EPICS

Peter Zumbruch

Experiment control systems group GSI
(KS/EE)

EPICS

GSI

Outline

- Part I

  EPICS

- Part II

  EPICS @ GSI

Part I

EPICS

(many slides taken from the presentation of B.Dalesio, BNL @ PAC 2009, Vancouver)

Outline (EPICS)

- Introduction: What is EPICS
  - The Collaboration
  - Architecture
  - Tool-Kit
- IOC Core
  - Channel Access
  - Process Database
- Conclusions

What is EPICS?

… **short answer**:

*EPICS: **E**xperimental **P**hysics and **I**ndustrial **C**ontrol **S**ystem*

… **a bit more elaborate**:

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 particle accelerators, telescopes and other large scientific experiments. (From the EPICS Home Page: http://www.aps.anl.gov/epics/)

… ***striking*** - is three things at once:

- A ***collaboration*** of major scientific laboratories and industry (> 150)
  - A world wide collaboration that shares designs, software tools and expertise for implementing large-scale control systems
- An ***architecture*** for building scalable control systems
  - A client/server model with an efficient communication protocol (Channel Access) for passing data
  - The entire set of Process Variables establish a distributed Real-time Database of machine status, information and control parameters
- A ***Software Toolkit*** of Open Source code and documentation
  - A collection of software tools collaboratively developed which can be integrated to provide a comprehensive and scalable control system

What is EPICS?

(Bob)

- A ***collaboration*** of the controls groups of many research organizations that use the EPICS tool-kit.
- A distributed ***architecture*** that supports a wide range of solutions from small test stands to large integrated facilities.
- A set of ***tools*** that reduces software application and maintenance costs by providing:
  - Configuration tools in place of programming
  - A large installed base of tested software
  - A modular design that supports incremental upgrades
  - Well defined interfaces for extensions at every level

What is EPICS? The Collaboration

- Over 150 independent projects in North America, Europe, Africa, Australia, South America and Asia
- Applications in particle physics, astronomy, and industrial control
- Distribute software over the network
- Independent development, co-development and incremental development of code done by members
- Problem reporting and resolution via e-mail exploders (tech-talk / core-talk)
- Documentation available on WWW sites
- Large collaboration meetings to report new work, discuss future directions, explore new applications, and explore new requirements for existing codes
- Small design groups from multiple labs meet to discuss design issues on significant codes: Channel Access, Archiving and MMI

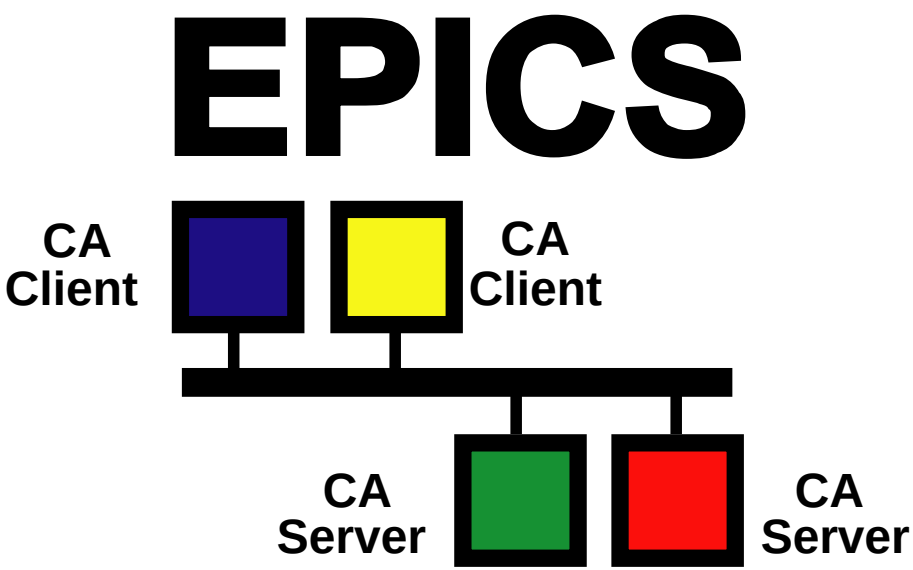
What is EPICS? Distributed Architecture

- **EPICS is physically a flat architecture of front-end controllers and operator workstations that communicate via TCP/IP and UDP**
  - System scales through the addition of new computers
  - Physical hierarchy is made through bridges, routers, or a gateway
  - Network bandwidth is the primary limiting factor
- **EPICS software architecture is client/server based - with independent data stores providing read/write access directly between any two points**
  - Local name services mean automatic integration of new components
  - Point-to-point communication supports automation

What is EPICS? (*Getting Started with EPICS: Introductory Session I*)

A Control System Architecture

Network-based “client/server” model (hence the EPICS logo)

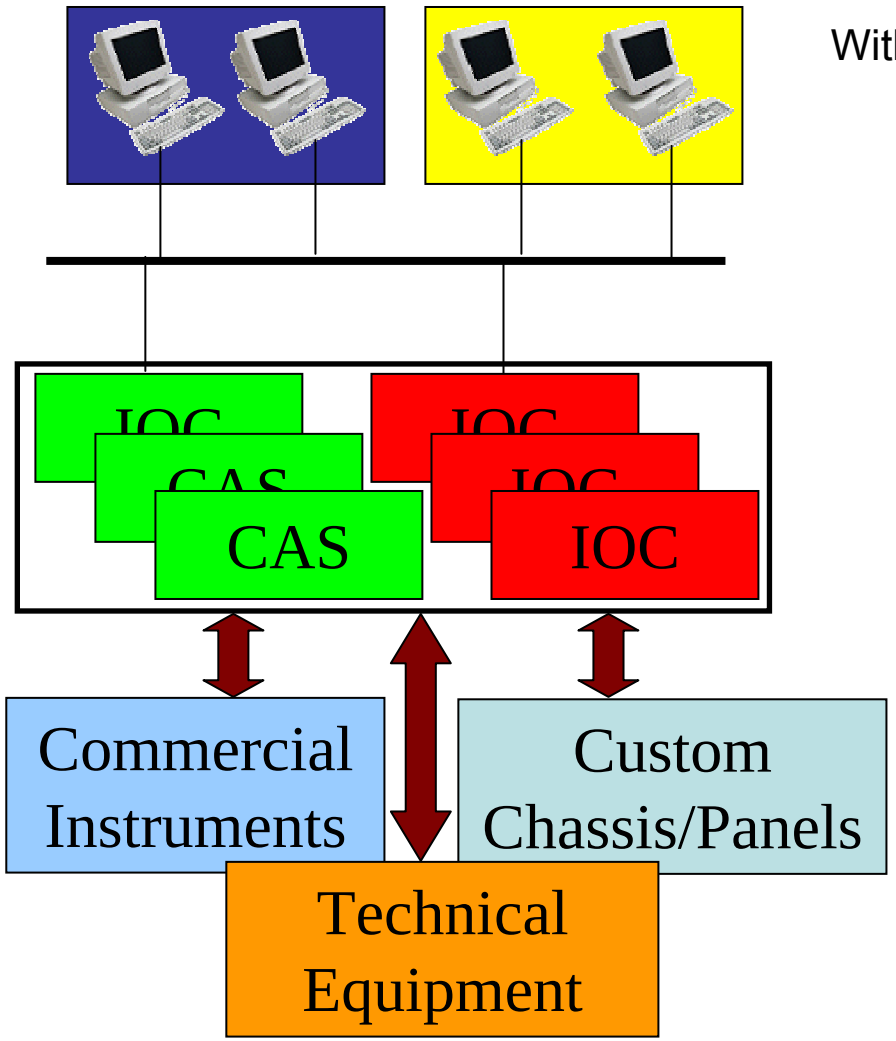


For EPICS, *client* and *server* speak of their Channel Access role

i.e. Channel Access Client & Channel Access Server

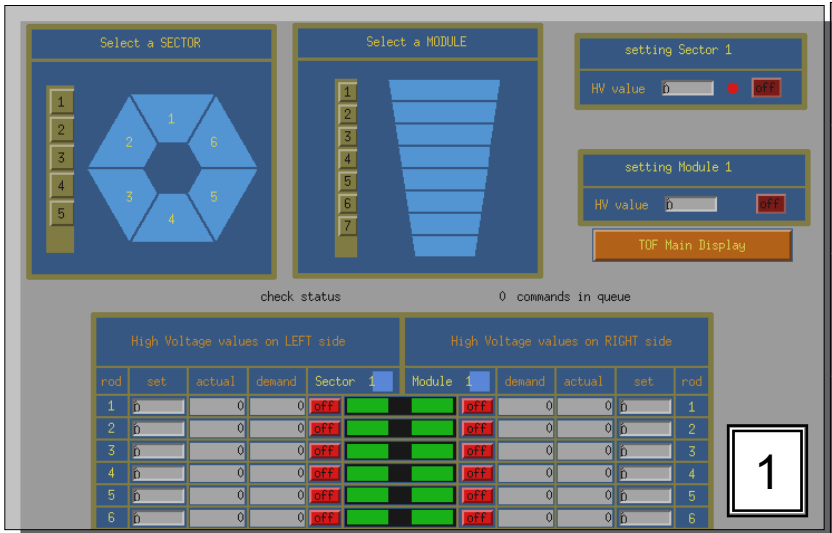
Typical Realizations of an EPICS System

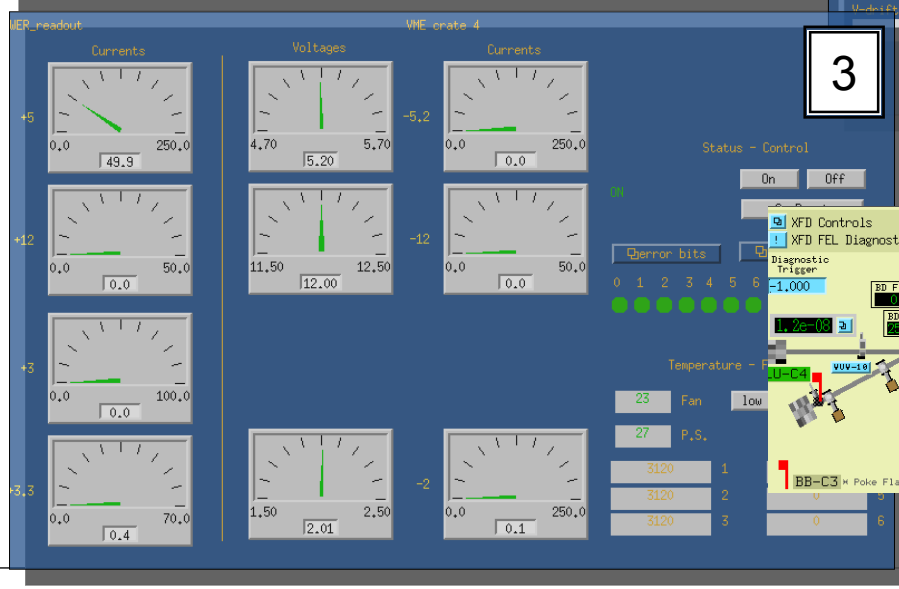
(Getting Started with EPICS: Introductory Session I)



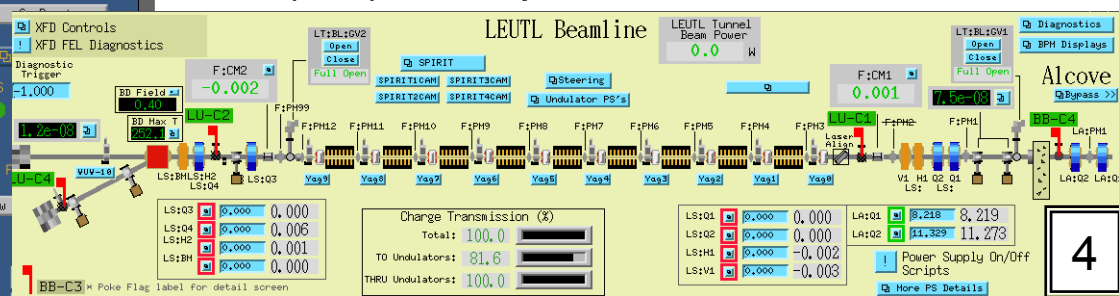
With Release 3.14, the operating system limitations for iocCore have been removed.

Displays and Controls





(1-3) Examples from HADES



interplay: CSS

Control System Studio

Eclipse and Java based

Integrated Development Environment

Developed at DESY

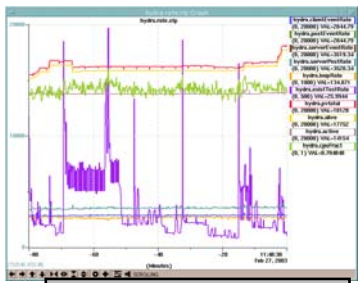
From the first principle independent of EPICS

… but via DAL (cosylab: DATA Access Layer) access too many different control systems

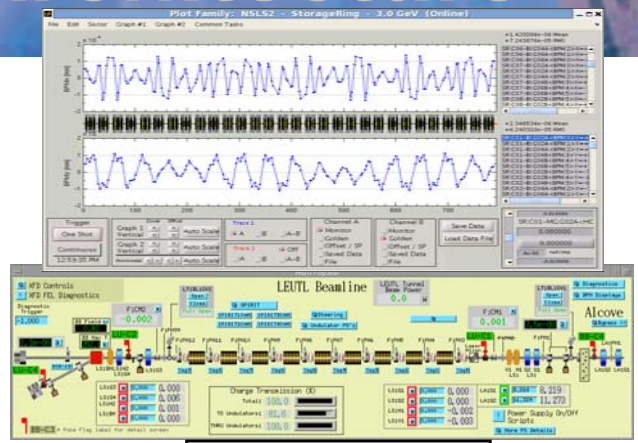
- EPICS, TINE, (GSI beam controls,) TANGO, …
- replaces (soon) the Motif based, old EPICS GUIs
- good collaboration with developers
- http://css.desy.de http://ics-web.sns.ornl.gov/css/products.html

Control System Studio
File CSS Quickstart Window Help
CVS Repository Exploring
CSS Standard
Navigator
LabVIEW
SDS
SDS Cursors
SDS Demo Display
SDS Demo Display 2
SDS Script Rules
Temperaturen [lxdv18.gsi.de]
.project 1.1
24h_plot_p4.css-plt 1.1
24h_plot_p4_archive.css-plt 1.
Copy of 24h_plot_p4_archive.c
Copy of plot_p4_archive.css-plt
Temperaturen.css-sds 1.1
Temperaturen_p4.css-sds 1.1
XY.css-sds 1.1
dis.css-sds 1.1
hades.css-plt 1.1
plot.css-plt 1.1
plot_p4.css-plt 1.1
plot_p4_archive.css-plt 1.2
scales-small.css-sds 1.1
tab.css-pvtable 1.1
table.css-pvtable 1.1
table_p4.css-pvtable 1.1
waage-hades.css-plt 1.1
Temperaturen 2
*plot_p4_archive.css-plt
Run Mode View
Temperatures
40.31
etrax048:MCM
31.19
room
31.00
outside #1
30.75
outside #2
Pressure
Run Mode View
Time [s]
24h_plot_p4_archive
24h_plot_p4
Not logged in

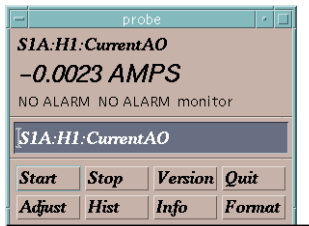
Overview of the Architecture



Channel Access Client



Channel Access Client



Channel Access Client

Network (Channel Access Protocol)

Channel Access Server

Process variable:

S1A:H1:CurrentAO

S1:P1:x

S1:P1:y

S1:G1:vacuum

Power Supply Cntrl → **Power Supply** → S1A:H1:CurrentAO

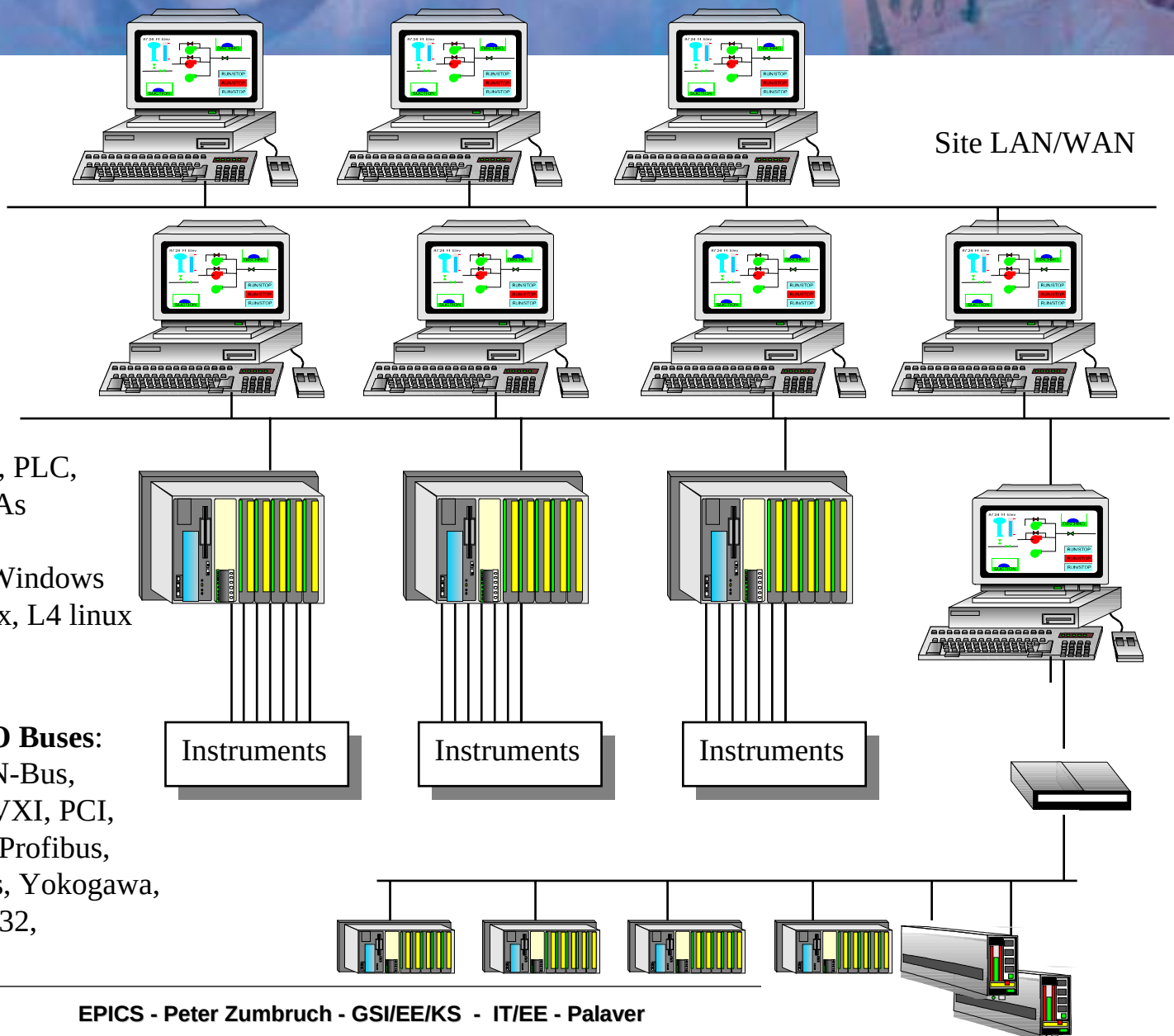
Brilliance ← **Beam Position Monitor** ← S1:P1:x, S1:P1:y

PLC/RS232 ← **Vacuum Control** ← S1:G1:vacuum

EPICS Supports a Standard Control System Architecture

OS:

Unix
Windows
MAC



I/O Controllers:

VME,VXI
PCI, PCIe, uTCA, PLC,
Softcores on FPGAs

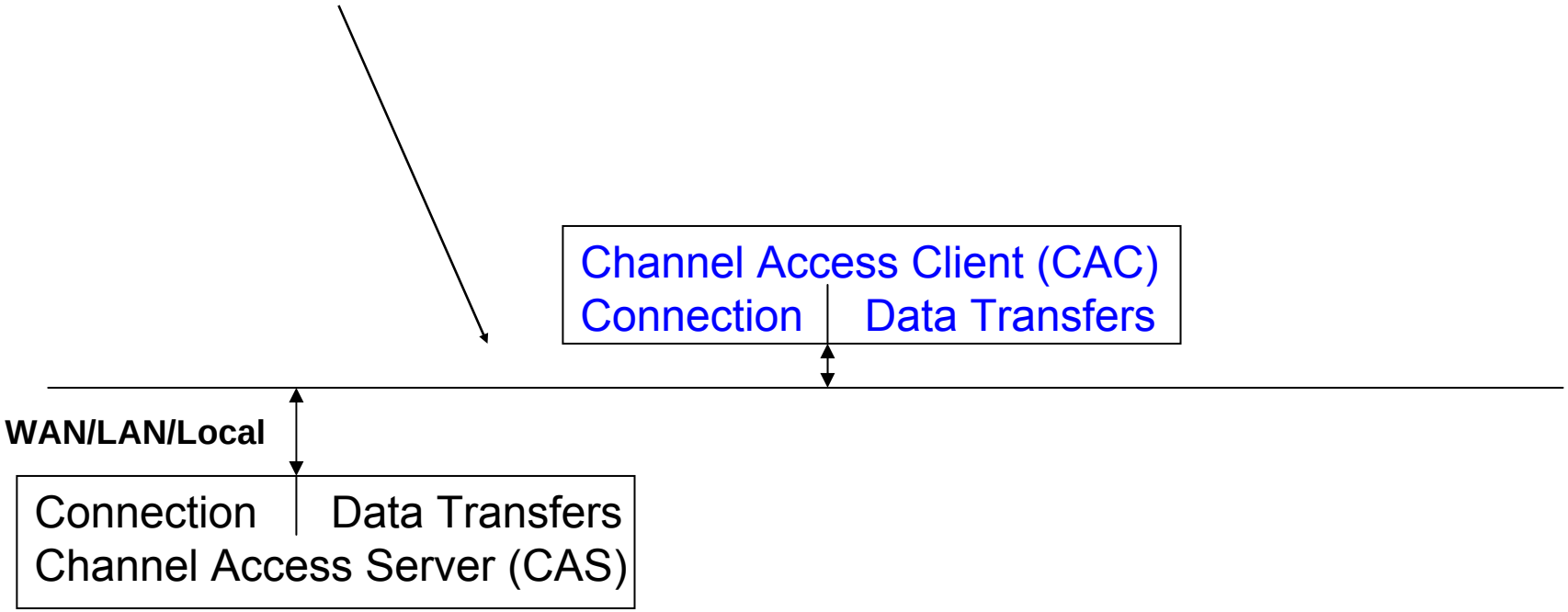
OS:

vxWorks, Unix, Windows
RTEMS, RTLinux, L4 linux

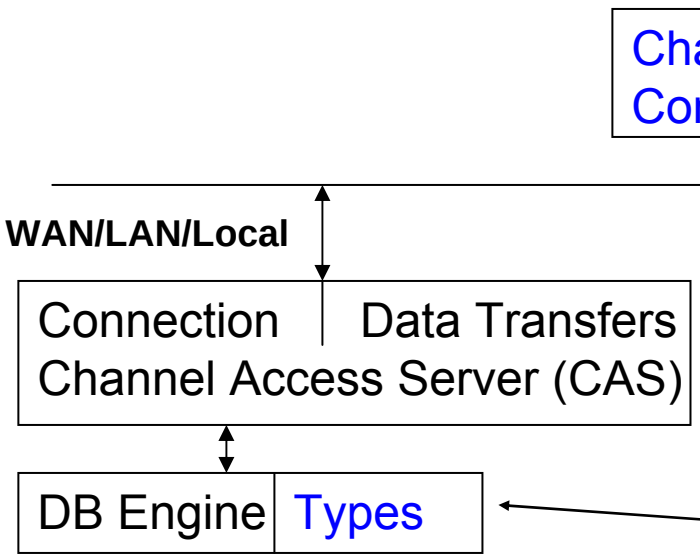
Remote and Local I/O Buses:
Control Net, PCI, CAN-Bus,
Industry Pack, VME, VXI, PCI,
ISA, CAMAC, GPIB, Profibus,
Bitbus, Serial, Modbus, Yokogawa,
G-3, Ethernet/IP, RS232,
Many more….

EPICS Architecture

Is based on the channel access protocol



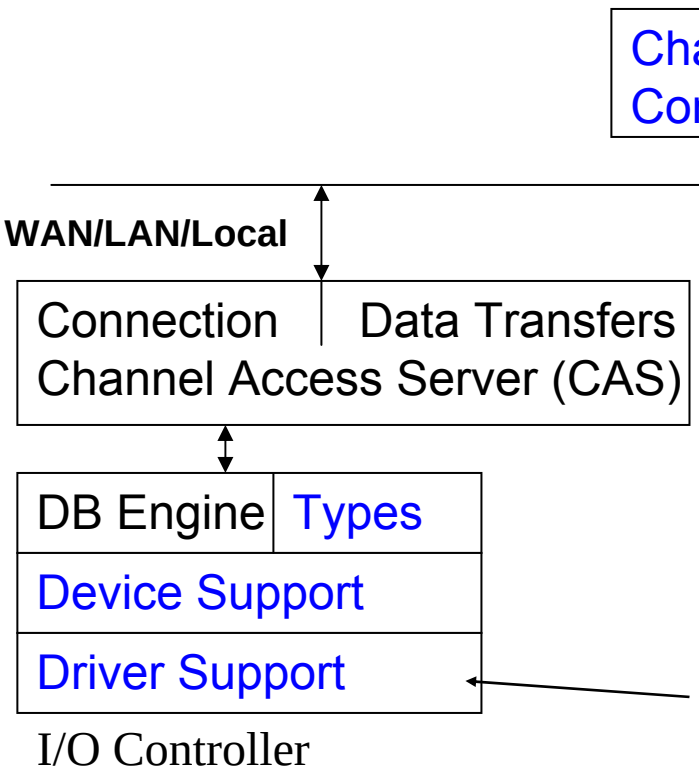
EPICS Architecture



With a process database engine that describes conversions to engineering units, alarm limits, steady state logic for PID control, interlocks, and complex data conversions or filtering

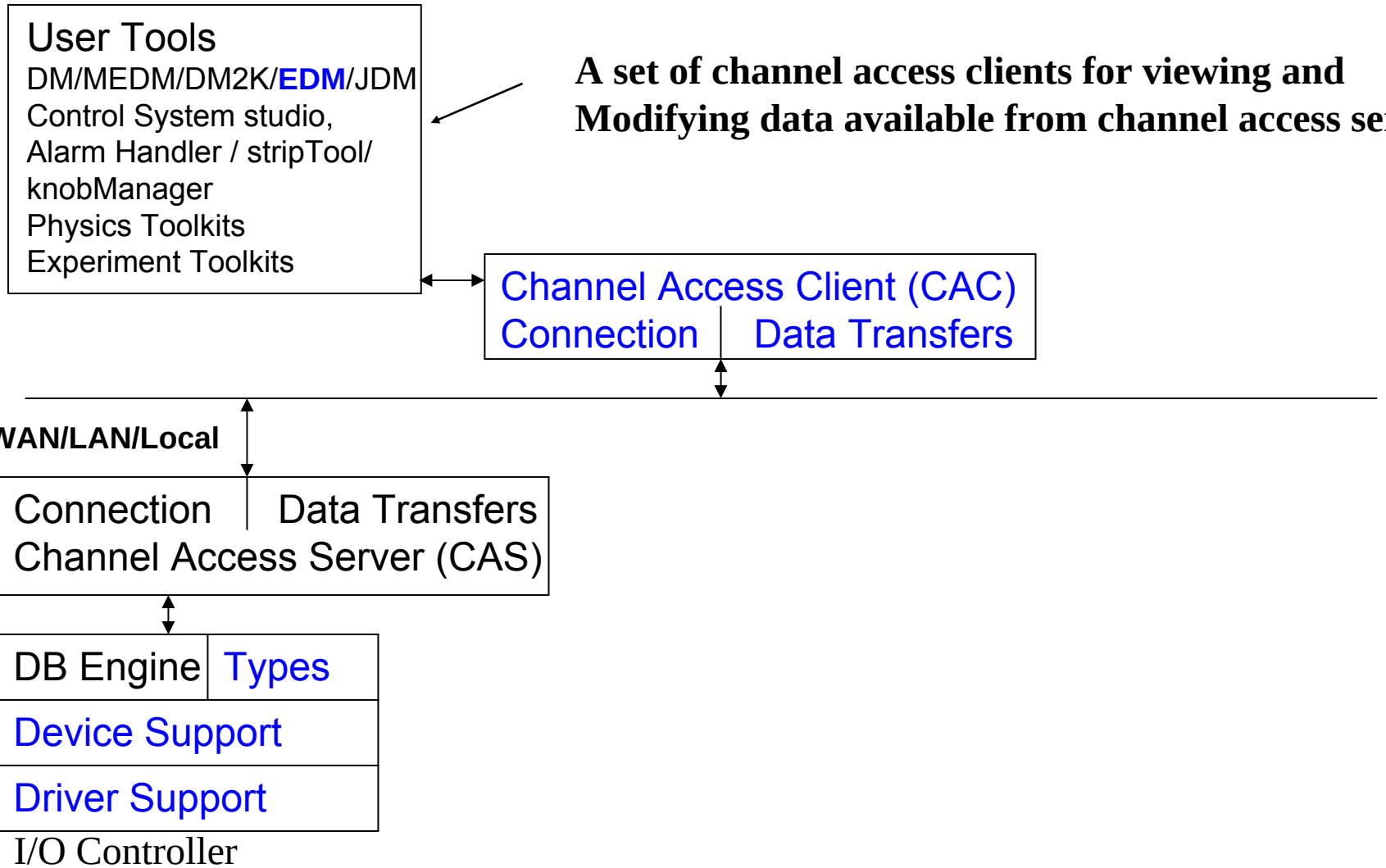
I/O Controller

EPICS Architecture

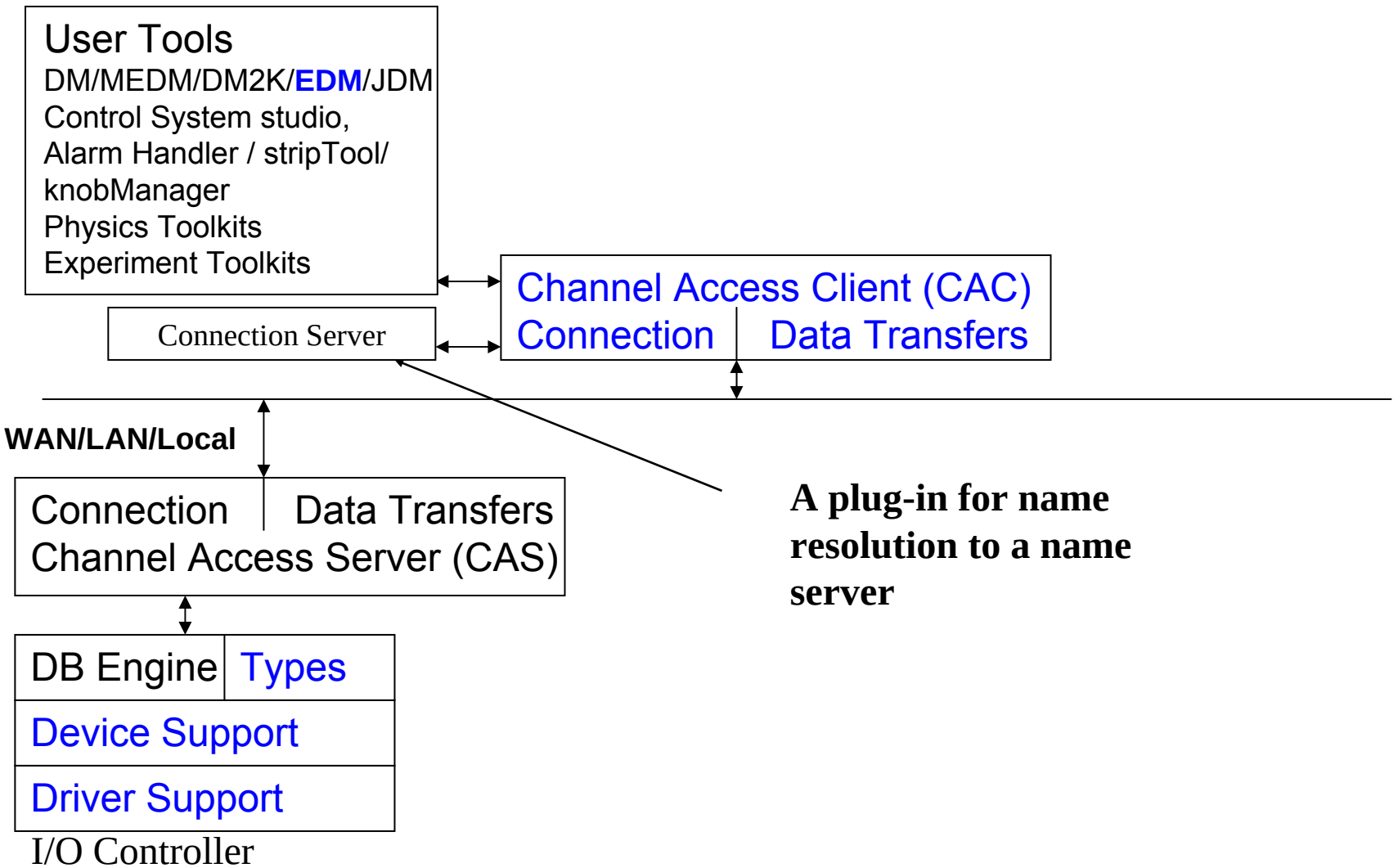


Device and driver support to integrate protocols required to communicate with a large variety of instrumentation and intelligent devices.

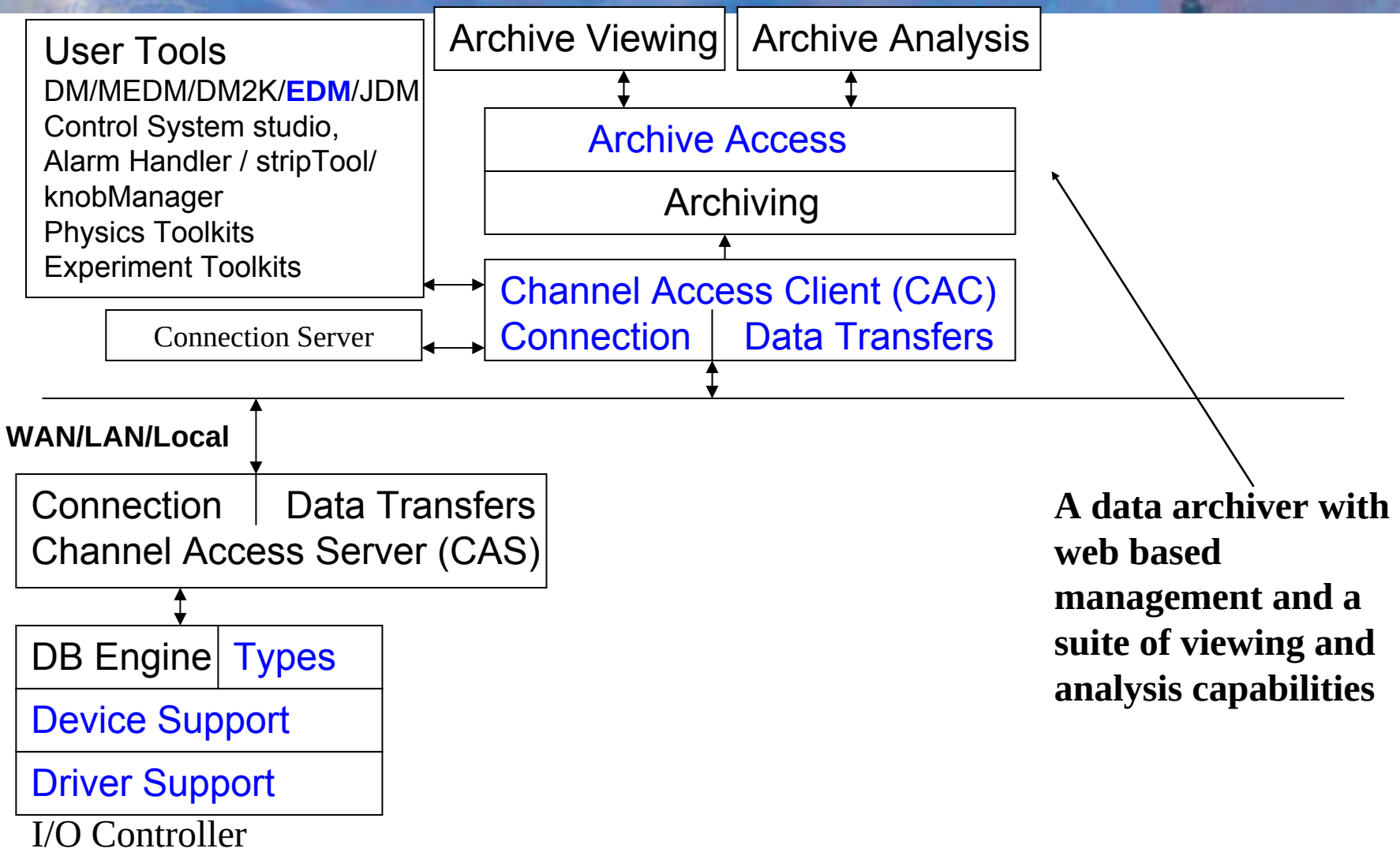
EPICS Architecture



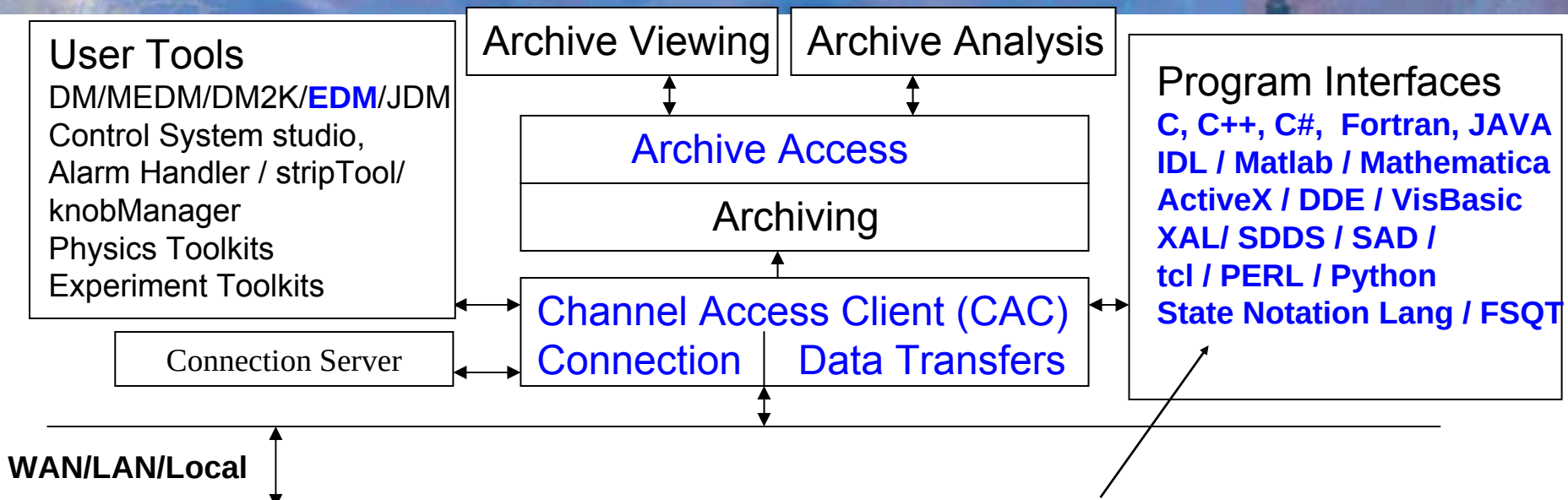
EPICS Architecture



EPICS Architecture

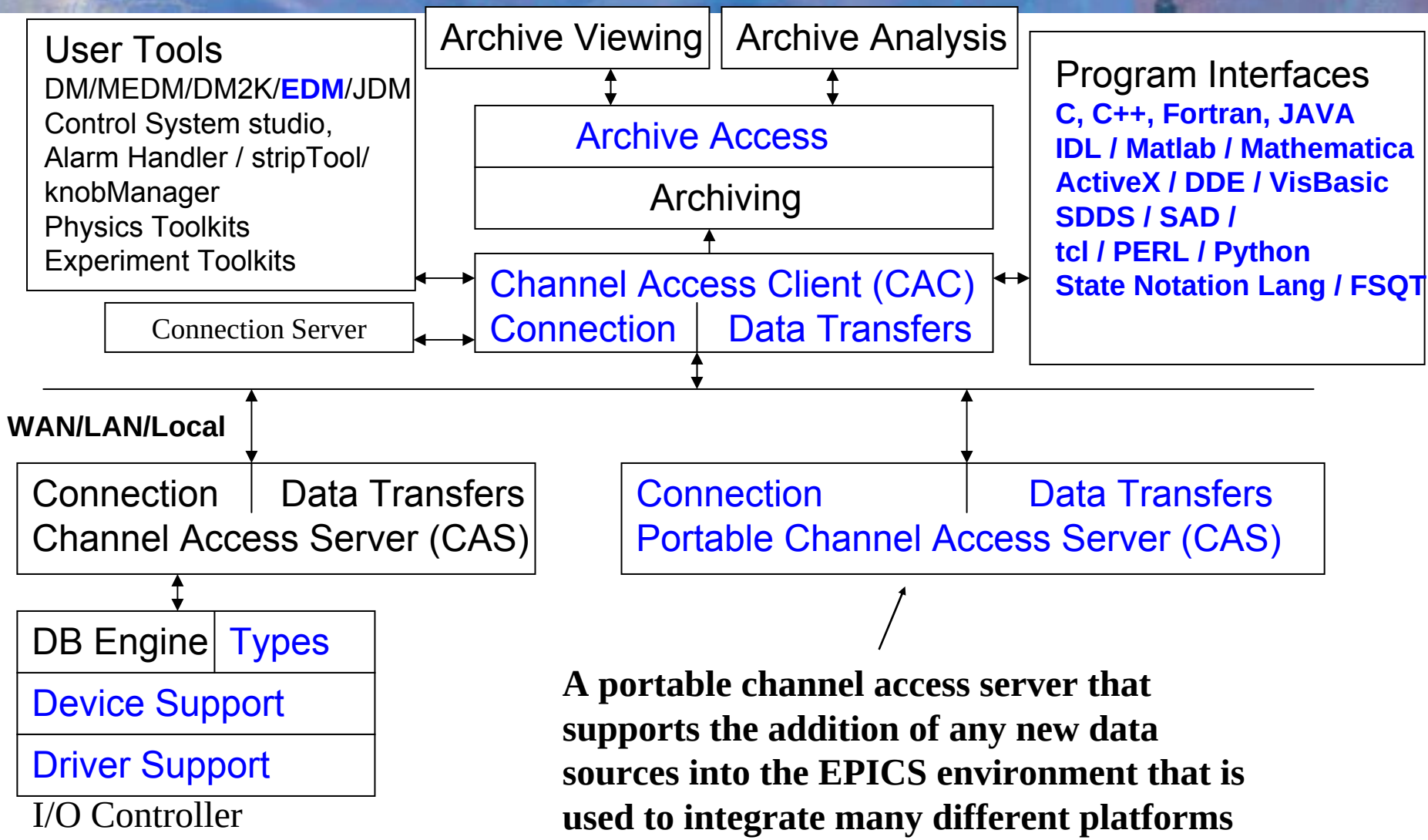


EPICS Architecture



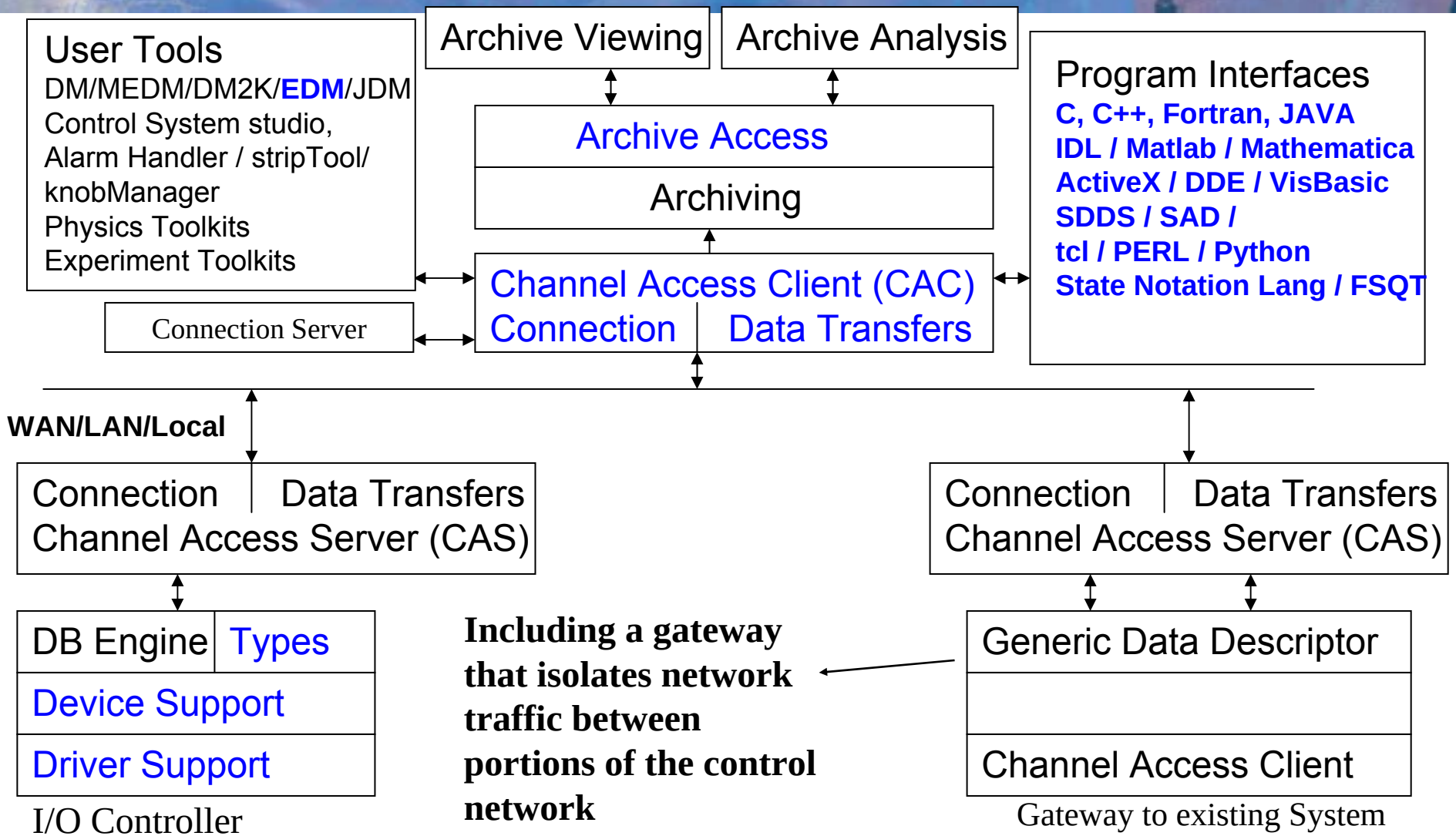
Programming interfaces for creating new channel access clients through standard language interfaces, scripting languages, specialized languages for state transition implementations, mathematical and modeling languages, and specialized accelerator analysis and control platforms

EPICS Architecture

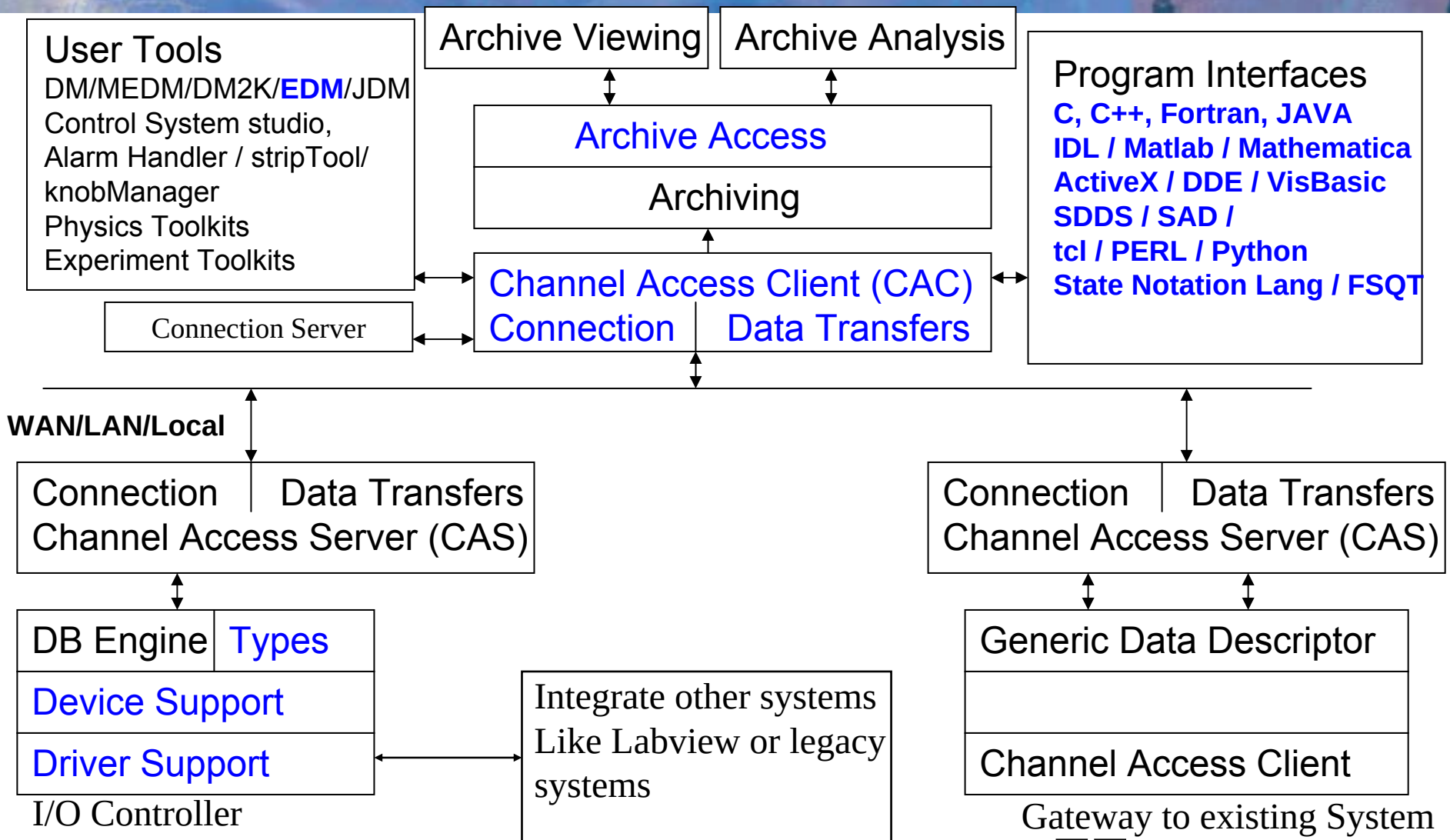


A portable channel access server that supports the addition of any new data sources into the EPICS environment that is used to integrate many different platforms

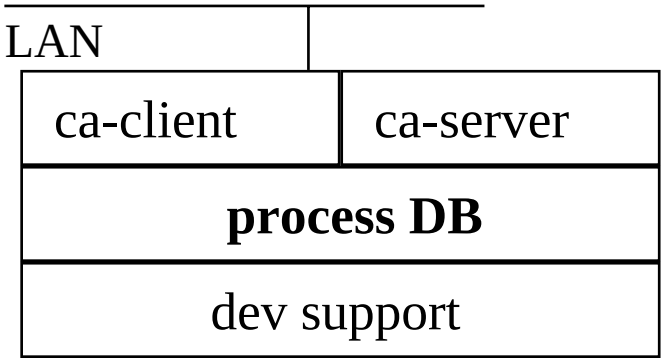
EPICS Architecture



EPICS Architecture



IOC Core: Process Database



A Channel Access server provides connection, get, put, and monitor services to this database

A Channel Access client provides access to process DBs in other IOCs

Process Blocks are the basic elements: AI, AO, BI, BO, Motor, CALC, PID, SUB, etc.…

Process Blocks consist of fields for: SCHEDULE, I/O, CONVERT, ALARM, MONITOR

They hold runtime values: VALUE, TIMESTAMP, ALARM CONDITION, etc.…

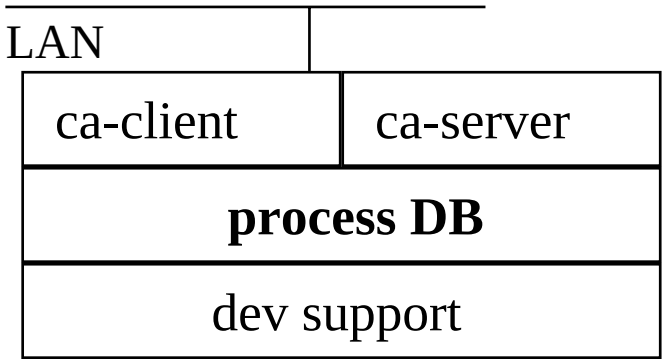
New process block are easily added

Configured using VDCT, GDCT, Relational DB, Text Editor at the workstation

Loaded as ASCII records at boot time

All fields can be read/written through the channel access client interface during operation

IOC Core: Process Database



Process Block execution time varies from block type to block type

AI on a PowerPC ~100,000/second (50% idle)
AI on a 68060 is ~18,000/second (50% idle)
AI on a 68040 is ~6,000/second (50% idle)

Fastest periodic scan rate is dependent on vxWorks clock tick – 1 KHz

Interrupt scanning is limited by the CPU bandwidth (interrupt delay ~15usec)

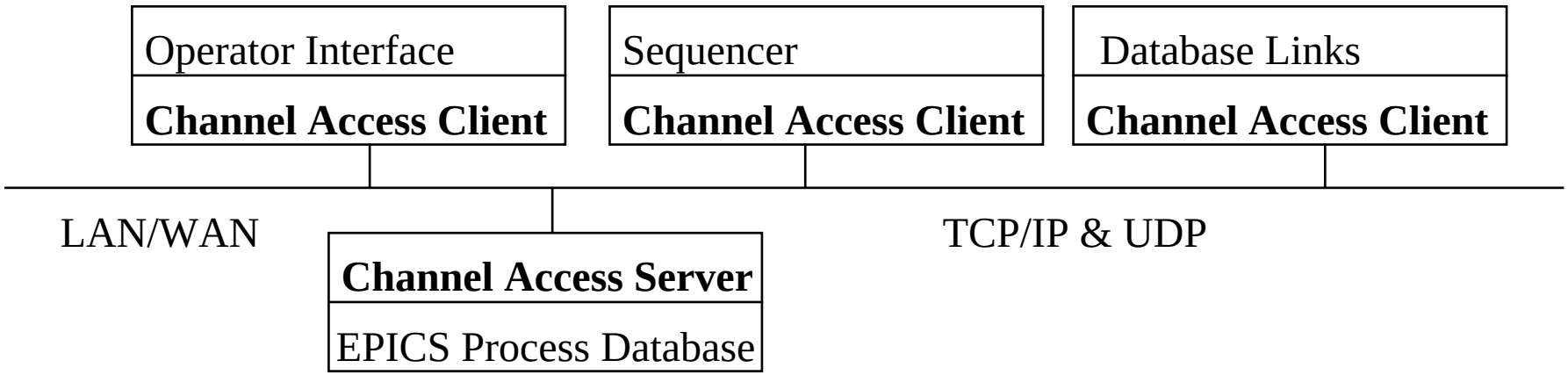
Name resolution - 10,000/second – on an 040 - runs at the lowest priority

2,500 Process blocks use around 1 Megabyte of memory

Support to particular physical I/O is distinct from process logic - ASCII device type
Platforms: UNIX, vxWorks, VMS (Client only), Windows NT, RTEMS, rtLinux

IOC Core

Channel Access Client/Server Libraries



Client: Provides read/write connections to any subsystem on the network with a channel access server

Server: Provides read/write connections to information in this node to any client on the network through channel access client calls. The data resides here!

Services: Dynamic Channel Location, Get, Put, Monitor
Access Control, Connection Monitoring, Automatic Reconnect
Conversion to client types, Composite Data Structures

Platforms: UNIX, vxWorks, VMS (Client only), Windows NT, RTEMS, rtLinux

IOC Core: Mapping Records to Channels

AI:
Name
SCAN
VAL
STAT ACK
SEVR ACKT
HOPR EGU
LOPR
HIHI HHSV
HIGH HSV
LOW LSV
LOLO LLSV

Channel Access Client:
Connect to “Name”
Add Event to
- alarm change
- monitor change
- archive change

Make data type request
- Value
- Status & Severity
- Time Stamp
- Display and Control Information

Channel Access Client:
Connect to “AI<.VAL>”
- AI.VAL
- AI.STAT
- AI.SEVR
- AI.TS
- AI.HOPR
- AI.LOPR
- AI.EGU

Connect to AI.SCAN
- AI.SCAN
- AI.STAT
- AI.SEVR
- AI.TS
- # Choices
- List of choices

The Learning Curve for EPICS

- **Requires system administration skills**
  - Installing EPICS –
  - Setting up the application environment to automatically build databases
  - Setting up the IOC to boot from the workstation
- **Requires Programming Skills**
  - Installing the new drivers
  - Knowledge of how to debug the application - is needed by everyone
  - Implementing state transition logic
- **Some Instrumentation Knowledge**
  - Learning to use the process database to implement data acquisition and steady state control such as PID control or interlocks.
- **Must Have an Opinion**
  - Choosing and learning which client tools to use

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
  - 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

Ten really neat things about EPICS

(Getting Started with EPICS: Introductory Session I)

- It's free
- It's Open Source
- There are lots of users
- All a client needs to know to access data is a PV name
- You can pick the best tools out there …
- … or build your own
- The boring stuff is already done
- There is a lot of expertise available close by
- A good contribution becomes internationally known
- By following a few simple rules, you get a lot for free …

“Configuration instead of Programming“

EPICS Database

... in 1 hour?!

Kay Kasemir,
SNS/ORNL

PAC 09, Vancouver

IOC Database

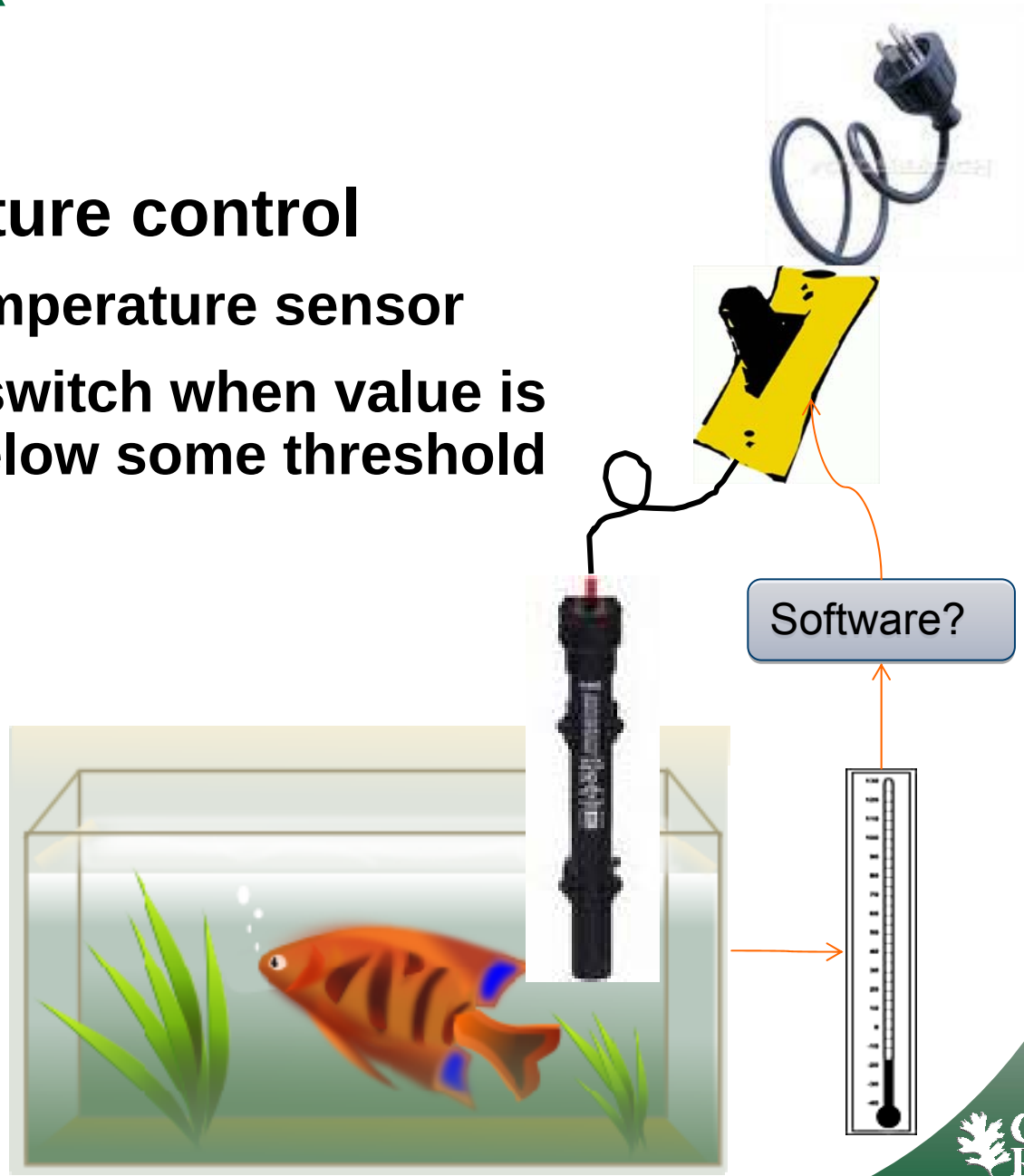
- **'iocCore' software loads and executes 'Records'**
  - **Configuration of records instead of custom Coding**
- **All control system toolboxes have (better?)**
  - **GUI tools**
  - **Network protocols**
  - **Hardware drivers**

  but few have a comparable database!

OAK RIDGE National Laboratory

Example Task

- **Basic temperature control**
  - **Read some temperature sensor**
  - **Open/close a switch when value is above resp. below some threshold**



OAK RIDGE National Laboratory

The Example in simplified Code

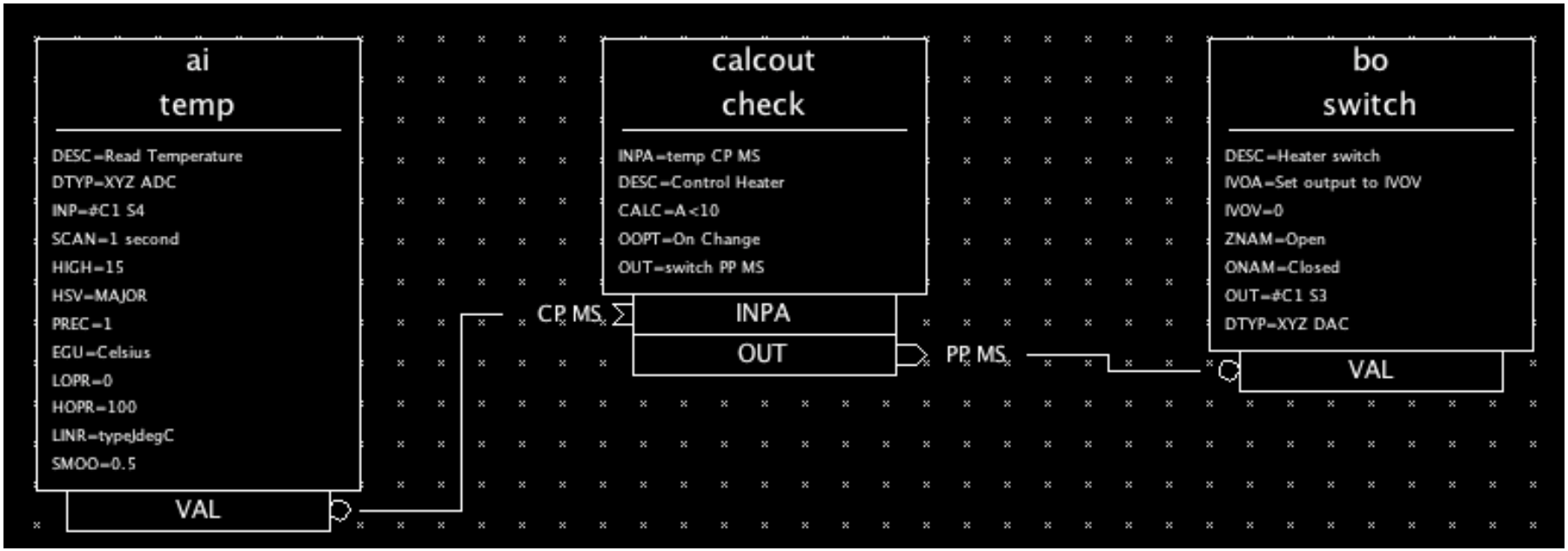
```
Sensor temp = open_device(…);

Switch switch = open_device(…);

Loop:

        if (temp.value() < 10)
            switch.close();
        else
            switch.open();

       delay(1.0);
```


OAK RIDGE National Laboratory

What we omitted

- **Error checking**
- **Code comments**
- **Apply some smoothing to the temperature reading to filter noise.**
- **Send current temperature and switch state to network clients (operator display).**
- **Attach a time stamp to the data, so that network clients can see for example when the switch was last opened.**
- **Send warnings if the temperature is close to the threshold, or an alarm when way above.**
- **Allow runtime changes of the threshold from the remote operator interface.**
- **Allow runtime changes to the scan rate.**
- **Maybe allow runtime changes to the device address?**
- **What if we have more than one fishtank?**

This IOC 'Database' does all that



- **At first glance, this might look much worse than the code, but…**
  - **that was simplified code.**
  - **there's no way the full code for the above would fit on one screen.**
  - **after learning more about the database (~2 days), this becomes much more readable than somebody else's custom code for the same functionality.**

Some Detail on EPICS 'Records'

```
record(ai, temp) {
  field(DESC, "Read Temperature")
  field(SCAN, "1 second")
  field(DTYP, "XYZ ADC")
  field(INP, "#C1 S4")
  field(PREC, "1")
  field(LINR, "typeJdegC")
  field(EGU, "Celsius")
  field(HOPR, "100")
  field(LOPR, "0")
  field(SMOO, "0.5")
  field(HIGH, "15")
  field(HSV, "MAJOR")
}


record(calcout, check) {
  field(DESC, "Control Heater")
  field(CALC, "A<10")
  field(INPA, "temp CP MS")
  field(OUT, "switch")
  field(OOPT, "On Change")
}


record(bo, switch) {
  field(DESC, "Heater switch")
  field(DTYP, "XYZ DAC")
  field(OUT, "#C1 S3")
  field(ZNAM, "Open")
  field(ONAM, "Closed")
  field(IVOA, "Set output to IVOV")
  field(IVOV, "0")
}
```

- **Configuration instead of Programming**
- **"SCAN=1 second" instead of starting periodic thread, delaying until next multiple of 1 second, locking required resources, …**
- **"SMOO=0.5" configures the smoothing algorithm.**
- **Almost any field in any record is accessible via network at runtime**
  - **Change scan rate, smoothing, …**

OAK RIDGE National Laboratory

IOC Database

- **A single analog record often handles the scanning, signal conditioning, alarming of a temperature, pressure, or similar analog reading.**
- **Combined with binary and computational records, it can express most of the data flow logic for a front-end computer**
  - **Avoiding the pitfalls of real-time, multithreaded and networked programming.**
- **One can have thousands of records in one IOC.**
  - **Well suited for systems with high signal counts, like vacuum or water systems with relatively simple logic but many, many sensors and valves.**
- **kHz-rate processing with record chains is doable**
  - **Of course limited by CPU. Not 1000nds of kHz rate-records…**

New Developments

Take Advantage of Changes in Technology and Improve Functionality, Ease of Use, and Performance While Reducing Cost

- Version 4 of EPICS that supports extended functionality in record support and network communication which has hampered the ability to significantly change the communication protocol and maintain a large record type base.
- Continue to Develop new database configuration tools (APS/Cosy Lab/Diamond/SNS)
- Develop embedded EPICS support (NI, ZTEch, ITech, Yokogawa)
- Support for IOC core and hardware drivers under new operating systems (All labs)
- New High Level Physics Application Platform and online modeling (ORNL/LANL/CosyLab/BNL)

Conclusions (EPICS)

EPICS Continues to Meet the Needs of Its Members Through Cooperative Development of a Scaleable, Flexible Tool-kit

- The fundamental performance and functionality is scaleable and easily configured.
- Clean interfaces have been used to extend EPICS on the client side for varied operational tools, on the driver and device support side to integrate a wide variety of I/O and legacy systems.
- Open software development supports cooperative collaboration and gives members laboratories a larger pool of talent to support their controls requirements.
- Continual improvements allow members to expand functionality, performance, reliability and function while taking advantage of latest technology.

Part II

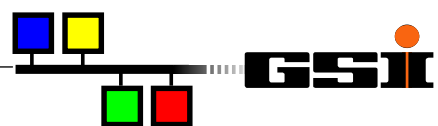
EPICS @ GSI

EPICS @ GSI

- Base Installation
  - “*. epics*“
- Interfacing
  - Technical and Social
- Embedded Target Platforms
  - ETRAX

http://wiki.gsi.de/Epics

epics@gsi.de



Playground

```
lxi009: /misc/zumbruch
> . epics -h
epics [Options] [version]
-- sets environment variables for EPICS
--
-- changes to corresponding $EPICS_HOME directory and calls login script for version
--
-- version: EPICS version
--          (default: "3.14.9")
-- options:
----- -h : prints this Help
----- -l : list all available versions
----- -s : stay at $EPICS_HOME directory
----- -q : quiet no output, but error messages
----- -v : verbose mode
----- -H argument : home directory of EPICS versions
-----                (default: "/u/epics")
----- -P argument : version prefix for PREFIXversion
-----                (default: "EPICS-")
-----
----- version: version of EPICS
-----
-- you have to call this script with a leading '. ', i.e. source it
--> i.e.: '. epics'
lxi009: /misc/zumbruch
```

Playground

```
lxi009: /misc/zumbruch
> . epics 3.14.10
sourcing `./myepicslogin.sh' seen from directory `/u/epics/EPICS-3.14.10'
MYEPICS_VERSION          set to 3.14.10
EPICS_VERSION            set to 3.14.10
sourcing /u/epics/EPICS-3.14.10/base-3.14.10/startup/Site.profile
EPICS_HOME               set to /u/epics/EPICS-3.14.10
EPICS_BASE               set to /u/epics/EPICS-3.14.10/base-3.14.10
EPICS_EXTENSIONS         set to /u/epics/EPICS-3.14.10/extensions
EPICS_MODULES            set to /u/epics/EPICS-3.14.10/modules
PSPRINTER                set to lp
EPICS_DISPLAY_PATH       set to /u/epics/EPICS-3.14.10/extensions/medm/adl
ORBITSCREENHOME          set to /u/epics/EPICS-3.14.10/extensions
EPICS_AR_PORT            set to 7002
EPICS_SNCSEQ_HOME        set to /u/epics/EPICS-3.14.10/modules/seq
EPICS_HOST_ARCH          set to linux-x86
added /u/epics/EPICS-3.14.10/extensions/bin/linux-x86 in front of PATH
added /u/epics/EPICS-3.14.10/base-3.14.10/bin/linux-x86 in front of PATH
added /u/epics/bin in front of PATH
added /u/epics/programs/bin in front of PATH
EPICS_CA_ADDR_LIST       set to lxi009
EPICS_CA_AUTO_ADDR_LIST  set to YES
--> Created for each variable EPICS_* a copy EPICS_*_COPY
    to be used in CONFIGURE scripts where EPICS_ variables might be overwritten
lxi009: /misc/zumbruch
```

Interface

- Socially (Mentoring / Networking)
  - "Since controls often isn't seen as important as it later comes out not much (man)power is invested into it."
  - Therefore as EPICS@GSI is known/active in several FAIR collaborations, we try to bring control people together to share ideas and work. Or at least learn from each other.
  - Call it Synergy, Networking, Mentoring, Interfacing, ...

Experiments @ FAIR

Experiment	Scientific Area	Research Program	Technical Facility	Members	Institutes
R3B	NUSTAR	Nuclear reactions in inverse kinematics reaction studies with relativistic radioactive ion beams	Large reaction set-up allowing complete kinematics reaction experiments	178	54
HIS DE					
LA					
MA					
ILI					
EX					
AI					
EL					
NC		site) neutron capture studies			
EXO-pbar	NUSTAR	Measurements of proton-neutron abundance at the nuclear surface of nuclei far off stability	Reaction experiment of very low-energy radioactive ions with antiprotons stored in a Penning trap	20	5
PANDA	QCD	QCD and hadron physics studies with cooled high energy antiproton beams at the HESR	Large state-of-the-art internal target detector system covering almost the full solid angle	344	70
CBM	QCD	Studies of the QCD phase diagram in high-energy nucleus-nucleus collisions	Large state-of-the-art fixed target detector system covering almost the full solid angle	357	63
PAX / ASSIA	QCD	QCD and hadron physics studies with polarized antiproton beams	State-of-the-art collider detector system covering a large solid angle	170/ 85	33 / 12
HEDge-HOB/ WDM	APPA	Investigations of warm and dense bulk matter produced by intense ion and/or laser pulses	Various plasma physics experimental stations	162 / 55	50 / 19
FLAIR	APPA	(Precision) studies with low energy or stopped antiproton ion beams	Various stations including an ultra-low energy electrostatic storage ring, a Penning trap, low energy antiproton target stations	142	54
SPARC	APPA	Atomic physics spectroscopy and collision studies with (stored) high energy ion beams	Various fixed target and in-ring experiments	218	108
BIOMAT	APPA	Applications of ion and antiproton beams in biophysics, biology, materials research and other disciplines	Various multi-purpose target stations	49	28

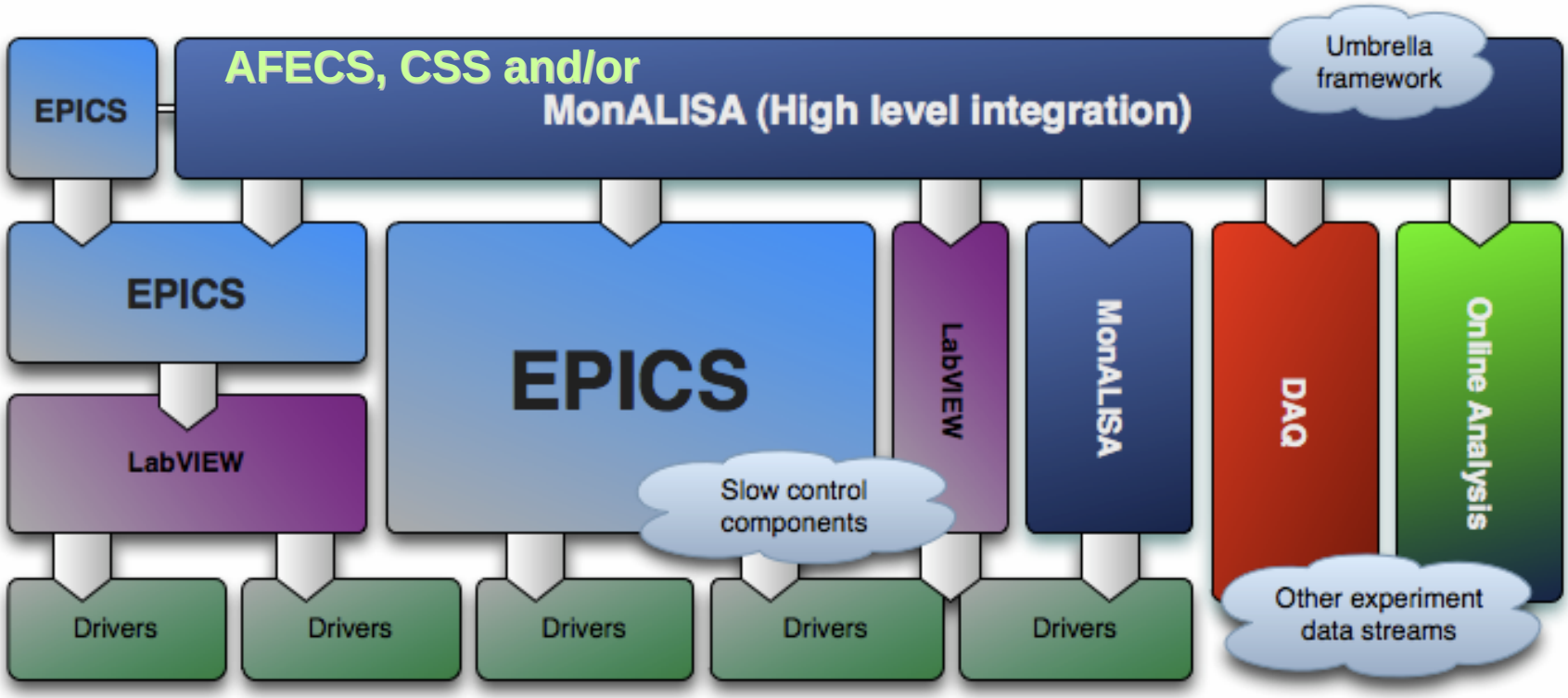
Experiments @ FAIR

Collaborations showing interest in EPICS (to my knowledge)

- NuSTAR
- PANDA
- CBM

Experiment	Scientific Area	Research Program	Technical Facility	Members	Institutes
R3B	NUSTAR	Nuclear reactions in inverse kinematics reaction studies with relativistic radioactive ion beams	Large reaction set-up allowing complete kinematics reaction experiments	178	54
PANDA	QCD	QCD and hadron physics studies with cooled high energy antiproton beams at the HESR	Large state-of-the-art internal target detector system covering almost the full solid angle	344	70
CBM	QCD	Studies of the QCD phase diagram in high-energy nucleus-nucleus collisions	Large state-of-the-art fixed target detector system covering almost the full solid angle	357	63
PAX / ASSIA	QCD	QCD and hadron physics studies with polarized antiproton beams	State-of-the-art collider detector system covering a large solid angle	170/ 85	33 / 12
HEDge-HOM/ WDM	APPA	Investigations of warm and dense bulk matter produced by intense ion and/or laser pulses	Various plasma physics experimental stations	162 / 55	50 / 19
FLAIR	APPA	(Precision) studies with low energy or stopped antiproton ion beams	Various stations including an ultra-low energy electrostatic storage ring, a Penning trap, low energy antiproton target stations	142	54
SPARC	APPA	Atomic physics spectroscopy and collision studies with (stored) high energy ion beams	Various fixed target and in-ring experiments	218	108
BIOMAT	APPA	Applications of ion and antiproton beams in biophysics, biology, materials research and other disciplines	Various multi-purpose target stations	49	28
		site) neutron capture studies			
EXO-pbar	NUSTAR	Measurements of proton-neutron abundance at the nuclear surface of nuclei far off stability	Reaction experiment of very low-energy radioactive ions with antiprotons stored in a Penning trap	20	5

q.v. DCS (PANDA)



AFECS for FAIR Workshop: http://www.doodle.com/participation.html?pollId=ef8dwz8weba4wqma

Technical interfacing of EPICS

Idea: "*Let EPICS talk to other systems*"

- Many Interfaces already available for EPICS,
  but DIM was missing

→ EPICS – DIM Interface

What is DIM?

"DIM is a communication system for distributed / mixed environments. It provides a network transparent inter-process communication layer."

- **Protocol**
- Distributed Information Management System
- Originally built for DELPHI
- http://dim.web.cern.ch/dim/

Some Properties:

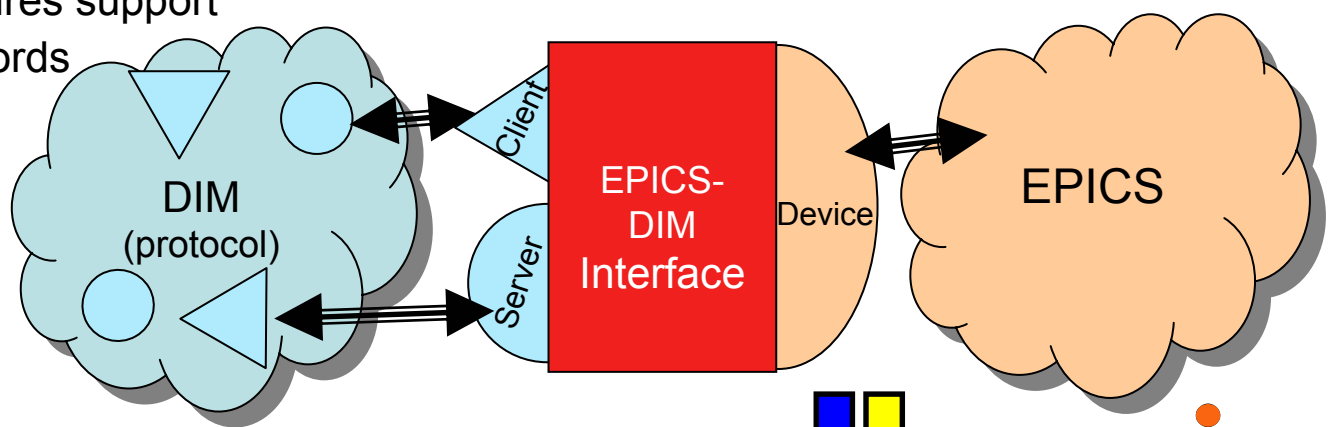
- Small / Tiny
- Many platforms
- No (not yet) inherent access security
- No logic
- Dynamic

- name based publisher/subscriber mechanism for services and commands

Used as network protocol for *CS Control system (GSI) and DABC Gateway to LabVIEW*

EPICS DIM Interface

- Implementation as "*device support module*"
- Running
  - DIM SERVER
    - Providing read/write access to EPICS variables
  - DIM CLIENT
    - Interfacing DIM services and commands for single variables to EPICS process variables
  - Successfully used 5 weeks continous HADES beam time
- On demand
  - String transport mode (DIM provides strings converted by the Interface to single data types, easier to handle by EPICS)
  - Array and structures support
  - More EPICS records



DAL DIM Interface

Outlook: (Martin Feldmann, GSI)

DAL – Data Access Layer

- Data Abstraction Framework to abstract connections to several control systems (TINE, TANGO, EPICS, ...) in order to access them all the same way.
- Used in CSS Control System Studio (css.desy.de)
- Plan: DAL DIM Interface

connecting to LabVIEW

http://wiki.gsi.de/Epics/ConnectingLabVIEWandEPICS

LabVIEW DIM Interface ⇔ EPICS - DIM Interface

LabVIEW Data Logging and Supervisory Control Module, NI

LabVIEW Shared Memory Interface to EPICS IOC by SNS

LabVIEW ActiveX CA by Kay Uwe Kasimir, ORNL

EPICS embedded

- Project:

  EPICS running on embedded systems

- Aiming at this architecture
  - ETRAX based CRIS architecture
    - HADControl (aka: **HAD**ES **SHO**WER **PO**WER **MO**NITOR)
    - TRBv2
    - HADES, CBM, Panda

Platform: ETRAX 100LX MCM / FS by AXIS

For HADES GSI's Experimental Digital Electronic group (M.Traxler) has developed:

- HADControl (HADSHOPOMO (HADES SHOWER POWER MONITOR))

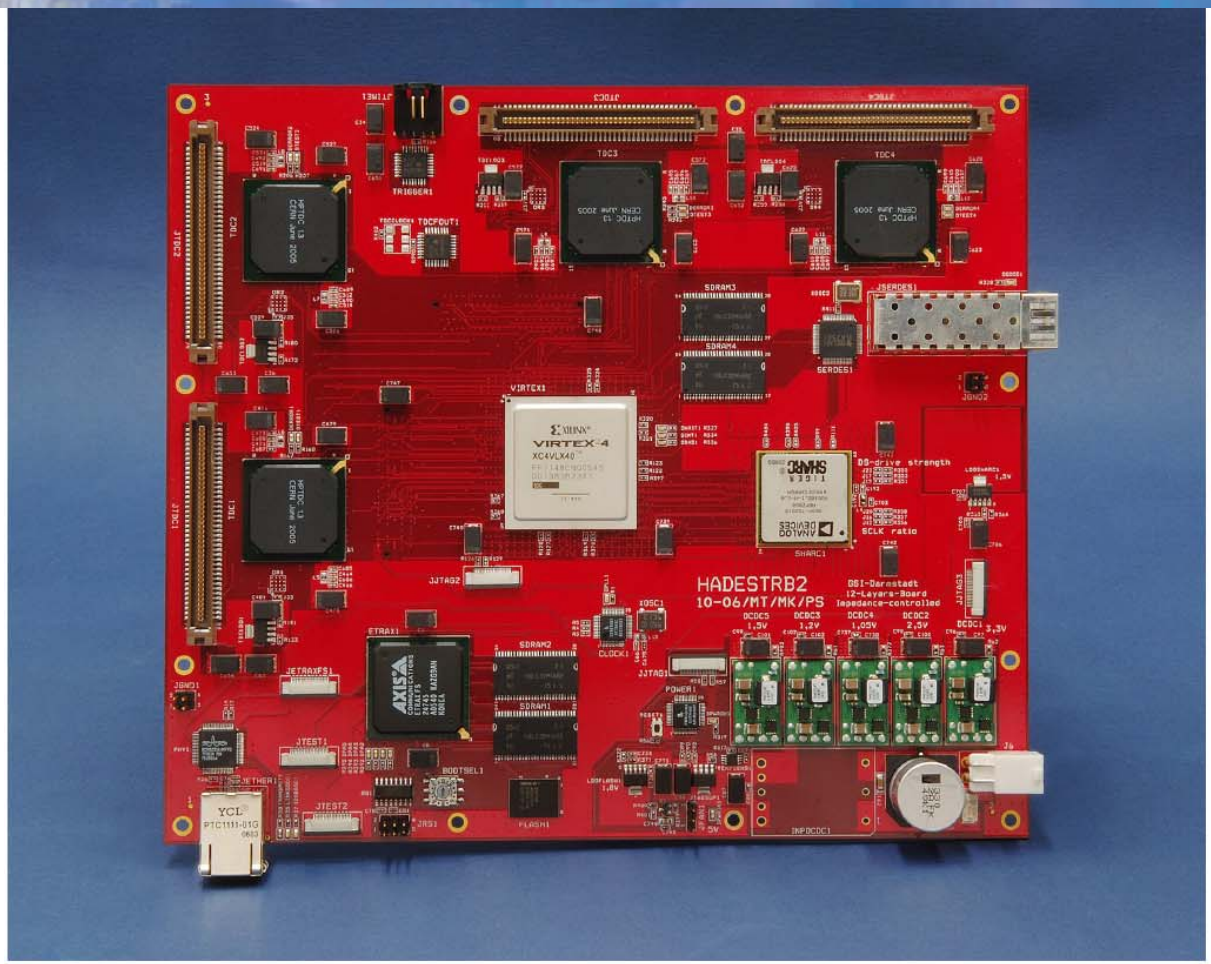
"Multi-purpose control/monitor device developed for HADES […] is based on the ETRAX 100LX MCM4+16 and runs the "Experimental Physics and Industrial Control System, EPICS".

http://developer.axis.com/showroom

  - some implementations:
    - single wire bus temperature measurement for HADES
    - Monitor system for Driftchamber pressure

- HADES TRB - Trigger Board, DAQ and Slow Control (ETRAX)

TRBv2



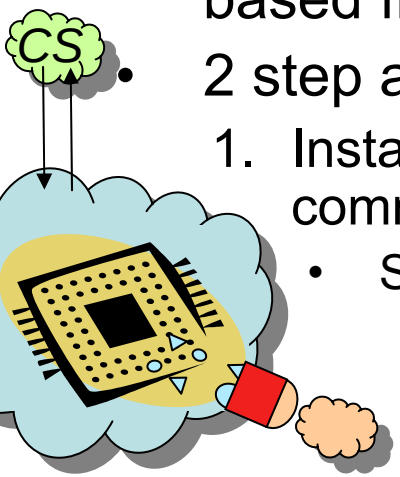
- successor of TRBv1, which is used in the experiment
- larger FPGA
- faster CPU (x3)
- Tiger-Sharc DSP
- 2 GBit/s optical link for trigger and data
- Add-on connector
- TRBv1 functionality given

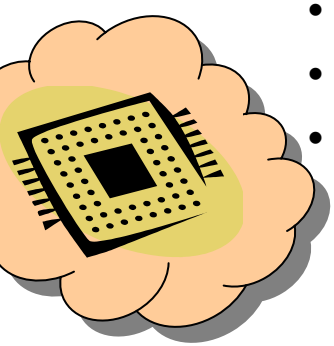
2007-06-04 Michael Traxler, GSI

http://www-linux.gsi.de/~traxler/GSIScientificReport2006_TRB/TRBv2_2006.pdf

Embedded EPICS on ETRAX

- install embedded Linux on ETRAX chip CPU (axis.com) based front-end systems
- 2 step approach:
  1. Install DIM on ETRAX and use EPICS-DIM Interface to communicate via network with external EPICS clients or IOCs
     - Suitable for development:
       - DIM protocol also accessible via other controls software, i.e. LabVIEW, or *CS,* etc.
       - But locally no (EPICS) logic (database, (fast) sequencing, alarming) provided
  2. Install EPICS Embedded on ETRAX
     - Provides all features of EPICS
     - Local fast EPICS based logic, network independent
     - By „turning the direction of the interface“ users may still see a DIM device, mimicked by EPICS using the EPICS – DIM interface

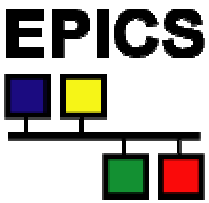




ETRAX based applications (GSI)

- Temperature Sensor Readout (HADES) based on Single wire bus
  - M.Feldmann, PZ, G.Ortona, R.Trebacz
- Threshold control (RPC detectors) based on SPI protocol
  - A.Gil, R.Trebacz, PZ
- CANbus controller
  - L.Fouedjio, R.Trebacz, PZ

Summary Outlook



EPICS is
…an grown-up, mature, portable control system architecture,
…a world wide active and very responsive collaboration and
…a rich collection of Open Source code and documentation

which allows to build up a large scale,
scalable control system.

EPICS at GSI

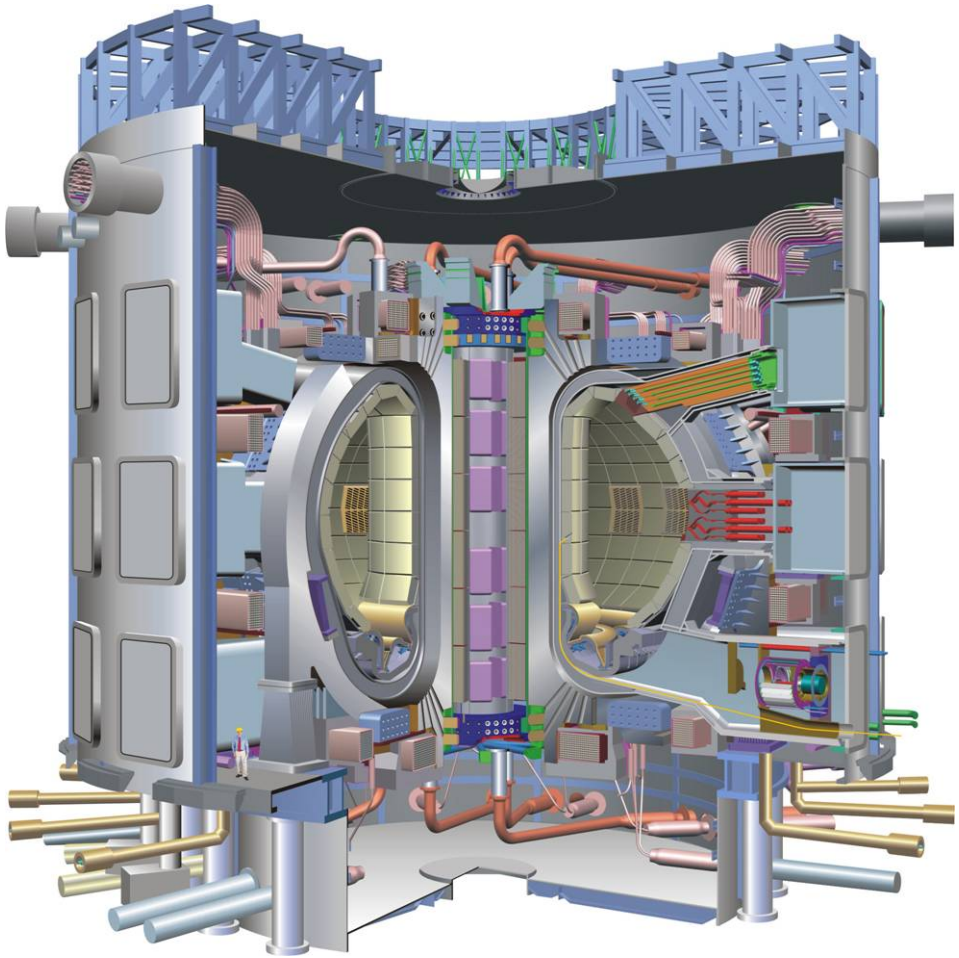
… can provide knowledge and information to build up FAIR experiment control systems or parts of it

… offers multi purpose tools which may be used

… projects/activities:

- Interfacing/Networking
- EPICS embedded (cris architecture (ETRAX), Xilinx FPGA)
- In future/today: CSS (Control System Suite):
  - IDE based on Eclipse

latest NEWS: ITER will use EPICS



To: "tech-talk@aps.anl.gov" <tech-talk@aps.anl.gov>
Subject: ITER will use EPICS
From: Di Maio Franck <Franck.DiMaio@iter.org>
Date: Tue, 10 Feb 2009 08:35:52 +0100

Dear all

We would like to share with you the following decision taken by the group in charge of the controls for ITER (www.iter.org).

The CODAC group initiated a number of actions in the first half of 2008 in order to select a software environment as a part of moving from the conceptual design to an engineering design of CODAC.

The conclusion of all these activities is that ITER, **being an experimental facility with a very long timeline, is better suited using an open source solution as compared to a commercial solution.** Further, the reports conclude that technically, any of the candidate open-source solutions would work. However, due to market share and proven record the preferred solution would be EPICS.

As a consequence, the CODAC group announces that EPICS will be used as the baseline for the software environment for the ITER control system within the scope of PCDH (Plant Control Design Handbook).

Anders Wallander, 01-Feb-2009

Notes:

- CODAC means COntrol, Data Access & Communications.
- The Plant Control Design Handbook (PCDH) is a contractual document that specifies the ITER standards for the instrumentation and control of the ITER plant systems.

ITER parties are: China, Europe, India, Japan, Korea, Russia and USA.

So, it means new users (& hopefully contributors) from these regions.

Best regards,

Franck

Franck DI MAIO
ITER Organization
CHD Department / CODAC & IT Division

http://wiki.gsi.de/Epics
epics@gsi.de

Thank you for your attention.

For more information …

Have a look at the extra slides.

Extra Slides

What is EPICS?

- Process Variable
  - A ***Process Variable*** is a named piece of data with a set of attributes
  - Examples of Attributes:
    - 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)

Vocabulary

(Getting Started with EPICS: Introductory Session I)

- 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

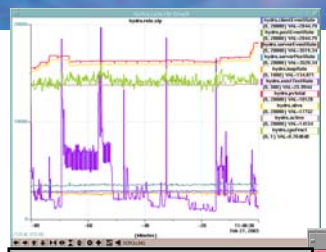
Vocabulary

(Getting Started with EPICS: Introductory Session I)

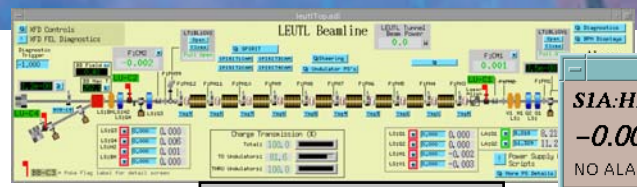
- IOC – Input Output Controller
  - A computer running *iocCore,* a set of EPICS routines used to define process variables and implement real-time control algorithms
  - *iocCore* uses database records to define process variables and their behavior
- Soft IOC
  - An instance of *iocCore* running as a 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

How does it do it?

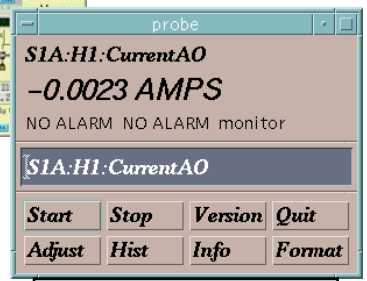
(Getting Started with EPICS: Introductory Session I)



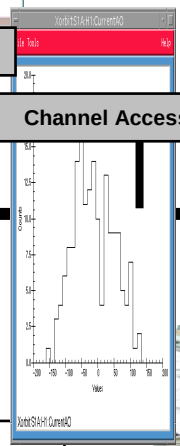
Channel Access Client



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?

(Getting Started with EPICS: Introductory Session I)

Remote Control & Monitoring

Alarm Logging/Reporting

Closed-loop Control

Operational Constraints

Automatic Sequencing

Data Trending/Archiving

Modeling/ Simulation/Analysis

Configuration Control

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

Data Conversion/Filtering

Alarm Detection

Access Security

Closed-loop Control

Computer Interface

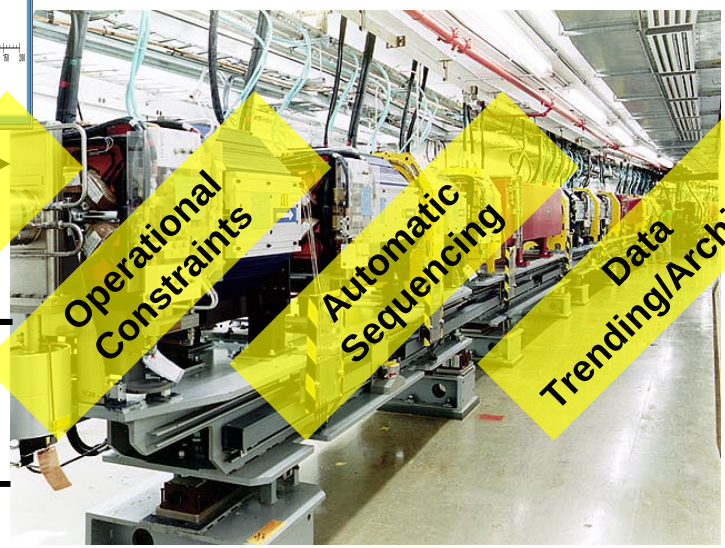
Computer Interface

Computer Interface

Power Supply

Beam Position Monitor

Vacuum Gauge



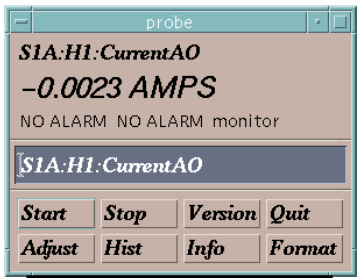
Operational Constraints

Automatic Sequencing

Data Trending/Archiving

Channel Access in One Slide

(Getting Started with EPICS: Introductory Session I)



Channel Access Client

CA Client

CA Server

Channel Access Server

Process Variables:

S1A:H1:CurrentAO

S1:P1:x

S1:P1:y

S1:G1:vacuum

"connection request" or "search request"

"get" or "caGet"

"put" or "caPut"

"set a monitor"

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

I do.

What is its value?

25.5 AMPS

Change its value to 30.5

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

Notify me when the value changes

It is now 20.5 AMPS

It is now 10.5 AMPS

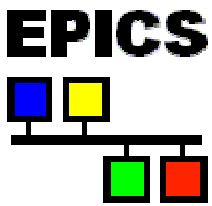
It is now -0.0023 AMPS

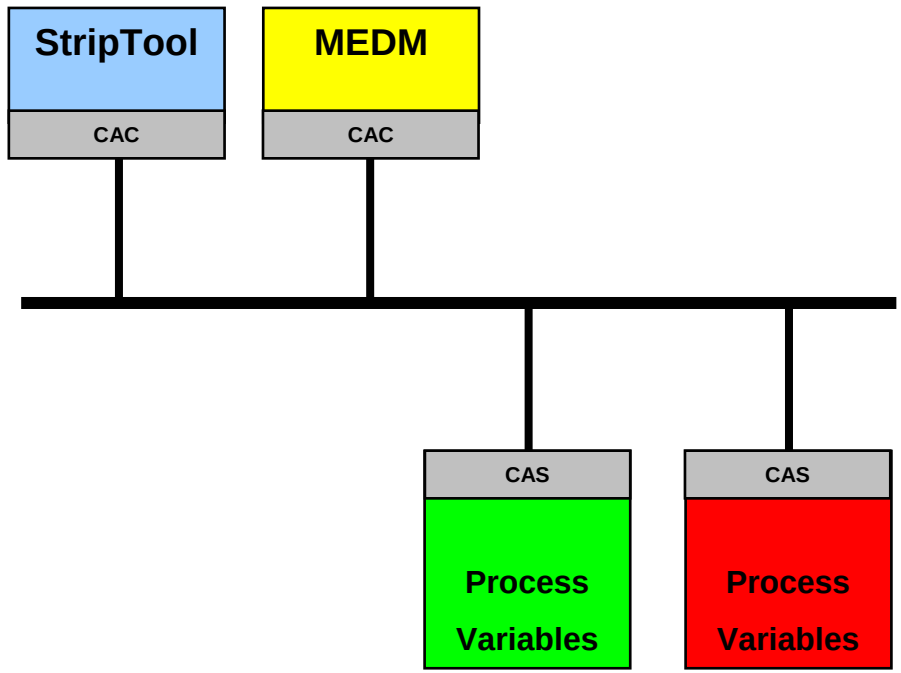
"post an event" or "post a monitor"

What is EPICS?

(Getting Started with EPICS: Introductory Session I)

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****

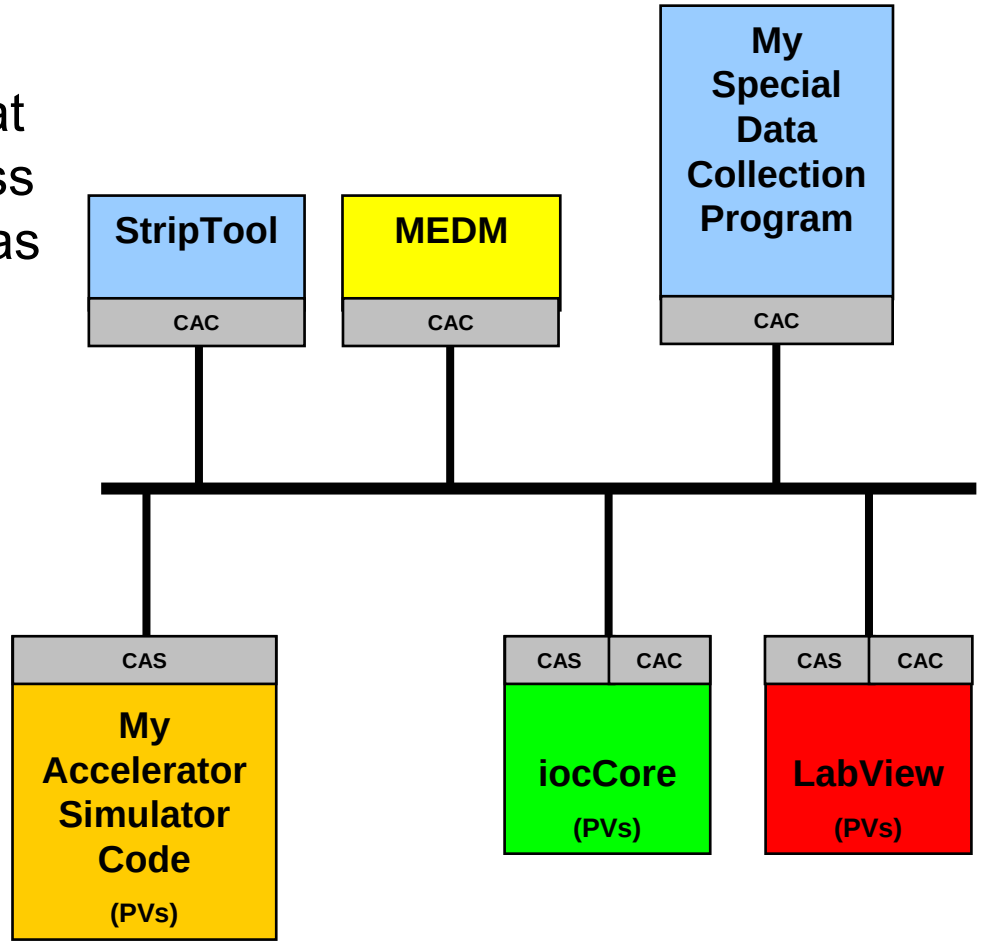
*** A** *Process Variable* **(PV) is a named piece of data.**

What is EPICS?

(Getting Started with EPICS: Introductory Session I)

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.



Introducing the IOC

Input Output Controller

A computer running software called "*IOC Core*"

The computer can be:

- VME based, running vxWorks (only choice until Release 3.14) or RTEMS
- PC running Windows, Linux, RTEMS
- Apple Mac running OSX
- UNIX Workstation running Solaris, OSF

Usually has Input and/or Output devices attached

An EPICS control system must contain at least one Channel Access Server (usually an IOC)

An IOC loads one or more *databases,* which tell it what to do

- What Process Variables to serve
- What I/O devices to connect to
- What other Pvs to connect to
- What processing to do