Keeping track of things with MeasureD

Poul-Henning Kamp

phk@FreeBSD.org

Things I do for a living...

- “Can you make our 20 year old VOR transmitter talk SNMP ?”

NAVAIDS

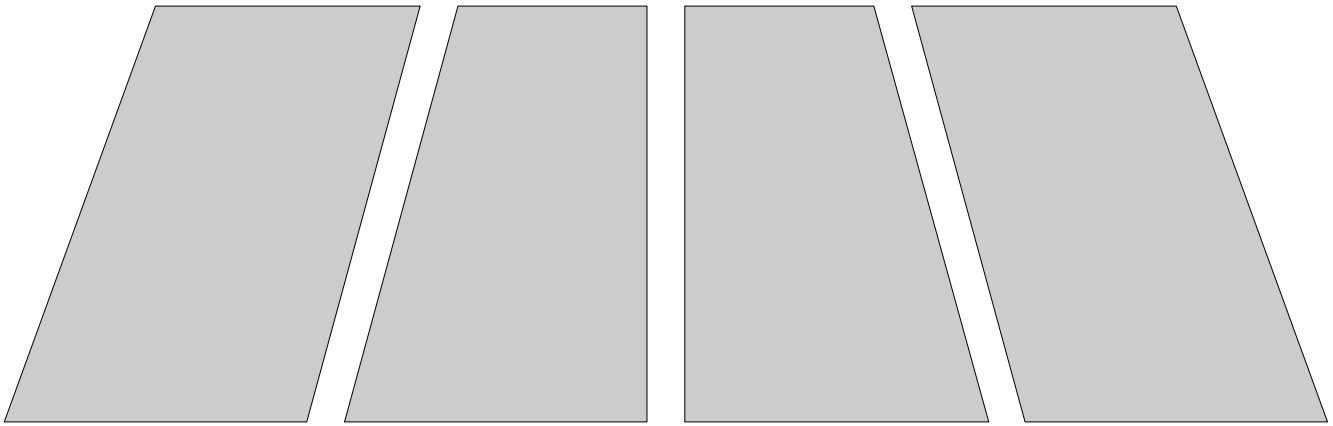
- ILS
  - Multisector beam guides plane to strip
- VOR
  - Phase difference gives compas angle
- DME
  - Ping-pong tells difference to DME
- RNAV = Improved VOR+DME
  - Backup for GPS/Galileo

There's also some other kit...

- Diesel Generator
- Diesel Tank
- Diesel engine starter battery
- Main batteries
- Air Condition
- &c

...and another thing...

- “This is part of the CAT3 landing system...”



CAT3 landing (fog), as seen from cockpit of 747

...and...

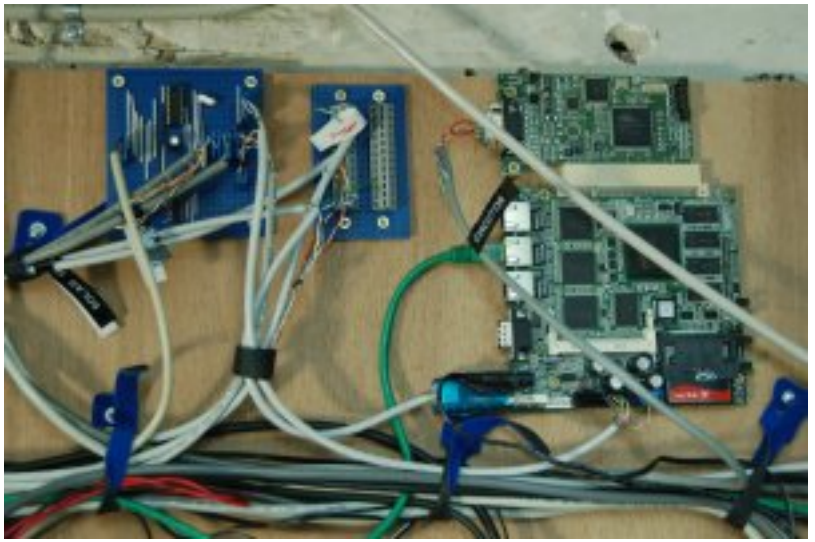
- ...a few other details, such as:
  - Must be low power (battery backup)
  - High Reliability (no rotating parts)
  - Secure
- ...the usual stuff:
  - User friendly, Windows Compatible, Standards Compliant, Configurable, General purpose, Maintainable, Open Source, Extensible, ...

Bad serial Protocols

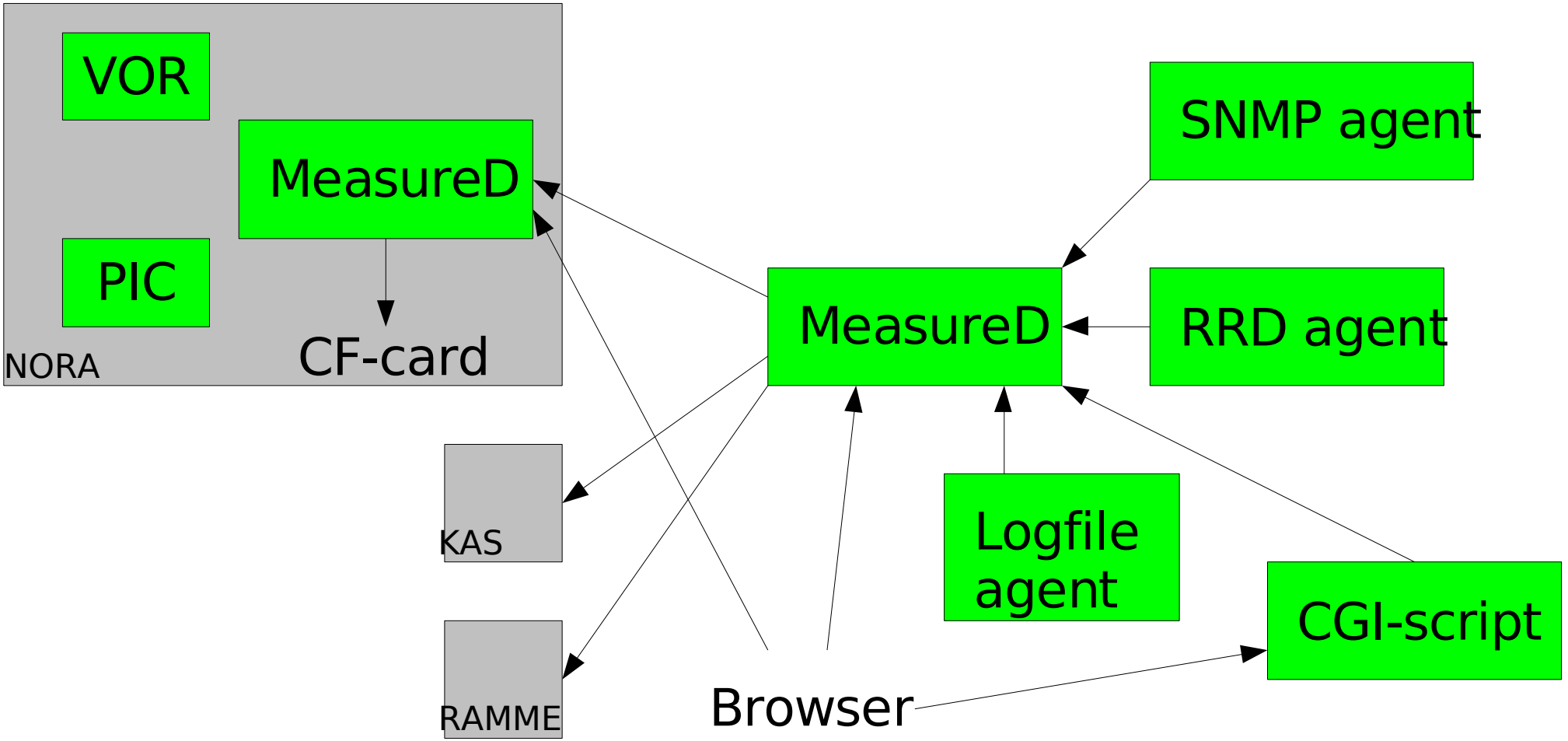
```
16              frame header 1
130             frame header 2
n + 5           number of bytes in frame - 1
destination     - 1 or 2- Cpu.
                - 0 Local/Remote PC
source          as for destination
message type    D = normal dialogue. A = alarm
data char 1
data char n
checksum byte 1
checksum byte 2
13              end of frame 1 ) carriage-return
10              end of frame 2 ) line-feed
```

Things I do for fun...

- Monitor my old house
  - Electricity
  - Natural Gas
  - Water
  - Solar water heating
  - Temperatures
  - Humidity



The MeasureD concept



Hardware

- Soekris NET4801 (with NanoBSD)
- Industrial 4GB CF card
- Opto-isolated dual serial PCI card
- DC/DC converter psu
- Home built 1U rack boxes
- Custom design PIC card

General Purpose I/O

- Digital/Analog inputs/outputs
- Use PIC18F8722 microcontroller
  - Serial interface to computer
  - 64 I/O pins, 16 with A/D converter
  - Eeprom for state storage
  - Fast power-on
  - Electrically Robust

PIC firmware

- Written in small C
  - Using SDCC compiler
- Functionality
  - Define pin function
    - (ain,din,dout,counter,temp)
  - 1-wire temp sensor protocol
  - Self- / mfg- test routines

Make interfaces simple

- External constraints:
  - Real-Time requirements
  - Odd-ball protocols
- There will be many off them
  - 9 so far, typically 200-500 lines of code
- Where future extension will happen

Threads or events ?

- I generally prefer eventdriven
- Writing to FIFO log (raw device, CF card) will sleep.
  - Bad for high-speed protocols
- Writing web-pages to TCP sockets in eventsized bits is nasty programming

- Threads are more convenient

ConfigKit

- Compiler and library for implementing cisco style CLI/config-file
- Saves tons of trivial code
  - Does argument syntax checking
- Supports multiple input sources
  - TELNET/SSH server
  - Files
  - Internally generated

ConfigKit

```
INSTANCE prs10 UINT {
    name    cfg_prs10
    new     cfg_prs10_new
    destroy cfg_prs10_destroy
    desc    "SRS PRS10 Rubidium Frequency Standard"

    WORD serial WORD {
        desc    "serial port device name"
        func    cfg_prs10_serial
    }
    WORD point WORD ... {
        desc    "point configuration"
        func    cfg_prs10_point
    }
    WORD debug UINT {
```

How it works...

- MeasureD's core contains:
  - Data-point management
  - Configuration parser
  - Event dispatcher
  - HTTP server

Extensions

- FIFO record/log buffer
  - Zlib compressed with timeout
- MASTER protocol server
  - And SLAVE “interface”
- ALARM facility
  - For service/off-line/fall-back &c.

Interfaces

- Talks via some interface to $real_world
  - Creates points
  - Updates points
  - Controls (output) points
  - Destroys points
  - Configuration functions

Addressing points

- Points have three part addresses:
  - Site (###.---.---)
  - Group (---.###.---)
  - Point (---.---.###)
- 21.4.34
  - Site NORA, SEL4000 VOR, Battery OK

Data point types

- Analog Input (“Battery voltage”)
- Analog Output (“Transmitter power”)
- Digital Input (“Diesel running”)
- Digital Output (“Start Diesel”)
- Counter (“Door opens”)

Data point attributes

- Label -- “Battery Voltage 48V”
- Format -- “%.2f”
- Units -- “Volt”
- Limits (high, raise, sag, low)
  - Hysteresis & limitdelay
- Timeout, Change
- SNMP trap number

Events

- Create/Destroy site, group, point
- Attribute changes to points
- Limit/Alarm changes to points
- New measurements on points

Built in Web-server

Home Logfile(old->new) Logfile(new->old) Configuration Alarm-State Alarm-Log(old->new) Alarm-Log(new->old)

site 000

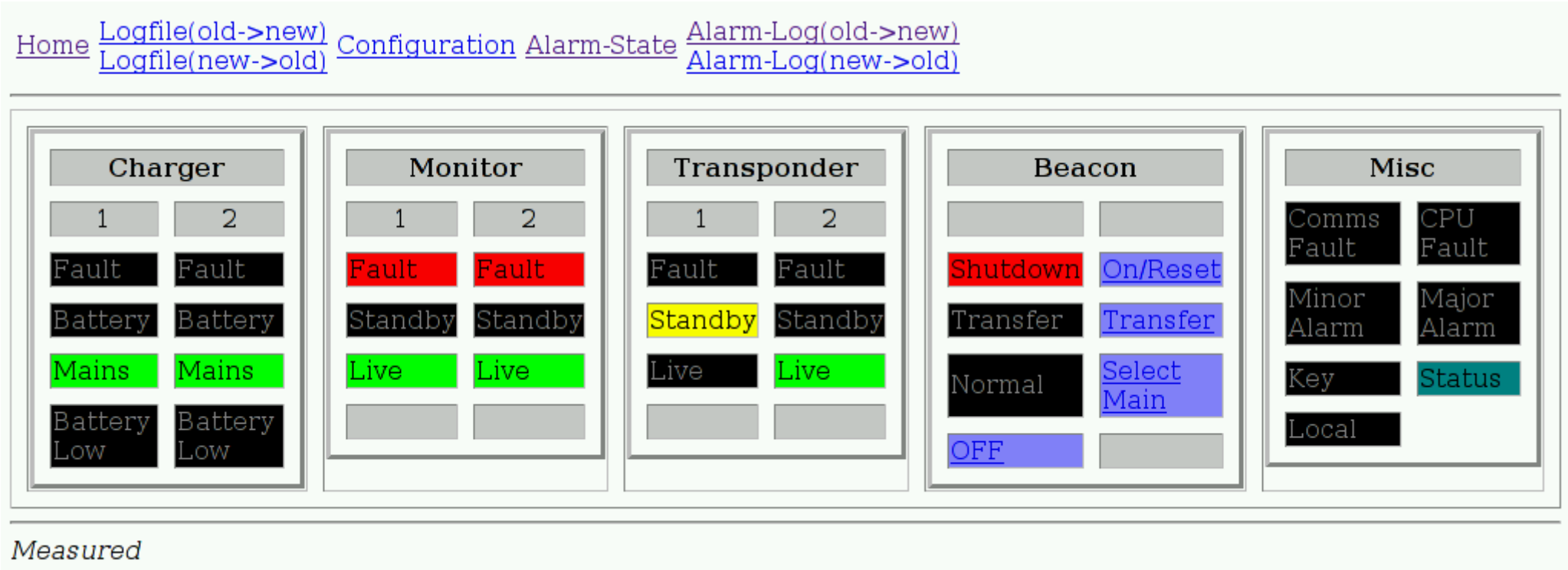
group 000.022 prs10

Index	Type	Label	Value	Units	Low	Sag	Raise	High
000.022.001	ain	Serial Number	nan					
000.022.002	aout	Lock mode	nan					
000.022.003	dout	Lock Loop status	nan					
000.022.004	ain	Frequency Control	nan					
000.022.005	ain	Power Cycles	nan					
000.022.006	ain	FC Saves	nan					

Macros for web-pages

```
.macro col 3
.if value($2) > .5
  <td bgcolor="$3">$1</td>
.else
  <td bgcolor="#000000">
     <font color="#888888">$1</font>
  </td>
.endif
.endmacro
```

Custom pages



MASTER/SLAVE protocol

- ASCII based
  - Easy interface from scripts
- Subscribe to events for some set of points
- Send “CONTROL” events back
- Built on top of HTTP server functionality

Master/Slave station

- Master subscribes to points from slave
  - NORA:0.4.* -> MASTER:7.4.*
  - KAS:0.4.* -> MASTER:8.4.*
- Only local points (site==0)

Project Status

- KAS DME goes live in a few weeks
  - CAT3 landing in Copenhagen Airport
- Roll-out throughout autumn/winter
  - 7 DME
  - 8 VOR
  - My house :-)