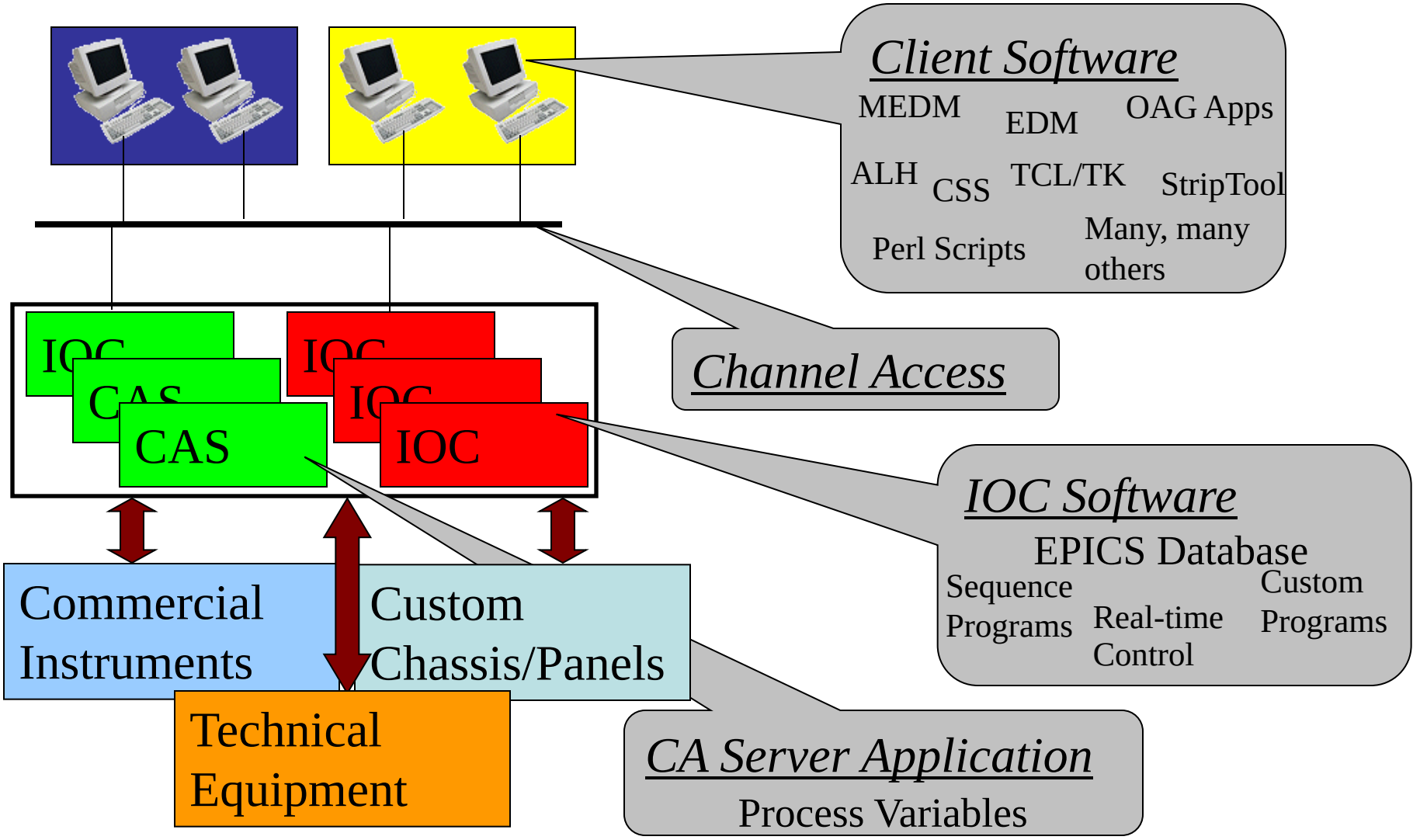
Standalone CA Clients
(from EPICS Website)

- ALH: Alarm Handler
- BURT: Backup and Restore Tool
- CAEX: Channel Access Examples
- CAPod: Channel Access projects for Apple iOS devices (SF)
- CASR: Host-based Save/Restore
- CA Watcher: Channel Access monitor and alarm handler (BESSY)
- Channel Archiver (SF)
- Channel Watcher (SLAC)
- CSS: Control System Studio (SF)
- EDM: Extensible Display Manager (ORNL)
- MEDM: Motif Editor and Display Manager
- NAL: Nagios Alarm Handler (INFN)
- Probe: Motif Channel Monitoring program
- StripTool: Strip-chart plotting tool

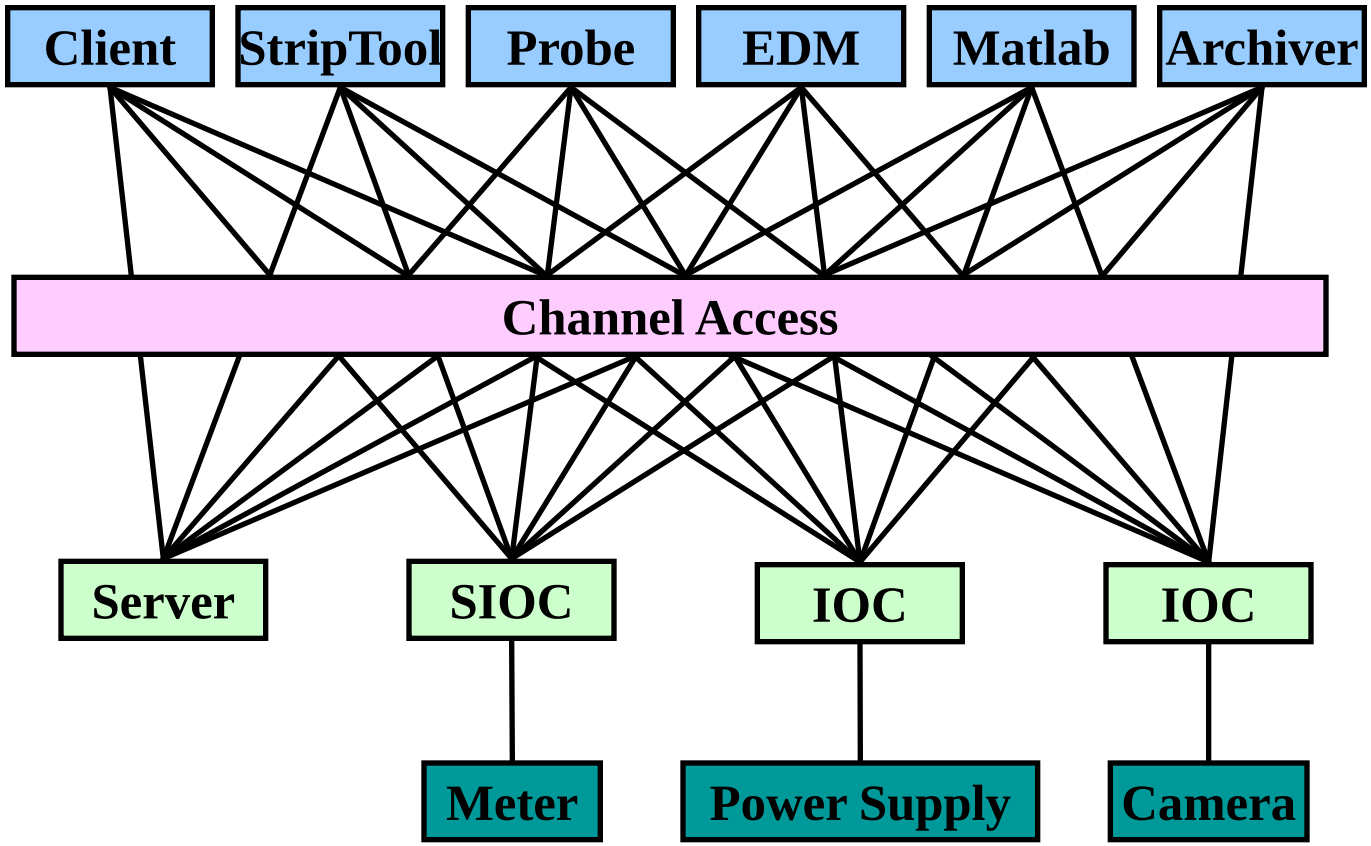
CA Client Interfaces to other tools and languages (from EPICS Website)

- C/C++
  - EZCA: Easy Channel Access interface library for C programs
  - EzcaScan: Easy Channel Access for arrays of channels
  - EPICS QT: Channel Access for the QT framework (SF)
  - SCA: Simple Channel Access (LBL/ALS)
- Java
  - CAJ: Pure Java Channel Access Client (Cosylab)
  - JCA: Channel Access client for Java using JNI (Cosylab) (old APS/BCDA)
- LabVIEW
  - CA_Lab: Channel Access client for Labview (BESSY)
  - LabVIEW: CA Client and Shared Memory interface to IOC (SNS)
  - LabVIEW IO Server: CA client for LabVIEW (NI)
- Matlab/Octave/Scilab
  - MCA: Channel Access client library for Matlab (SF)
  - LabCA: Matlab & Scilab interface to Channel Access (SLAC)
- Perl
  - CAP5: Channel Access for Perl5 (part of Base since R3.14.10)
  - PEZCA: A Perl binding to EZCA (BESSY)
- Python
  - Cothread: A CA interface to Python (Diamond)
  - PyEpics3: Pure Python interface to Channel Access (UChicago)
  - NPEI: New Python EPICS Interface (PSI)
- Other
  - EpicsSharp: CA Client library in native C# (PSI)
  - CAML: Channel Access Markup Language (ORNL)
  - IDL: CA client libraries and scripts for IDL via EZCA
  - IGOR2EPICS: CA client library for WaveMetrics IGOR Pro (SF)
  - PHP_EPICS: CA interface for PHP-based web applications (SLS)
  - SDDS: The Self-Describing Data Sets analysis package
  - WebCA: Channel Access web browser plug-in (CosyLab)

Canonical Form of an EPICS Control System

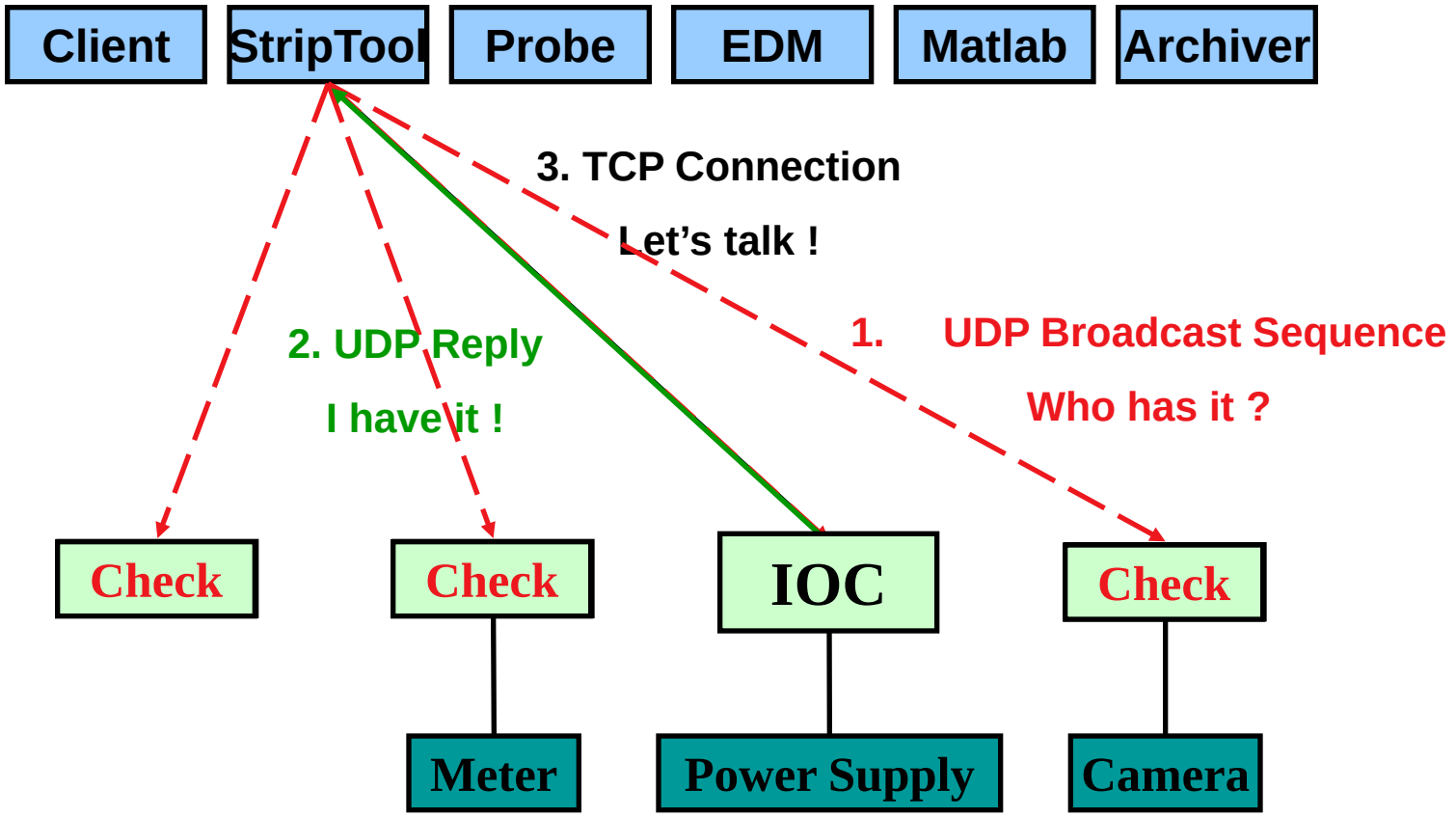


EPICS CA Overview



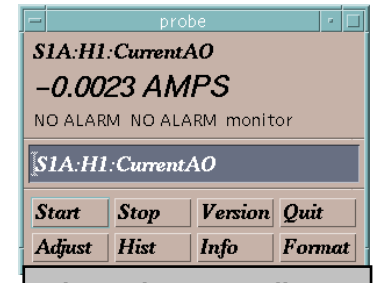
Based On Getting Started with EPICS Lecture Series "Introduction to Channel Access Clients"
Kenneth Evans, Jr.

CA Search and Connect Procedure



Based On Getting Started with EPICS Lecture Series "Introduction to Channel Access Clients"
Kenneth Evans, Jr.

Channel Access Overview



Channel Access Client

"connection request" or "search request"

"get" or "caGet"

"put" or "caPut"

"set a monitor"

Who has a PV named "S1A:H1:CurrentAO"?

What is its value?

Change its value to 30.5

Notify me when the value changes

CA Client

CA Server

Channel Access Server

Process Variables:

S1A:H1:CurrentAO

S1:P1:x

S1:P1:y

S1:G1:vacuum

I do.

25.5 AMPS

OK, it is now 30.5

"put complete"

or

30.5 is too high. It is now set to the maximum value of 27.5.

or

You are not authorized to change this value

It is now 20.5 AMPS

It is now 10.5 AMPS

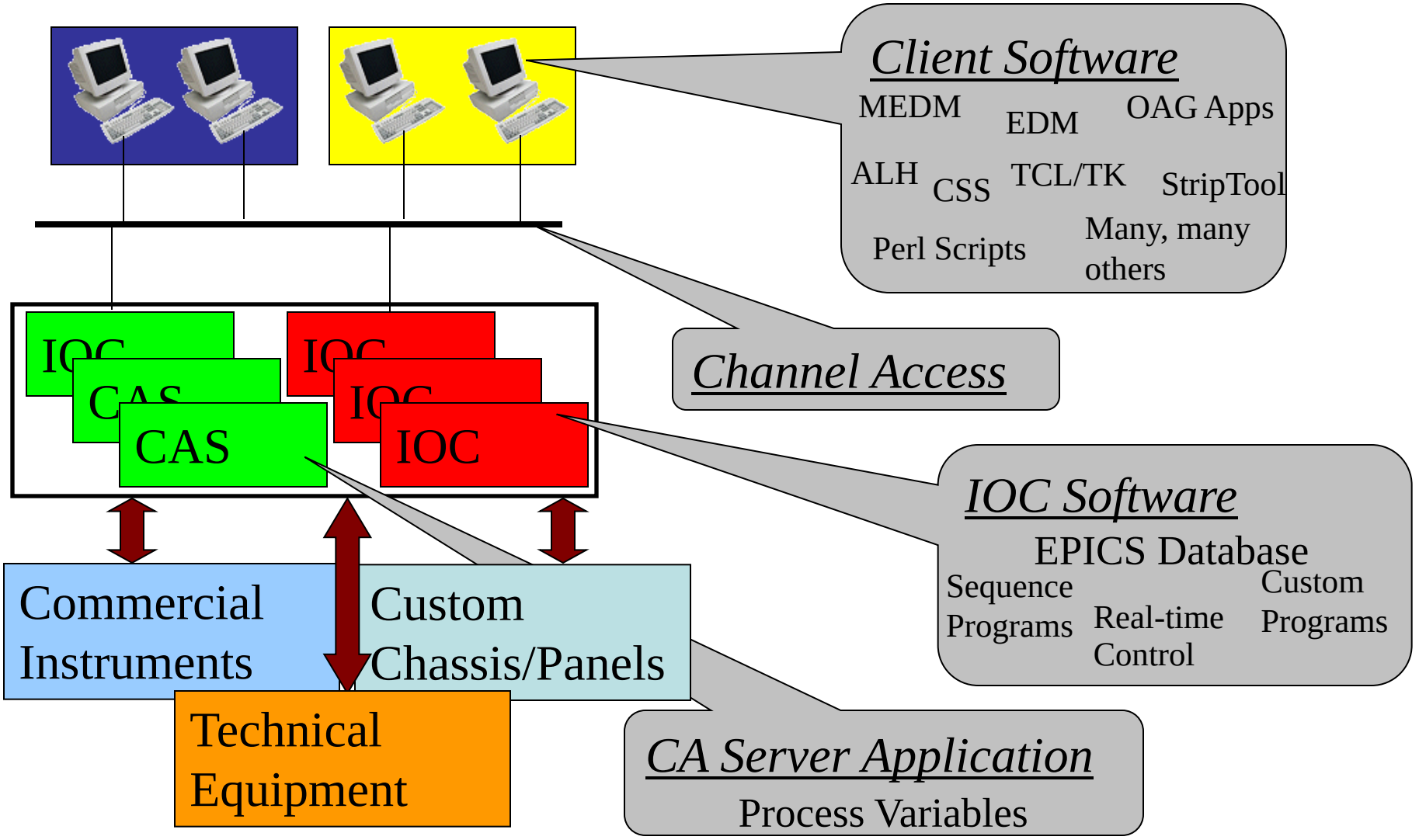
It is now -0.0023 AMPS

"post an event" or "post a monitor"

Key Features of Channel Access …

- Clients broadcast PV names to find the server in which they exist
- Channel Access Security can be applied to limit access to Process Variables
- Clients can wait until a ‘put request ‘ is completed before proceeding
- Clients can ‘set monitors’ on PVs and will then be notified when the value changes

Canonical Form of an EPICS Control System

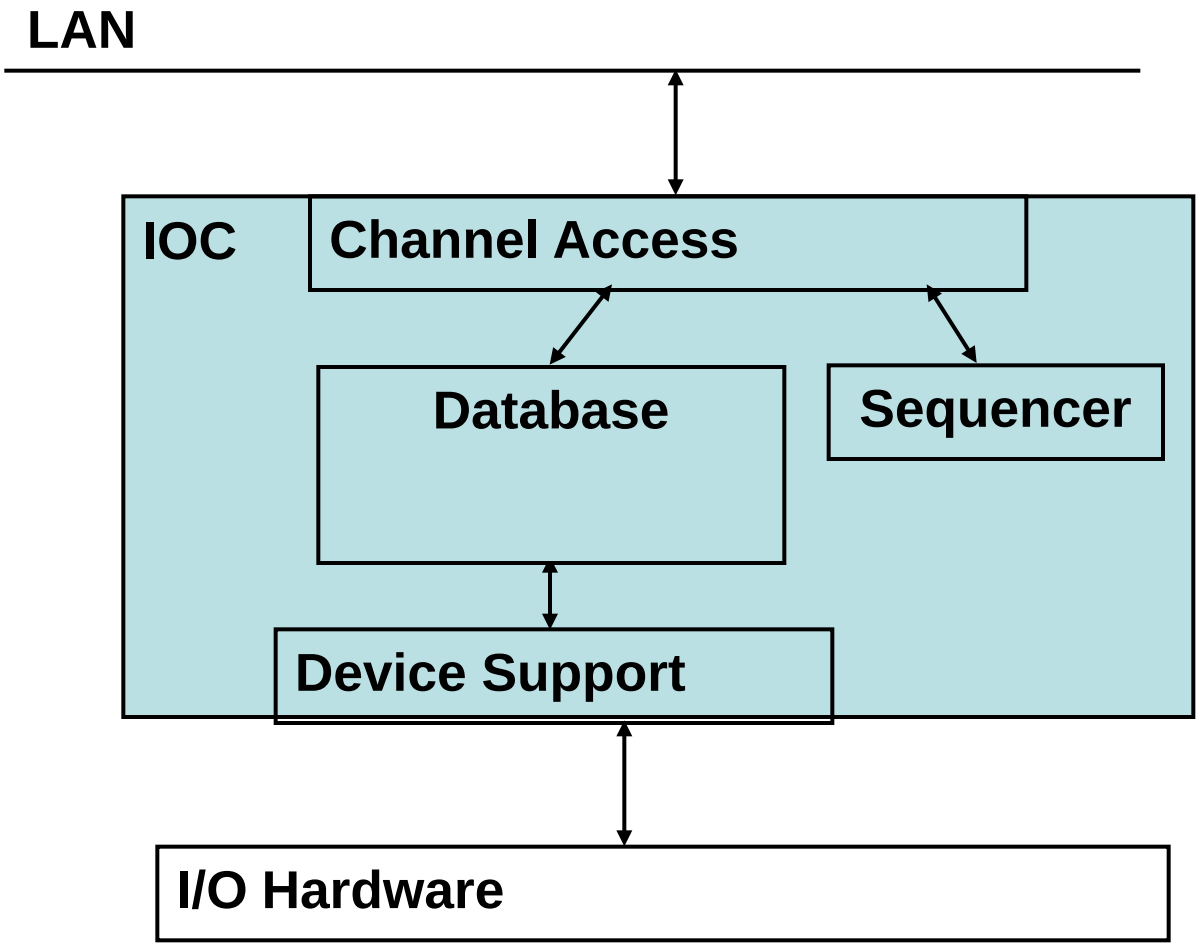


Key Features of IOC Software

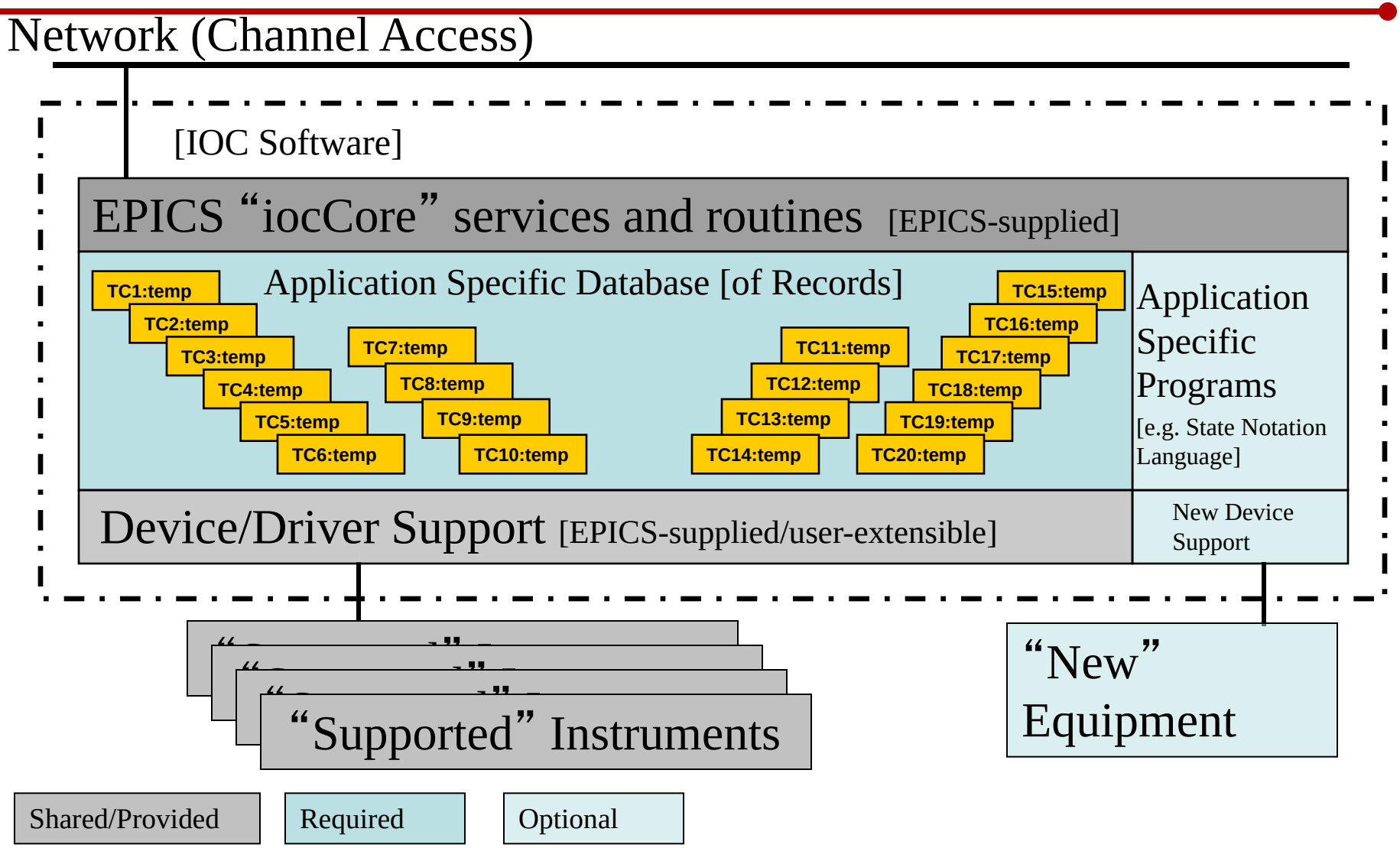
- Two primary application specific components:
  - The real-time database of records (required)
  - State Notation Language programs used to implement state oriented programs (finite-state machine)
- Machine status, information and control parameters are defined as “records” in the application specific database.
- The data within a record is accessible via Process Variables.
- Records have some functionality associated with them (scaling, filtering, alarm detection, calculations, etc). Different record types have different functions and uses.
- Records are frequently associated with I/O equipment that requires unique “device support” for that instrument.

Inside an IOC

The major software components of an IOC (IOC Core)

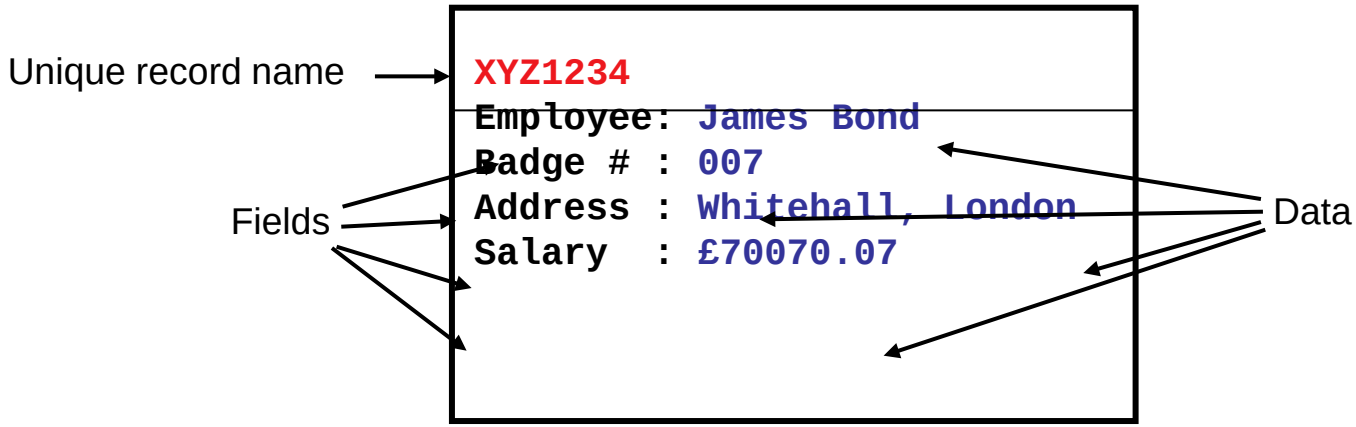


IOC Software in One Slide



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

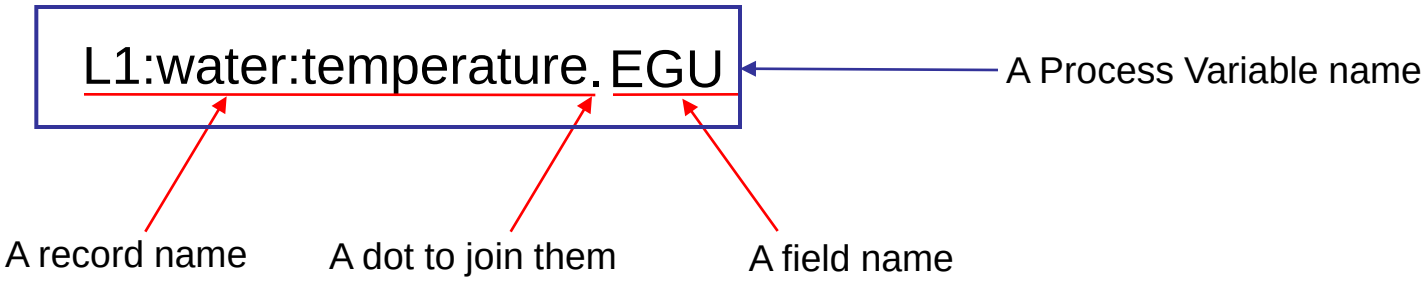


What are EPICS records?

- 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

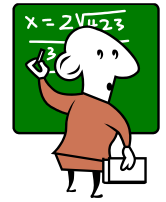
- A PV name is comprised of two parts
  - The record name, and
  - The name of a field belonging to that record
- For example…



- **Note that if no field name is given, Channel Access will default to using the .VAL field**
- **i.e. to CA, "`L1:water:temperature`" = "`L1:water:temperature.VAL`"**
- **PV names must be unique**

What do records do?

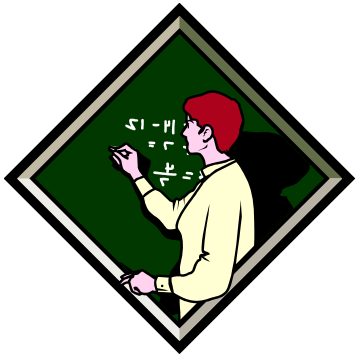
- Records are active, they do things
  - Get data from other records or from hardware
  - Perform calculations
  - Check values are in range and raise alarms
  - Put data to other records or to hardware
  - Activate or disable other records
  - Wait for hardware signals (interrupts)



- 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 (genSUB)
  - Output: e.g.
    - Analog Out (AO)
    - Binary Out (BO)
  - Custom: e.g.
    - Beam Position Monitor
    - Multi Channel Analyzer



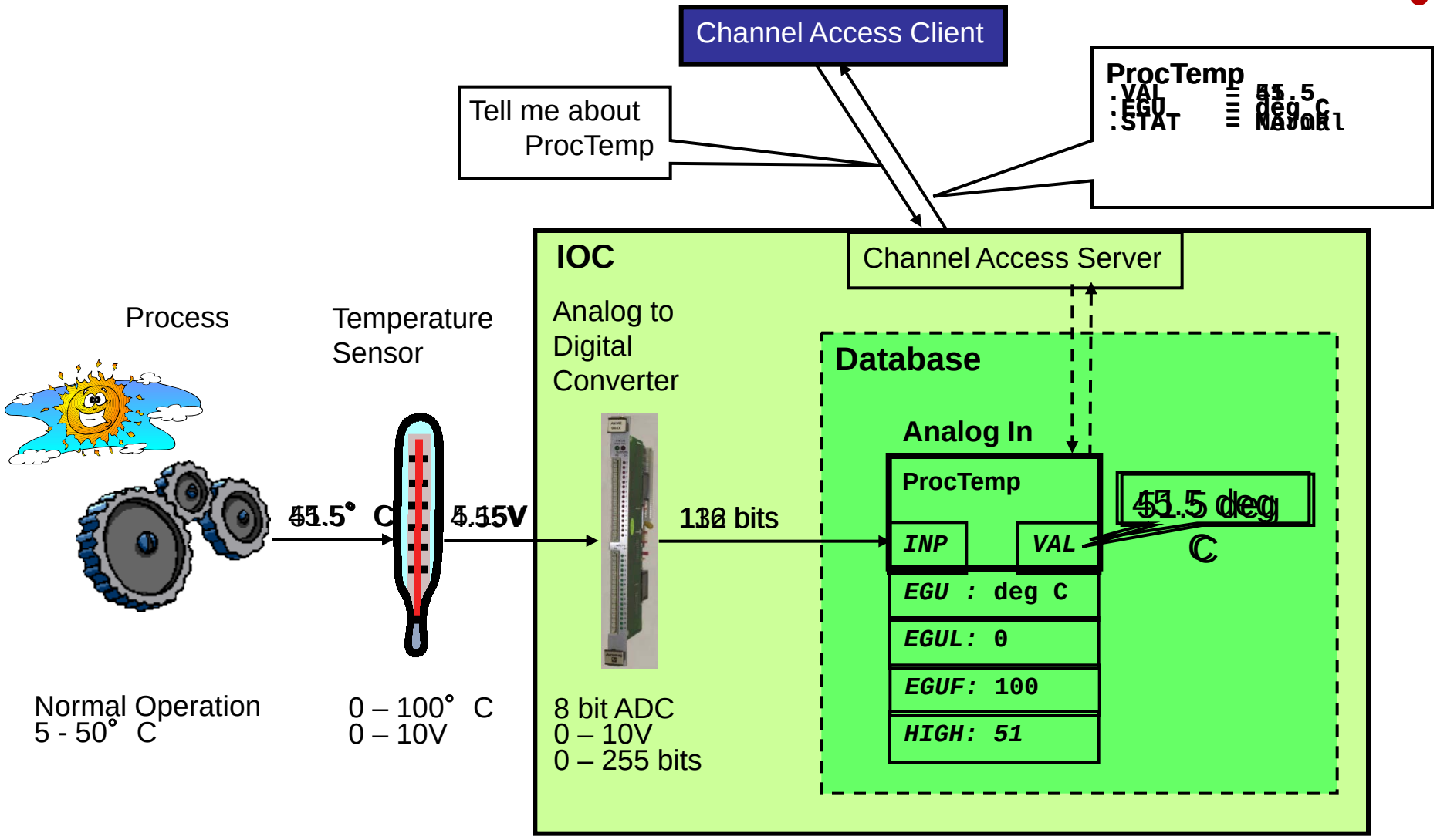
Some record types

- 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

EPICS Databases – What are they?

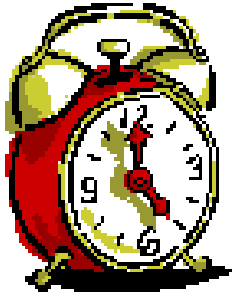
- A collection of one or more EPICS *records* of various types
- Records can be interconnected and are used as building blocks to create applications
- A data file that's loaded into IOC memory at boot time
- Channel access talks to the IOC memory copy of the database

Our First Database



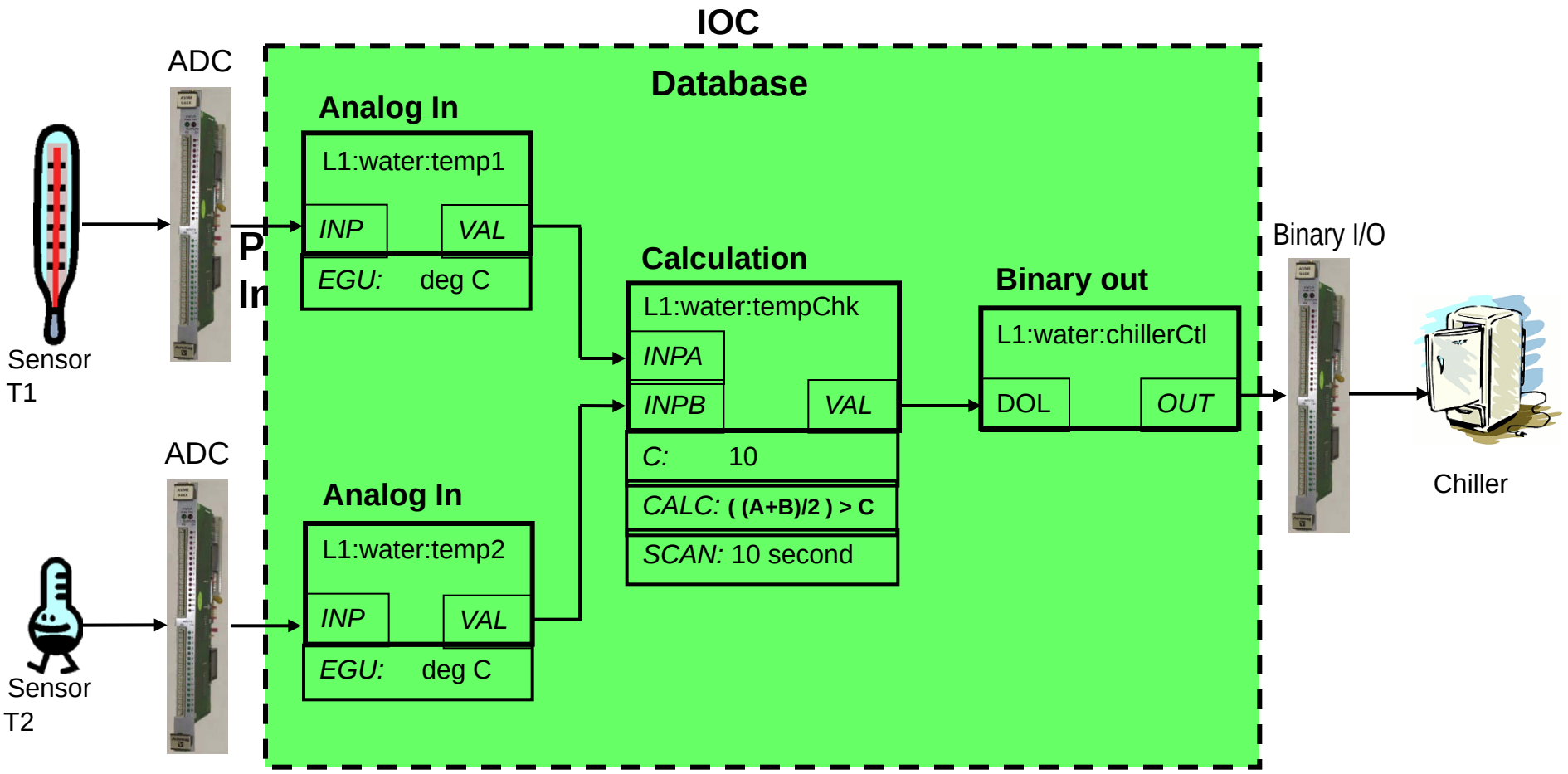
Record Processing

- 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

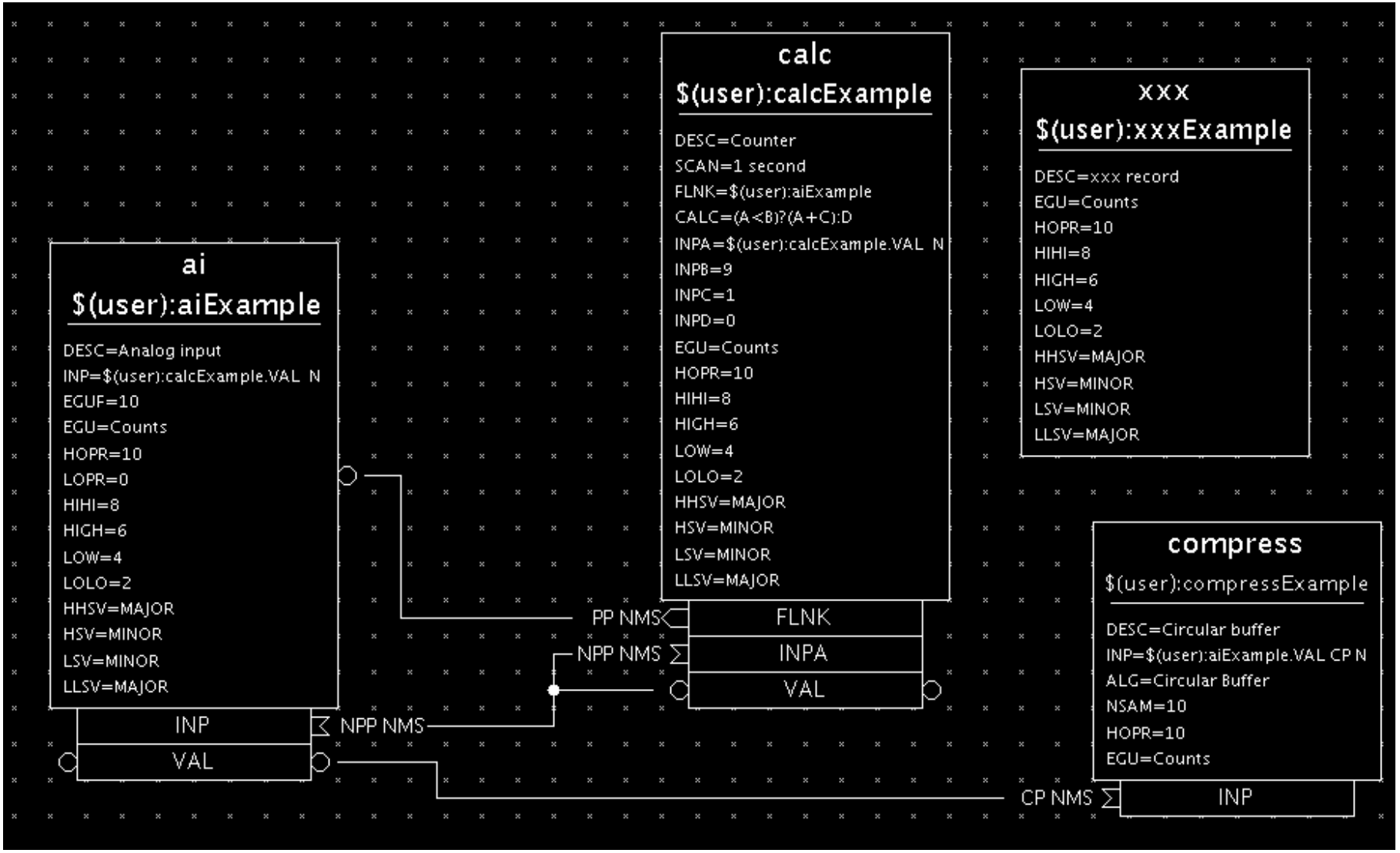


  - Event driven: Events include
    - Hardware interrupts
    - Request from another record via links
    - EPICS Events
    - Channel Access Puts

Database Processing

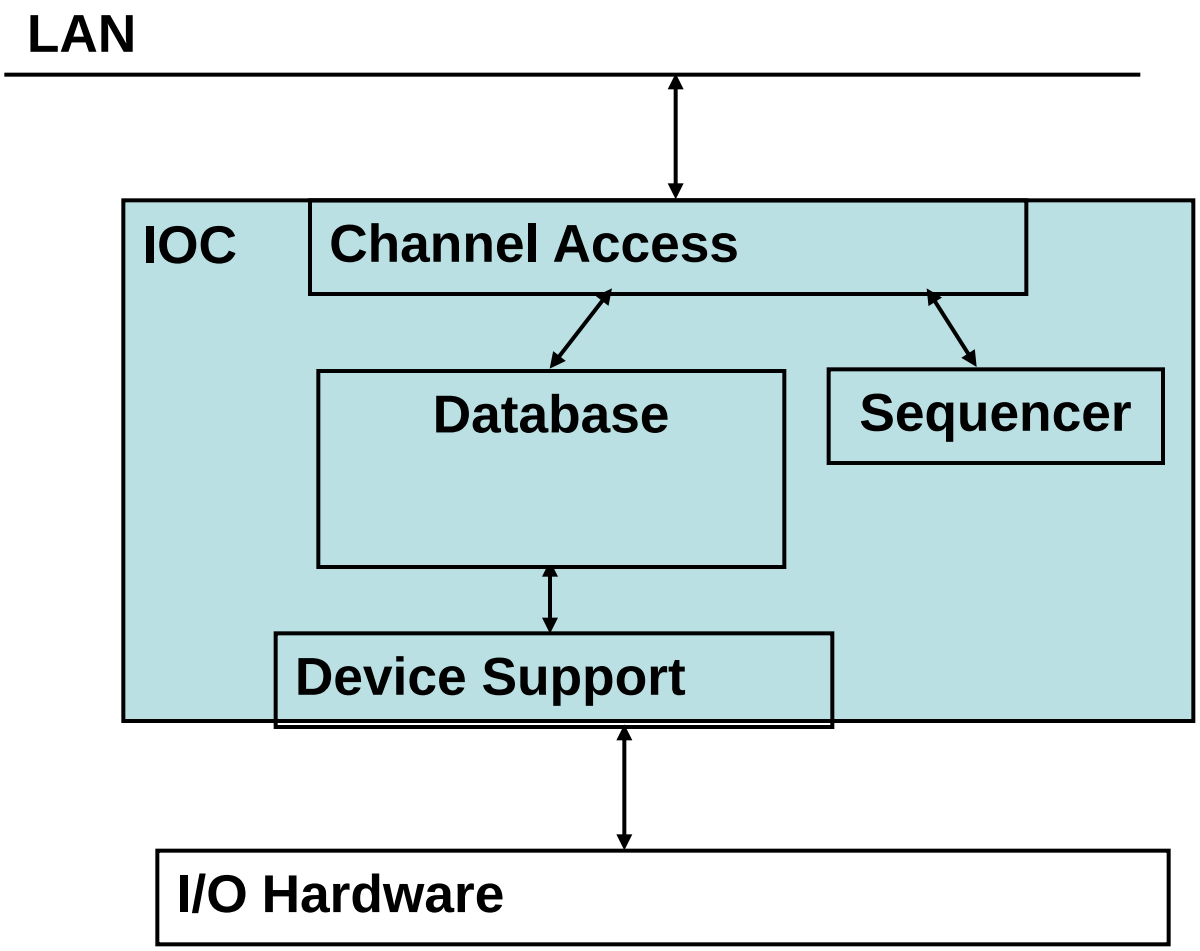


Simple IOC Database



Inside an IOC

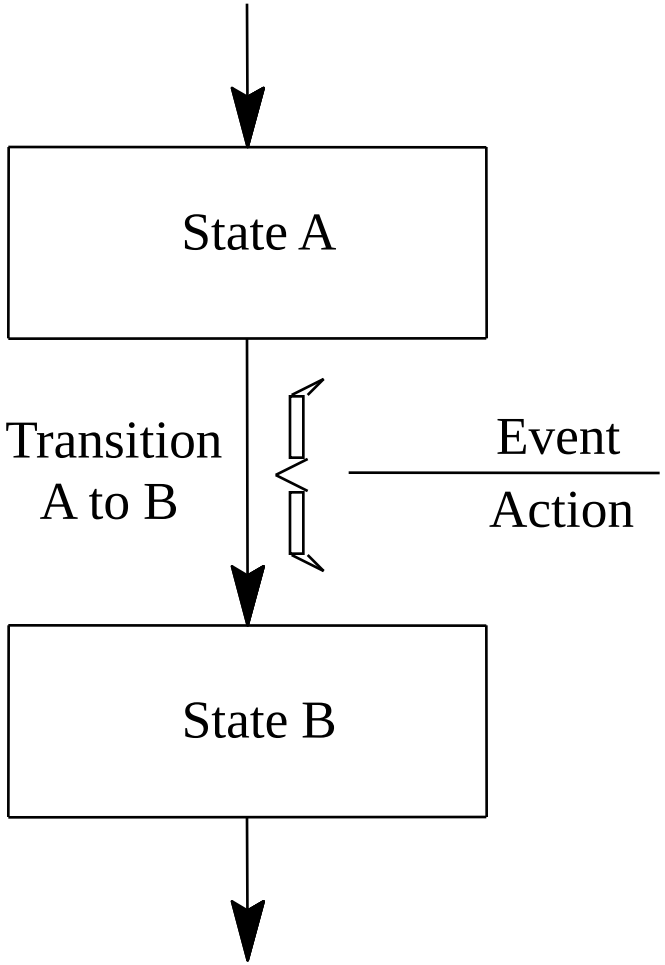
The major software components of an IOC (IOC Core)



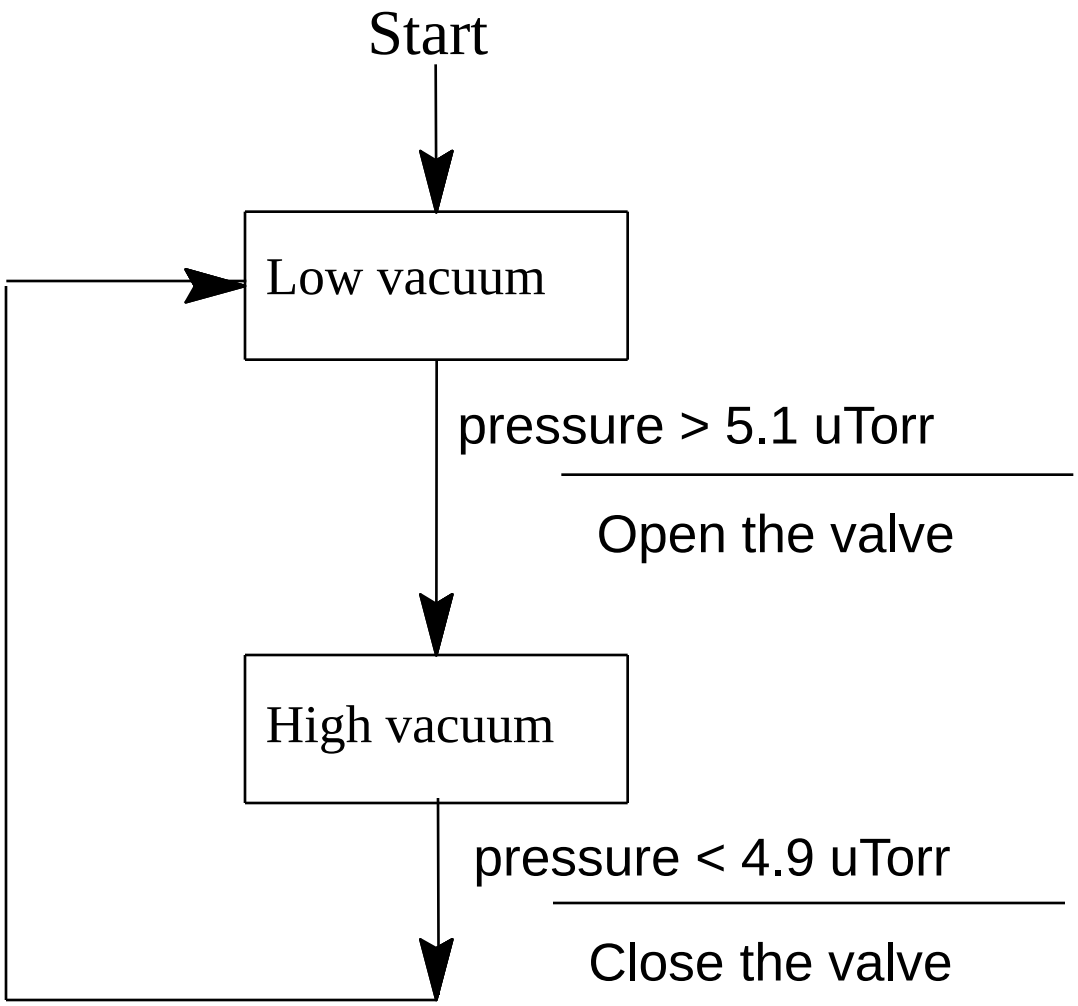
The Sequencer

- 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

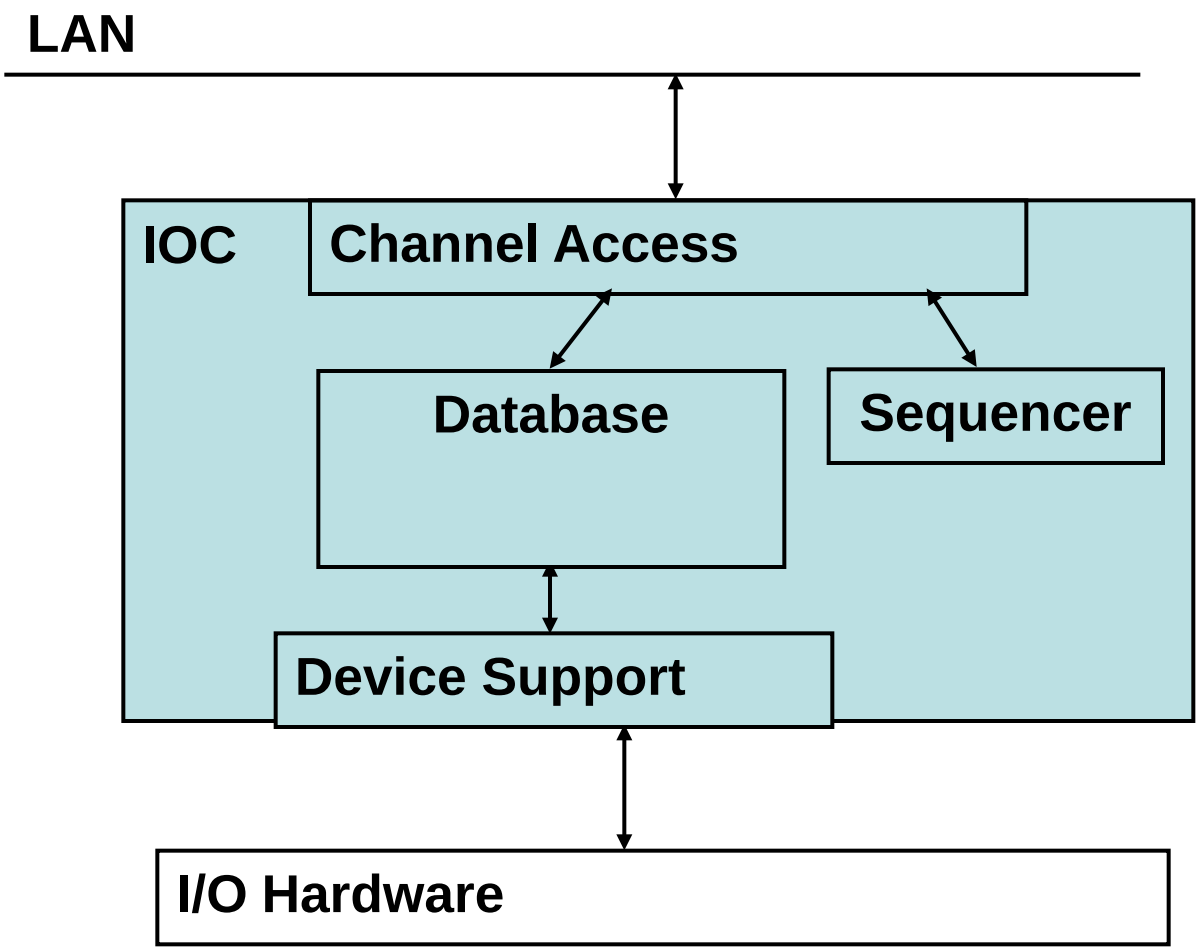


STD Example


Start
Low vacuum
pressure > 5.1 uTorr
Open the valve
High vacuum
pressure < 4.9 uTorr
Close the valve

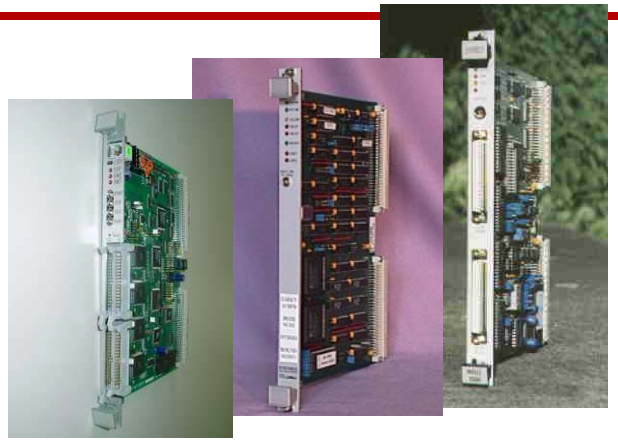
Inside an IOC

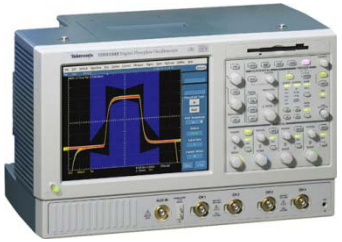
The major software components of an IOC (IOC Core)



Device Support

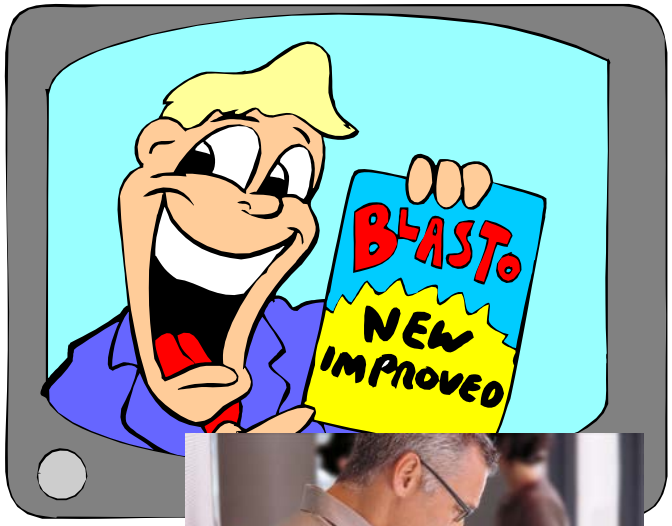
- Device and driver support interface hardware to the database
- Examples of devices….
- VME cards: ADC, DAC, Binary I/O e.t.c.
- Motor controllers
- Oscilloscopes
- PLCs

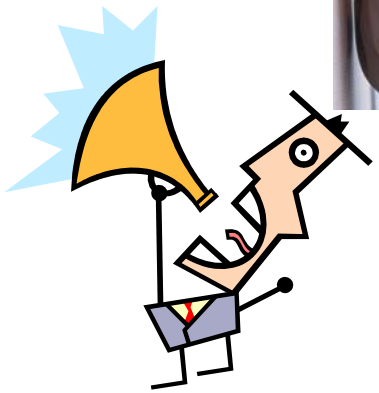




Device Support

- 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





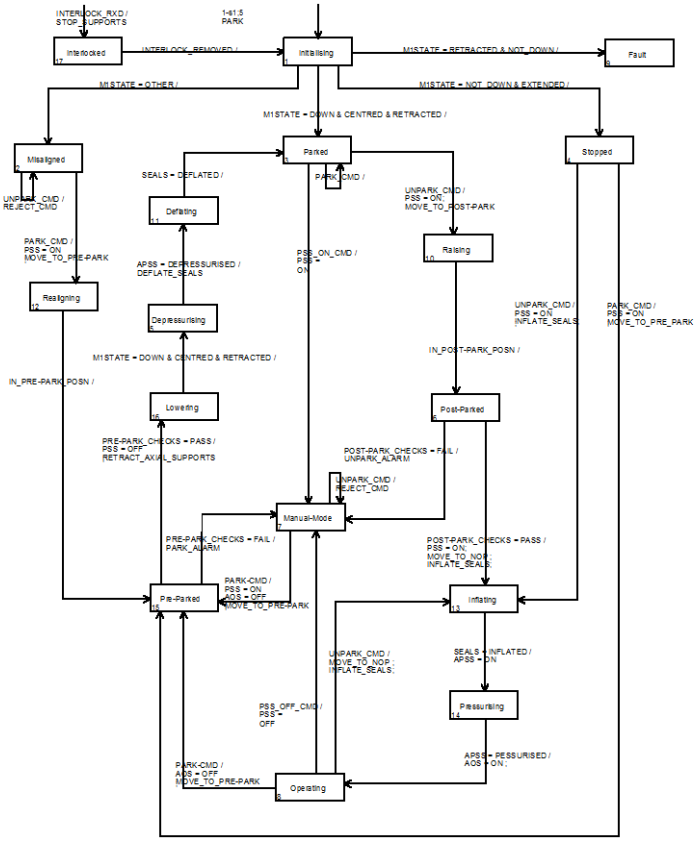
When to use databases

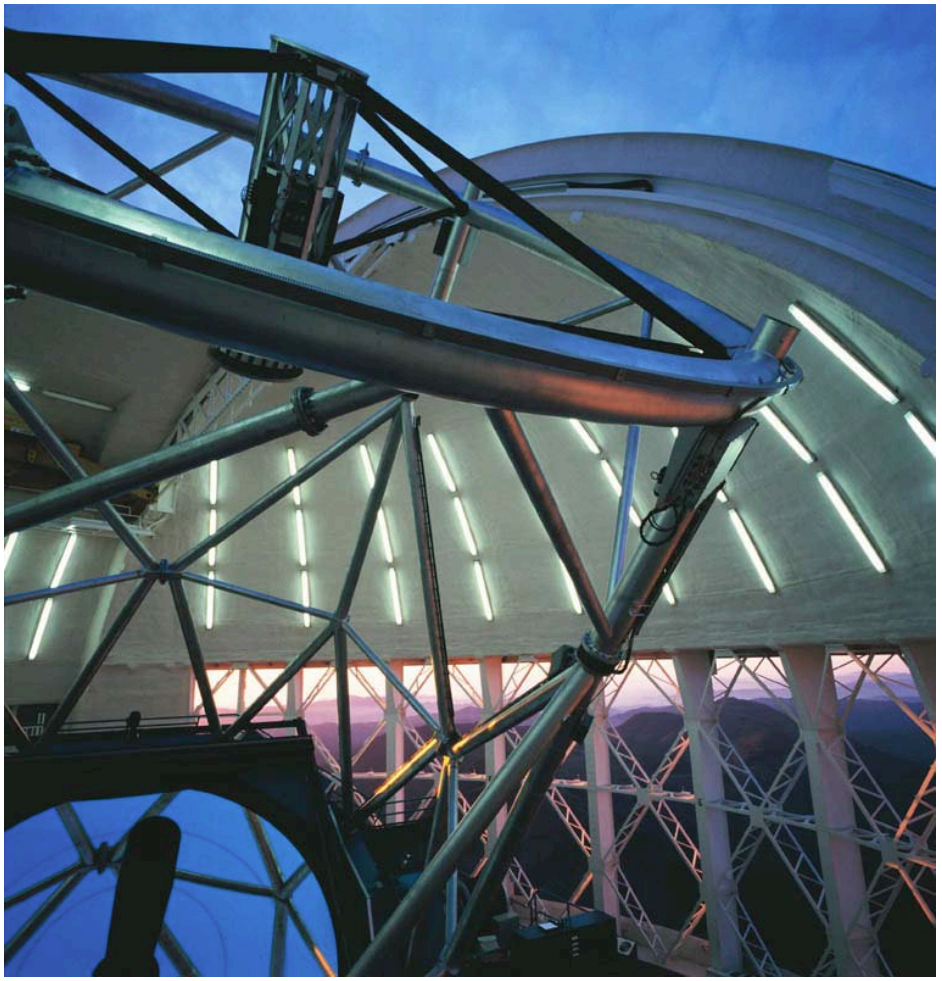
- Hardware connection
- Real time performance – no network latencies
- Whenever a database is good enough

Advantages	**Disadvantages**
Simplify hardware connection	If you have device support
Configuring not programming	You need to understand database use
Database is easily understood by other EPICS developers	
Speed - All processing (often) in same machine	

When to use the sequencer

- For sequencing complex events
- E.g. Parking and unparking a telescope mirror



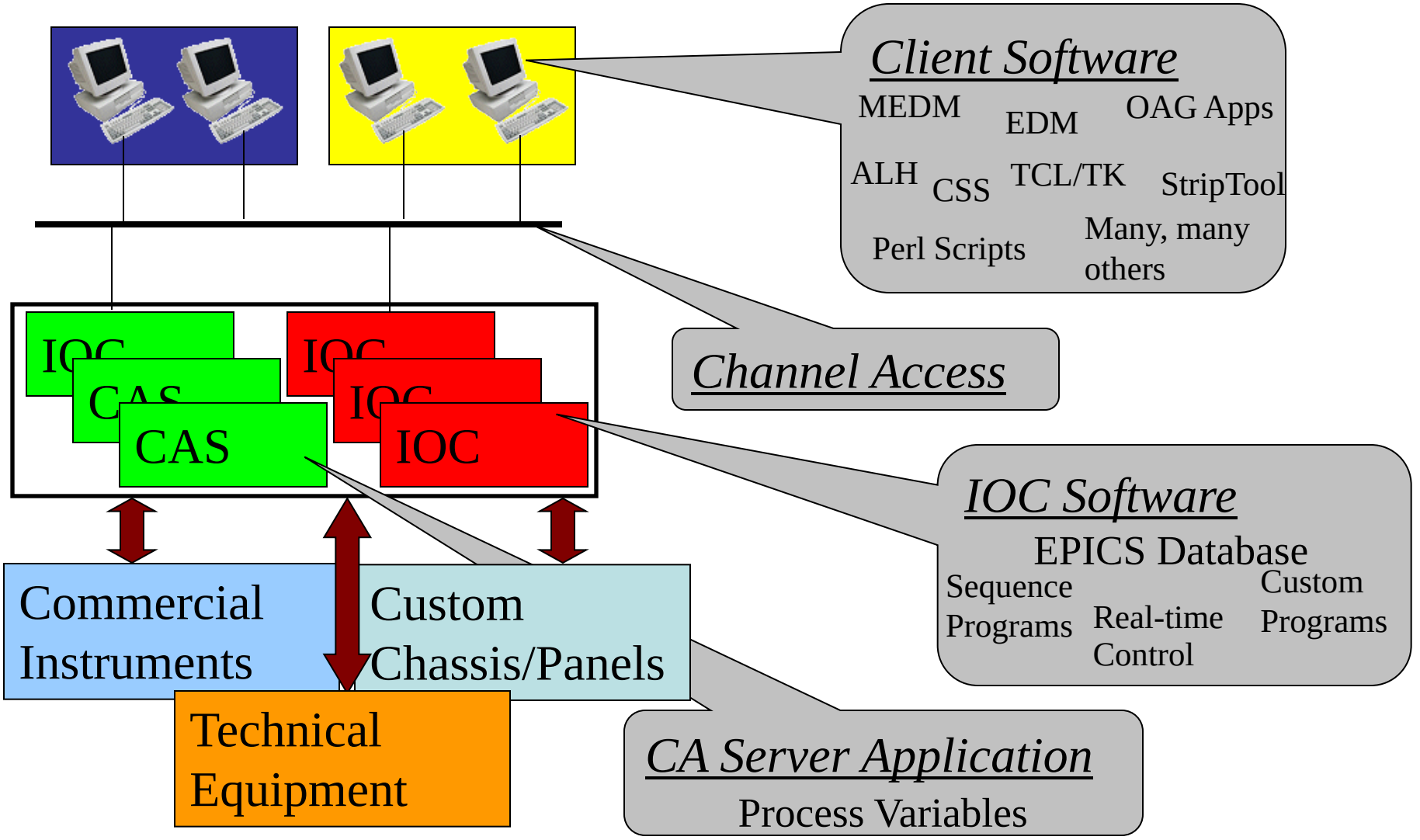


Photograph courtesy of the Gemini Telescopes project

When to use clients

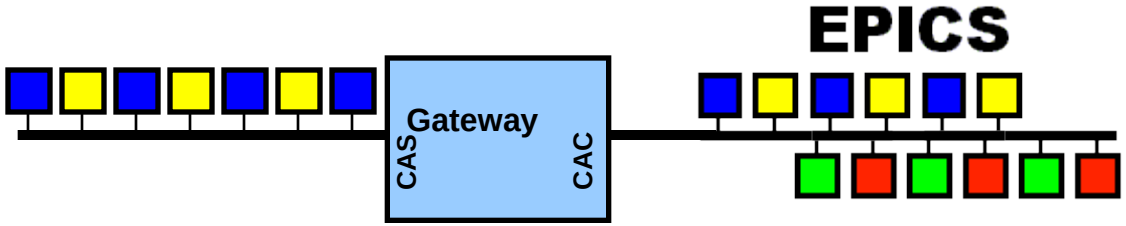
- To interact with the control system
- Many already exist – MEDM, ALH, Strip Tool, archiver etc.
- For data analysis or visualization
- Supervisory control
- E.g. to manage an accelerator

Canonical Form of an EPICS Control System

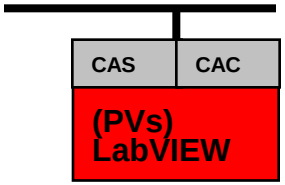


Popular CA Server Applications

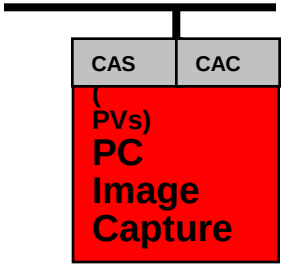
- IOC Core
- PV Gateway



- CA Server Interface to LabVIEW



- CA Server Interface to PC Image Acquisition Systems



Ten really neat things about EPICS

- It is free. No license fees, no new payment for every upgrade. You can download EPICS free of charge from the web.
- It is Open Source (i.e. the source code is accessible). Adaptions and changes due to a special environment are therefore possible.
- There are lots of users. It is tested and most bugs are already found.
- All a client needs to know to access data is a PV name. No single point of failure due to a nameserver and no messing around with fixed addresses.
- You can pick the best tools out there ...
- ... or build your own.
- The boring stuff is already done. For example, the communication with Channel Access is stable and well tested.
- There is a lot of expertise available close by.
- A good contribution becomes internationally known.
- It doesn't matter whether you need 10 PVs or 10 Million PVs. You can scale EPICS almost freely.

Vocabulary

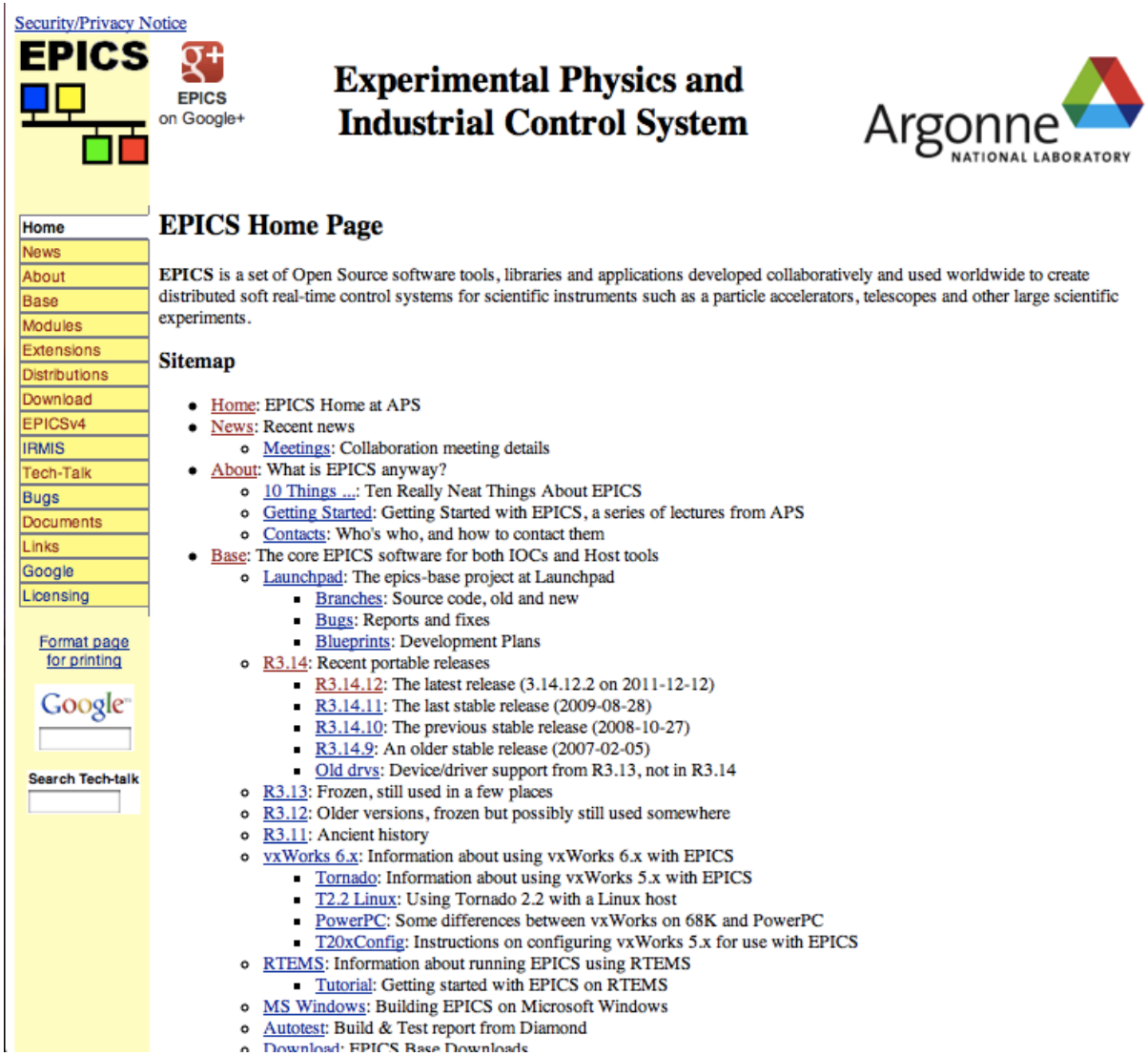
- EPICS
  - Experimental Physics and Industrial Control System
- Channel Access
  - The communication protocol used by EPICS
- Process Variable
  - A piece of named data referred to by its PV name
  - The primary object of the Channel Access Protocol
- Channel
  - A synonym for Process Variable
- Channel Access Server
  - Software that provides access to a Process Variable using the Channel Access Protocol
- Channel Access Client
  - Software that requests access to a Process Variable using the Channel Access Protocol

Vocabulary

- IOC – Input Output Controller
  - A computer running *iocCore,* a set of EPICS routines used to define process variables and implement real-time control algorithms
  - *iocCore* uses database records to define process variables and their behavior
- Soft IOC
  - An instance of *iocCore* running as a process on a “non-dedicated” computer (i.e. a computer that is performing other functions as well)
- Record
  - The mechanism by which a Process Variable is defined in an IOC (using *iocCore*)
  - Dozens of record types exist, each with it’s own attributes and processing routine that describe its functionality

Collaboration Website

http://www.aps.anl.gov/epics/index.php



Security/Privacy Notice

EPICS

EPICS on Google+

Experimental Physics and Industrial Control System

Argonne NATIONAL LABORATORY

- Home
- News
- About
- Base
- Modules
- Extensions
- Distributions
- Download
- EPICSv4
- IRMIS
- Tech-Talk
- Bugs
- Documents
- Links
- Google
- Licensing

Format page for printing

Google

Search Tech-talk

EPICS Home Page

EPICS is a set of Open Source software tools, libraries and applications developed collaboratively and used worldwide to create distributed soft real-time control systems for scientific instruments such as a particle accelerators, telescopes and other large scientific experiments.

Sitemap

- Home: EPICS Home at APS
- News: Recent news
  - Meetings: Collaboration meeting details
- About: What is EPICS anyway?
  - 10 Things ...: Ten Really Neat Things About EPICS
  - Getting Started: Getting Started with EPICS, a series of lectures from APS
  - Contacts: Who's who, and how to contact them
- Base: The core EPICS software for both IOCs and Host tools
  - Launchpad: The epics-base project at Launchpad
    - Branches: Source code, old and new
    - Bugs: Reports and fixes
    - Blueprints: Development Plans
  - R3.14: Recent portable releases
    - R3.14.12: The latest release (3.14.12.2 on 2011-12-12)
    - R3.14.11: The last stable release (2009-08-28)
    - R3.14.10: The previous stable release (2008-10-27)
    - R3.14.9: An older stable release (2007-02-05)
    - Old drvs: Device/driver support from R3.13, not in R3.14
  - R3.13: Frozen, still used in a few places
  - R3.12: Older versions, frozen but possibly still used somewhere
  - R3.11: Ancient history
  - vxWorks 6.x: Information about using vxWorks 6.x with EPICS
    - Tornado: Information about using vxWorks 5.x with EPICS
    - T2.2 Linux: Using Tornado 2.2 with a Linux host
    - PowerPC: Some differences between vxWorks on 68K and PowerPC
    - T20xConfig: Instructions on configuring vxWorks 5.x for use with EPICS
  - RTEMS: Information about running EPICS using RTEMS
    - Tutorial: Getting started with EPICS on RTEMS
  - MS Windows: Building EPICS on Microsoft Windows
  - Autotest: Build & Test report from Diamond
  - Download: EPICS Base Downloads

Tech Talk EPICS Mailing List

Tech-talk Mailing List

The EPICS mailing list is called tech-talk, and is used for technical discussions, questions and announcements. See below for subscription instructions. There is another mailing list called core-talk which is for use by the developers working on the EPICS core software.

The tech-talk list accepts posts from subscribers; messages from an email address that is not subscribed to the list are still able to post but these messages will be delayed pending moderator approval, and if judged to be Spam may be discarded without notice. Members of the EPICS community who wish to be exempted from the moderation requirement without subscribing can request this by emailing the list administrator.

When posting messages to the list, please observe conventional netiquette. In particular, if you are starting a new line of discussion, please do **not** reply to an older tech-talk message and change the subject line — your mail program will attach a hidden reference header to the message, linking it to the message you are replying to and causing other mail programs and the tech-talk archives to think your message is part of that same thread. Start by hitting "New Message" whenever you want to start a new conversation.

Please don't attach tarfiles or zipfiles or directly include files with the extension `.cmd`, such as IOC startup scripts files `st.cmd`. A `.cmd` extension is used on Microsoft Windows for a kind of command script. Many spam filters nowadays search through tarballs and zipfiles and will either reject such messages or remove the file from it, but in either case they result in many rejection messages coming in to the list manager.

List Archives

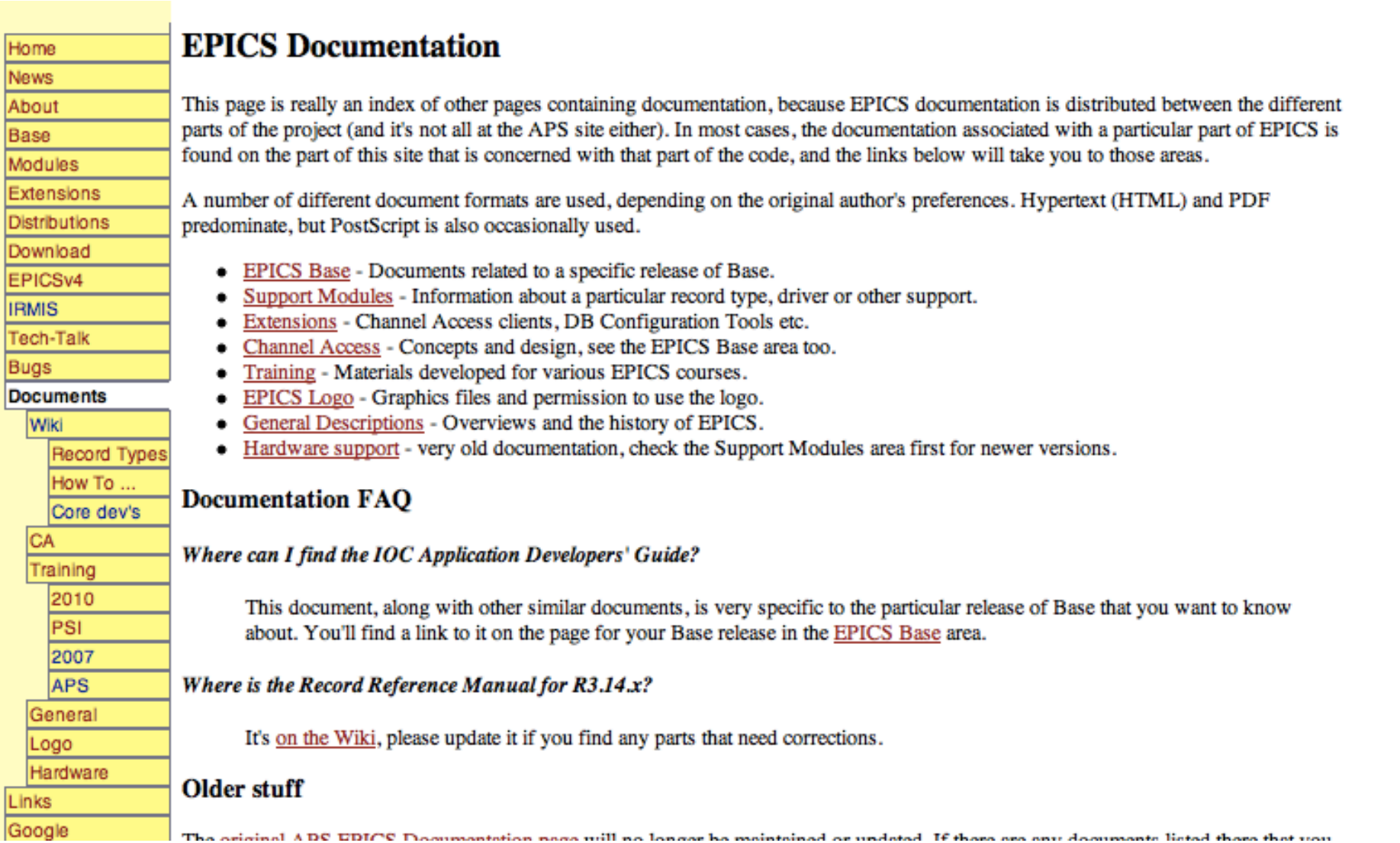
Messages sent to the list are archived annually.

Indexed by Date:
1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012

Indexed by Thread:
1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012

The tech-talk archives are searchable. The archive search index is updated overnight, so today's messages won't appear in any search results. Give a space- or comma-separated list of words to search for. Only whole words will be matched. Check the box to return a list of messages containing **any** of the search words, othewise only messages containing **all** the words will be returned.

Online Resources and Additional Training



- Home
- News
- About
- Base
- Modules
- Extensions
- Distributions
- Download
- EPICSv4
- IRMIS
- Tech-Talk
- Bugs
- **Documents**
  - Wiki
    - Record Types
    - How To ...
    - Core dev's
  - CA
  - Training
    - 2010
    - PSI
    - 2007
    - APS
  - General
  - Logo
  - Hardware
- Links
- Google

EPICS Documentation

This page is really an index of other pages containing documentation, because EPICS documentation is distributed between the different parts of the project (and it's not all at the APS site either). In most cases, the documentation associated with a particular part of EPICS is found on the part of this site that is concerned with that part of the code, and the links below will take you to those areas.

A number of different document formats are used, depending on the original author's preferences. Hypertext (HTML) and PDF predominate, but PostScript is also occasionally used.

- EPICS Base - Documents related to a specific release of Base.
- Support Modules - Information about a particular record type, driver or other support.
- Extensions - Channel Access clients, DB Configuration Tools etc.
- Channel Access - Concepts and design, see the EPICS Base area too.
- Training - Materials developed for various EPICS courses.
- EPICS Logo - Graphics files and permission to use the logo.
- General Descriptions - Overviews and the history of EPICS.
- Hardware support - very old documentation, check the Support Modules area first for newer versions.

Documentation FAQ

Where can I find the IOC Application Developers' Guide?

This document, along with other similar documents, is very specific to the particular release of Base that you want to know about. You'll find a link to it on the page for your Base release in the EPICS Base area.

Where is the Record Reference Manual for R3.14.x?

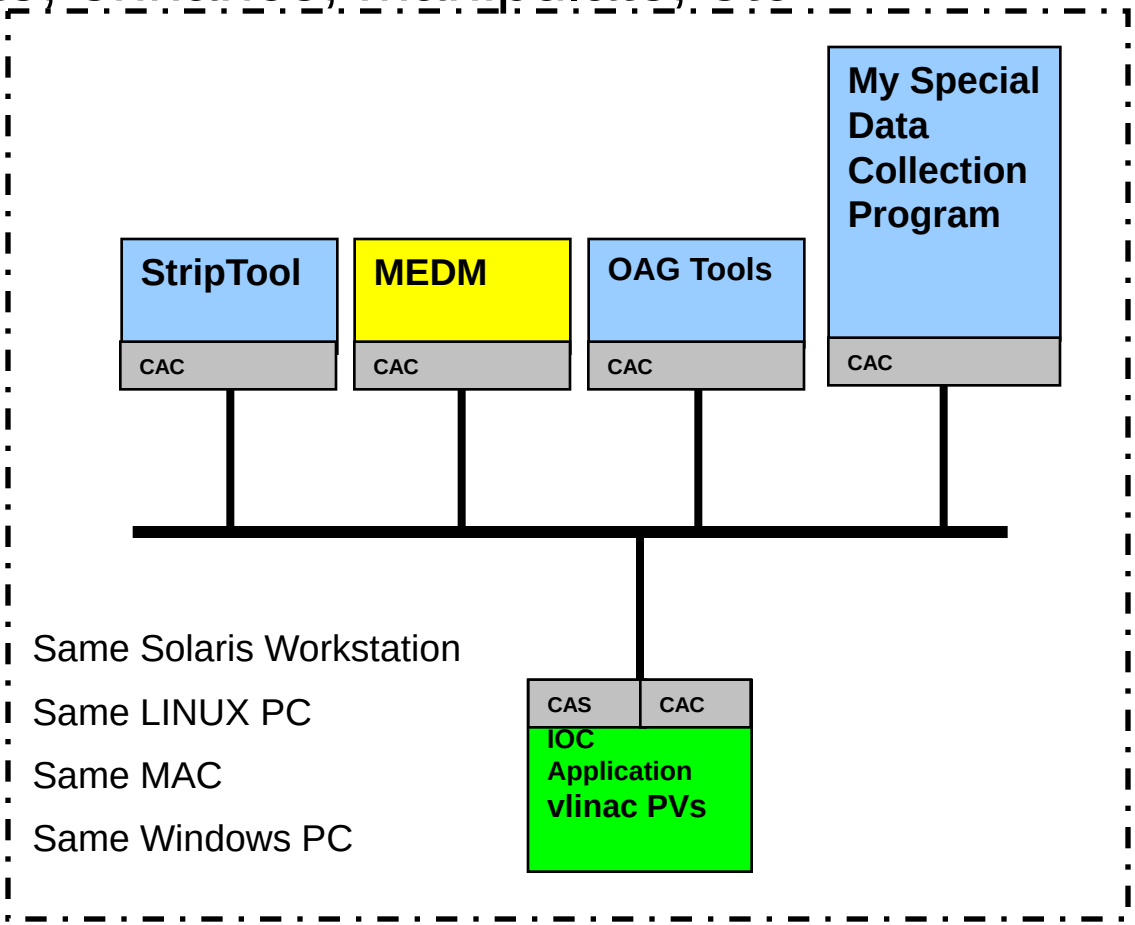
It's on the Wiki, please update it if you find any parts that need corrections.

Older stuff

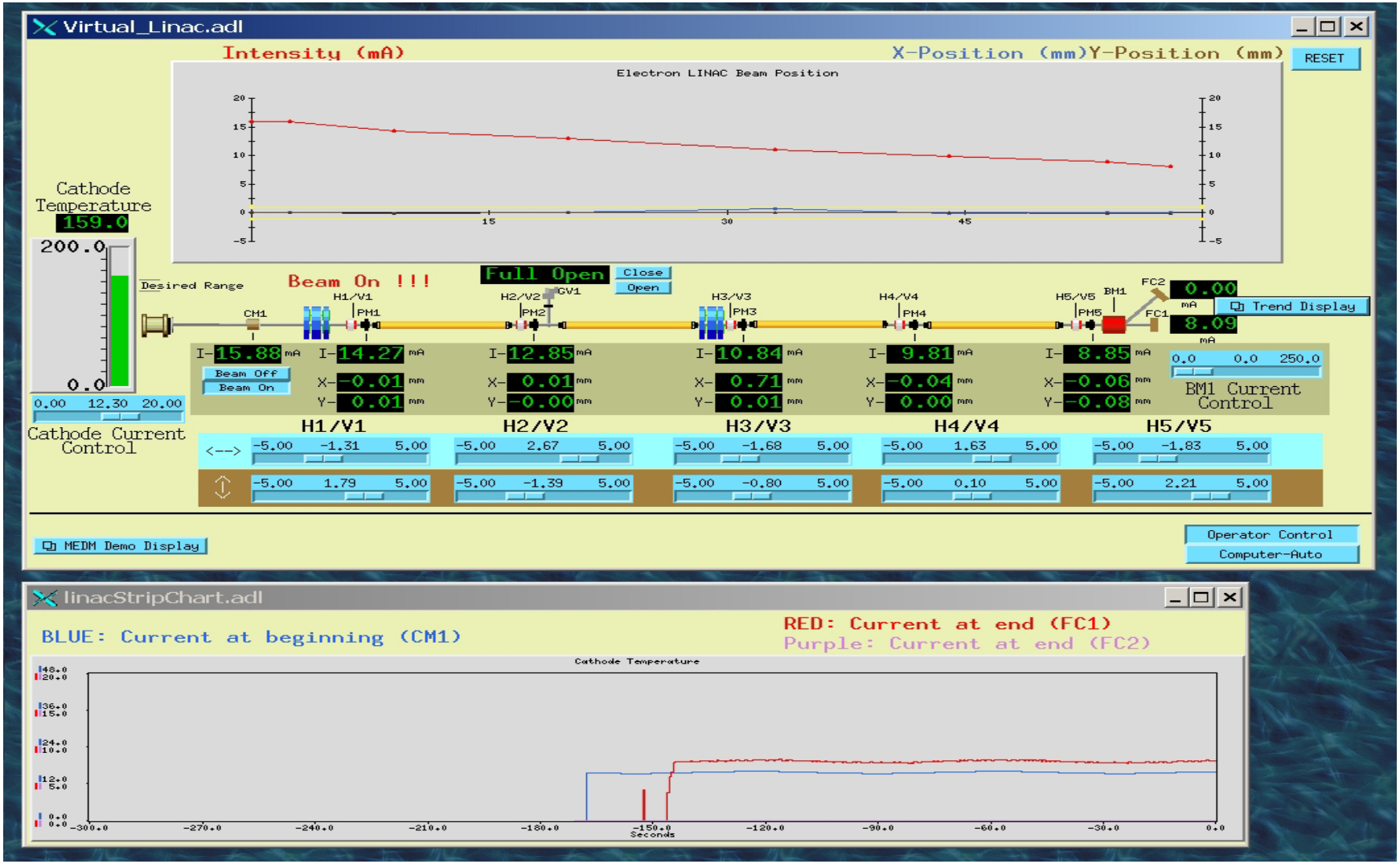
The original APS EPICS Documentation page will no longer be maintained or updated. If there are any documents listed there that you

Virtual LINAC Application

- A pre-packaged EPICS application for you to install, operate, enhance, manipulate, etc.



Virtual LINAC Application



Virtual LINAC Application

EPICS on Google+

Experimental Physics and Industrial Control System

Argonne NATIONAL LABORATORY

- Home
- News
- About
- Base
- Modules
- Extensions
- Distributions
- Download
  - Base
  - Modules
  - Extensions
  - Distributions
  - Examples
  - Browse CVS
- EPICSv4
- IRMIS
- Tech-Talk
- Bugs
- Documents
- Links

EPICS Examples Downloads

This page contains pointers to some example EPICS IOC applications which are distributed by APS under the terms of the EPIC Open license.

- Point Releases
- CVS Snapshots

Point Releases

The following releases of example EPICS IOC applications are available for download.

Application Name	Version or Date	File Size	Click to Download
Virtual Linac ISO CDROM image (gzipped)	V0.7	80479 KB	vlinac0-7.iso.gz
Virtual Linac source code	2004-08-18	59 KB	vlinac_src.tgz
Virtual Linac demo for Sun Solaris on sparc	2004-08-18	2669 KB	vlinac_solaris-sparc.tgz

LivEPICS from INFN

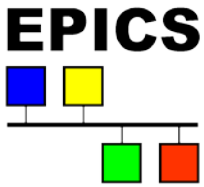
- **LivEPICS(c)** is a Linux Live CD (and a USB bootable all-in one solution)

that includes: Epics base, Extensions tools, introductory documents and manuals. It has the complete functionality to develop a small control system, but it is mainly intended for training classes or to monitor and supervise an EPICS network

- All the Epics software (sources and built) is placed into */opt* folder:
  - base-3.14.10 (default)
  - base-3.14.9
  - Extensions:StripTool2.5.12.0, ALH1.2.23,MEDM3.1.2.1,Probe,msi1.4,VDCT
  - Modules:Asyn4.10(default), Asyn4.9,Asyn4.8, AutoSave, ProcServer2.4.0
  - And some additional software
- Copies Available Here

Next Steps in Training

- **Introduction to EPICS**
  - What is EPICS? What are the lectures about? **DONE**
- Getting Started with Input/Output Controllers (IOCs)
  - **EPICS Databases**, VisualDCT, **State Notation Language**, **Record and Device Support**, **ASYN**, etc.
- Getting Started with Using EPICS Tools
  - MEDM, EDM, **CSS**, Alarm Handler, Channel Archiver, etc.
- Getting Started with Developing EPICS Tools
  - Tcl/Tk, Perl, IDL, JAVA, Python, **CA Server Applications**
- Applications/Special Topics
  - synApps, motors, scans, data visualization, etc.



Thank You