Hall B Slow Controls

By Hovanes Egiyan Jefferson Lab

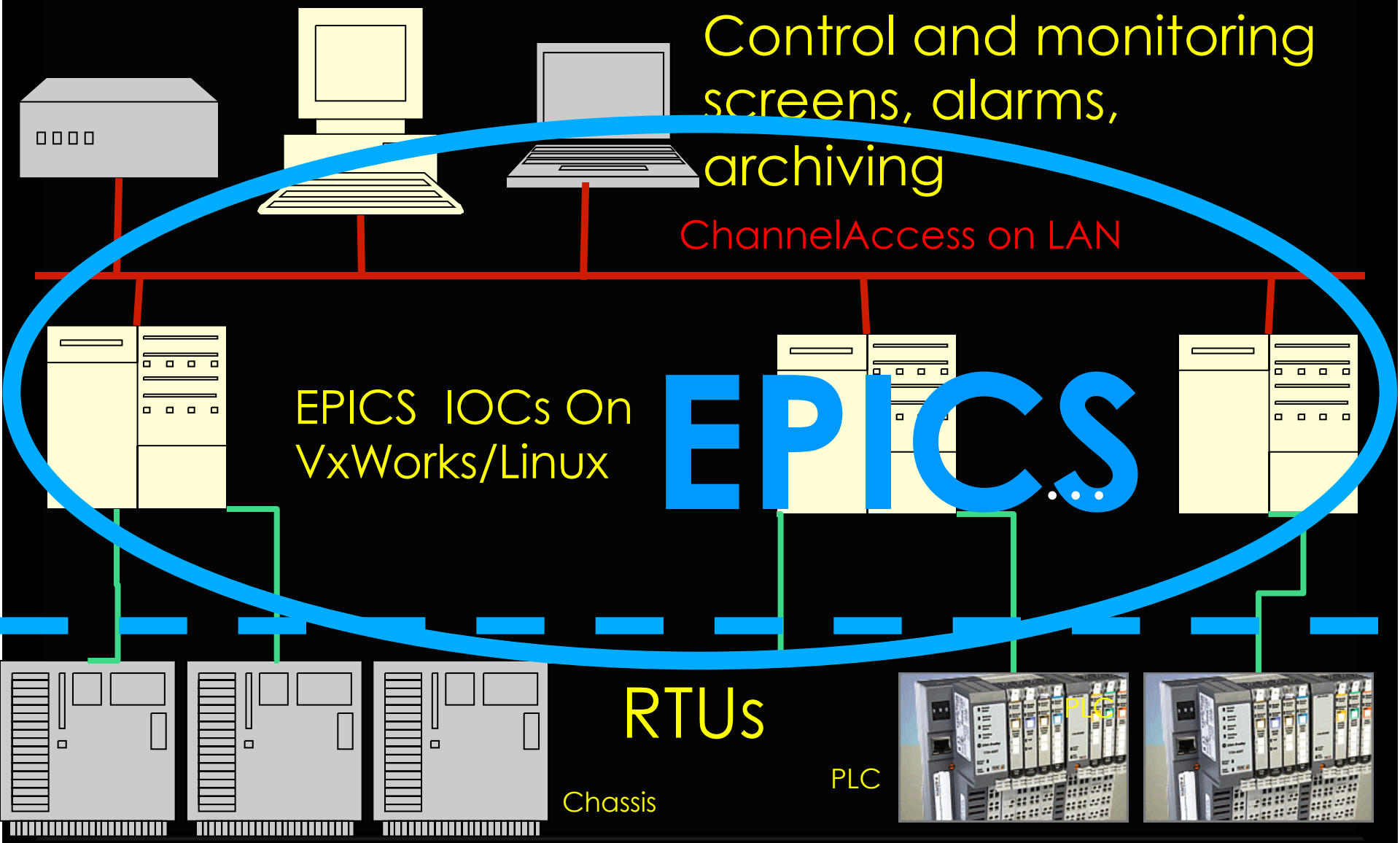
What Is This Talk About?

- Introduction to EPICS
- EPICS applications in today's Hall B
- Brief discussion HPS needs
- Summary

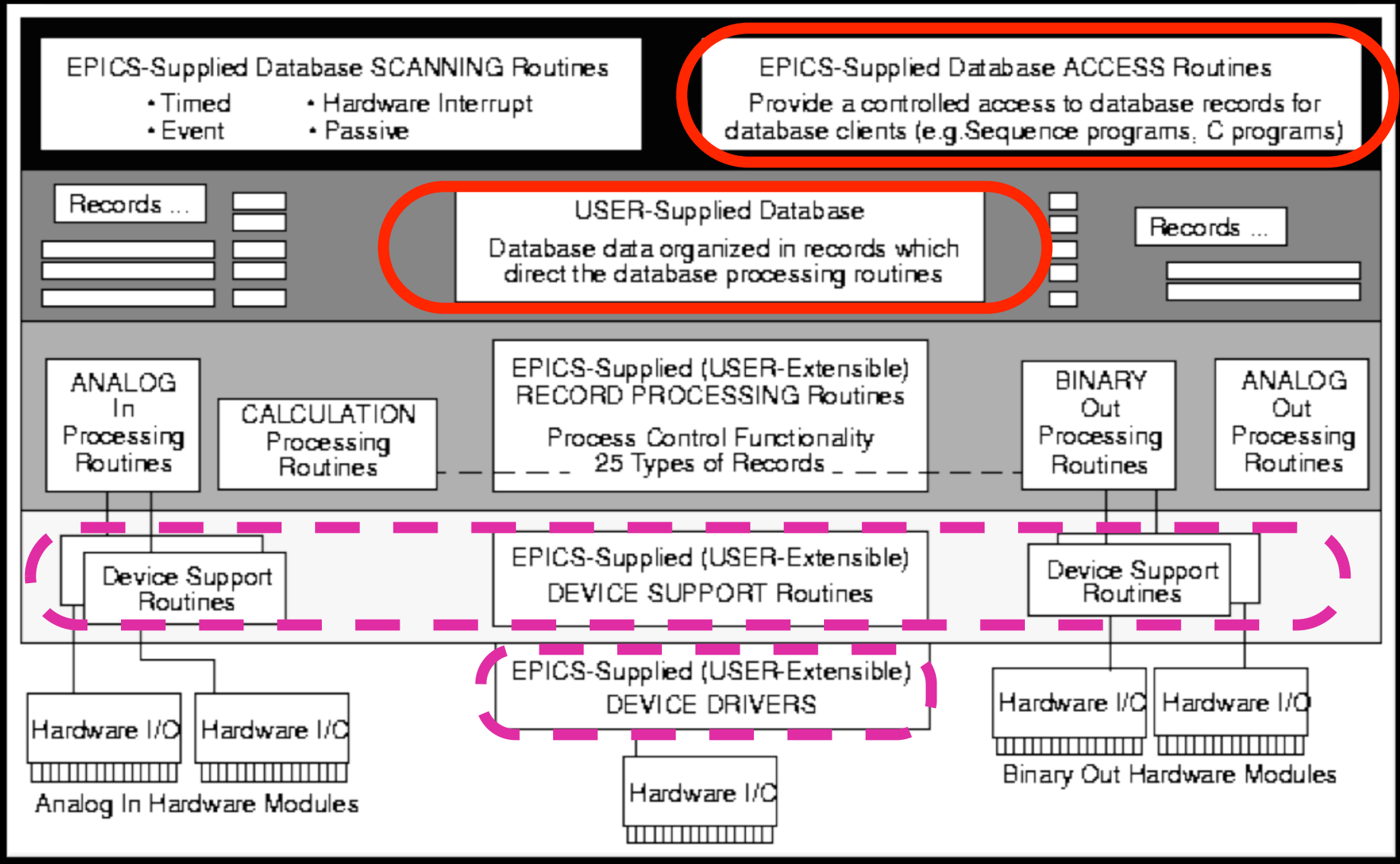
What is EPICS?

- Experimental Physics and Industrial Control System
- Free Open Source software based on C with a large user base
  - Main controls framework of CEBAF
- Server/Client model:
  - Input/Output Controller (IOC) serves variables over Ethernet using ChannelAccess protocol.
  - Clients on different hosts communicate with IOC(s) displaying, modifying, archiving values.
- Applications are usually written in form of **EPICS record** database
  - Each EPICS record type has a variety of fields, like *VAL*, *SCAN*, *STAT*.
  - Each record type has a set of functionalities, hence function block programming
  - Individual EPICS record instances (PVs) can have different processing
    - Periodic, S/W Event, Passive, H/W Interrupt
- EPICS community provides various tools, like display management, alarm handling, archiving of variables.

What EPICS does



How EPICS Software Works



EPICS in Hall B

Almost all of the EPICS IOCs in the halls are on VME chassis
- VxWorks operating system running on MVME 2306

softIOCs running on Linux can be setup on computers in the counting house

A couple of applications utilized embedded Linux
- Can use PC104 bus for I/O

LabView-to-EPICS interface is also used
- Targets, IC temperatures

Main EPICS Extensions in Hall B

MEDM for display management

Really old, MCC uses EDM instead

StripTool is used for making strip-charts

EPICS **Alarm Handler** for alarms framework

BURT package is used for backing-up and restoring

Some Applications in Hall B

Motion control

Harps, collimator, pair spectrometer target

Temperature monitoring

DVCS temperature monitoring

High Voltage control (CAEN SY1525)

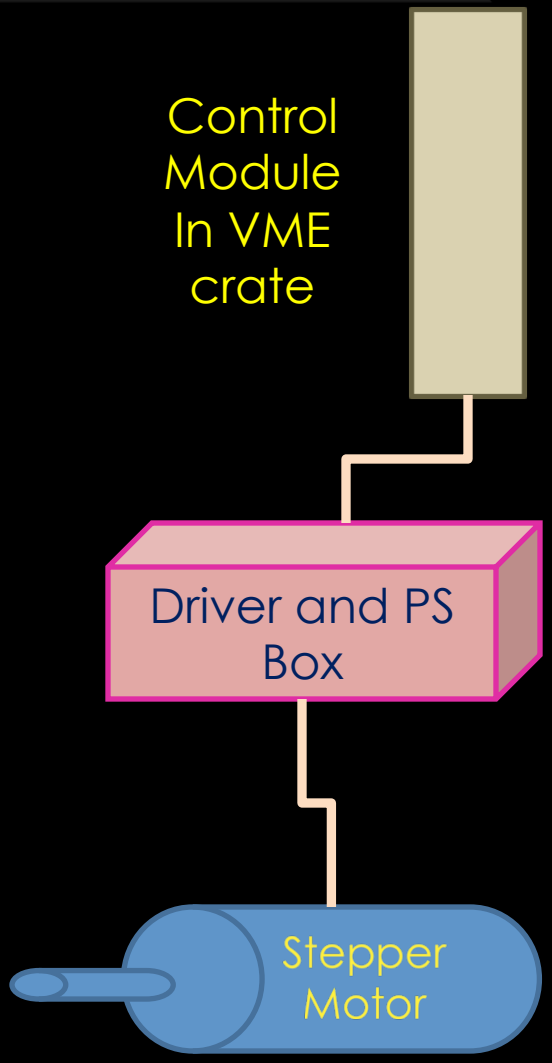
Beamline, IC, EC, TOF, DC

Monitoring of scalers (Struck SIS3801)

Main feedback from detector components

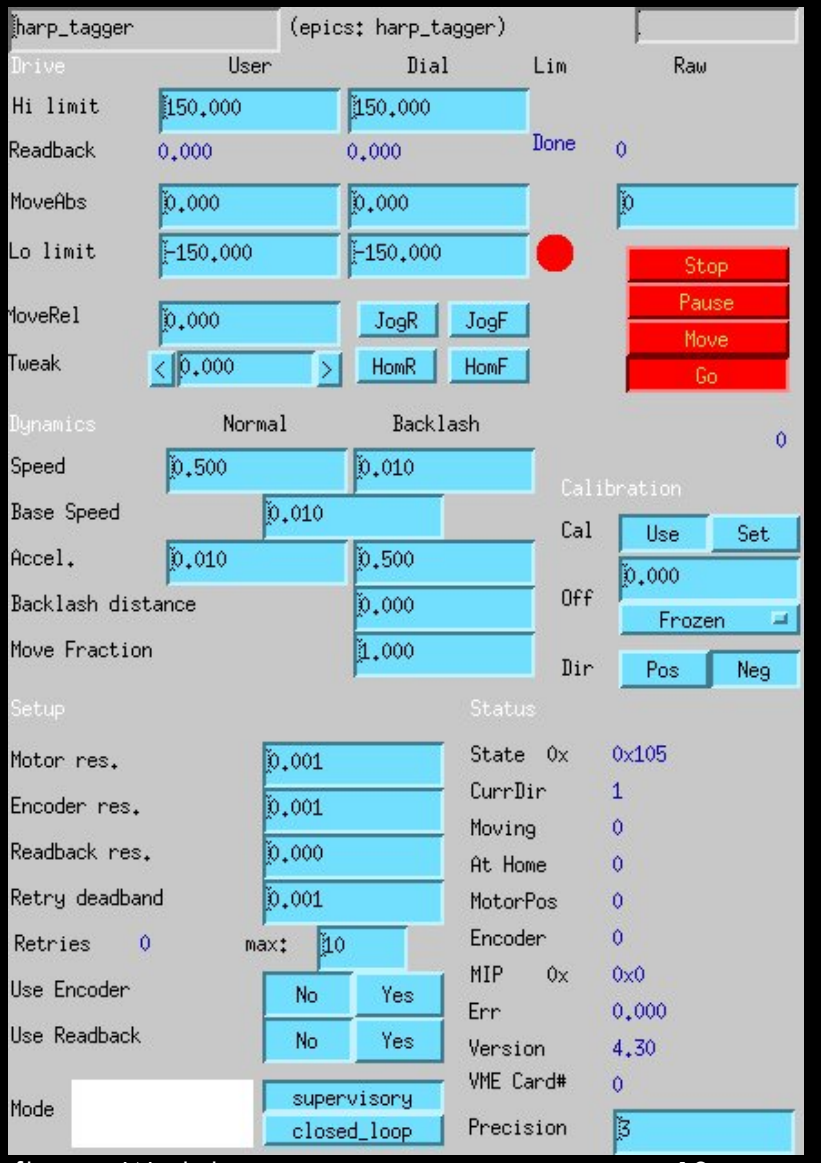
Stepper Motors

- The control modules from OMS has EPICS interface.
  - Record type ***motor***
- We need to define one record of type ***motor*** per stepper motor.
- An application needs to be written if the motor is expected to do any automated motion
  - Create EPICS database or state code to set the appropriate fields of the ***motor*** record at appropriate times.



Motor GUI

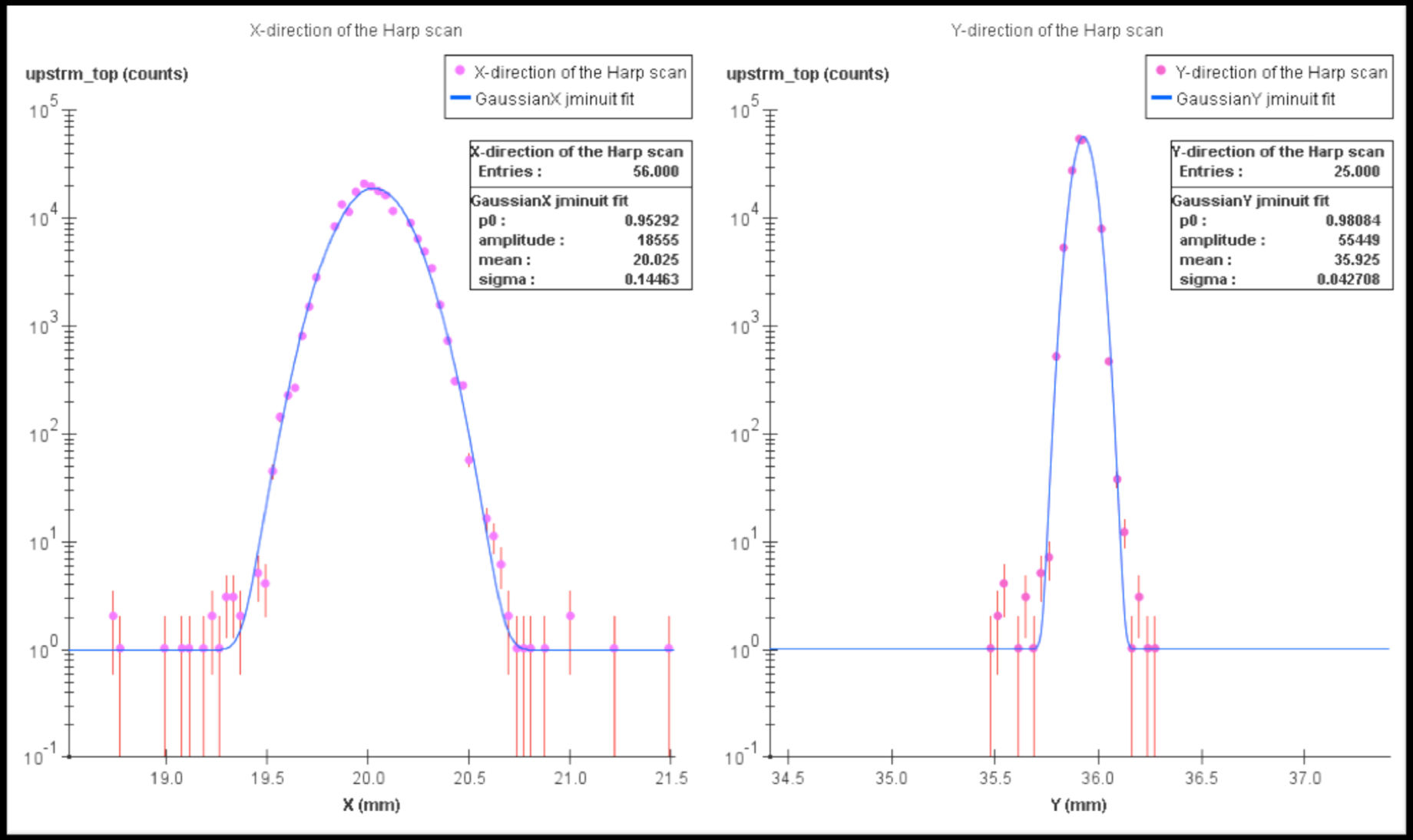
- All Motor GUIs in Hall B look similar.
- Ambiguity in the units of the motor interface
  - Both inches and mm are used.
- Should try to keep it this way.



Harp Scans

- A harp is moving at ~0.5mm/s speed across the beamline
  - The control modules for motors are in a VME crate.
- Scalers attached to the beam halo detectors are continuously read out.
  - In Hall B scaler readout happens on VME bus.
- Every few millisecond both motor position and scaler readings are written to the disk.
- The scaler readout and harp motion needs to be synchronized.
  - The level of synchronization depends on the speed of the harp and required precision in the beam position.
  - Having both motors and scalers and motor controller on the same VME bus is sufficient.
  - If these are in different places, then the motor control module needs to send out strobe signals for scaler latching.

Scan Result


X-direction of the Harp scan
upstrm_top (counts)
X-direction of the Harp scan
GaussianX jminuit fit
X-direction of the Harp scan
Entries : 56.000
GaussianX jminuit fit
p0 : 0.95292
amplitude : 18555
mean : 20.025
sigma : 0.14463
X (mm)
Y-direction of the Harp scan
upstrm_top (counts)
Y-direction of the Harp scan
GaussianY jminuit fit
Y-direction of the Harp scan
Entries : 25.000
GaussianY jminuit fit
p0 : 0.98084
amplitude : 55449
mean : 35.925
sigma : 0.042708
Y (mm)

IC Temperature Sensors

- Hall B uses NI hardware for IC temperature monitoring
  - FieldPoint cFP-20xx network module
  - FieldPoint cFP-RTD-124 temperature sensors
- The readout is done using LabView
  - Usually requires a standalone computer
- Using LabView module EPICS variable are regularly updated
  - In principle works
  - Not the most reliable way of doing this
- Alternatively, we could use direct EPICS readout.
  - Analog input through XYCOM-560 modules used in Hall B.
  - There are some problems upgrading to newer version of VxWorks and EPICS .
  - EPICS driver may need some work.
  - Can find another EPICS I/O module, there are options.
  - Krister Bruhwel is the Hall B engineering contact for controls.

Request

- Krister Bruhwel, Nerses Gevorgyan and myself will work on integrating the HPS and existing Hall B control systems.
  - We are familiar with Hall B setup, but unfamiliar with HPS requirements.

- Need input from detector components
  - Description of the hardware and its functionality.
  - Parameters to monitor and control, and how.
  - Existing software/firmware description.
  - Parameters that need archiving
  - Alarm conditions.
  - Software interlocks.
  - Backup/restore requirements.

Summary

EPICS is a very flexible and capable framework with a large user base, but the learning curve for EPICS can be steep.

Hall B has a slow controls framework based on EPICS, it may get upgraded for 12 GeV running.

Integrating a component in EPICS can take a week, or can take two months, depending of the choice and the availability of hardware.

We need adequate time before start of the run to make sure the controls software is ready.