



# Introduction to EPICS

For J-K Joint project  
June 20,2002  
by N. Yamamoto  
KEK, JAPAN

# EPICSとは何か？

## EPICS

Experimental Physics and Industrial Control System.

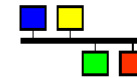
- ▶ ネットワーク分散型の制御システム構築のためのツールキット
- ▶ EPICSソフトウェアを共有する国際共同研究

# ネットワーク分散型制御システム

US  
Particle  
Accelerator  
School

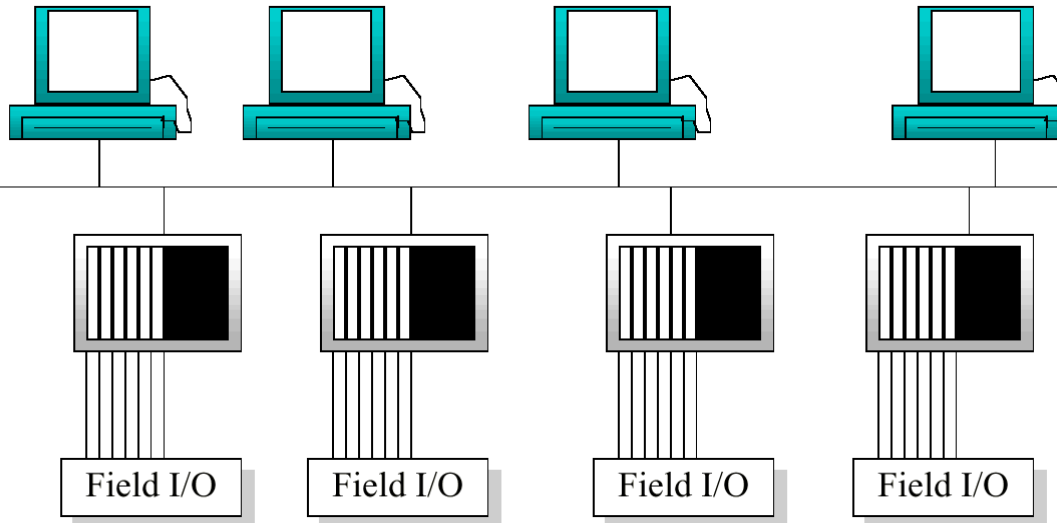
## Distributed Hardware Configuration

EPICS



Workstations:

Sun  
HP  
DEC/Alpha  
Silicon Graphics  
PC



I/O Controllers:

VME, VXI,  
PCI, ISA

Remote I/O and Signal  
Conditioning:

CAN-Bus, Industry Pack  
VME, VXI, PCI, ISA  
CAMAC, GPIB  
Profibus, Bitbus, Serial  
Allen-Bradley, Modbus

オペレータ・インタフ  
ェース (OPI)および  
制御アプリケーション

ネットワーク

分散配置されたハ  
ードウェア制御計  
算機(IOC)

ハードウェア・イン  
タフェース

# EPICSの提供する機能

## 制御システムの基本機能

- ✓ デバイスの状態をGet/Put/Monitor
- ✓ システムの状態を運転員に提示
- ✓ 新規の制御指令値を運転員から受け取りデバイスに設定する。
- ✓ 状態の異常を運転員に通知する。
- ✓ システムの初期化
  - 以前の運転状態の復元
  - 運転状態の保存
  - 制御アルゴリズムを制御アプリケーションに実装

EPICS ではこれらの機能は EPICS run time database と Channel Access(CA) プロトコルを組み合わせて実現される。

# EPICSで何が出来、なにが出来ないか。

- ☑ EPICSを使って制御システムを構築することが出来る。(EPICSはツールキットであって制御システムその物ではない。)
- ☑ プログラムを使わずに操作画面を作成できる。EPICSの汎用アプリケーション (CAクライアントソフトウェア) は設定ファイルを用意するだけで利用できるものがほとんどである。
- ☑ 制御対象のデバイスがEPICSで既にサポートされているものであれば、プログラムの手間無しにこれらのデバイスを制御出来る。(もしサポートされていなければ、いくらかのC/C++によるプログラムを作成する必要がある。)
- ☑ 加速器制御に必要とされるほとんどのデータをEPICSを使って収集することが出来る。(KEKB-Belleの様な高速なデータ収集システムには適用出来ない。)

# EPICSの基本構成

## ハードウェア制御層

- VME Single Board Computer as a controller
- VME/VXI and field busses as communication bus
- EPICS プロセス・データベースが定期的な機器データの読み出し、書き込みなどの基本機能を実現

## 通信・ネットワーク

- 標準的なネットワーク技術(TCP&UDP/IP, Ethernet/FDDIなど)を採用。
- ハードウェア制御層と制御アプリケーションとのデータ交換のために Channel Access プロトコルを定義
- Monitor :ハードウェア制御層から制御アプリケーションへのeventの通知を行うCAの機能

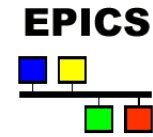
## 操作インタフェース

- 制御システム用の汎用アプリケーション

# Basic Frame Work

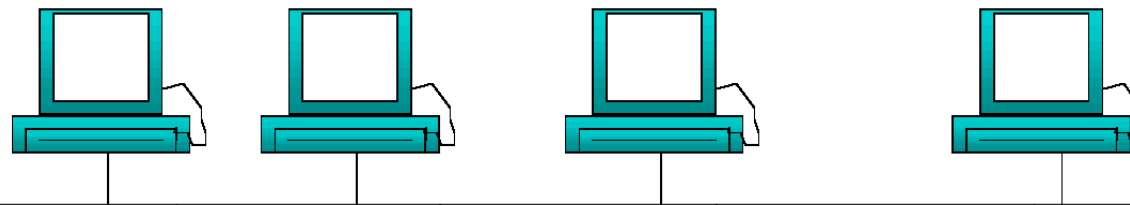
US  
Particle  
Accelerator  
School

## *Distributed Hardware Configuration*



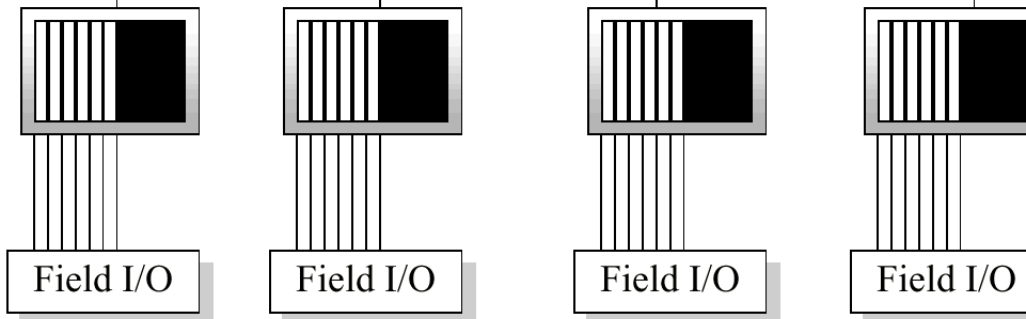
### Workstations:

Sun  
HP  
DEC/Alpha  
Silicon Graphics  
PC



### I/O Controllers:

VME, VXI,  
PCI, ISA



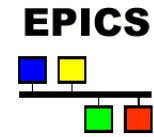
### Remote I/O and Signal Conditioning:

CAN-Bus, Industry Pack  
VME, VXI, PCI, ISA  
CAMAC, GPIB  
Profibus, Bitbus, Serial  
Allen-Bradley, Modbus

# Operating Systems

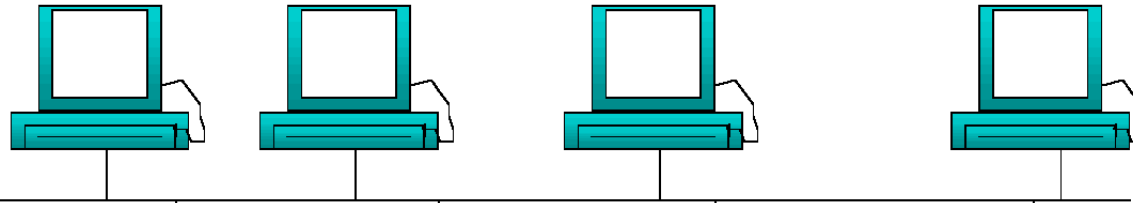
US  
Particle  
Accelerator  
School

## Standard Operating System Software Configurations



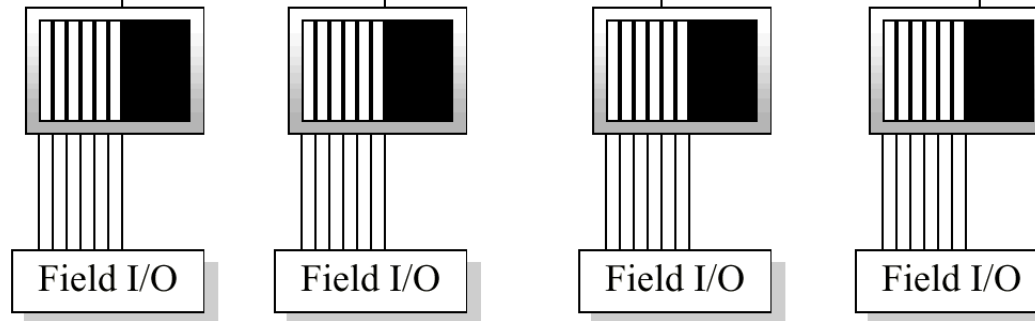
### Workstations OS:

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



### I/O Controllers OS:

- vxWorks
- RTEMS
- Linux





# Core Software

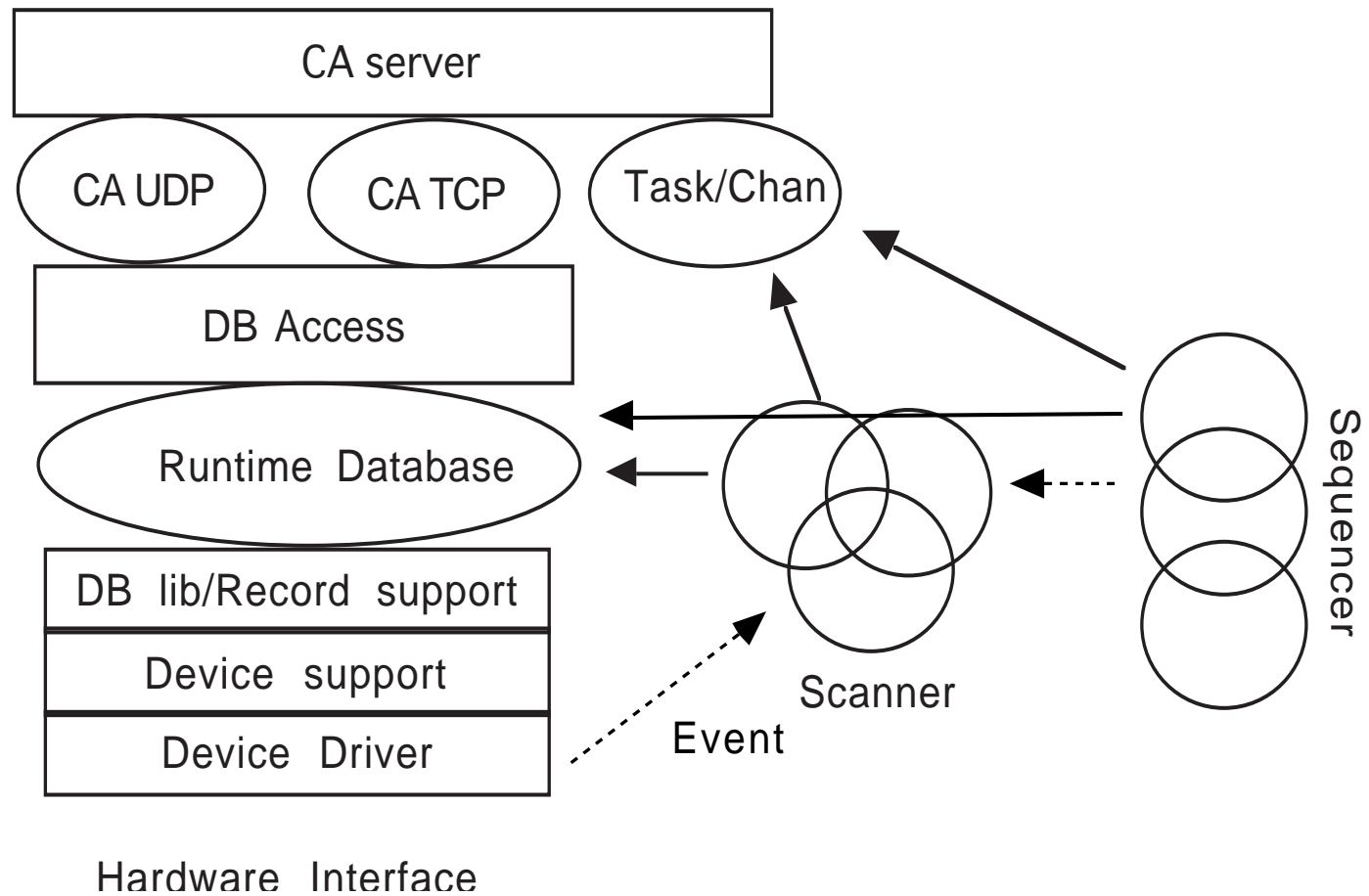
## Channel Access (CA): データ交換のためのプロトコル

- TCP/IP,UDP/IPの上に定義されている : "Follows the standard" 原理
- 非同期のプロトコル
- Server/Client model : 読み出し(get)/書き込み(put) 、 処理終了通知機能 (callback)がある。
- + Publisher/Subscriber model:monitor 機能、状態変化の通知機能
- 制御システム内のデータへの一元的なアクセス方法を提供する。

## EPICS プロセス・データベース: ハードウェア制御の経由点

- CA server : チャンネル名の解決/ CA データ要求の応答 /アクセス制御
- Scan mechanism : 定期的・イベント
- データベース・アクセス・ライブラリ
- レコード・サポート : 様々な標準レコード型、レコード型の拡張
- デバイス・サポート/ドライバ・サポート : ハードウェア固有

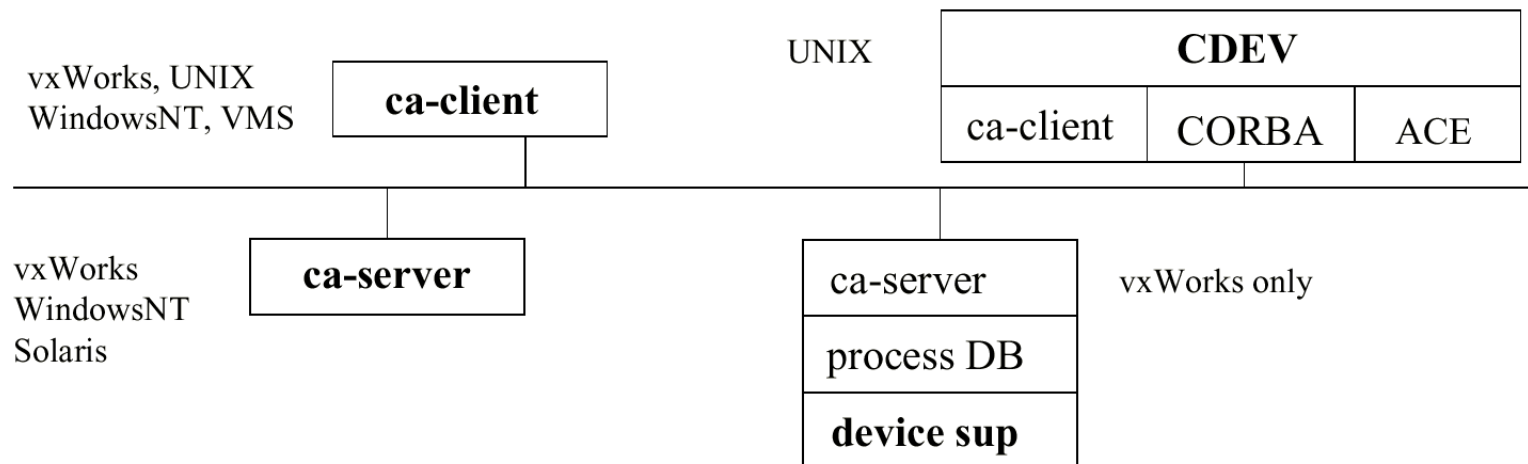
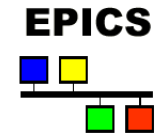
# Processes on IOC



# Distributed Software Architecture

US  
Particle  
Accelerator  
School

## *Distributed Software Architecture*



# EPICSに含まれるその他のソフトウェア

## EPICS Processデータベース構築のためのツール

- dbLoadTemplate/dbLoadDatabase,... : データベースをIOCにダウンロードするためのツール
- gdct/Capfast+e2db /jdct ,... : データベース設定ファイルの作成支援
- Database デバッグ用コマンド

## 汎用アプリケーション（設定ファイルによる動作）

- EDD/DM, medm, dm2k : 操作画面作成ツール
- ALH : アラーム状態通知・表示プログラム
- Archiver, Channel Archiver: 制御データを files/RDBに保存また、保存データを引き出すためのツール
- その他 :

# CA client applications

## Standalone CA Clients

- ALH: Alarm Handler
- BURT: Backup and Restore Tool
- Channel Archiver (LANL)
- DM2K: Display Manager 2000 (BESSY)
- DP: Parameter Display Page
- EDD/DM: Editor and Display Manager (LANL)
- KM: Knob Manager and KnobConfig, interface to SunDials
- MEDM: Motif Editor and Display Manager
- StripTool: Strip-Chart display client (JLAB)
- XARR: Archive Data Browsing (JLAB)
- XMCA, PROBE: Motif Channel Monitoring programs
- and more.

# CA Interfaces to other tools and languages

## CA Interfaces to other tools and languages

- ActiveX-CA: CA client for Active-X programs (LANL)
- CA-ActiveX: CA server of Active-X objects (LANL)
- CaMath: CA client interface for Mathematica
- CaWave: CA client for Precision Visuals' analysis package
- CDEV: Common Device application framework and API (JLAB)
- EZCA: Simplified Channel Access interface library for C
- JCA: Channel Access client class library for Java
- Gateway: Channel Access proxy server (BESSY)
- PADL: Portable CA Server interface to IDL (LANL)
- PCAS: Channel Access Portable Server (LANL)
- PEZCA: CA interface to Perl, using EZCA (BESSY)
- SCA: Simple Channel Access (LBL/ALS)
- SDDS: The Self-Describing Data Sets analysis package
- Tcl Interfaces: ET and IT
- Python/Tk
- SAD/Tk

# EPICS プロセスデータベース

A *database* はレコードの集合

A *record* is はフィールドの集まり

A *field* はCAによるデータアクセスの基本単位(channel)となる。

Channel はその名前によって区別される。

Channel の名前は形式: <record 名>.<field 名> を持つ。

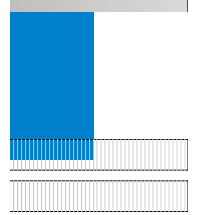
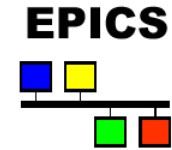
record名は31文字以内 (変更可能ではあるけれど,...)

field名は四文字以内



US  
Particle  
Accelerator  
School

## EPICS Process Database Provides Data Acquisition and Control



LAN	
ca-client	ca-server
<b>process DB</b>	
dev support	

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

Record types are the basic elements: AI, AO, BI, BO, Motor, CALC, PID, SUB etc....

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

Fields hold runtime values: VALUE, TIMESTAMP, ALARM CONDITION, etc....

New record types are easily added

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

Loaded as ASCII records into vxWorks at boot time

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



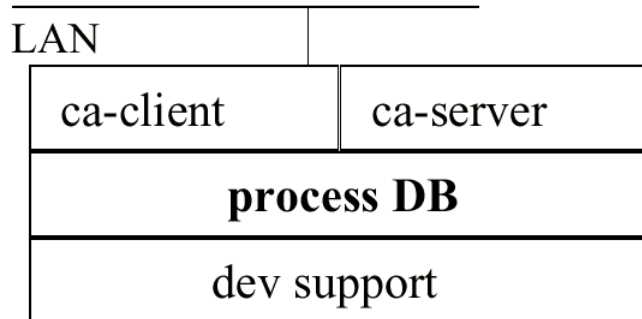
# DB: EPICS レコード

## レコード型

- AI/AO, LI/LO, BI/BO, Waveform, ... : 基本データ型の入出力.
- CALC, Subroutine, ... : データの変換.
- sequence, fan out, event, ... : レコードプロセスの流れ

## Record processing :

- レコードは静的なデータの集合では無く、レコードプロセスによってデバイスと相互作用する。
- 定期的/ Event/ I/O interrupt などのSCAN機構あるいは操作員の要求などによってレコードのプロセスが開始される。
- デバイスサポート中のprocess() 関数がRecordプロセスの間にcallされる。



Record execution time varies from record type to record type

AI on a 68060 is ~12,000/second (50% idle)

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

Fastest periodic scan rate is dependent on vxWorks clock tick (typically 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 records use around 1 Megabyte of memory

Assignment to particular physical I/O is distinct from process logic - ASCII device type

## DB: DTYP フィールドと IN/OUT フィールド

DTYP field : レコードとデバイスサポートを結び付ける。

IN/OUT field : デバイスのハードウェア・アドレスを保持.

<hw address>@<parm field>

### ■ IOタイプ

- ▶ Generic
- ▶ VME :
- ▶ GP-IB :
- ▶ CAMAC :

アドレス指定中のparm フィールドを使って デバイス・サポートにパラメータを渡す事が可能。parmフィールドの解釈はデバイス・サポートに依存する。

# DB:既に稼働実績のあるHardware/Field busses

<http://www.aps.anl.gov/epics/modules/bus.php>

- ▶ VME boards
- ▶ VXI boards/VME-MXI/VXI-MXI
- ▶ CAMAC
- ▶ GPIB
- ▶ Serial/RS-232
- ▶ IPAC ( Industrial Pack)
- ▶ Allen Bradley : AB1771 / AB1794
- ▶ BitBus
- ▶ Arcnet
- ▶ CAN
- ▶ Group 3 control( a fiber optically linked control system using distributed I/O modules )
- ▶ Modbus/Modbus+
- ▶ WE : from Yokogawa. Used in JHF

\*) See "Supported Hardware" by Ned D. Arnold in USPAS course for more information

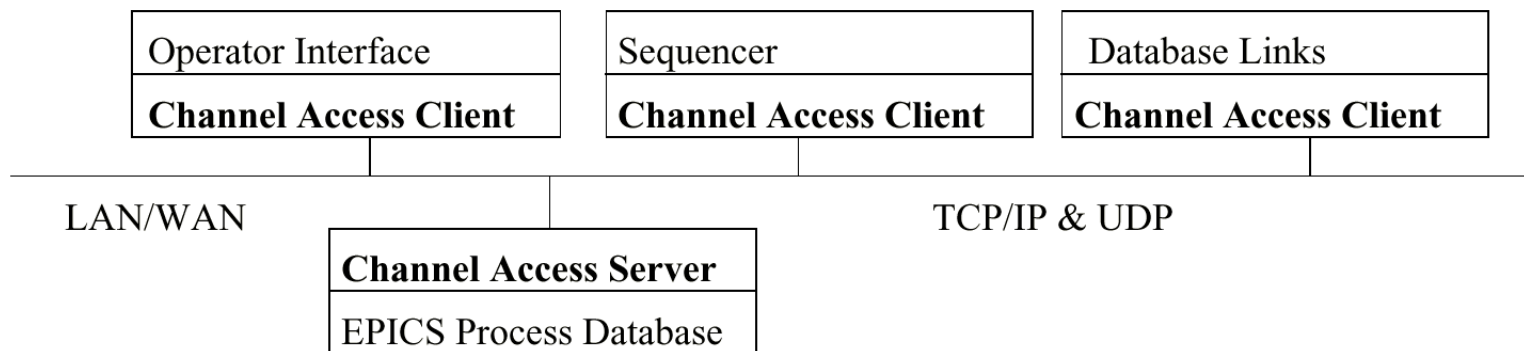
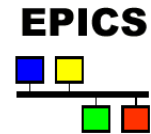
## DB : アラーム

- ▶ 各レコードはアラーム状態(Normal, Minor, Major, Invalid)を示す SEVRフィールドを持つ。
- ▶ CA クライアントはこのSEVR フィールドをモニタすることで、レコードのアラーム状態の変化を知る。
- ▶ STATフィールドは詳細なアラームステータスの情報を保持する。
- ▶ STAT フィールドのとりうる値は<alarm.h>に定義されている。
- ▶ Alarm Handler は汎用のアラーム状態モニタプログラムである。

# CA: Channel Access

US  
Particle  
Accelerator  
School

## Channel Access Client/Server Libraries



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

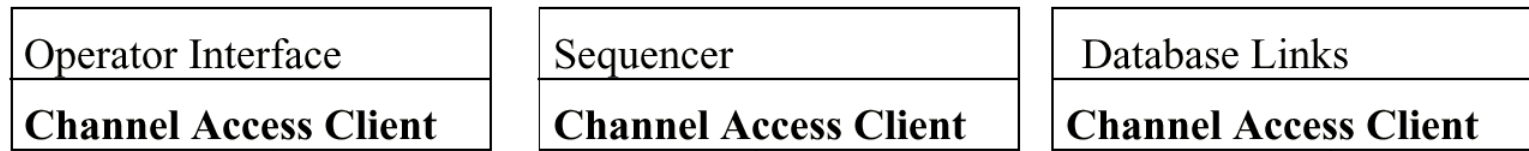
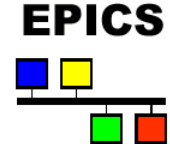
Server: Provides read/write connections to information in this node to any client on the network through channel access client calls

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

US  
Particle  
Accelerator  
School

## Channel Access



TCP/IP & UDP

LAN/WAN

Performance:

68040 over 10 Mbit Ethernet

Gets

Propagation Delay	2 mS
Throughput	7.8K /sec

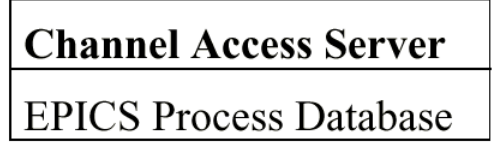
Puts

Propagation Delay	1 mS
Throughput	17K /sec

Monitors

Propagation Delay	Dependent
Throughput	10K / sec

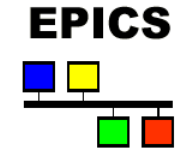
(Typically 10% channels have monitors)  
(memory use in IOC - 2 Meg / 60 connections)  
(30% network load for 10K monitors / second)



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



## Channel Access Client Codes Provide a Variety of Functionality



Operator Interface
<b>Channel Access Client</b>
LAN

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

Alarm Handler: **ALH**

State Programming: **SNL**

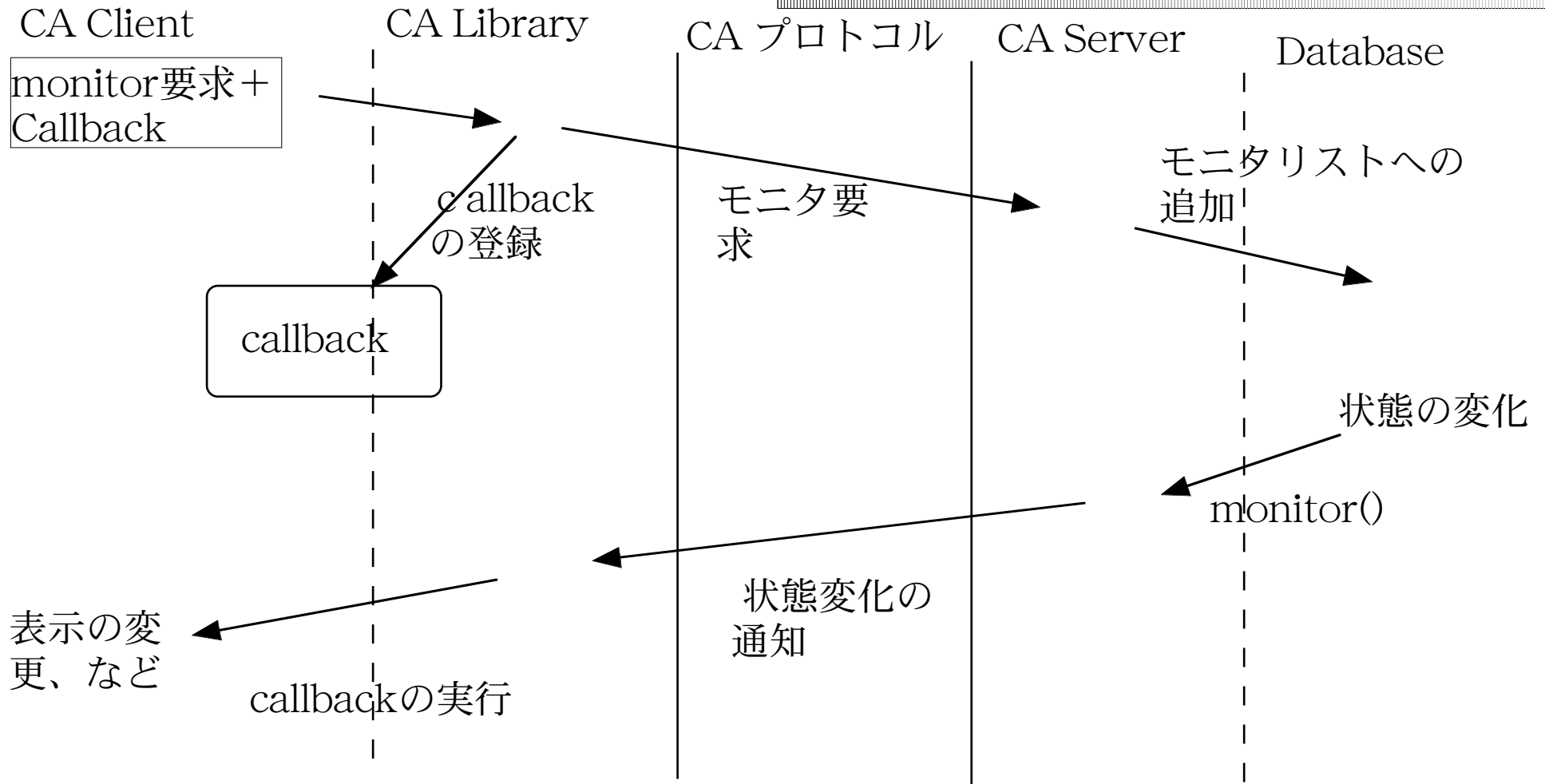
Other Clients: PV-Wave, Mathematica, Matlab, Probe, Orbit Lock,  
Correlation Plots, Striptool, X-Orbit, SAD, CNLS-net  
SDDS, ARTEMIS, Archivers, Save/Restore



## CA: Monitor機能

- ▶ CA クライアントはモニタ要求をIOC上のCA サーバに送ると同時に、コールバック関数をCAライブラリに登録する。
- ▶ モニタ登録したレコードの状態の変化がモニタ条件に合致すると、CAサーバはクライアントに状態変化の通知を送る。
- ▶ CA client側では通知を受けてコールバック関数が実行される。

# CA:モニタ機能



# CA:Data のロギング/IOC 状態の復帰

## Data Archiving/Logging

- 要求毎のスナップショット
  - ▶ BURT :
  - ▶ Save/Restore routines on IOC
  - ▶ CaSave in KEKB
- 定期的なデータのロギング
  - ▶ Channel Archiver
  - ▶ kekblogger in KEKB
- イベント
  - ▶ CaSave in KEKB

# アクセス・セキュリティ

許可されていないCA経由のアクセスを制限する。

アクセスの権限は以下の項目を条件とすることができる:

- Who: Channel accessを発行しているユーザのユーザーID
- Where: Channel Access要求が発行されたホスト計算機のHost id..
- What: Channel access 要求の対象となるフィールドのアクセス・セキュリティ・レベルとそのレコードが所属するアクセス・セキュリティ・グループ(レコードのASGフィールドに指定)
- When:アクセス・セキュリティ・グループに指定したレコードの値によって選ばれるアクセス・ルール

# Access Security の例

```
UAG(uag) {user1,user2}
HAG(hag) {host1,host2}
ASG(DEFAULT) {
  RULE(1,READ)
  RULE(1,WRITE) {
    UAG(uag)
    HAG(hag)
  }
}
```

These rules provide read access to anyone located anywhere and write access to user1 and user2 if they are located at host1 or host2.

An ASG field of a record specifies ASG to which this record belongs.

# Documentation

## EPICS Home on WWW

- <http://www.aps.anl.gov/epics/>

## USPAS(US Particle Accelerator School) EPICS course text

- <http://www.aps.anl.gov/epics/docs/training.php>

## Application Developer's Guide

- R3.13 : <http://www.aps.anl.gov/epics/modules/base/R3-13.php>
- R3.14 : <http://www.aps.anl.gov/epics/modules/base/R3-14.php>

## Record Reference Manual

- <http://www.aps.anl.gov/asd/controls/epics/EpicsDocumentation/AppDevManuals/RecordRef/Recordref-1.html>

# Documentation

## Channel Access Library Reference Manual

- <http://mesa53.lanl.gov/lansce8/Epics/ca/ca.htm>

## Recommended EPICS Documentation on LBL-Website

- <http://csg.lbl.gov/EPICS/RecommendedDocs.html>