



KEKB and Linac Controls with EPICS

(Control System Examples)

Kazuro Furukawa, KEK

for KEB Controls Group

for Linac Controls Group

<kazuro.furukawa@kek.jp>

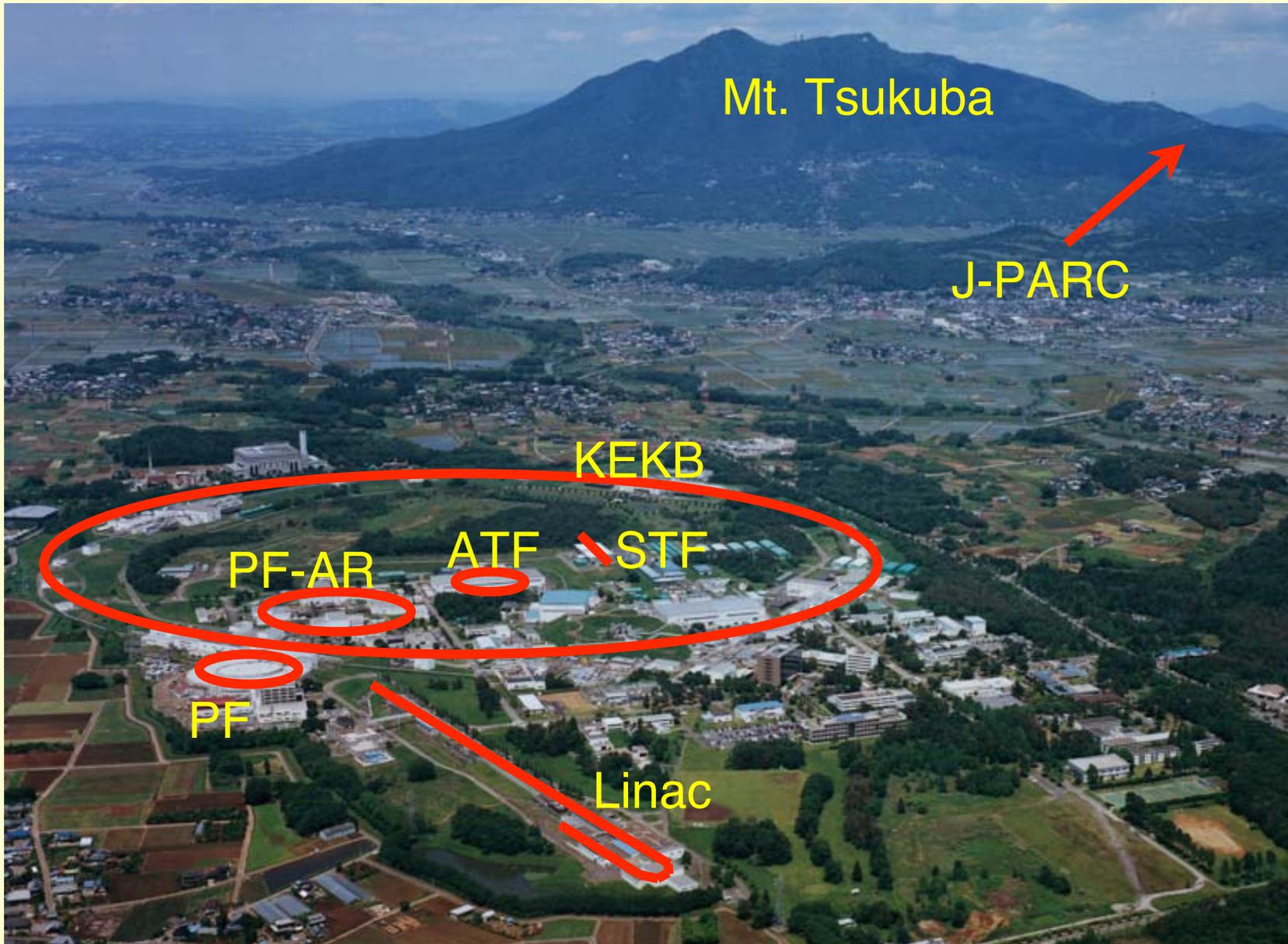
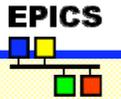
Jul. 11. 2006.

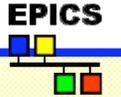




- ◆ Accelerator Control Systems in KEK
- ◆ KEKB & Linac Accelerator
- ◆ KEKB Controls
- ◆ Linac Controls
- ◆ Operational Software







Control Systems in KEK

◆ Operational Presently

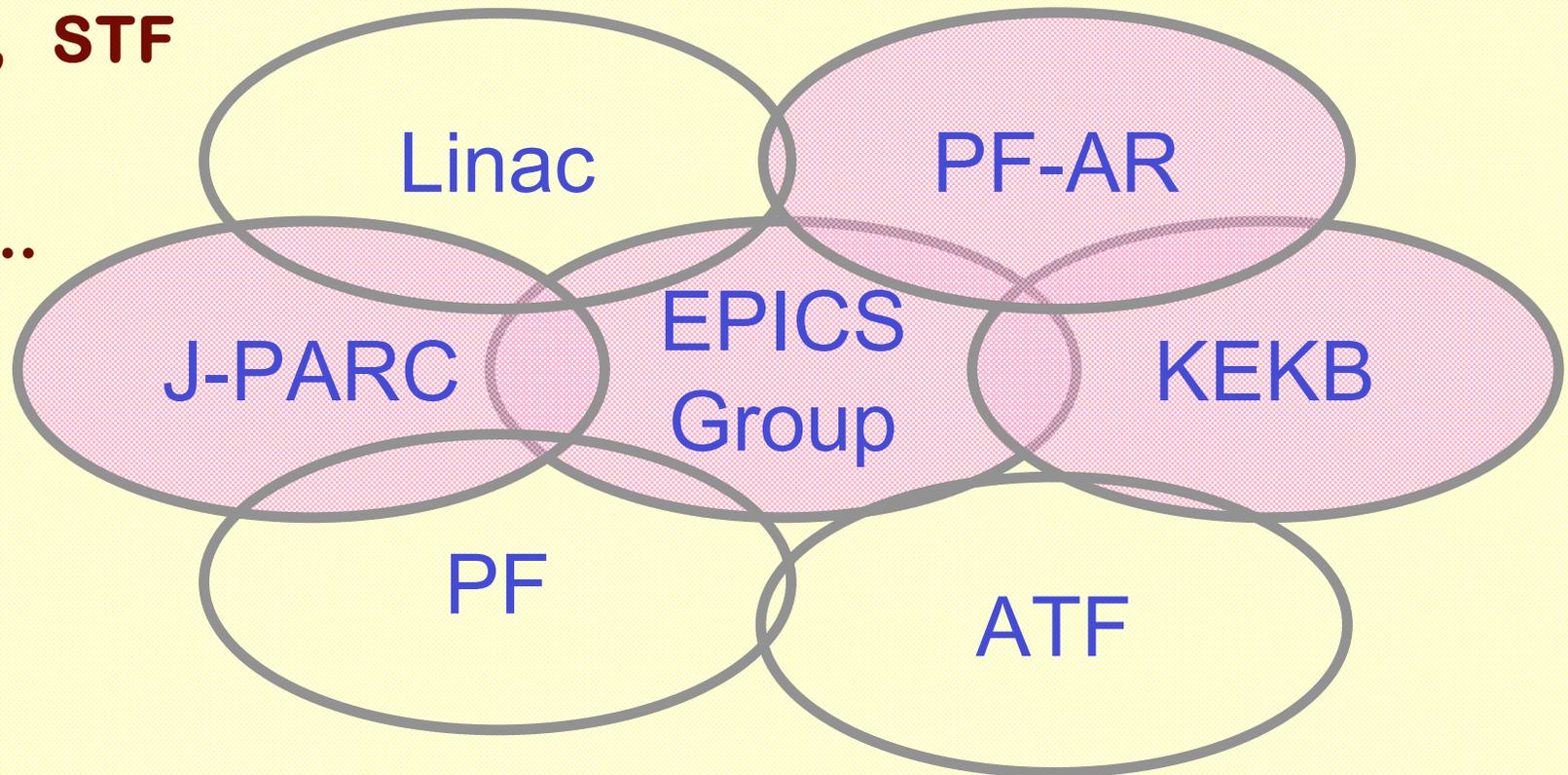
- ❖ Linac, PF, PF-AR, ATF, KEKB

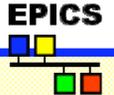
◆ Under Construction

- ❖ J-PARC, STF

◆ EPICS

- ❖ KEKB, ...





Control Systems

- ◆ 12GeV Proton Synchrotron (PS): Hardwire + Mini-computers
- ◆ 2.5GeV Electron Linac (Linac): 8 Mini-computers + >200 Micro-computers, Optical Networks
- ◆ Photon Factory (PF): Mini-computers -- Workstations
- ◆ TRISTAN ~33GeV: ~20 Mini-computers, CAMAC + NODAL Interpreter
- ◆ Upgraded PS: VME/VersaDOS + MAP
- ◆ Upgraded Linac: Unix servers, VME, PLC, CAMAC + TCP/IP + Home-grown RPC, Tcl/Tk, Gateway to EPICS
- ◆ ATF: VMS, CAMAC + V-System (Vista)
- ◆ KEKB: VME, CAMAC, VXI, ARCnet + EPICS + Python, SADscript
- ◆ PF-AR: the Same architecture as KEKB
- ◆ PF: Many of the Components with EPICS (Obina-san's Talk)





KEKB





KEKB

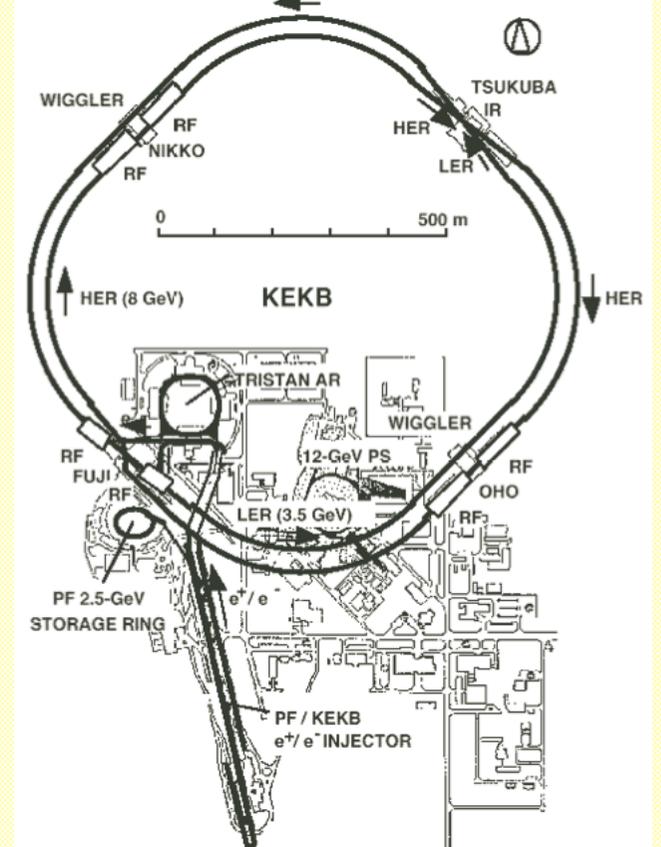
◆ **B-factory: Electron/Positron Asymmetric Collider for CP-violation Study**

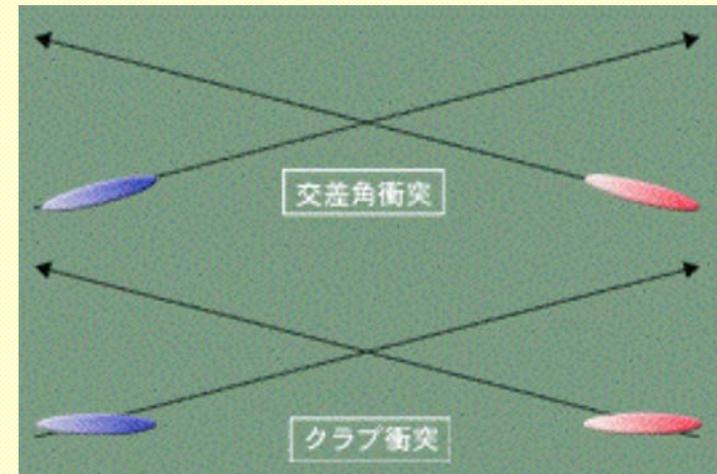
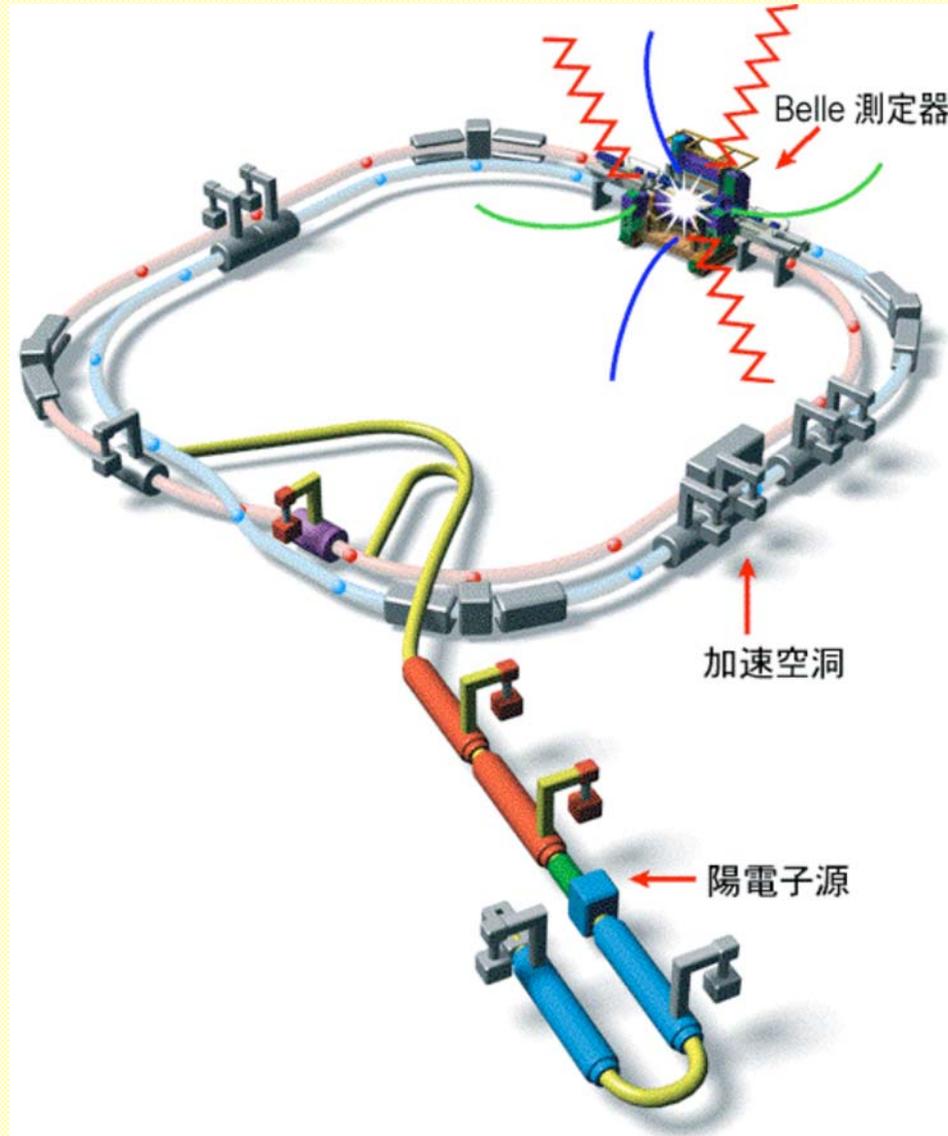
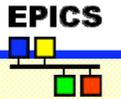
◆ **~3km Dual-rings: Electron(8GeV - 1.4A) / Positron(3.5GeV - 1.8A)**

◆ **Stable and Robust Operation**

◆ **Many Active Operation Parameters**

◆ **Controls are Important**





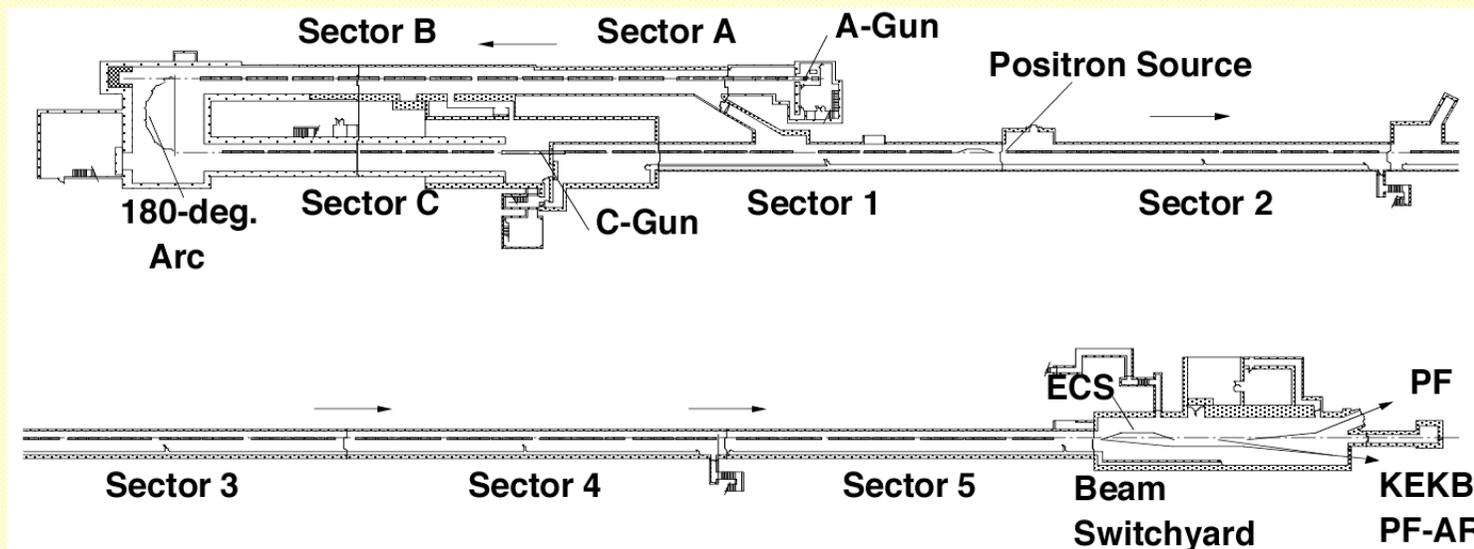
Electron/Positron Injector Linac

◆ Machine Features

- ❖ 600m Linac with 59 S-band rf Stations, most of them Equipped with SLED to provide 20MeV/m
- ❖ Dual Sub-Harmonic Bunchers to achieve 10ps for 10nC, and Energy Compression System for Positron

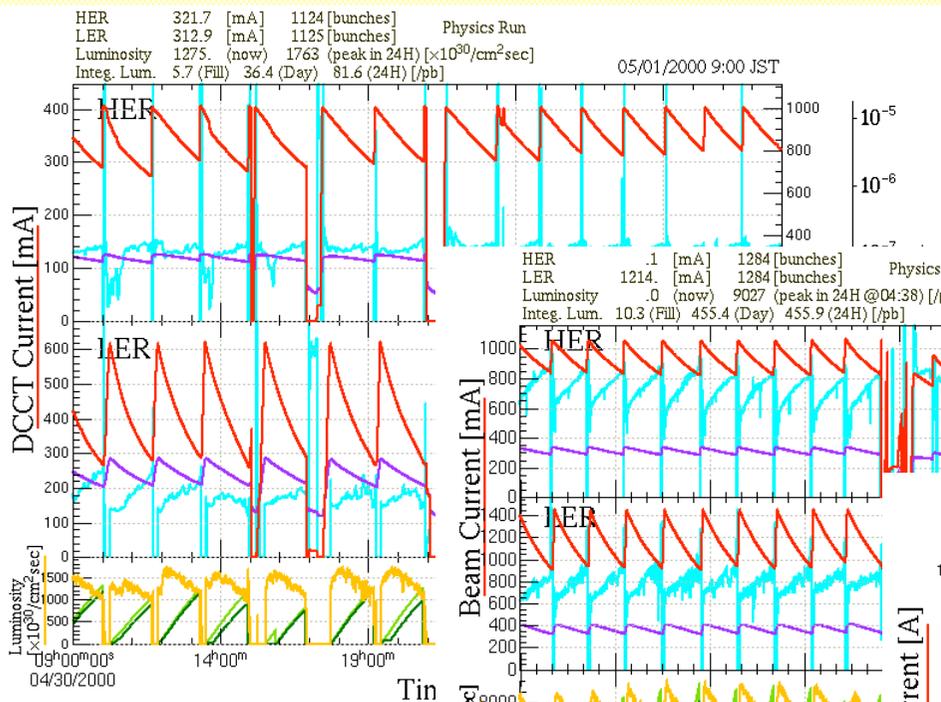
◆ Beam Characteristics

- ❖ 8GeV 1.2nC Electron and 3.5GeV 0.6nC x2 Positron for KEKB
- ❖ 2.5GeV 0.2nC for PF, 3.0GeV 0.2nC for PF-AR

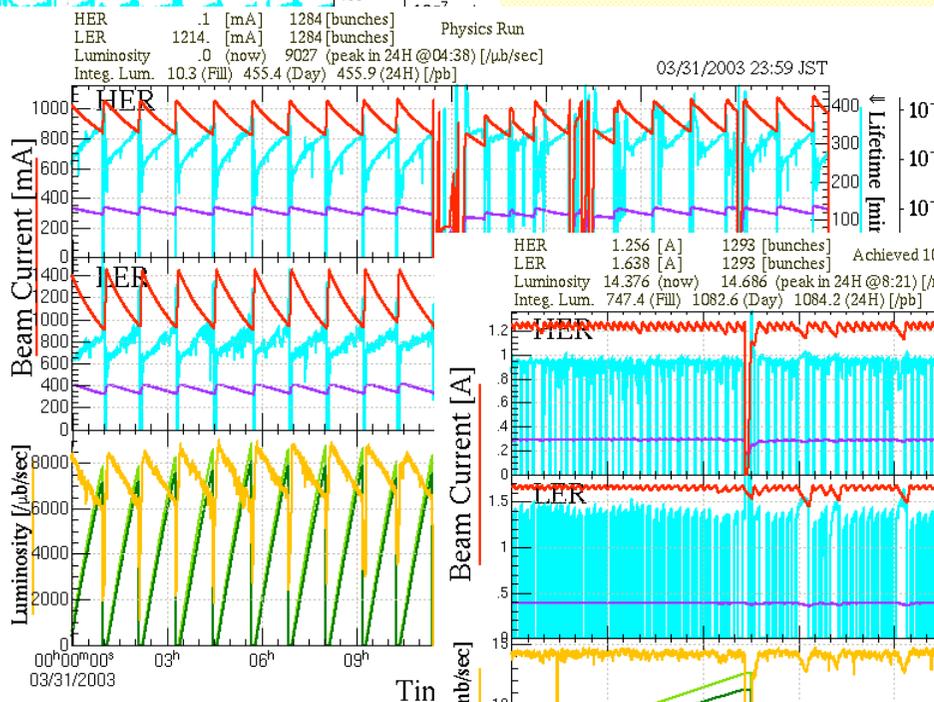




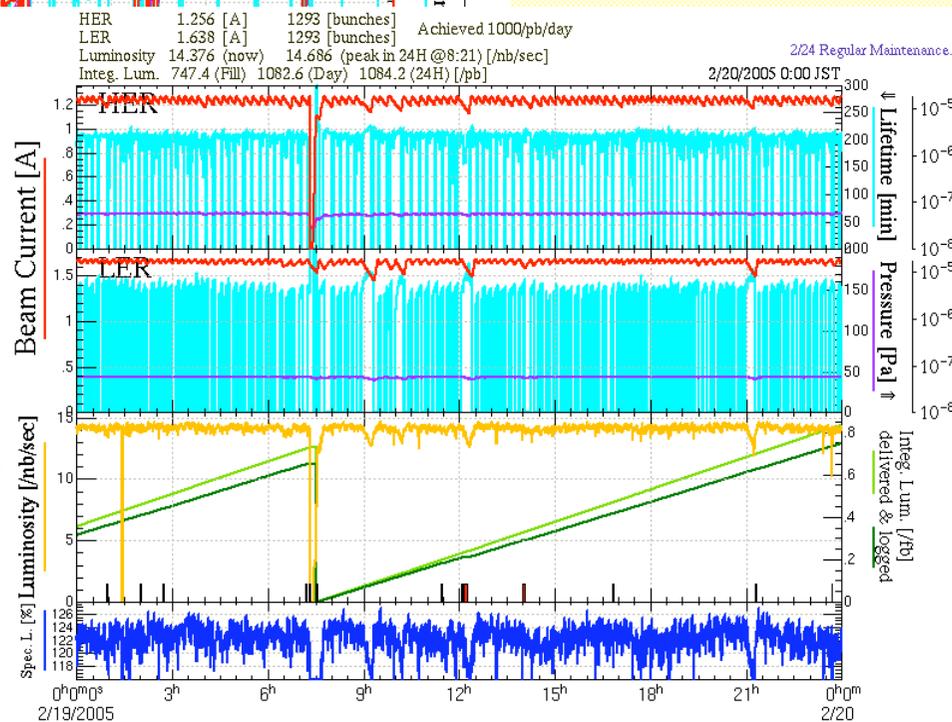
Increase of the Luminosity



May.2000



Apr.2003
Dual Bunch e⁺



Feb.2005
Continuous
Injections



Performance of KEKB

Staffs are
always
Interested in
performances
in KEKB/Belle

Here is the
KEKB daily
Performance
Page updated
Every minute





KEKB Control System (Hardware)

◆ FDDI-GbE Fiber Optic Networks

- ❖ Single Broadcast Domain
- ❖ Central Control Room and 26 Local Control Rooms

◆ VME/IOC

- ❖ ~100 VME/IOC mostly with PowerPC CPU

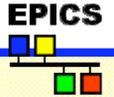
◆ Fieldbus

- ❖ VXI thru MXI for BPM Instrumentations
- ❖ CAMAC for rf and Vacuum (inherited from TRISTAN)
- ❖ ~180 ArcNet network segments for Magnet Power Supplies, and other field Controllers
- ❖ GPIB for Instrumentations, RS232C, Modbus+ for PLCs

◆ Host Computers

- ❖ HP-UX/PA-Risc, Linux/x86 Controls Server
- ❖ 3 Tru64/Alpha with TruCluster
- ❖ Several Linux
- ❖ Many MacOSX
- ❖ (Solaris/Sparc for VxWorks)

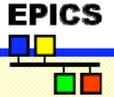




KEKB Control System (Software)

- ◆ **EPICS 3.13.1 mainly, with Some 3.14**
- ◆ **VxWorks 5.3.1 mainly**
 - ❖ **Hope to upgrade EPICS/VxWorks Shortly**
- ◆ **IOC Development**
 - ❖ **CapFast, Perl, SADscript for Database Configuration**
 - ❖ **Oracle as a backend Database (Partially)**
- ◆ **Operational Application Development**
 - ❖ **MEDM(DM2k) for Startup**
 - ❖ **Python/Tk for Equipment Controls**
 - ❖ **SADScript/Tk for Beam Operation, etc**





IOC and Records

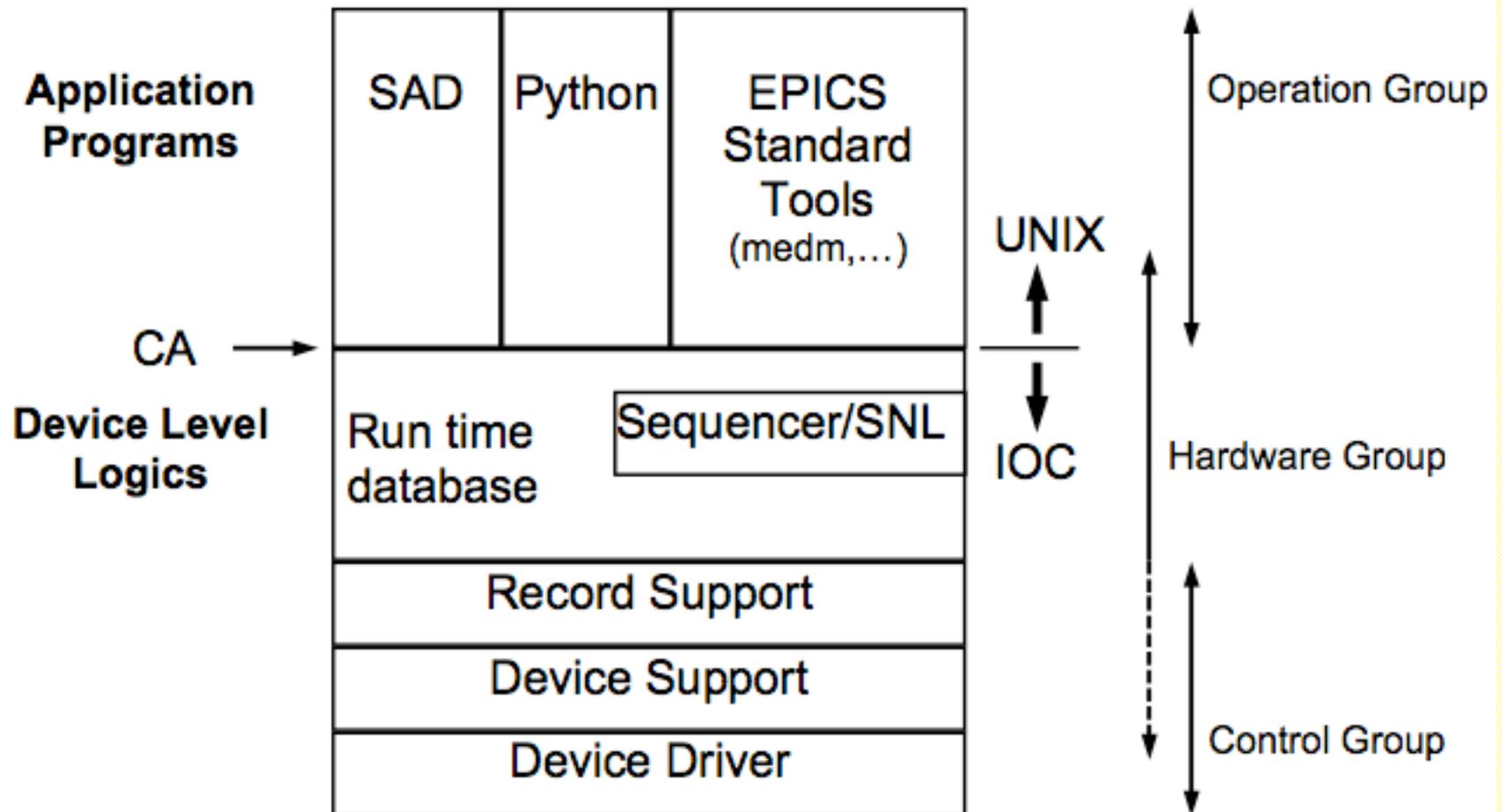
- ◆ **There Several Groups in KEKB**
 - ❖ **Each IOC is Assigned to a Group**
 - ✧ **MG, RF, VA, BM, etc**
 - ❖ **About 100 in Total**
 - ❖ **About 250k Records in Total**
 - ❖ **Most Groups Follow Naming Convention**
 - ✧ **GroupName Accelerator DeviceType**
 - ✧ **DeviceName**
 - ✧ **Property**
- ◆ **Some EPICS Databases are Generated from Oracle Database**





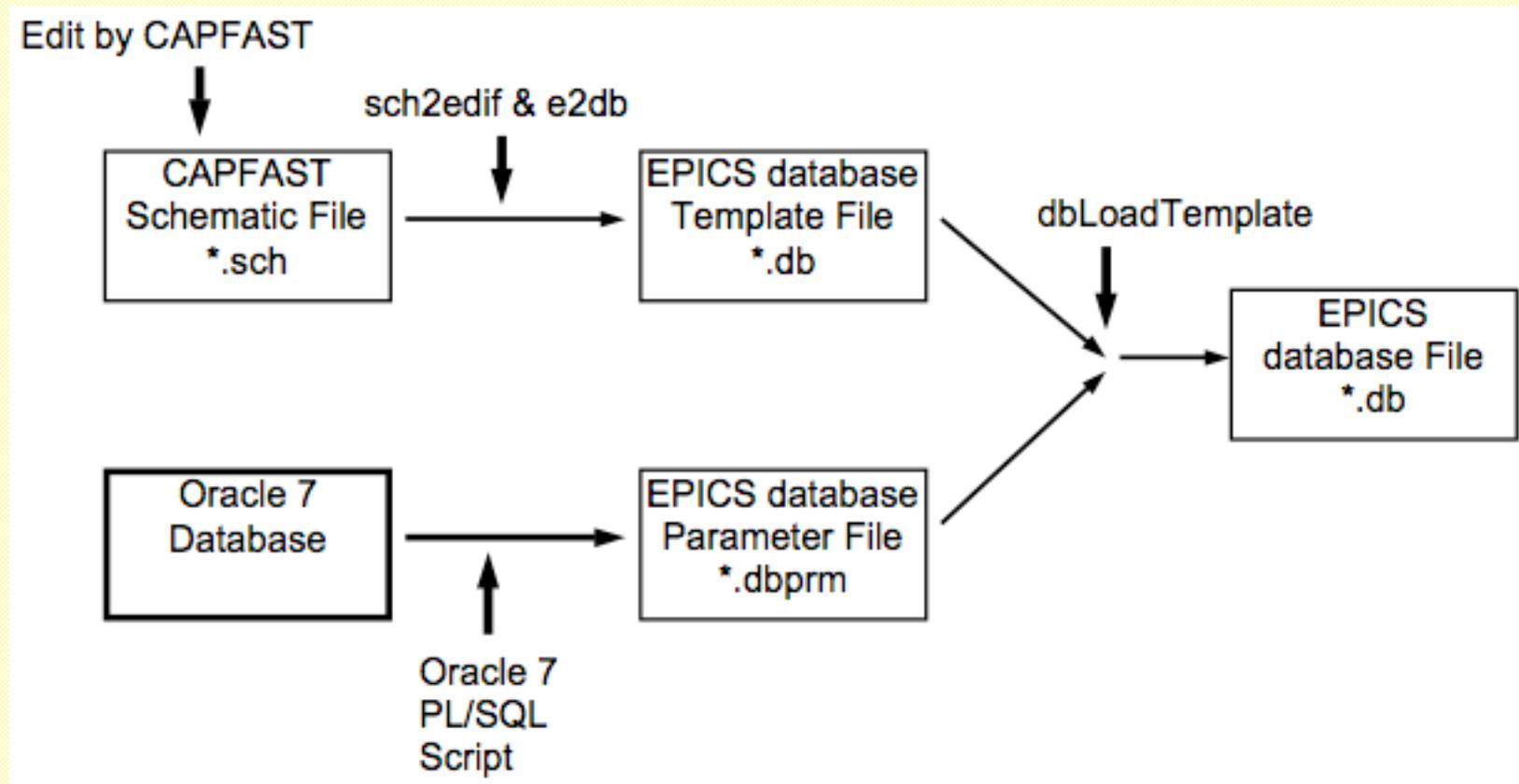
IOC Configuration

Structure of Software





IOC Database with Oracle

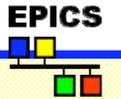




Magnet Controls

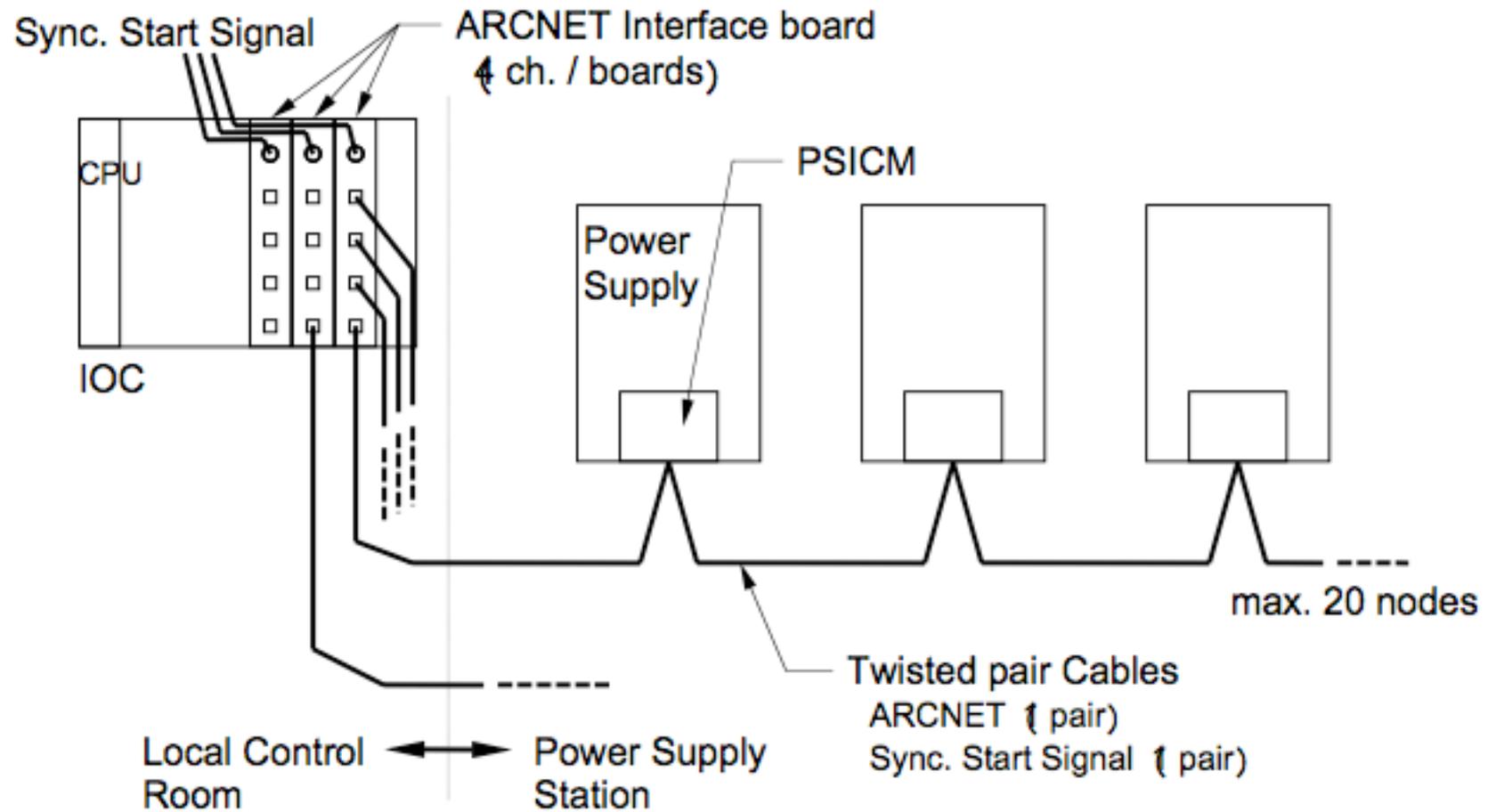
- ◆ **~5000 Magnets, ~2500 Power Supplies**
 - ✧ **No One-to-one Correspondence**
 - ❖ **Basic Controls thru ARCnet**
 - ❖ **Voltmeter Scanner for Analog Read-out**
 - ❖ **PLC for Interlocks**
- ◆ **IOC Manages All Controls**
 - ❖ **Interlock Status, On/Off, etc**
 - ❖ **Energy - Field - Current**
 - ❖ **Synchronous Operation**
 - ✧ **for Tune change, Orbit Correction, etc.**
 - ❖ **Standardization**
- ◆ **Special Large Subroutine Record**
 - ❖ **“hugesub” Record and C Routine**





Magnet Controls

Configuration of Magnet Power Supply Control System





KEKBLOG and ZLOG

◆ KEKBlog/kblog Archiver is Used from the Beginning of the Commissioning

- ❖ Just less than 2GB / day

- ❖ Several Viewer Tools

 - ✧ Very often Used to Analyze the Operation Status

◆ Zlog Operation Log

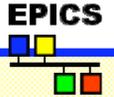
- ❖ Zope, Python, PostgreSQL

 - ✧ Most of the operation logs

 - ✧ In Mostly Japanese

 - ✧ Waiting for the Figure Storing Integration





Near (?) Future

- ◆ **Catching up with Recent Versions of EPICS, VxWorks, etc**
- ◆ **Replacement of CAMAC with PLCs**
- ◆ **Replacement of RAS (Reliability, Availability, Serviceability) boards**
 - ❖ **Instead of RS232C but Ethernet**
- ◆ **New Hardware for Crab Cavities**
- ◆ **Python Related Developments**





Linac





Linac Controls

- ◆ **KEKB = Factory Machine => Reliable Operation**
- ◆ **Controls should be Robust and Flexible**
- ◆ **~1000 devices and ~10000 signals**
- ◆ **Frequent Beam Mode Switches**
 - ❖ **Four very Different Beam Modes, 360 times/day**
- ◆ **Precise Controls of Beam Parameters**
 - ❖ **Energy, Orbit, Emittance, Charge, Energy spread, Timing, etc.**





History and Design Concept

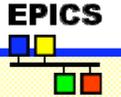
◆ History

- ❖ 1978-1982: Construction of First Computer-controlled System with 8 mini-computers, >200 micro-computers, >30 optical loop networks
- ❖ 1989-1992: Design of the next system
- ❖ 1993-1997: Installation and expansion for KEKB

◆ Design Concept

- ❖ Use of International and/or de-facto Standards
- ❖ Use of Optical IP Networks for every Device controllers
 - ✧ No new field Networks, only IP Network (to be inherited by J-PARC)
- ❖ Both of above should make future upgrade easier
- ❖ (EPICS was not available widely at that time)

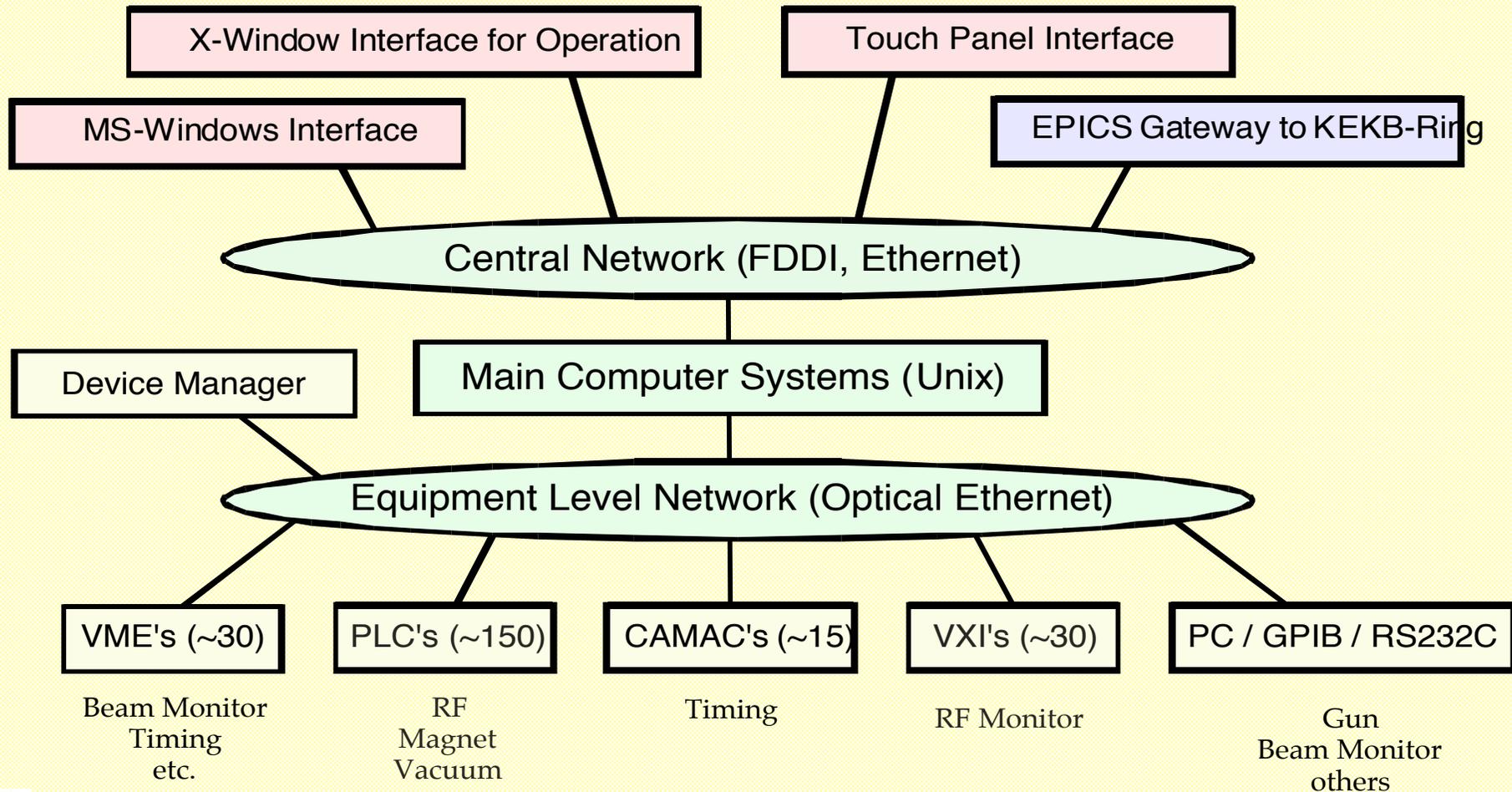


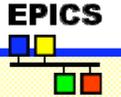


Physical Structure

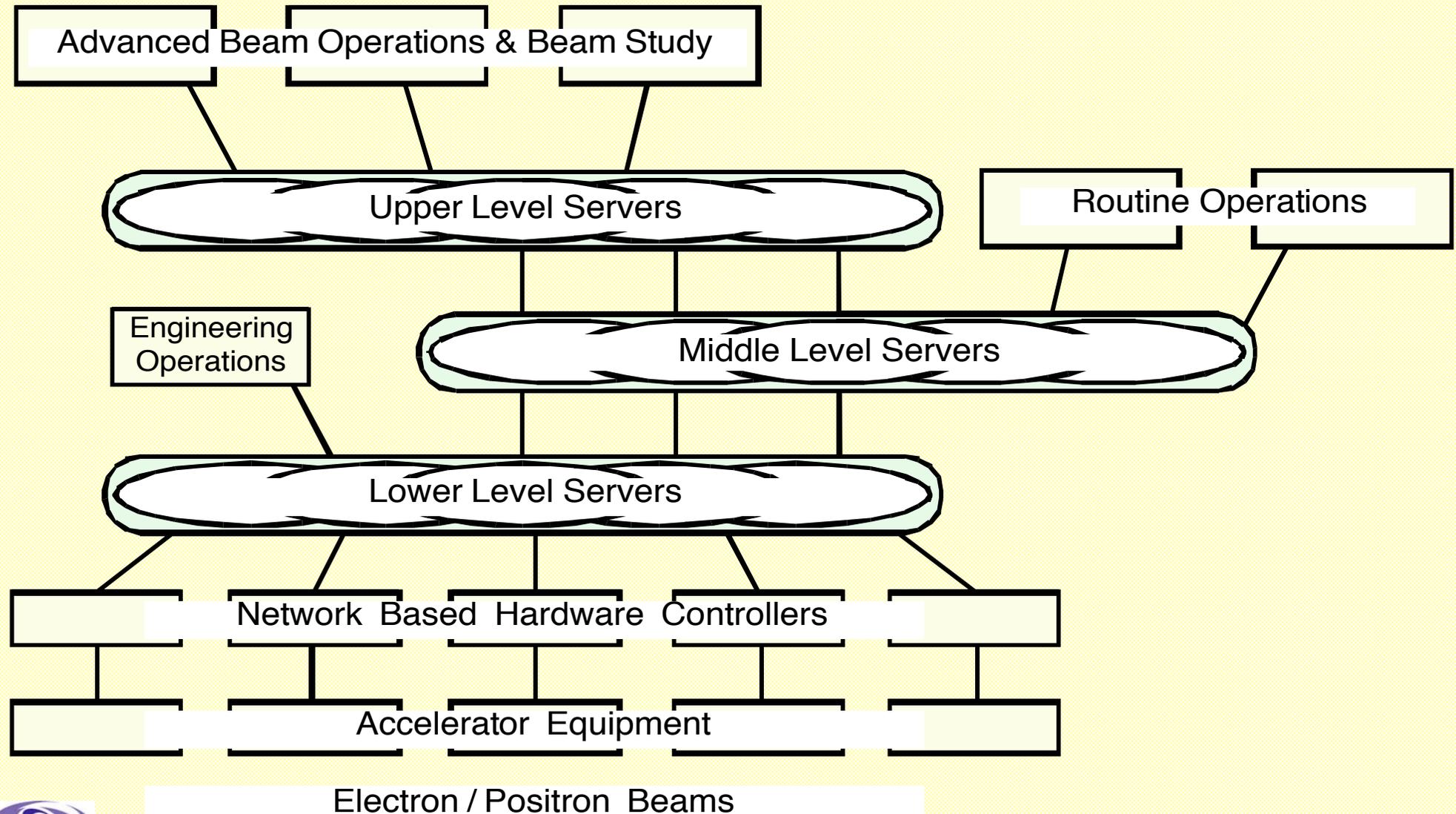
◆ Multi-tier, Multi-hardware, Multi-client,

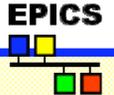
...





Multi-tier Logical Structure

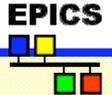




Software Architecture

- ◆ **Base control software structure for Multi-platform**
 - ❖ any Unix, OS9, LynxOS (Realtime), VMS, DOS, Windows, MacOS
 - ❖ TCP - UDP General Communication Library
 - ❖ Shared-Memory, Semaphore Library
 - ❖ Simple Home-grown RPC (Remote Procedure Call) Library
 - ❖ Memory-resident Hash Database Library
- ◆ **Control Server software**
 - ❖ Lower-layer servers (UDP-RPC) for control hardware
 - ❖ Upper-layer server (TCP-RPC) for accelerator equipment
 - ❖ Read-only Information on Distributed Shared Memory
 - ❖ Works redundantly on multiple servers
- ◆ **Client Applications**
 - ❖ Established applications in C language with RPC
 - ❖ Many of the beam operation software in scripting language,
 - ✧ Tcl/Tk
 - ✧ SADscript/Tk

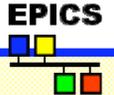




Why EPICS in my case

- ◆ We made too much effort on duplicate development on many control systems
- ◆ Our goal is to achieve high performance in the accelerator and the physics experiments
- ◆ Reuse of available resources is preferable
- ◆ Devices in Linac have been modernized, and development of EPICS device supports became possible
- ◆ Anyway we need interface to down-stream accelerators esp. KEKB
- ◆ Want to merge several archive formats in Linac
- ◆ May expect (?) man-power from other groups
- ◆ May contribute to world-wide EPICS collaboration

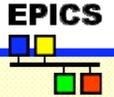




Building EPICS Gateway

- ◆ **Common Control System at the Top (of Linac and Ring)**
 - ❖ Needs too much resources
- ◆ **Port EPICS onto our VME/OS9-LynxOS (1994)**
 - ❖ Failed to get support/budget for LynxOS at Linac
 - ❖ (EPICS Maintenance with an unsupported Platform ?)
- ◆ **Special Gateway Software, which interfaces to both the Linac Controls and EPICS IOCs as a Client**
 - ❖ Built to ensure the feasibility at 1995
- ◆ **Portable Channel Access Server**
 - ❖ Implemented with EPICS 3.12 and being used on HP-UX since 1996
 - ❖ It is being used for several application software including Alarm display
- ◆ **Software IOC with 3.14**
 - ❖ Being used and being extended on Linux since 2003





Use of Existing EPICS IOC (Gateway IOC)

◆ Software availability

- ❖ Portable Channel Access Server was not available at around 1995

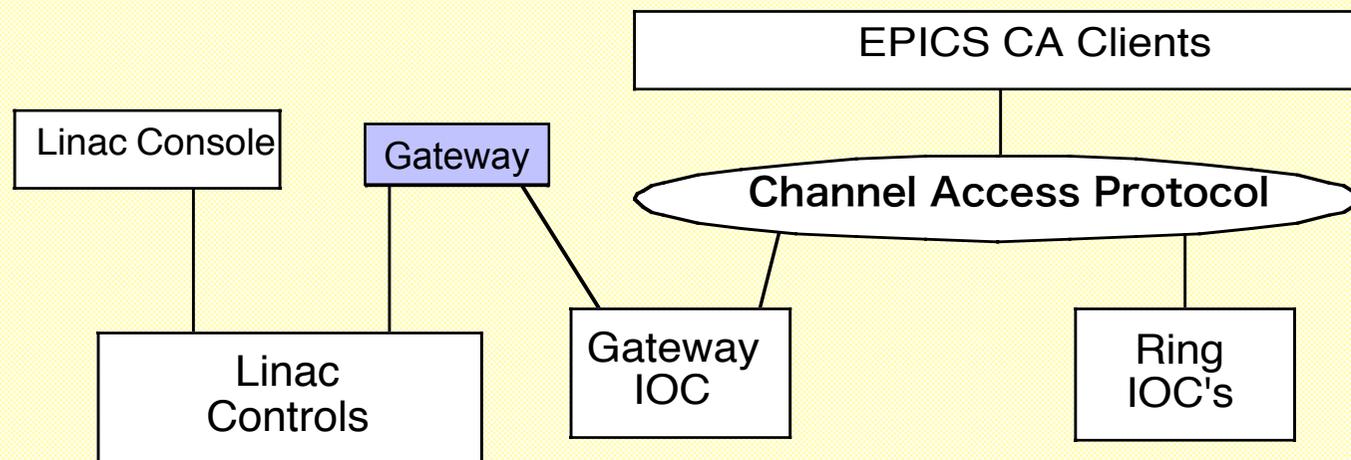
◆ Channel Access Server Emulation with Available Software Components

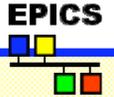
- ❖ New gateway software which is clients to the both Linac and EPICS, and group of EPICS soft records

- ❖ Real-time Operation is possible both ways using Monitors

◆ Tested for Magnet Controls

- ❖ MEDM panels were written





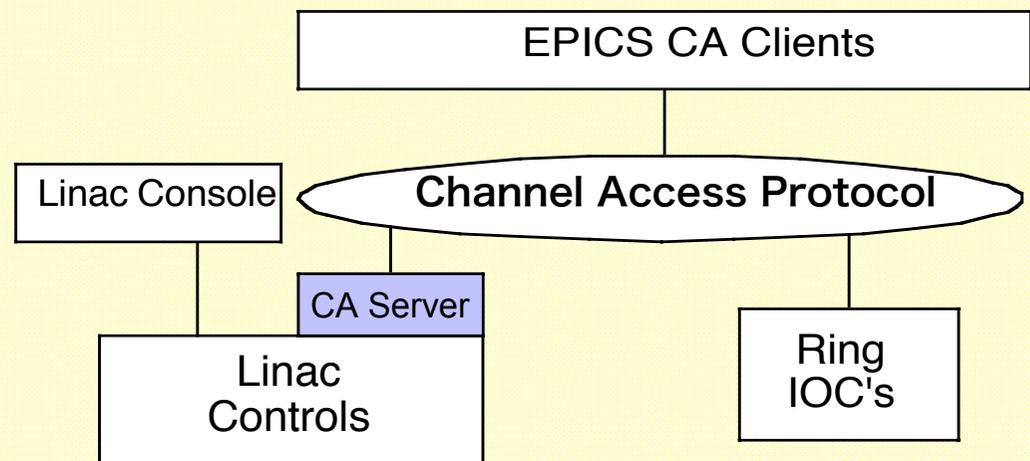
Portable Channel Access Server (PCAS)

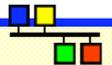
◆ Protocol Conversion

- ❖ Client to Linac Controls with Home-grown RPC and Cache Memory, Interface to Upper-level Servers (not directly to Lower-level Hardware Servers)
- ❖ Server to EPICS environment, with some Name wrapping

◆ Implemented for Linac in 1996-

- ❖ for Magnets, RF, Beam Instrumentations
- ❖ >4000 Records are available
- ❖ Write-access Possible, normally Read-only
- ❖ Still used for KEBB Unified Alarm, Operation Status, etc.

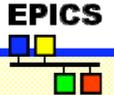




Soft IOC

- ◆ **IOCcore is available on Unix in EPICS 3.14**
 - ❖ We have Tru64unix, Linux, HP-UX
- ◆ **Simple**
 - ❖ IOCcore hides the complexity of Channel Access, etc
 - ❖ We design the device support to Upper-level Linac Servers, as we access to hardware in normal IOC
- ◆ **All standard EPICS facilities are available**
 - ❖ Alarms, Operation Limits, Links, Periodic processing, Monitors, etc.
- ◆ **Implemented for Linac on Linux since 2003**
 - ❖ For RF, Beam Instrumentation, Vacuum, etc.
 - ❖ >5000 Records are available and extending
- ◆ **Most records are archived in Channel Archiver and KBlog**
 - ❖ KBlog is used to analyze correlations between Linac/Ring
 - ❖ (Developed Java viewer of the archive at One Time)
 - ❖ Channel Archiver/Viewer from SNS is used for Linac Internals





General Comparisons

◆ Symmetry

- ❖ Gateway IOC is Symmetric between outside and inside of EPICS
 - ✧ Accessing from/to EPICS goes thru the same Gateway
- ❖ Others are (somewhat) asymmetric

◆ Name Resolution

- ❖ PCAS can resolve names dynamically (at run-time)
 - ✧ Consumes less memory (?)
- ❖ SoftIOC has to be prepared with static database
 - ✧ May be expected to give better response
 - ✧ Can be impossible for a large installations

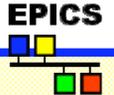
◆ Database processing and associate fields

- ❖ SoftIOC provides EPICS database Facilities like Limits, Alarms, Links, etc.
 - ✧ If we archive them, Archive Deadband is most necessary

◆ Implementation of Gateway

- ❖ SoftIOC is relatively straight forward
 - ✧ Simply adding device supports

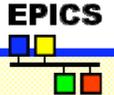




Application software

- ◆ **Most records from the Linac Soft IOC are archived both in Channel Archiver and in KBlog**
 - ❖ KBlog is used to analyze correlations between Linac/Ring
 - ❖ (Developing Java viewer of the archive)
- ◆ **KEKB Alarm is connected to Linac PCAS**
 - ❖ May migrate to Linac SoftIOC
(Linac PCAS is currently based on EPICS 3.12)
- ◆ **Some other applications utilize PCAS as well**
 - ❖ (Many others access Linac Controls directly now)
- ◆ **Small number of Records are going thru Gateway IOC, historically**





Performance

◆ EPICS Gateway and Channel Archiver

❖ are Running on Linux 2.4.20 (Redhat) with Intel Xeon 2.4GHz and Memory of 2GB

✧ About 10% of CPU usage

✧ Monitors/Archives all of ~2200 Channels (partial in Kblog)

✧ Can process 5400~6600 Channel Access Requests over Network

❖ Archive size is about 400MB/day (300MB/day in Kblog)

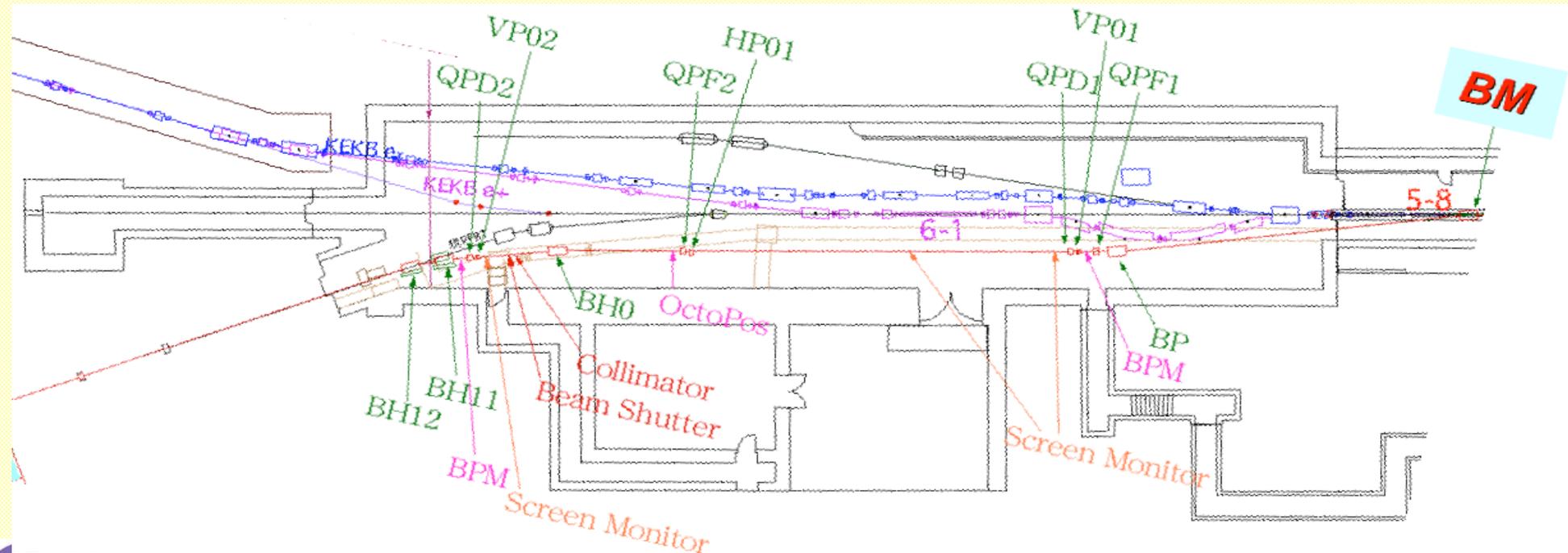
✧ Both Channel Archiver and KBlog collect Data

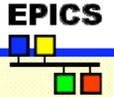




Linac Upgrade Overview

- ◆ **Fast Beam-switch (pulse-by-pulse) will be Carried out.**
- ◆ **Upgrade would be Carried in 3 Phases**
 - ❖ **Phase-I: Construction of New PF-BT Line Summer 2005**
 - ❖ **Phase-II: Simultaneous Injection between KEKB e⁻ and PF e⁻**
 - ❖ **Phase-III: Simultaneous Injection including KEKB e⁺ (,PF-AR)**
- ◆ **Control / Timing Systems will be upgraded during Phases**

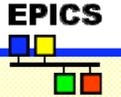




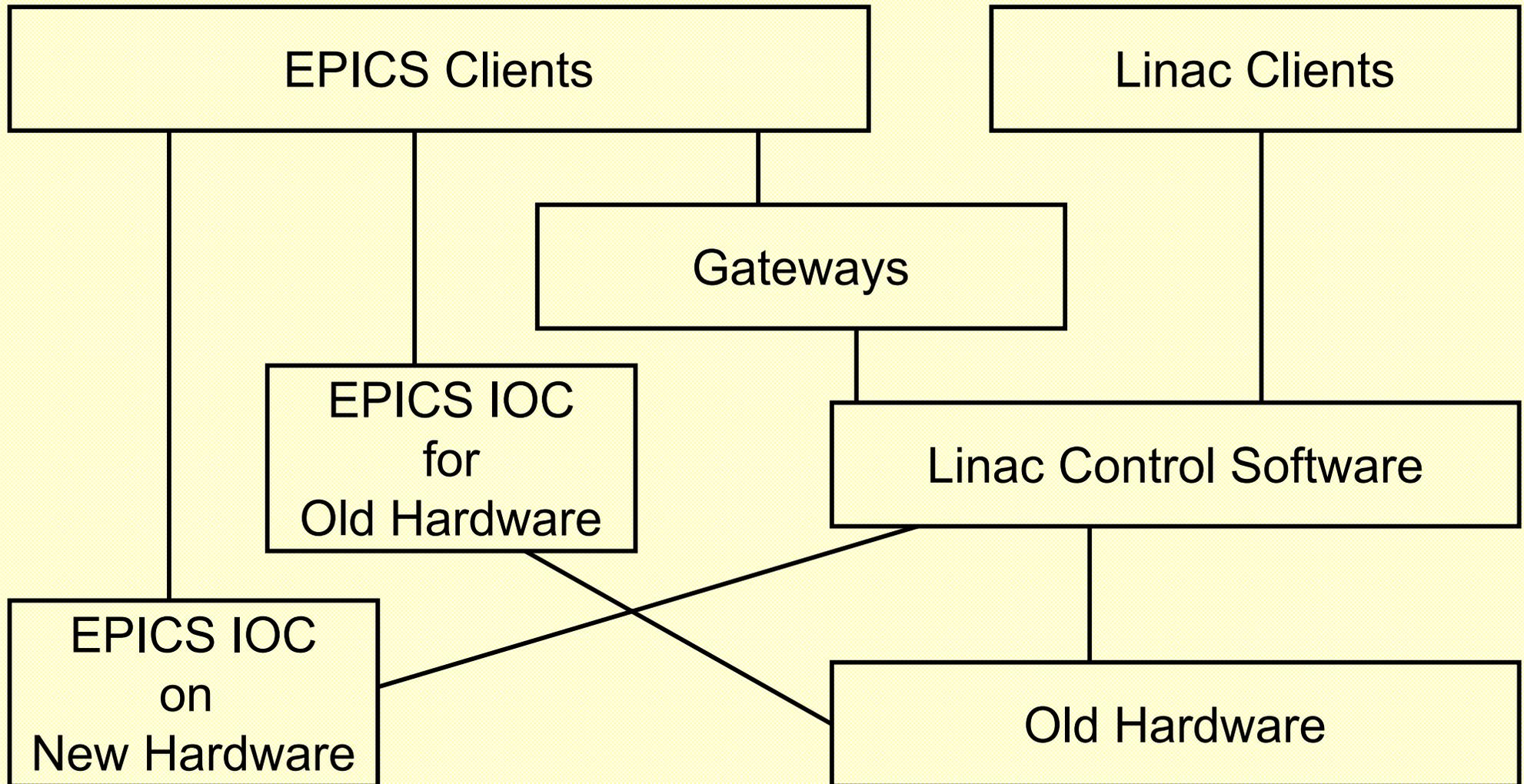
Controls Upgrade

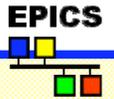
- ◆ Stimulated us to Utilize Hardware IOC
- ◆ VME/IOC with Timing Event System
 - ❖ Will Simplify the System
- ◆ BPM Read-out with Tektronix DPO7104
 - ❖ Will Improve the Performance
- ◆ Both Systems will Utilize





Interim Mixed Architecture





Timing Event System

◆ Present Timing System

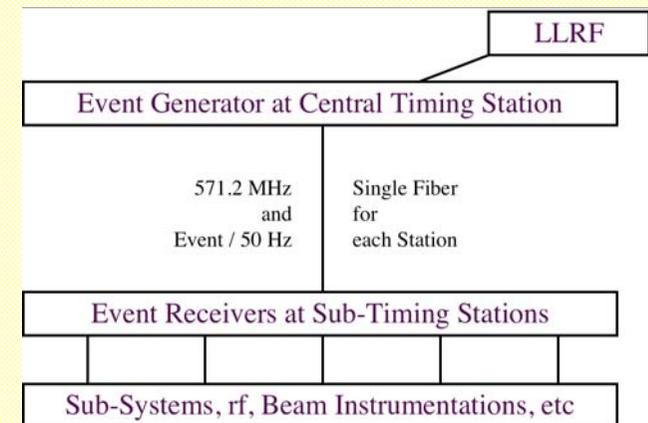
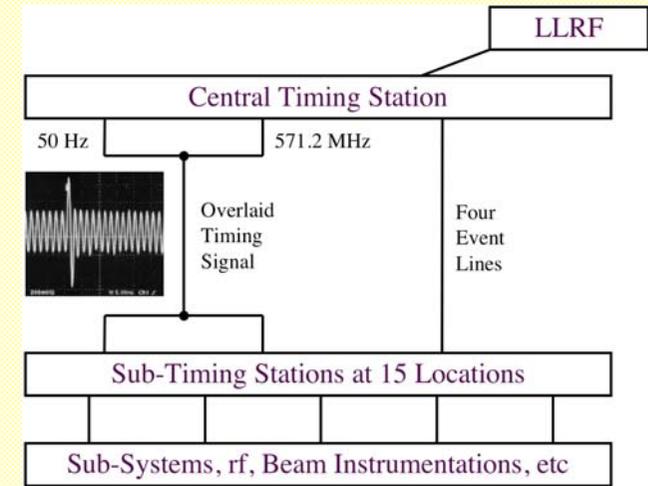
- ❖ Provides ~3pico-second Timings to ~150 Devices
- ❖ Only 4 Events can be Distinguished
- ❖ VME(x6) and CAMAC(x10)

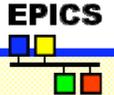
◆ Diamond Event System

- ❖ Single Fiber can Transfer Clock, Delayed-Timings, Events (256), Data Buffers (2k-bytes)

◆ New IOC

- ❖ MVME5500
- ❖ RTEMS (developed at BNL)
 - ✧ (May migrate to VxWorks if KEKB upgrades Vxworks)
- ❖ EPICS Driver/Device Support from SLS/Diamond/SLAC/LANL

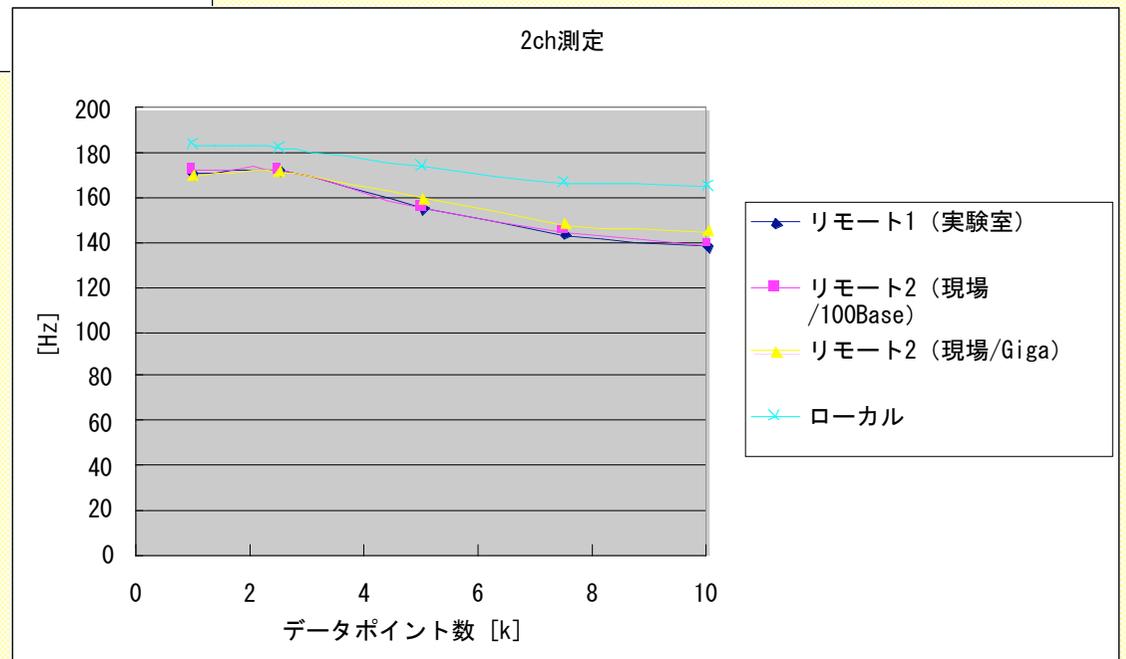
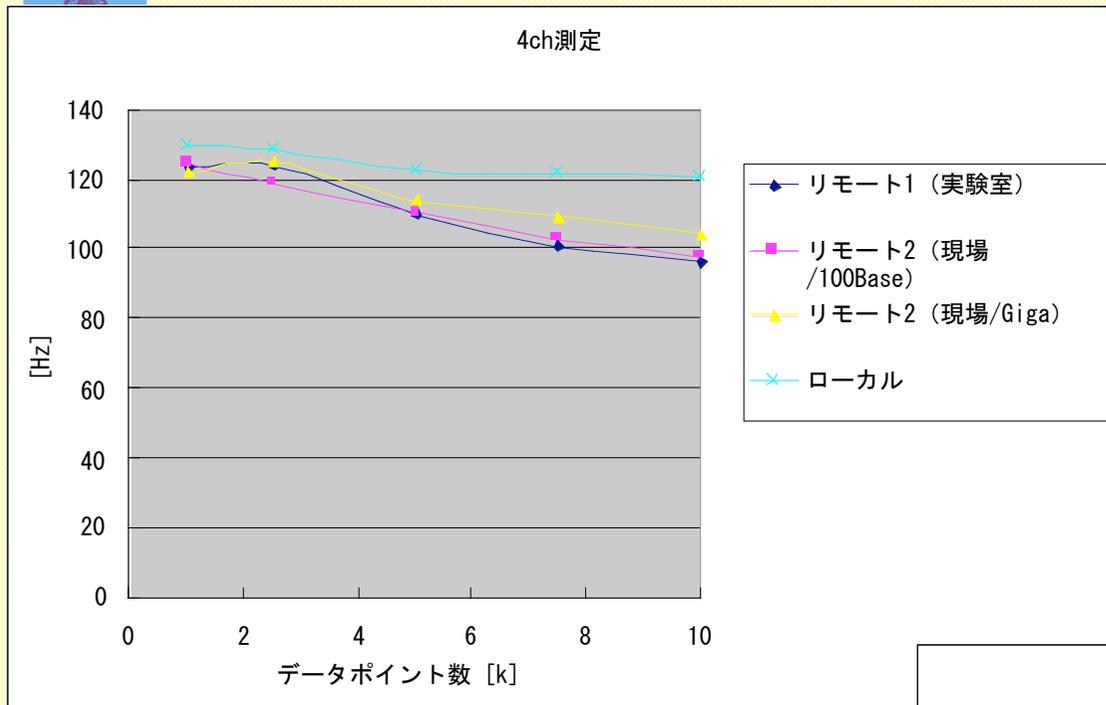
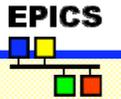




BPM Read-out

- ◆ **Current System Utilize 20 Oscilloscopes for 100 BPMs**
- ◆ **50-Hz Read-out is Required**
 - ❖ **For Fast Beam Feedback, Better Resolution, etc**
- ◆ **Evaluated Many Solutions Last 5 Years**
- ◆ **Tektronix DPO7104 Satisfies Us**
 - ❖ **10k-byte x 4ch in 100Hz**
 - ❖ **About One Data Loss in 10 Million Samples at 50Hz**
 - ❖ **Windows XP/Pentium 3GHz Enables Preprocessing of the Data with ~30 coefficients per BPM**
 - ❖ **Visual Studio 2005 with Cygwin Tools for the Oscilloscope/EPICS Development**

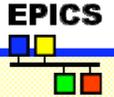






Operation





KEKB Commissioning Groups

◆ Formation of Commissioning Group (KCG)

❖ Linac Commissioning (LCG)

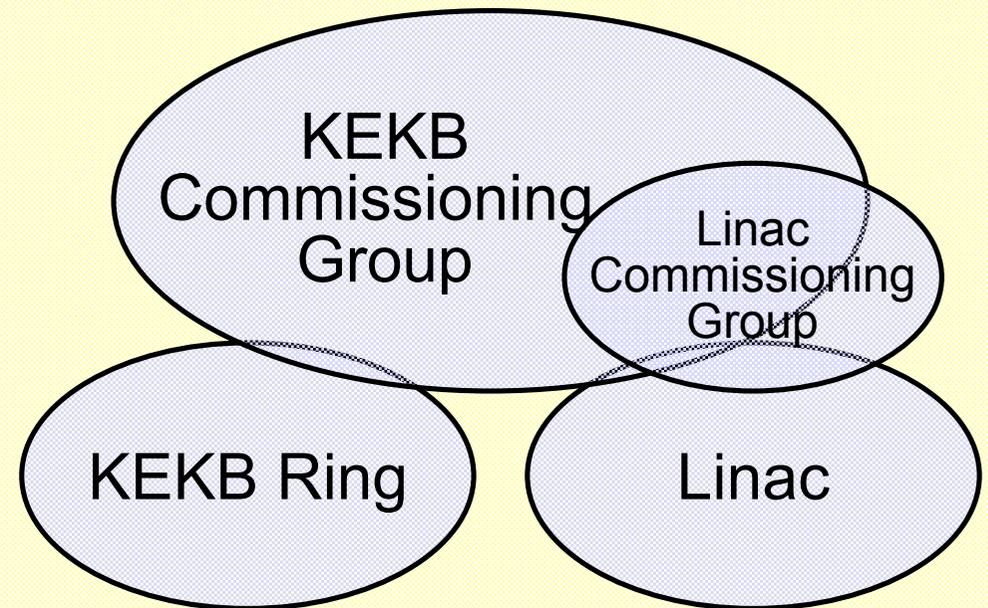
- ❖ 7 from Linac
- ❖ ~10 from Ring

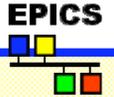
❖ KEBB Ring Commissioning Group (KCG)

- ❖ All LCG
- ❖ ~20 from Ring
- ❖ Several from Detector (BCG)

❖ Commissioning software base was formed during Linac Commissioning (1997~)

Tcl/Tk, SAD/Tk, Python/Tk





SADScript

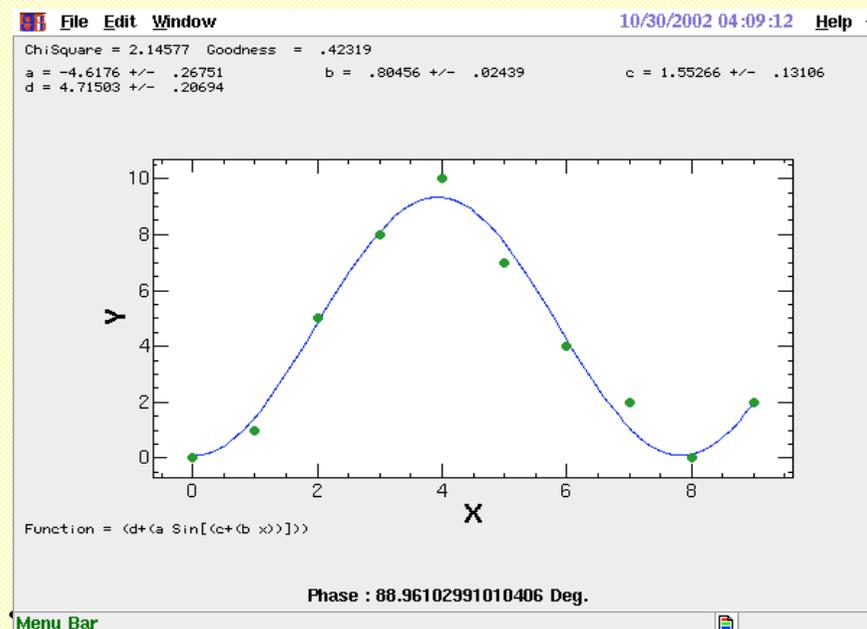
- ◆ **Mathematica-like Language**
 - ❖ **Not Real Symbolic Manipulation (Fast)**
 - ❖ **EPICS CA (Synchronous and Asynchronous)**
CaRead/CaWrite[], CaMonitor[], etc.
 - ❖ **(Oracle Database)**
 - ❖ **Tk Widget**
 - ❖ **Canvas Draw and Plot**
 - ❖ **KBFrame on top of Tk**
 - ❖ **Data Processing (Fit, FFT, ...)**
 - ❖ **Inter-Process Communication (Exec, Pipe, etc)**
System[], OpenRead/Write[], BidirectionalPipe[], etc.
 - ❖ **Greek Letter**
 - ❖ **Full Accelerator Modeling Capability**
 - ❖ **Also Used for non-Accelerator Applications**



SADScript

◆ Example

```
FFS;  
w=KBMainFrame["w1",fm,Title->"t1"];  
$DisplayFunction=CanvasDrawer;  
W1=Frame[fm];  
c1=Canvas[w1,Width->600,Height->400,  
  Side->"top"];  
Canvas$Widget=c1;  
data = {{0,0}, {1,1}, {2,5}, {3,8}, {4,10}, {5,7},  
fit = FitPlot[data,a Sin[x b + c] + d, x, {a,5},{b,1},{c,1},{d,5},  
  FrameLabel->{"X","Y"}];  
phase = StringJoin["Phase : ", (c/.fit[[1]]) 180/Pi, " Deg."];  
f1=KBComponentFrame[w1,Add->{KBFText[Text->phase]}];  
TkWait[];  
Exit[];
```

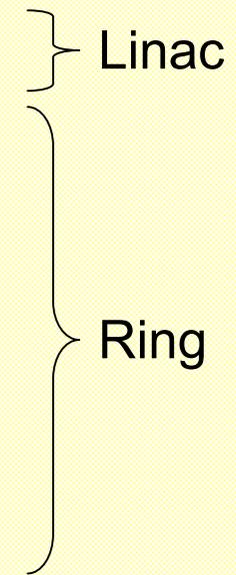




KEKB Alarm Panel

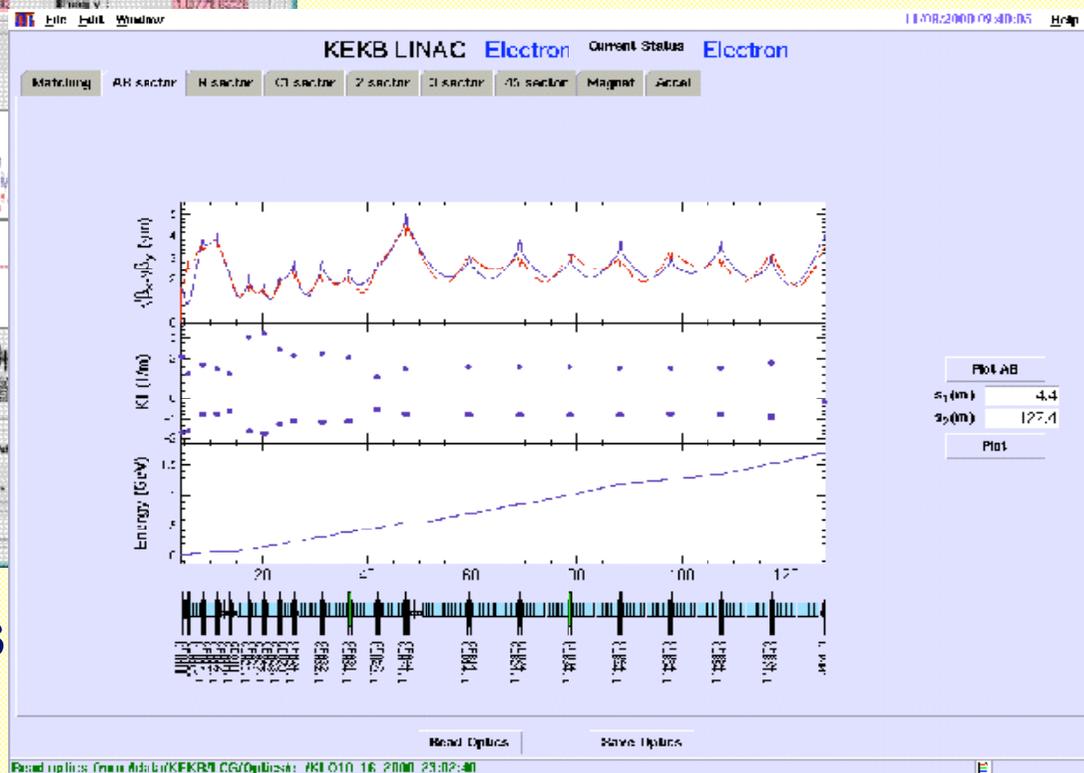
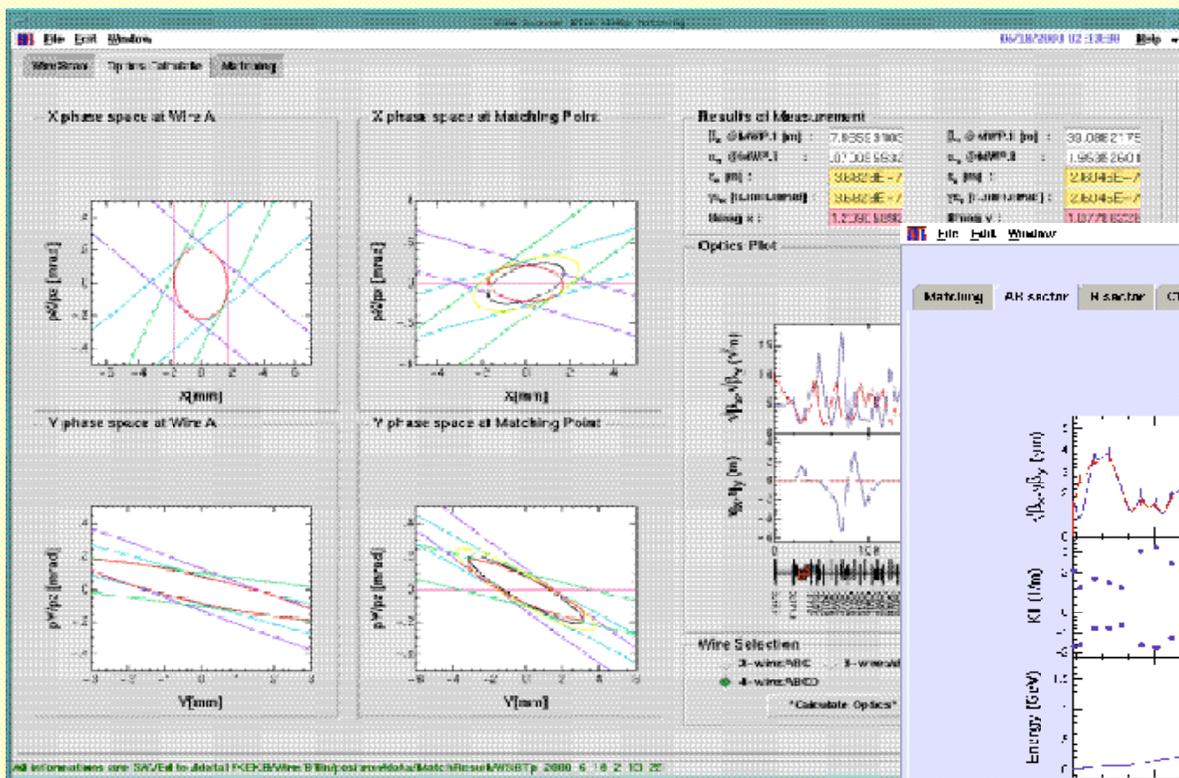
◆ KEKB Alarm Main Panel covers Linac Alarms as well. ~10,000 Records are Monitored in One Panel. Detailed alarm information/history is available in a separate panel

KEKB Alarm Status		01/05/2001 15:08:13
Linac transient		Linac(RF)
BT(p)		BT(e)
MG(LER)		MG(HER)
RF(LER)		RF(HER)
VAC(LERp)		VAC(HERe)
Safety		BM
New Alarm		01/05/2001 14:43:57
Super D10A CAVITY Pirani		
SF2NLE_1 : Magnet Water Stop		
QEAE_13 : Magnet Water Stop		
QEAP_13 : Magnet Water Stop		



Beam Optics Panels in SAD

◆ Beam Optics Matching and Optimization Panels in SADscript

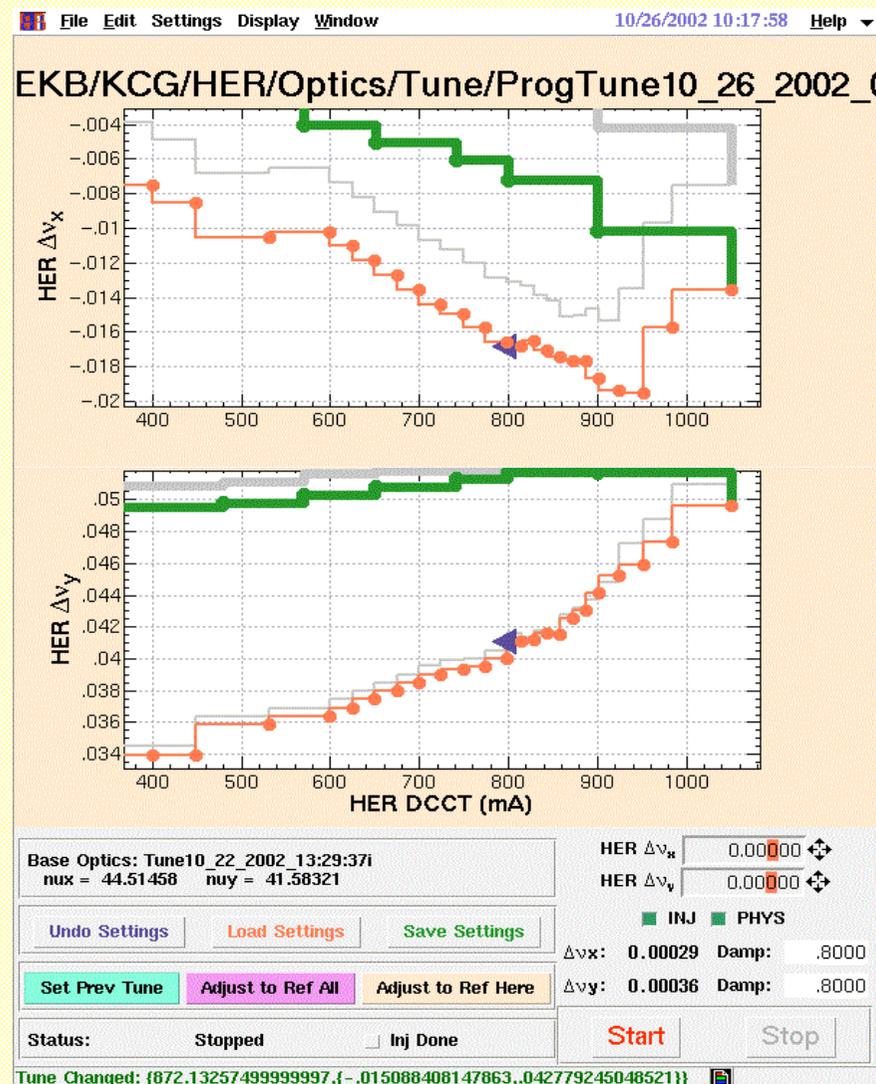
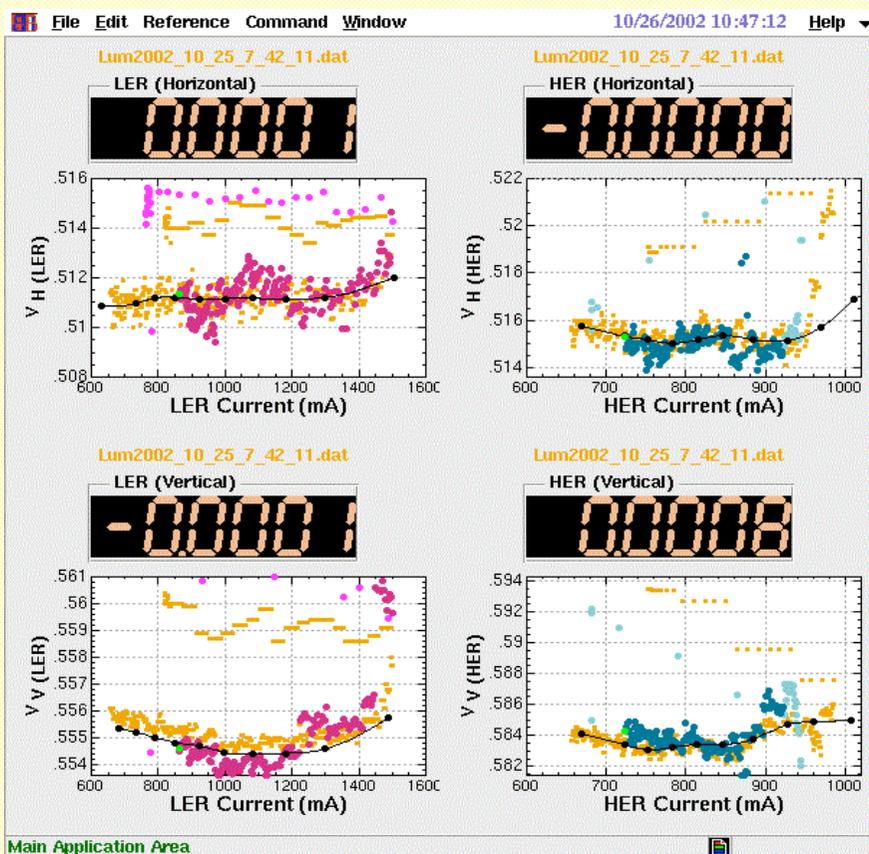


◆ Some Parameters goes thru EPICS Gateways, others directly to Linac



KEKB Operation Panel Examples

◆ Tune Measurement and Tune Changer

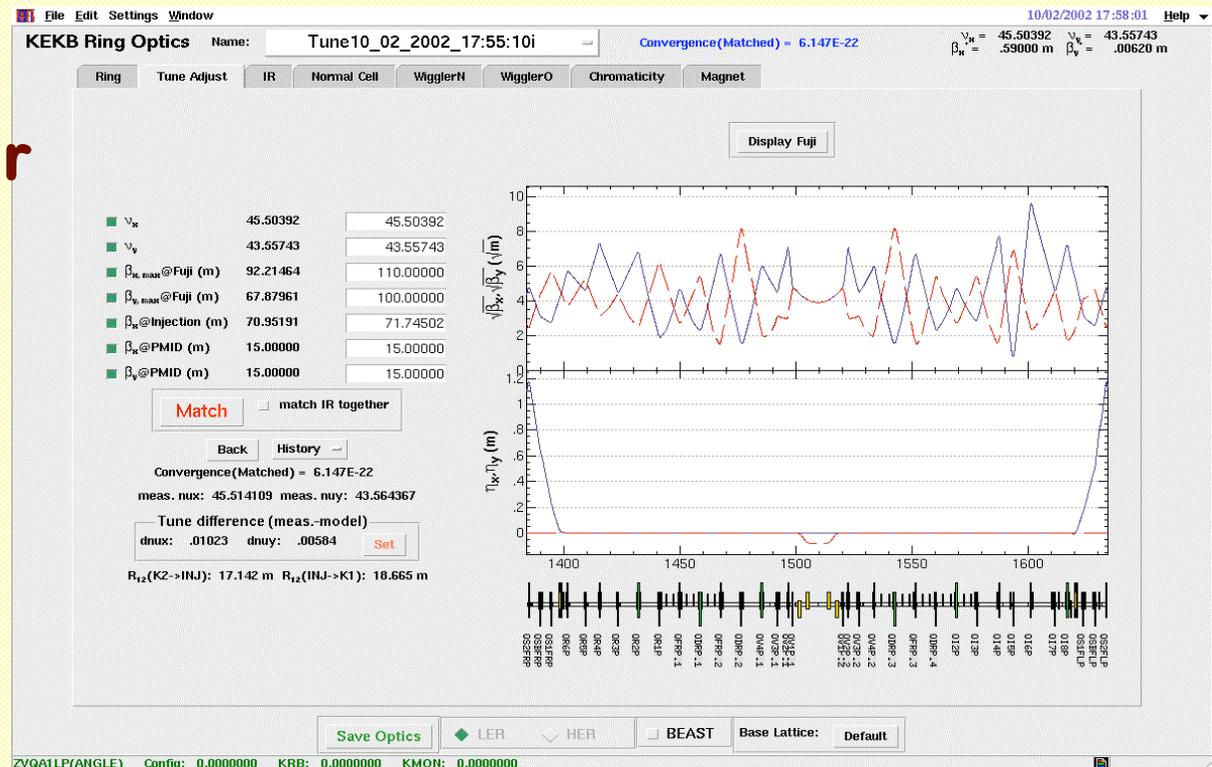


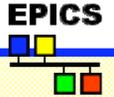


Virtual Accelerator in KEKB

◆ Tune/Optics Server

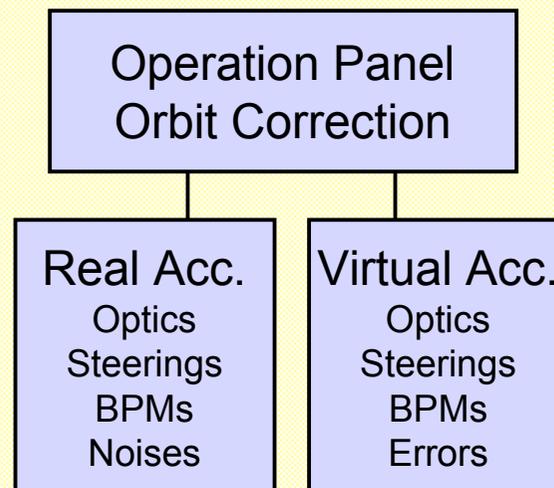
- ❖ Keep A Model of Real Accelerator
- ❖ Can Change Tune, Chromaticity, etc, on Request by Other Panels
- ❖ Act as a Virtual Accelerator

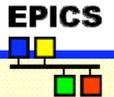




Example Virtual Accelerator

- ◆ **Virtual Accelerator may Provide**
the Both Fake Steerings and Fake BPMs
Maybe with Simulated Errors/Noises
- ◆ **Orbit Correction Application may Work**
On Those Fake Information

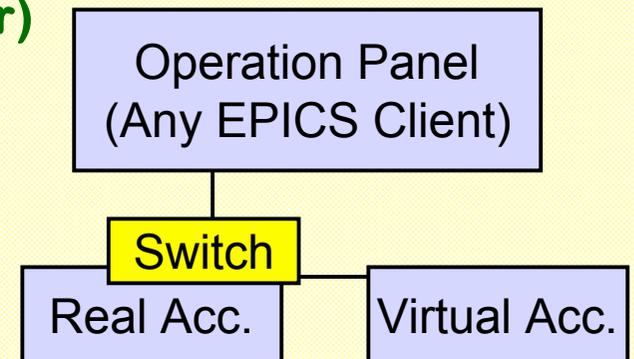
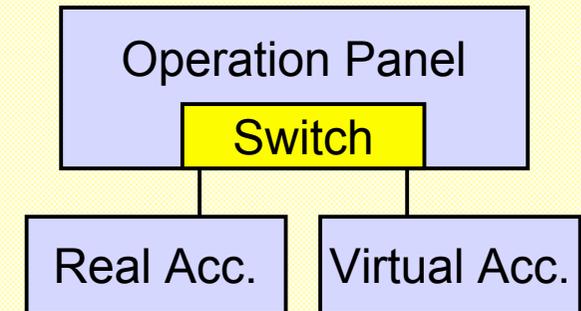
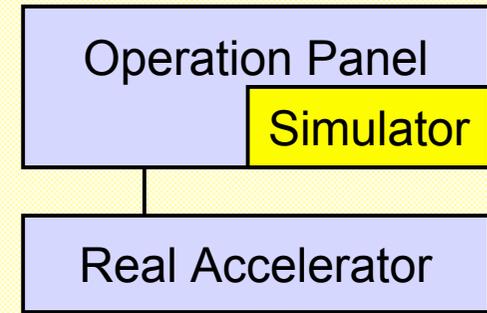




Virtual Accelerator with EPICS

◆ Fake Accelerator Implementation

- ✧ With EPICS Channel Access
- ✧ In A Single SAD Application
 - ✧ Built-in Simulator in Operation Panel
 - ✧ Only SAD Applications
- ✧ Separate Simulator (Virtual Accelerator)
 - ✧ Needs Some Switching Mechanism
- ✧ Separate Simulator (Virtual Accelerator)
 - ✧ In EPICS Semantics (EPICS Simulation Server)
 - ✧ Any Operation Panel (not only SAD)
 - ✧ SAD Simulation Server should Act as EPICS Channel Access Server



Virtual Accelerator

◆ Other Implementation Possibilities

❖ Upper Level Protocol Like

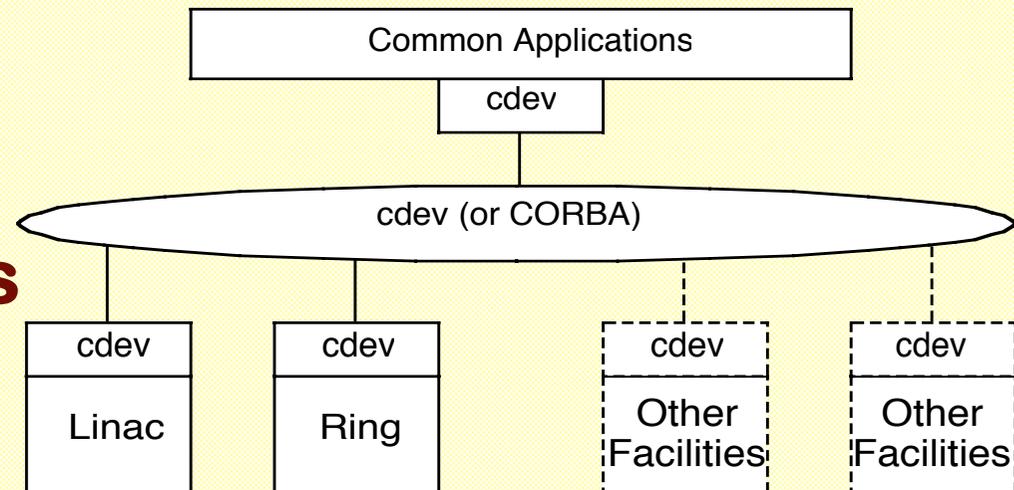
✧ CORBA

- ◆ Used in Several Lab.

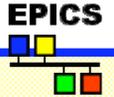
✧ Cdev

- ◆ May be Used in LHC (?)

❖ May Cover Systems Not On EPICS



❖ Not Covered in This Talk



EPICS Simulation Mode

◆ EPICS Database - Simulation Mode

A set of fields to support simulation are supplied on all hardware input records.

SIMM = YES makes this record a simulation record.

A link to a database value to put the record into simulation mode is specified

in the SIML. A non-zero number puts the record into simulation mode.

SVAL is the engineering units value used when the record is in simulation mode.

SIOL is a database location from which to fetch **SVAL** when the record is in simulation mode.

SIMS is the alarm severity of the record if it is in simulation mode.

❖ **That is, EPICS Records may have Proxy Records**





EPICS Simulation Mode

✧ **SIMM - Simulation Mode**

» This field has either the value YES or NO. By setting this field to YES, the record can be switched into simulation mode of operation. While in simulation mode, input will be obtained from SIOL instead of INP.

✧ **SIML - Simulation Mode Location**

» This field can be a constant, a database link, or a channel access link. If SIML is a database or channel access link, then SIMM is read from SIML. If SIML is a constant link then SIMM is initialized with the constant value but can be changed via dbPuts.

✧ **SVAL - Simulation Value**

» This is the record's input value, in engineering units, when the record is switched into simulation mode, i.e. when SIMM is set to YES.

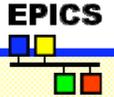
✧ **SIOL - Simulation Value Location**

» This field can be a constant, a database link, or a channel access link. If SIOL is a database or channel access link, then SVAL is read from SIOL. If SIOL is a constant link then SVAL is initialized with the constant value but can be changed via dbPuts.

✧ **SIMS - Simulation Mode Alarm Severity**

» When this record is in simulation mode, it will be put into alarm with this severity and a status of SIMM.

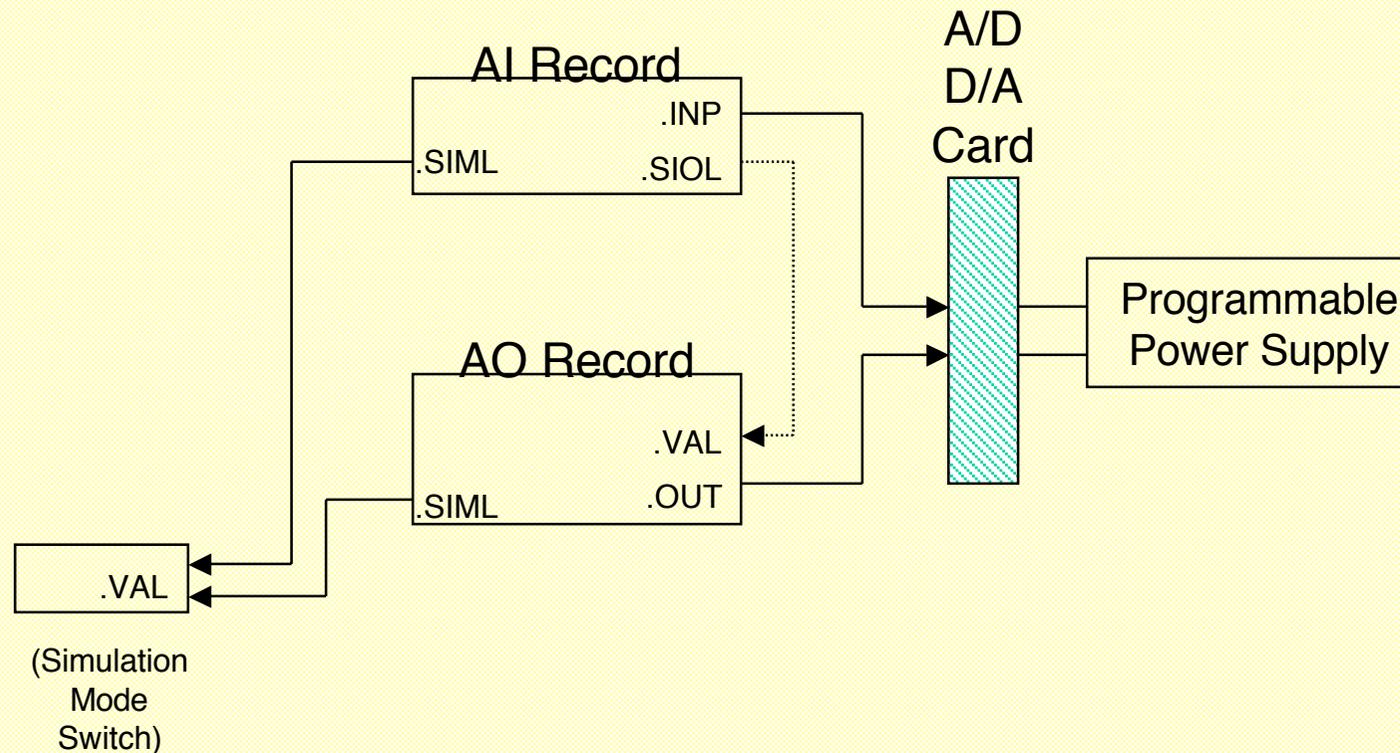


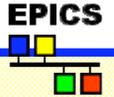


Simulation Mode

◆ EPICS Simulation Mode Simple Example

❖ Tests Logic without Hardware

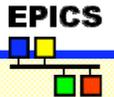




SAD as EPICS Simulator

- ◆ **Implementing a Virtual Accelerator**
- ◆ **SAD Simulator in Channel Access Server**
 - ❖ **Serves Channel Values Requested by Channels (Records) in Simulation Mode (SIOL), Acting as a Channel Access Server**
 - ❖ **Slightly more Difficult to Implement (at the First Stage)**
- ◆ **SAD Simulator in Channel Access Client**
 - ❖ **Provides Channel Values Needed by Channels (Records) in Simulation Mode (SVAL)**
 - ❖ **Easier to Implement (?)**
 - ✧ **Needs Some Studies**





Channel Access Server

◆ Using Portable Channel Access Server

- ❖ Needs Interface from Server-side Channel Access Library to SAD

✧ Written in C++

◆ Using IOC Core of EPICS-3.14 (or Later)

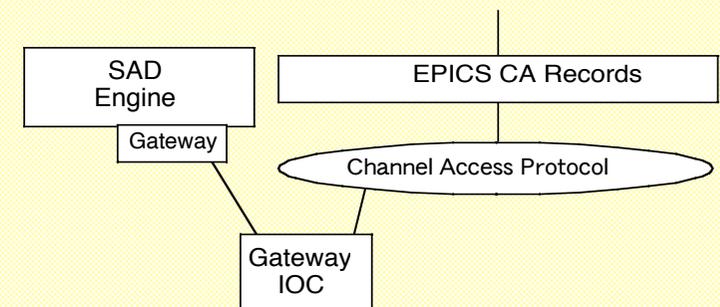
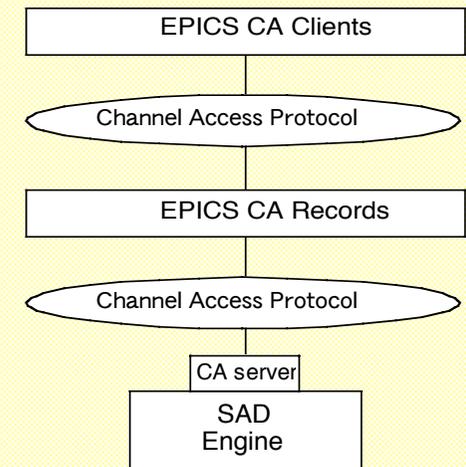
- ❖ Needs Device Support for SAD

✧ Maybe Easier

◆ Using Intermediate Soft Records

- ❖ SAD may act as an EPICS Client

✧ Maybe Easier





Summary

- ◆ **EPICS made KEKB a Great Success**
- ◆ **Other Accelerators Began to Use EPICS**
- ◆ **(We need to Catch up with Recent Versions)**
- ◆ **(ILC R&D may Start with EPICS)**
- ◆ **(More Reliability Features Needed)**
- ◆ **Transition of Controls from non-EPICS to EPICS can be Several Ways**
- ◆ **Let's Share more Experiences**





Thank you



