Experimental Physics and Industrial Control System (EPICS) Overview

Bob Dalesio, May 4, 1998

Outline

- Introduction: What is EPICS
- The Collaboration
- Software / Hardware Architecture
- The Client/Server Model over the Channel Access Protocol
- A Look at the Principal Data Store - Process Database
- A Look at Some Channel Access Clients
    - EDD/DM
    - Alarm Manager
    - State Notation Language
- Conclusions

Introduction: What is EPICS?

- 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

The EPICS Collaboration

- Over 90 independent projects in North America, Europe 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
- 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, CDEV, Archiving and MMI

Accelerator Programs Using EPICS

- The Advanced Photon Source and Beamlines @ ANL
- The Thomas Jefferson National Accelerator Facility @ Newport News, Va.
- Proton Storage Ring @ LANSCE
- Injection Linac for KEKB Factory @ KEK
- Advanced Free Electron Laser @ LANL
- Duke Free Electron Laser and Mark III Laser @ Duke University
- Heavy Ion Fusion Test Stand @ LBL
- Intense Pulsed Neutron Source @ ANL
- HERA Cryogenic Plant and Tesla Test Facility @ DESY
- RF and Beamline Control for the B-Factory @ SLAC
- Bates Linear Accelerator @ Bates MIT
- Racetrack Microtron @ University of Athens
- BESSY II @ BESSY

Astronomy Programs Using EPICS

- Gemini 8 Meter Telescopes @ NOAO
- Kitt Peak Observatory @ NOAO
- WFFOS Spectrograph @ William Herschel Telescope @ La Palma
- United Kingdom Infrared Telescope @ JAC
- LIGO @ Cal Tech
- GWFD @ Japanese National Large Telescope
- Keck II @ CARA

Detector Programs Using EPICS

- PHENIX Detector @ RHIC
- STAR Detector @ RHIC
- Detector Halls A, B & C @ TJNAF
- Gammasphere @ LBL
- Babar Detector @ PEP II
- Under investigation for D0

Industrial Use of EPICS

- GM Fuel Cell Test Stand @ LANL
- Agile Manufacturing Plant @ Allied Signal
- Plant Simulation @ Knolls Atomic Power Laboratory
- Flight Simulation @ Jet Propulsion Laboratory
- Liquefied Natural Gas Plant @ BG & E
- Western Lake Superior Waste Water Treatment Plant
- St.. Louis County - Potable Water Distribution Network
- Bechtel-Parsons-Brinkerhoff's Fire Ventilation Test Facility
- The Advanced Micro Devices Wafer Fabrication - Clean Room
- Citgo Oil Product Storage and Movement Facility
- Saphania Oil Field Well Head and Extraction
- Las Vegas - Potable Water Distribution Network

Data Acquisition Commercial Applications

- Pratt & Whitney Engine Test Facility
- Lockheed Martin
- NASA Langley Wind Tunnel
- United Technologies Engine Test Facility
- McDonnell Douglas Component Test Facility

Project Status Throughout the Collaboration

• APS	300,000	170 IOC	20 ws	reliable operation*
• HERA Cryo	6,000	4 IOCs	1 ws	reliable operation
• Tesla Test Inj	600	4 IOCs	2 ws	reliable operation
• KeckII	1,500	2 IOCs	2 ws	reliable operation
• PSR	2,500	4 IOCs	6 ws	reliable operation
• Duke FEL	2,500	6 IOCs	3 ws	reliable operation
• PEP II RF	8,400	8 IOCs	2 ws	reliable operation
• BESSY	15,000	15 IOCs	4 ws	reliable operation
• 1APS Beamline	15,000	22 IOCs	10 ws	reliable operation
• Jlab	180,000	81 IOCs	30 ws	Mostly reliable*

* Jlab has software problems under 1% of downtime - mostly at restart. Hardware problems are still over budget (2%) with power supplies and Bira Crate Controllers.

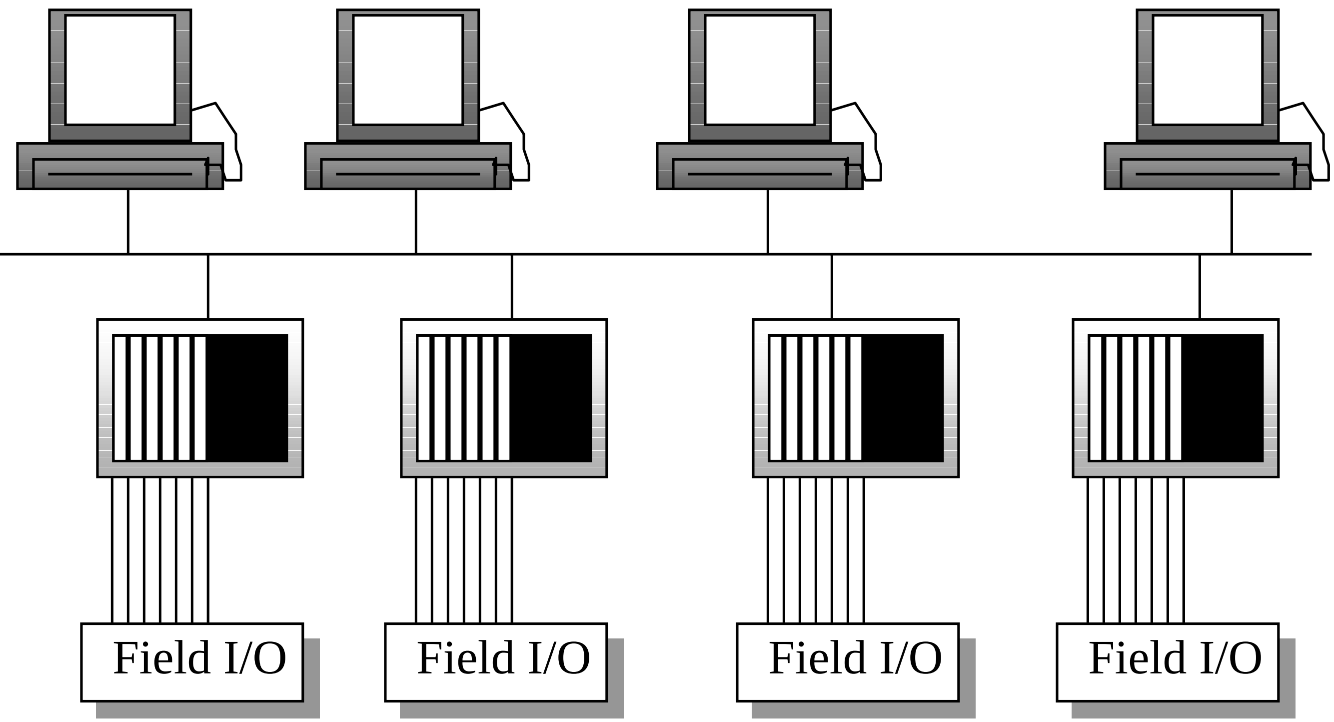
*APS has one IOC loss every 2 weeks - tracking down. Under 1% down time.

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

Distributed Hardware Configuration

Workstations:
- Sun
- Hp
- DEC/Alpha
- Silicon Graphics
- PC



I/O Controllers
- VME/VXI/PCI

Remote I/O and Signal Conditioning
- CAN-Bus, Industry Pack
- VME, VXI, PCI, ISA
- CAMAC, GPIB
- Profibus, Bitbus, Serial
- Allen-Bradley, Modbus

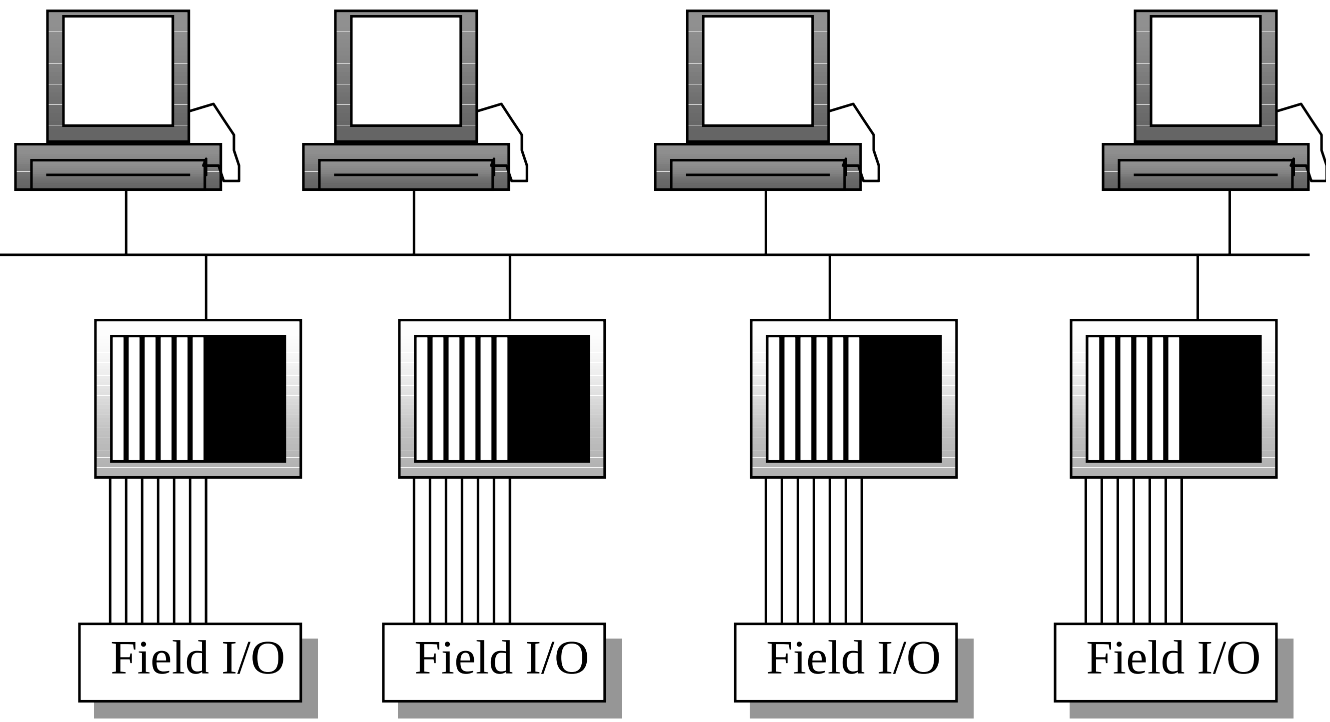
Standard Operating System Software Configurations

Workstations OS:

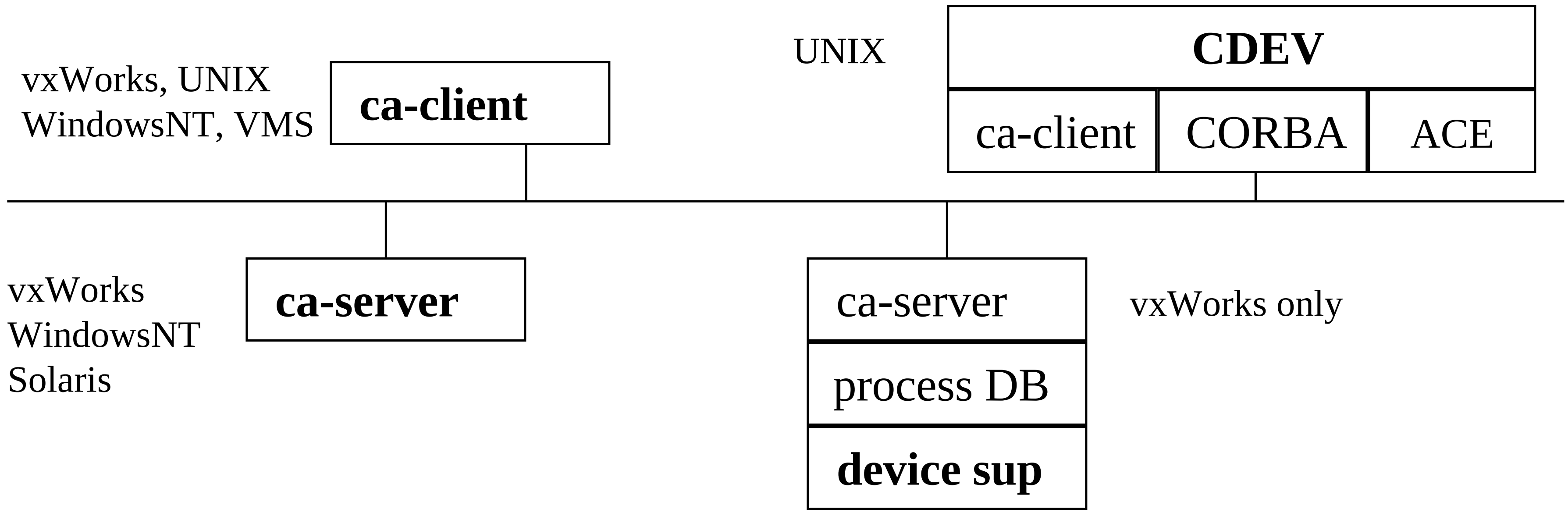
- SunOS, Solaris
- HPUX
- DEC-UNIX
- SGIX
- Windows NT*

I/O Controllers

- vxWorks
- LynxOS*



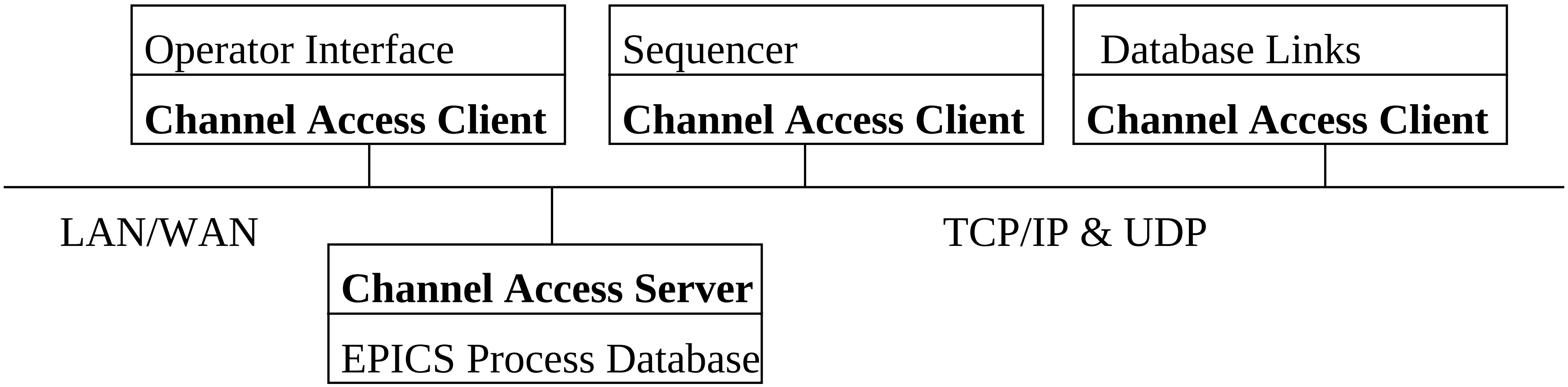
Distributed Software Architecture



Many tools are available in the EPICS tool-kit

- EPICS tools are connected via the Channel Access client/server libraries
- Server Interfaces:
  - Process Database
  - Gateway (CA-Client - GDD Library - Portable server on Solaris)
- Client Interfaces
  - Process Database Links
  - Sequential Control Language
  - Data Visualization Packages
  - Data Analysis Packages
  - Modeling and Automation Packages

Channel Access Client/Server Libraries



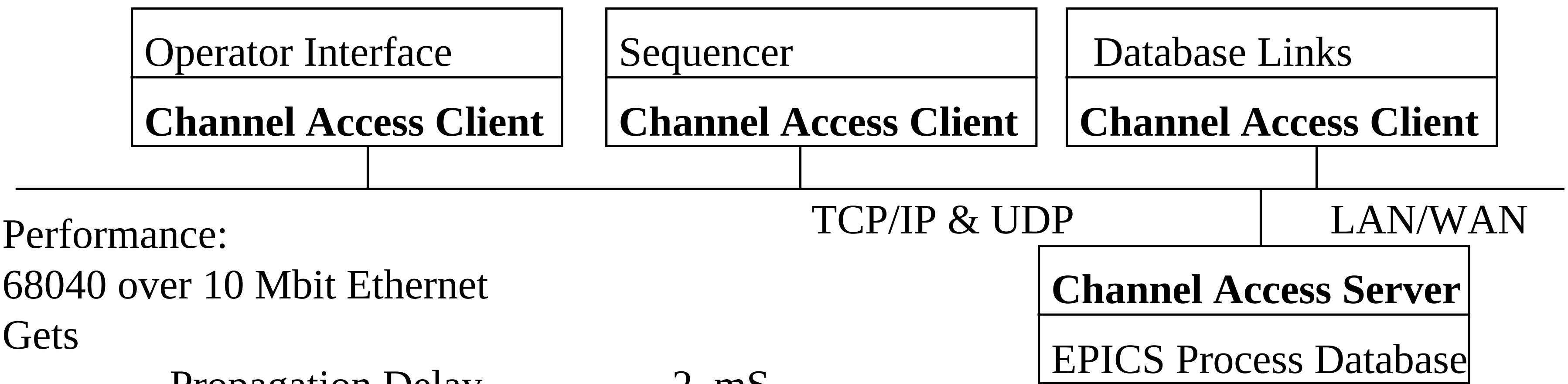
Client: Provides read/write connections to any subsystem on the network with a channel access server

Server: Provides read/write connections to information in this node to any client on the network through channel access client calls

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

Channel Access



Performance:

68040 over 10 Mbit Ethernet

Gets		
	Propagation Delay	2 mS
	Throughput	7.8K /sec
Puts		
	Propagation Delay	1 mS
	Throughput	17K /sec
Monitors		
	Propagation Delay	Dependent
	Throughput	10K / sec

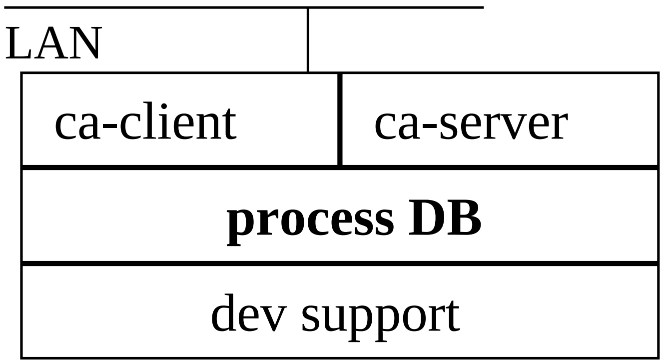
(Typically 10% channels have monitors)

(memory use in IOC - 2 Meg / 60 connections)

(30% network load for 10K monitors / second)

Increase bandwidth with Routers, Bridges, Higher speed networks and EPICS gateway

EPICS Process Database Provides Data Acquisition and Control



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.…

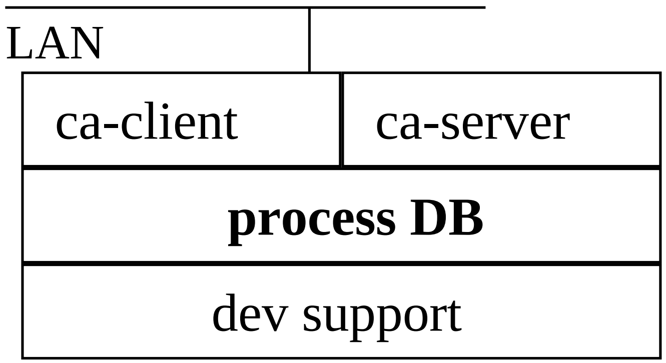
New process block are easily added

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

Loaded as ASCII records into vxWorks at boot time

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

EPICS Process Database



Process Block execution time varies from block type to block type

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 - 60 Hz

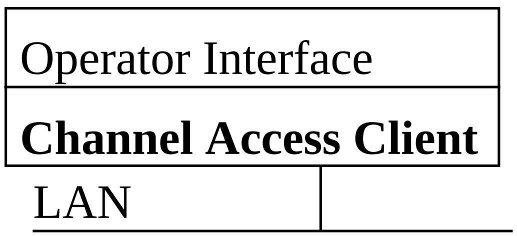
Interrupt scanning is limited by the CPU bandwidth (interrupt delay ~33usec)

Name resolution - 10,000/second - 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

Channel Access Client Codes Provide a Variety of Functionality



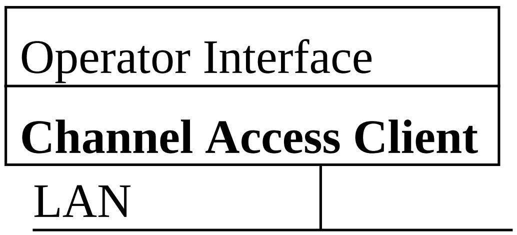
Visualization: **EDD/DM**, MEDM, Dataviews, SL-GMS, tcl/TK, Visual Basic, Visual C++, Labview, IDL, SAMMI JAVA

Alarm Manager: **ALM**

State Programming: **SNL**

Other Clients: PV-Wave, Mathmatica, Matlab, Probe, Orbit Lock, Correlation Plots, Striptool, X-Orbit, SAD, CNLS-net SDDS, ARTEMIS, Archivers, Save/Restore

EDD/DM Provides an Interactive Display Editor and High Performance Display Manager



The Channel Access Client provides name resolution, monitors and puts to any data stores with a CA server

EDD is an interactive display editor for creating operator screens

DM is a display manager to activate screens created using EDD

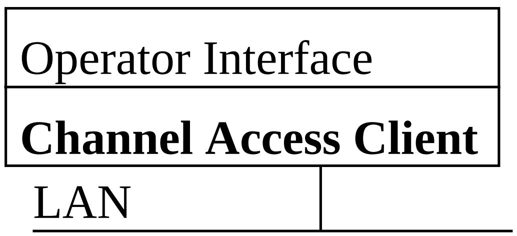
Connections are made to channels - typically *process-block<.field-name>*

DM monitors channels and only updates the screen when there are changes

Monitors from channel access are cached - not queued

ASCII display formats can be read/written for version upgrades/text editing

EDD/DM Provides an Interactive Display Editor and High Performance Display Manager



The Channel Access Client provides name resolution, monitors and puts to any data stores with a CA server

Static objects include: text, line, box, oval

Monitor objects include: text, bars, meters, indicators

Control objects include: text, slider, menu, buttons

Plot objects include: strip chart, Cartesian plots, smith charts

Related display call-up includes parameter passing

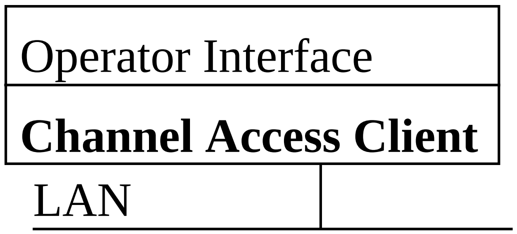
Color Rules are available for all objects independent of dynamic channel

Visibility modifiers are available for static objects

DM does 2,000 updates per second on a SPARC IPC

2 second display call-up for 100 dynamic and 1000 static elements

The Alarm Manager Provides Alarm Viewing, Logging and Annunciation



The Channel Access Client provides name resolution, monitors alarm changes, writes group alarm status

Alarms are detected in the data stores - most commonly the process database

Alarms are configured into an alarm hierarchy

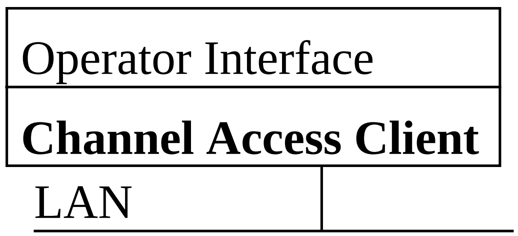
Logging, annunciation, acknowledgment and collection is configured for each group

All lower levels of the hierarchy inherit the disable status from above

Operator logging tracks alarm disable, alarm acknowledge and silence commands

500 Alarm condition changes per second can be handled on a SPARC IPC

The State Notation Language Provides a Mechanism for State Machine Implementation in EPICS



The Channel Access Client provides name resolution, monitors and writes to state programs

SNL is preprocessed to be converted to C code

Constructs are provided for easy channel access interface

Construct are provided to define state programs, state sets and transitions

SNL implements the Mealy Model - actions are only taken on transition

SNL runs in the vxWorks environment

Events are supported for synchronizing state sets within a state program

SNL does not have a server

The process database is used for variables to be exported

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

- Support large user facilities with real-time data to hundreds of workstations through the development of a gateway (ANL/LANL)
- Provide the ability to add/delete records in the process database during operation to support continuous operations (KEK/LANL/ANL)
- Support a single computer solution for small experiments and small portions of larger installations (DESY/LANL)
- Provide native windows NT capabilities to lower cost and provide data to every desktop (LANL)
- Create a modern JAVA based MMI attempting to maintain the performance while increasing the functionality and ease of use (LANL L-12)
- Provide a distributed archiving/archive retrieval capability to better support analysis and operations (TJNAF/LANL/DESY)

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 for clients, new record types, data stores and hardware promote independent development, support ease of reintegration, and protect against obsolescence
- Cooperative collaboration gives members laboratories a larger pool of talent to support their controls
- Continual improvements allow members to expand functionality, performance, reliability and function while taking advantage of latest technology