

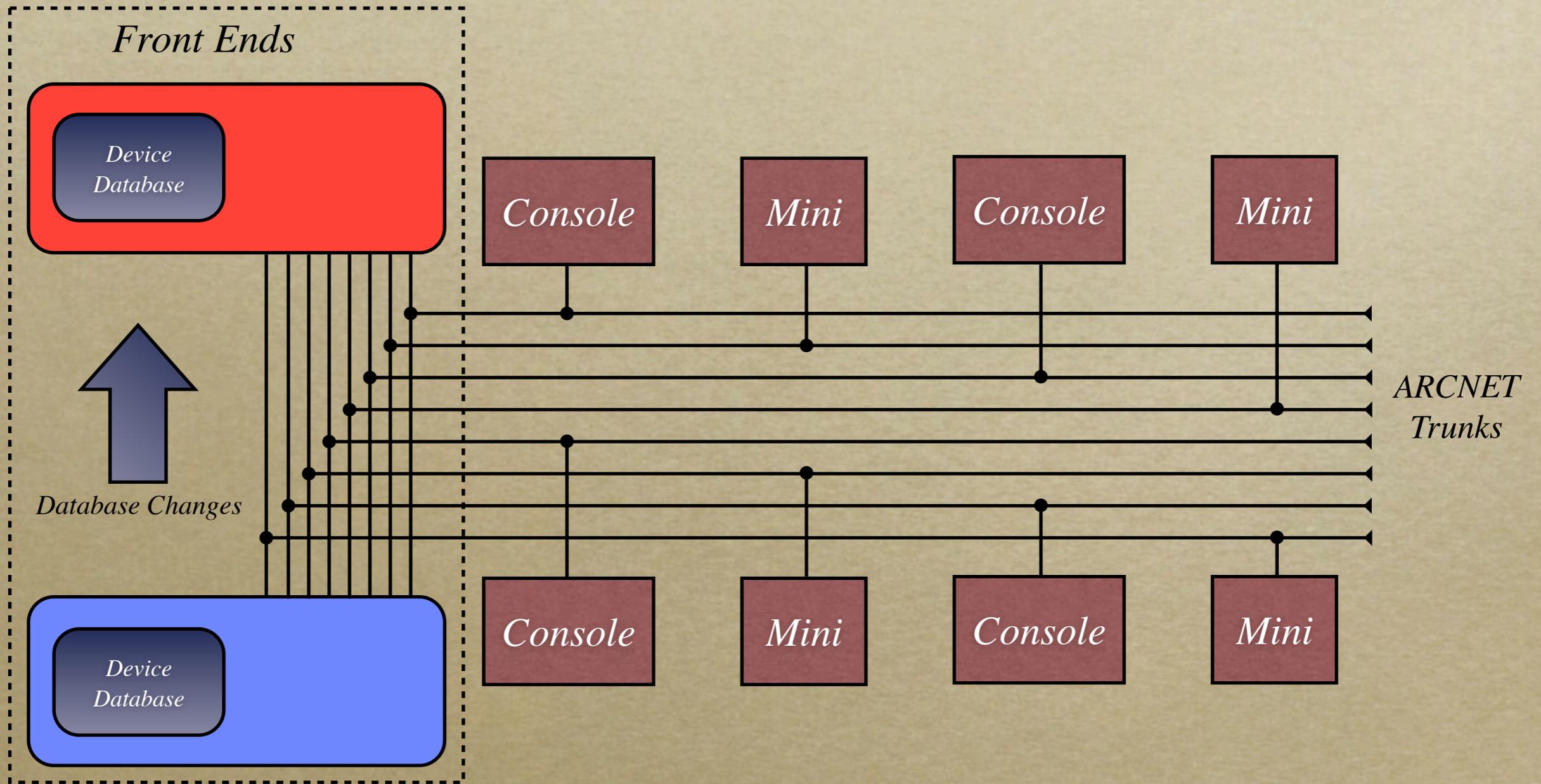
Upgrading the Fermilab Fire and Security Reporting System

Charlie King
Rich Neswold
Oct 11, 2011

FIRUS

- *Acronym for **F**ire Incident **R**eporting and **U**tility **S**ystem*
- *Uses its own private **ARCNET** network*
 - *Separate from Fermi's control system*
 - *No direct connection to the Internet*

FIRUS Topology



FIRUS Console in a Nutshell

- *Alarm acknowledgement and display*
- *Alarm logging*
- *Device database management*
- *Real-time parameter page display*
- *Data logging at multiple rates*
- *Real-time and logger plotting*
- *Synoptic picture displays*
- *Fully configurable*
- *Password protection for sensitive items*

Why Upgrade the Console?

- *Antiquated console hardware*
 - *Unable to find PC replacements*
 - *Obsolete ISA bus*
- *GEM interface*
 - *Low resolution and limited coloring*
- *MS-DOS/GEM memory issues*
 - *Program had to be split*
 - *Program changes were almost impossible*

Console Alternatives

- *GEM Emulator/Newer version of GEM*
- *Windows XP/7*
- *Linux with QT interface*
- *Java swing*
- *Apple Mac OSX*

Mac OS X Because...

- *Modern graphical interface*
- *Unix core*
- *Powerful set of free development tools*
 - *Xcode presented a learning curve but proved to be well worth it*
 - *Excellent tools for debugging*

Console Design Goals

- *Minimize user learning curve*
- *Retrofit (side by side verification)*
- *All one program*
- *Kiosk mode*
 - *Keep unprivileged users from switching away or quitting FIRUS*
- *Automatic software update*
- *Improve and add new features where needed*

Connecting to ARCNET

- *Desired Macs don't support additional hardware cards*
- *Found a USB to ARCNET bridge*
- *Wrote a network level driver thinking we could use TCP/IP*
 - *This almost worked but eventually would crash the FIRUS minis*
- *Fortunately OSX has the ability to access the USB subsystem from user level code*

Development Notes

- *Very rich and well thought out framework*
 - *Built on OPENSTEP which has been around since the days of NeXT Inc.*
- *Interface builder made it very easy to layout the user interfaces*
- *Some standard OSX features were not a good-fit for FIRUS*
 - *Don't allow ad-hoc sorting of current alarm screens*

Future FIRUS Development

- *Upgrade front-end software*
- *Remote access and mobile devices*

Front End Upgrades

- *Already have modern hardware*
- *Need a software rewrite*
 - *3 separate processes that don't communicate very well*
 - *Distributed database only in one direction*

FIRUS Erlang Front End

- *Erlang is a programming language and runtime environment*
 - *Was developed by Ericsson for use in telecommunications systems*
 - *Built-in support for concurrency, distribution and fault tolerance*
- *Distributed database is built in*
- *Erlang runtime supports many concurrent processes and message passing*
- *Minimal downtime*
 - *Upgrade software in a running system*

Remote Access/Mobile Devices

- *Currently have a web display of active alarms*
 - *Available offsite over a password protected encrypted connection*
- *Would like to develop IOS applications that do more*
 - *Developing for IOS would use the same tools as OSX development*
 - *iPad could become the new remote console*

Conclusion

- *New FIRUS console is completed and will be in use by the end of October*
- *Ready to start front-end update*
- *Looking forward to incorporating mobile devices*