

RICH Sbus Monitoring
Cornell University
Laboratory for Nuclear Studies

Nick Brönn
Georgia Institute of Technology

Georg Viehhauser
Ray Mountain
Syracuse University

August 7, 2000

Overview

1. The RICHSbus Component

- A C++ program that reads data from the databoards.
- Made modifications so that appropriate alarms are thrown to the Alarm Manager.
- It now writes data files to disk which can be read by mn_fit and the GUI.

2. The RICHSbus GUI

- A Java program that displays the current data from the databoards.
- Completely rewritten.
- Has a lot more features than before.
- Easy to monitor critical parameters.

A Comparison

1. Old GUI

- All in one file. This does not support the Object Oriented paradigm.
- Barely interactive
- 350 lines of code.

2. New GUI

- Expanded to support OO paradigm.
- Contains buttons that can be pressed for more information.
- Allows alarm data (i.e. nominal data and error ranges) to be easily modified via an interface.
- 2200 lines of code.

Dependencies: A Comparison

1. Old GUI:

RICHsbusGUI
JFrame (API)

2. New GUI:

RICHsbusGUI
Monitor
ADCWindow
History
Alarm
DataPoint
Canvas (API)
WindowListener (API)
ActionListener (API)
Frame (API)

So How Does the GUI Work?

1. Class RICHSbusGUI

- Main control of GUI.
- Establishes connection with Component in order to receive new data.
- Calls an instance of Class Monitor

2. Class Monitor

- Sets up the layout of the GUI.
- Displays grids of buttons that represent a cell (6 or 10 channels) of data.
- Adds listeners to the buttons.
- Colors each button according to how out-of-range the data is.

So How Does the GUI Work?

1. Class ADCWindow

- When a button on the monitor is pressed, an ADC (cell) window pops up.
- This displays the current data for each channel in the cell.
- Each channel is colored corresponding to how out-of-range its data is.
- Each channel is also a button that can be pressed to display approximately the past 24 hours of a channel.

So How Does the GUI Work?

1. Class History

- Reads files written by Component and plots the history of a specific channel.
- Displays percent off nominal data and percent variation.

2. Class DataPoint

- A data structure that holds a channel's data value and the time it was recorded.

3. Interface Alarm

- Declares nominal data values.
- Contains ranges and errors for the data.
- Specifies colors for each type of error.

Conclusion

1. Accomplished:

- Used data to fix constant-loading problem.
- Component source has been incorporated into the system.
- GUI is ready to be incorporated.

2. In the future:

- Fix memory error.
- Have shifters use GUI for monitoring RICH Sbus.