



# WIDAR Prototype Testing User Interface Software

NRAO-DRAO Face-to-Face Meeting  
April 3, 2006

Kevin Ryan



```

<baselineBoard>
  <recirculator id="x1" blockSize="2048" blockStart="0" blockEnd="127"
    timeCodeSource="data0_1">
    <internalStreamConfig id="0" stationId="110" subBandId="10" baseBandId="0"/>
    <internalStreamConfig id="1" stationId="111" subBandId="11" baseBandId="1"/>
    <internalStreamConfig id="2" stationId="112" subBandId="12" baseBandId="2"/>
    <internalStreamConfig id="3" stationId="113" subBandId="13" baseBandId="3"/>
    <state register=""/>
  </recirculator>
</baselineBoard>

```

Prototype testing will involve configuring\*  
FPGAs (and correlator chips) to various states  
and monitoring them along with dynamic  
statistics gathered by Bruce's low-level  
S/W.

This is done via the CMIB.

\*Configuring is not the same as programming  
an FPGA's 'personality' which is not discussed here.

XML  
over

Ethernet



CMIB

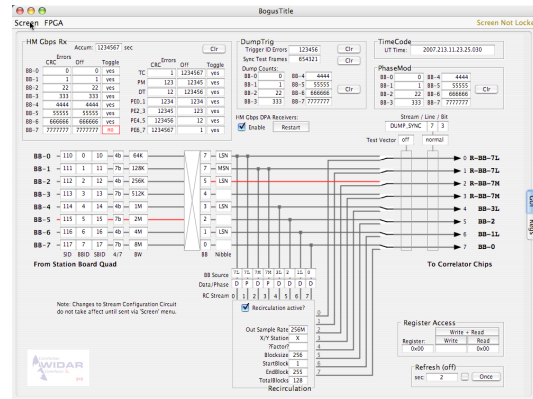
PCI Bus

PC-104+ Mezzanine Card

WIDAR MCB Bus

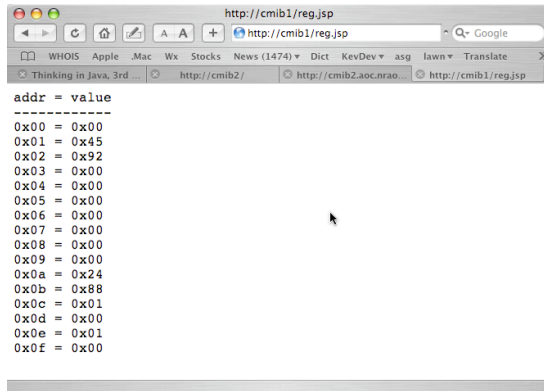
WIDAR Prototype Board Under Test

Special Purpose  
Applications  
GUIs can be general or  
specific in purpose.



Multiple ways to  
communicate.

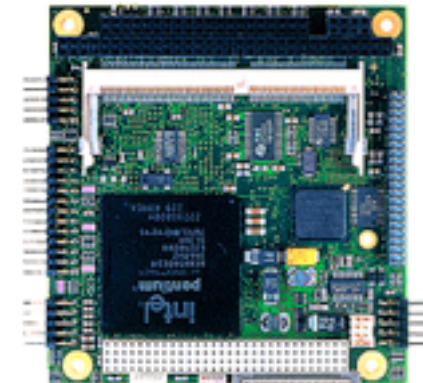
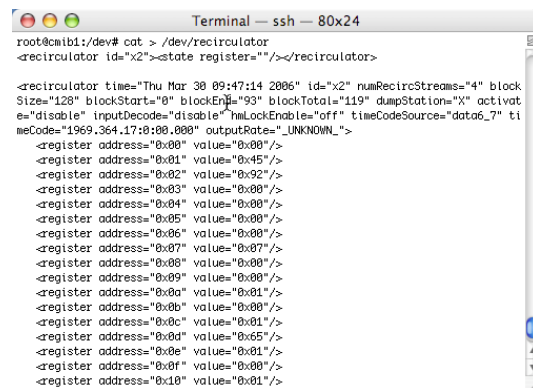
General Purpose  
Applications



XML  
over

Ethernet

command line



CMIB

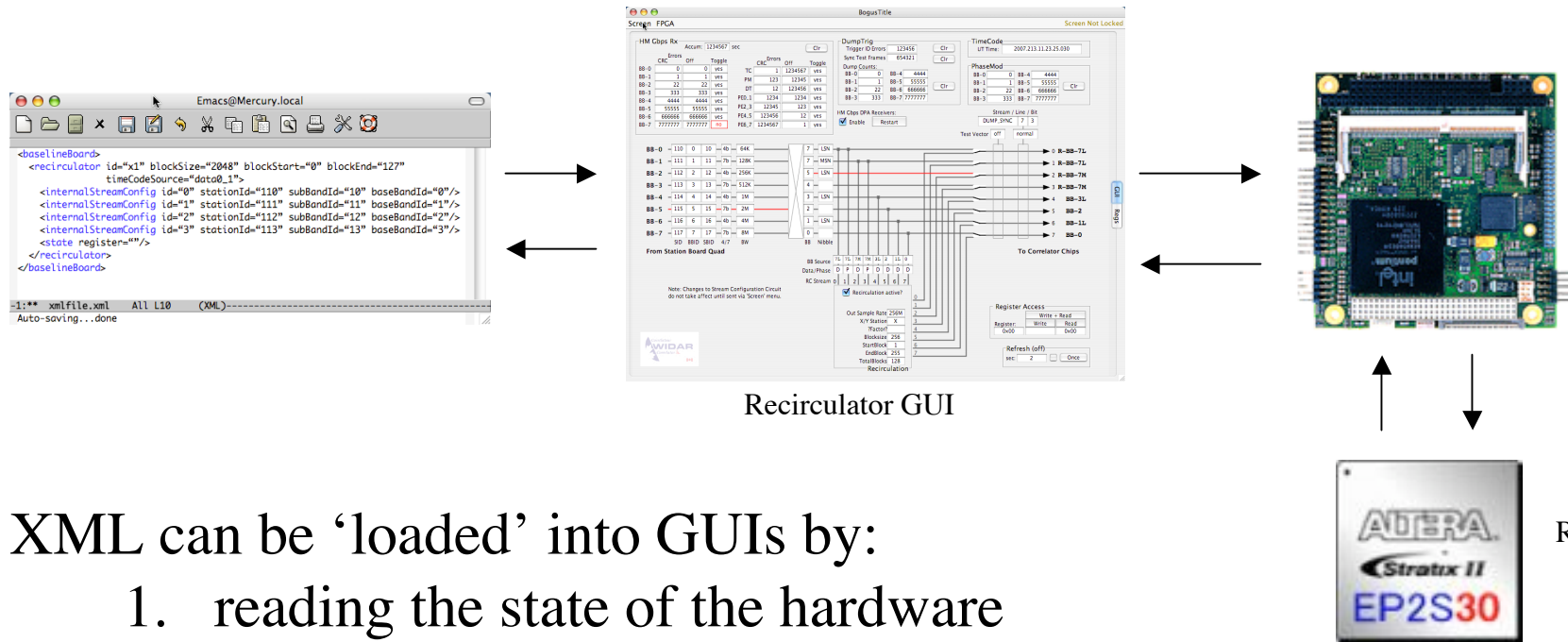
PCI Bus

PC-104 Mezzanine Card

WIDAR MCB Bus

WIDAR Prototyp Board Under Test

Special Purpose GUI's provide a 'device-centric' representation of the XML



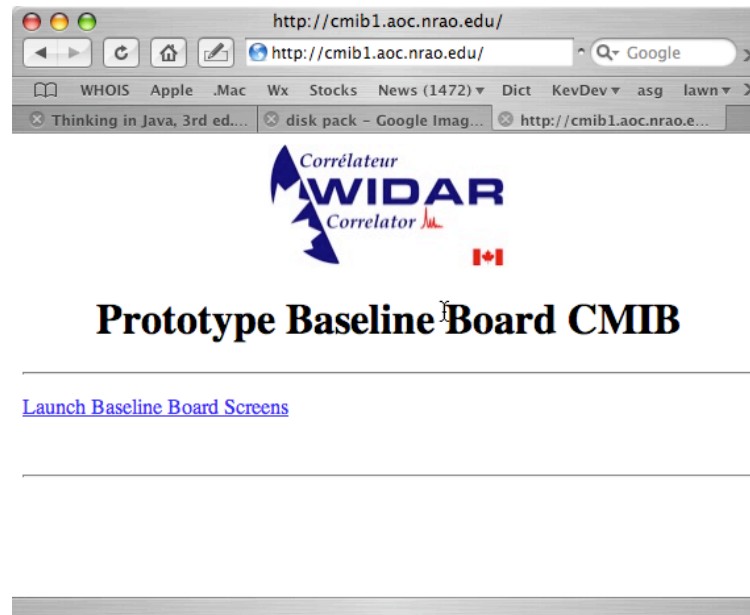
XML can be 'loaded' into GUIs by:

1. reading the state of the hardware
2. reading a configuration file
3. the operator can configure its various components by hand

GUIs can send XML representations of themselves to:

1. The H/W - the device is then configured to 'mirror' the GUI
2. A file - where the configuration can be saved for future use.

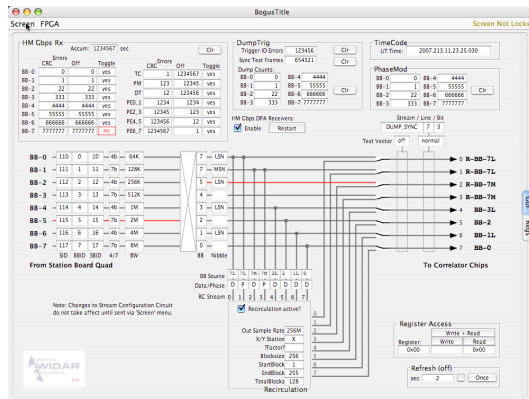
A GUI is obtained by the tester using a standard web browser pointing to the CMIB itself



The web browser and Java Web Start will get the GUI going on your local machine

JWS will

- Check your local machine for an existing version of the GUI application
- If a current version exists, it will start it
- If it is old or non-existent, a new version will be downloaded and started

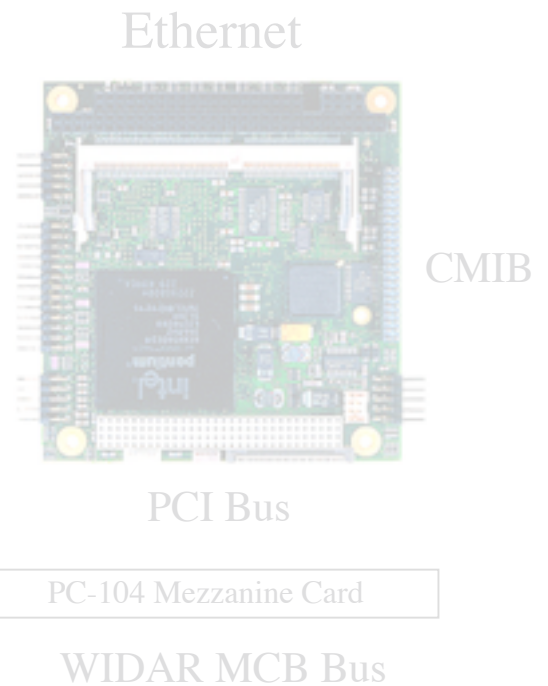
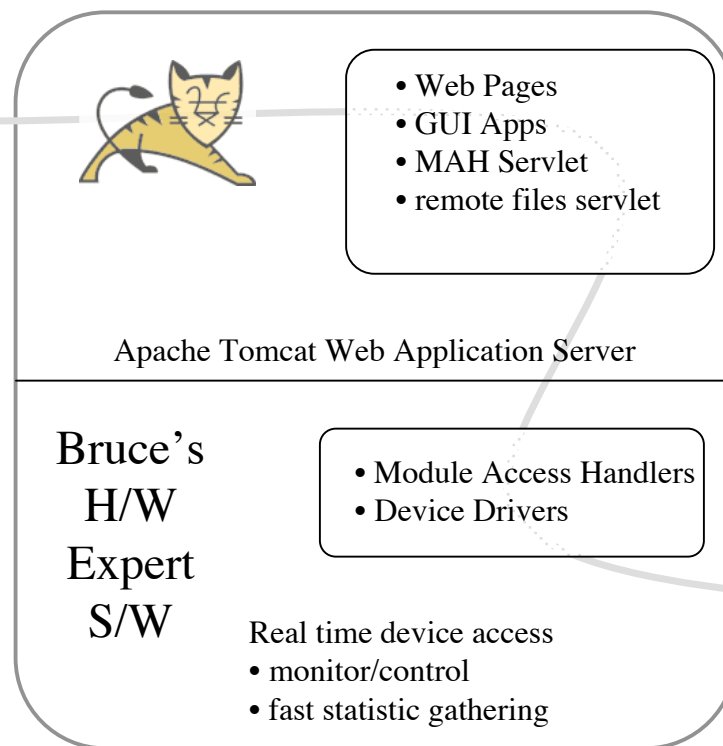


Once the GUI is running on your local machine it connects to the CMIB and you no longer use the web browser.

NRAO/DRAO proprietary S/W goes in Tomcat's *webapps* Directory

Inside each CMIB ...

GUIs use the same comm protocol as the WWW - handled for the CMIB by Tomcat



WIDAR Prototype Board Under Test

CMIBs don't have hard drives so  
how does all that stuff get in there?

CMIBs use a central file server.



Brunette loading IBM 2314 disk pack

## A Central Server

... well, actually it's a Dell  
(and Brunetteless)



NFS



- *cmibhost* in Bruce & Kevin's office
- *venus* at DRAO - for prototype testing
- ? - for OTS and full WIDAR deployment\*

- The CMIBs' Root File Systems reside on the server's hard drive and are mounted at boot-time via NFS

- A single web application directory is shared by all CMIBs (and the server itself)

\*Is it possible to have a hot-standby central server?



The webapps directory resides at:

**/export/home/cmibhost/widar/web/tomcat/webapps/ROOT**

As seen from the CMIB, this directory is mounted at:

**/opt/widar/web/tomcat/webapps/ROOT**

So browsers pointing at different CMIBs will actually be pointing to the same web site running on different machines.

- This is good since only one site will have to be maintained vs. ~300

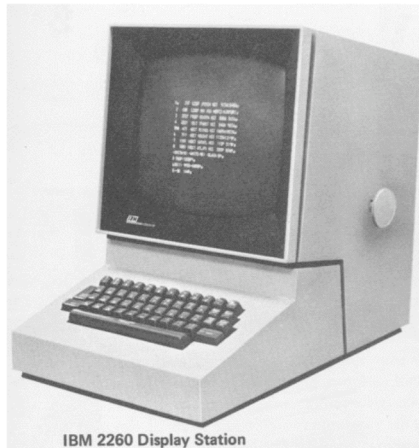
Normally when a browser points to the same web site (even when served from multiple machines) it will see the same web page.

But we don't want ours to work that way.

- We want a Station Board CMIB to show a Station Board web site, etc.

So when ...

## Ken Sowinski's Web Browser



## Points to

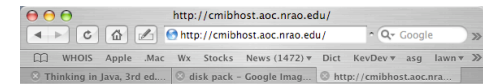
<http://cmibhost.aoc.nrao.edu>

<http://cmib1.aoc.nrao.edu>

<http://cmib2.aoc.nrao.edu>

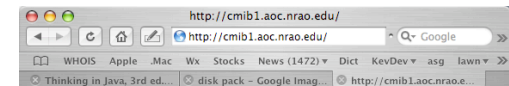
- Instead of using *index.html* for the main page, we use *index.jsp*
- JSP files can have program code in them that can obtain the server's IP address and display the appropriate page.

## It Will Show



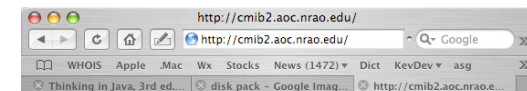
### Prototype Board Test Interface Site

[Go To Baseline Board](#)  
[Go To Station Board](#)  
[Go To Timecode Board](#)



### Prototype Baseline Board CMIB

[Launch Baseline Board Screens](#)



### WIDAR

### Prototype Timecode Generator CMIB

This page is being served from the WIDAR Timecode Generator CMIB

[Launch Timecode Generator Screens](#)

## GUI Application Deployment

- A Java ARchive 'jar' file exists for each of the different board types

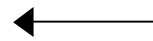
- On the server at:

`.../widar/web/tomcat/webapps/ROOT/app`

`baselineBoard.jar`

`stationBoard.jar`

`timecodeBoard.jar`



Revised GUIs are placed here and Tomcat and JWS see that they get delivered.

- The jar files contain the GUIs
- When a new version of an GUI is to be deployed, its jar file is placed in that directory and Tomcat and Java Web Start take care of the rest.
- One jar file for each board type.

## Configuration Files Deployment

- In addition to the web app each board type has its own directory
- On the server at:

.../widar/

web/...

baselineBoard

stationBoard

timecodeBoard

- To hold configuration files (and whatever)
  - For simplicity, it was decided to limit GUI access to files on the server only (rather than also on the tester's local machine).
- A RemoteFileChooser servlet is used by the GUI to access these files.
- This will (hopefully) now be demonstrated ...

On to a peek at some GUIs