Bio & Sensors in OpenBSD

Marco Peereboom <marco@openbsd.org>

NYCBSDCon 2006

Terminology

- RAID – Redundant Array of Inexpensive Disks
- bio – Block Input Output
- bioctl – Block Input Output ConTroL
- Vendor – Sensor and/or RAID stack company
- SES – SCSI Enclosure Services
- SAF-TE – SCSI Accessed Fault Tolerant Enclosures

The vendor adventure

- Vendor tools provide whatever features marketing dreamt up. Inherent to this there are several issues.
- Over the years this has created an extremely complex tool chain without uniformity across OS' and hardware.
- The Open Source movement at large has not held vendors responsible for writing correct and/or open code.
- These issues are universally true for management tools. The industry is overcome by so called architects and spec writers that prefer complexity over functionality.

Sensors defined

- Sensors are hardware probes that provide vital environmental information about a system's health. For example:
  - Temperature
  - Fan RPM
  - Chassis intrusion
  - Voltages

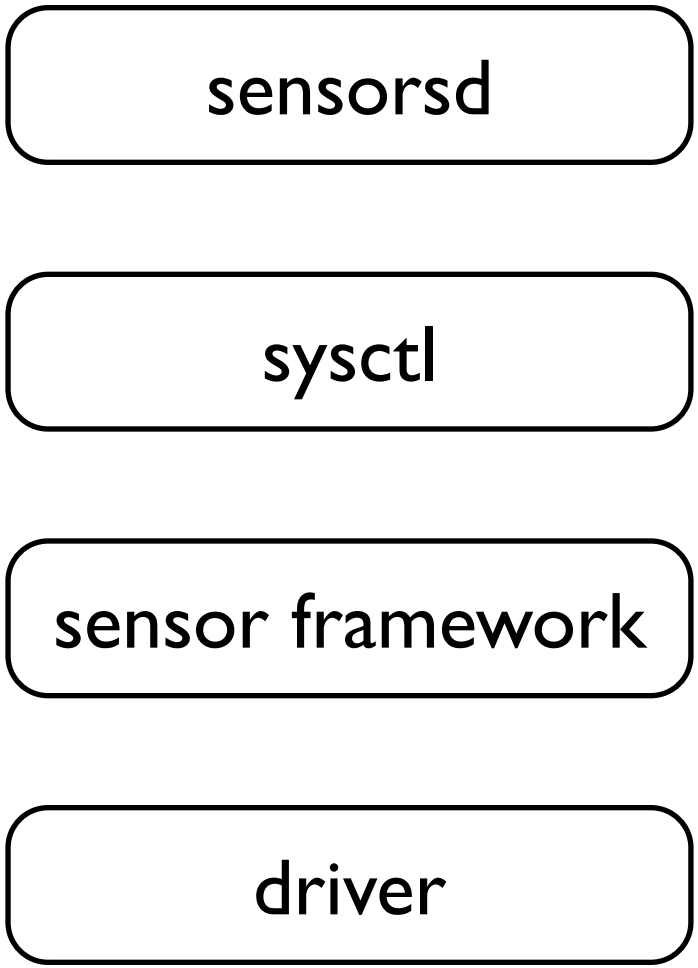
Why do we care?

- Environmental conditions might severely reduce the ability of a machine to run or worse, run reliably.
- Environmental readings aid in predicting potential future failures.
- Shutdown machines before component failure turns into irreparable machine failure.
- Correct redundant components without rebooting.

Sensor hardware

- Modern machines contain quite a variety of sensors. Here are some examples of what OpenBSD currently supports:
  - SCSI enclosures: SES & SAF-TE
  - Systems with BMCs: IPMI & ESM
  - I2C & SMBUS: adc, admcts, admlc, amdtemp, admtm, admtmp, admtt, adt, asbtm, asms, fcu, glenv, lmenv, lmtemp, maxds, maxtmp, pcfadc, tsl

Sensors in depth



The user-space daemon, sensorsd, obtains sensor values via the syctl interface. If necessary it acts upon configured thresholds in sensorsd.conf.

The driver is responsible for filling out the abstracted sensor framework values.

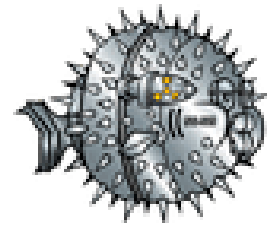
Driver

- The driver is responsible for retrieving, interpreting and abstracting the sensor values.
- The driver fills out the sensor-framework structure and periodically updates it's values and status.
  - The updates can be done via a timer interrupt.
  - Whenever a sensor driver requires process context to retrieve values and status it can register with the sensor-framework which provides a threaded interrupt mechanism.

Sensor-framework

- The sensor-framework is a collection of functions and APIs that do the following:
  - Add and remove a sensor from the kernel list.
  - Provide a callback mechanism for drivers to update values and status. This is done from a kernel-thread that calls the driver on a set interval. This enables sensors that require process context (e.g. setup a DMA transfer) to run. This in turn prevents “kernel-thread pollution”.

Sysctl



- The sysctl interface is where user-land and kernel meet.
- Sysctl is used to retrieve values and status from the kernel for user-land consumption.
- Barring appropriate privilege sysctl can also be used to set writable values.

Sensorsd

- Sensorsd retrieves sensor information via sysctl.
- Sensorsd can react upon defined threshold values (/etc/sensorsd.conf). For example if a temperature value exceeds 70C page the administrator.

Sensors example

```
# sysctl hw.sensors
hw.sensors.0=ipmi0, Phys. Security, On, CRITICAL
hw.sensors.1=ipmi0, Baseboard 1.5V, 1.51 V DC, OK
hw.sensors.2=ipmi0, Baseboard 2.5V, 2.51 V DC, OK
hw.sensors.3=ipmi0, Baseboard 3.3V, 3.34 V DC, OK
hw.sensors.4=ipmi0, Baseboard 3.3Vsb, 3.49 V DC, OK
hw.sensors.5=ipmi0, Baseboard 5V, 5.10 V DC, OK
hw.sensors.6=ipmi0, Baseboard 12V, 12.10 V DC, OK
hw.sensors.7=ipmi0, Baseboard -12V, -12.30 V DC, OK
hw.sensors.8=ipmi0, Battery Voltage, 3.14 V DC, OK
hw.sensors.9=ipmi0, Processor VRM, 1.47 V DC, OK
hw.sensors.10=ipmi0, Baseboard Temp, 30.00 degC, OK
hw.sensors.11=ipmi0, Processor 1 Temp, 36.00 degC, OK
hw.sensors.13=ipmi0, Baseboard Fan 1, 1980 RPM, OK
hw.sensors.14=ipmi0, Baseboard Fan 2, 2100 RPM, OK
hw.sensors.15=ipmi0, Baseboard Fan 3, 1410 RPM, OK
hw.sensors.16=ipmi0, Baseboard Fan 4, 1860 RPM, OK
hw.sensors.17=ipmi0, Baseboard Fan 5, 0 RPM, CRITICAL
hw.sensors.18=ipmi0, Baseboard Fan 6, 1950 RPM, OK
hw.sensors.19=ipmi0, Processor Fan 1, 5250 RPM, OK
```

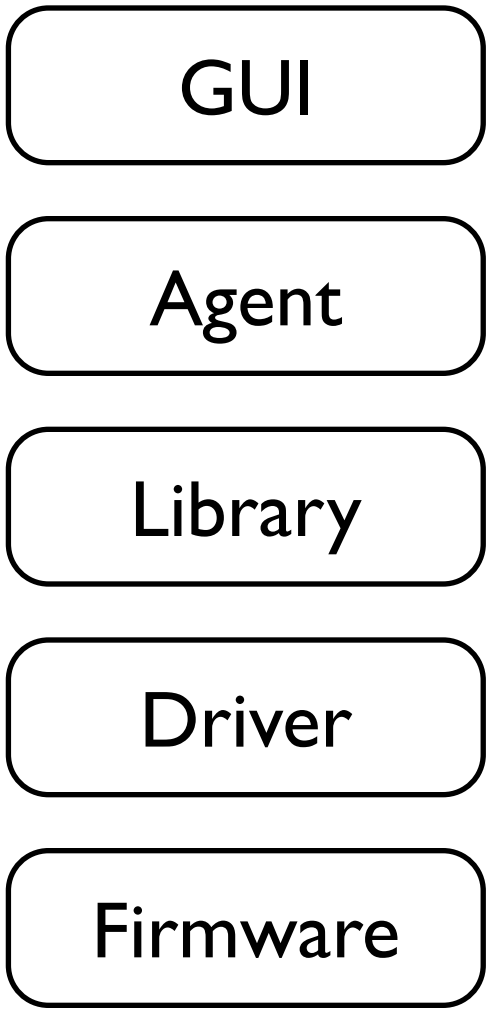
RAID and sensors

- So what do RAID and sensors have to do with each other?
  - They are actually intricately linked due to:
    - SES & SAF-TE
    - RAID volume status

RAID documentation

- OpenBSD's attempts to obtain documentation have stalled for several reasons:
  - Vendors do not posses current and accurate documentation.
  - Vendors do not want to support a product beyond regular channels.
  - Vendor's code is brittle in certain areas and they fear letting 3rd parties touching it and uncovering embarrassing details.
  - Vendors think their stack is unique.

Typical RAID management stack.



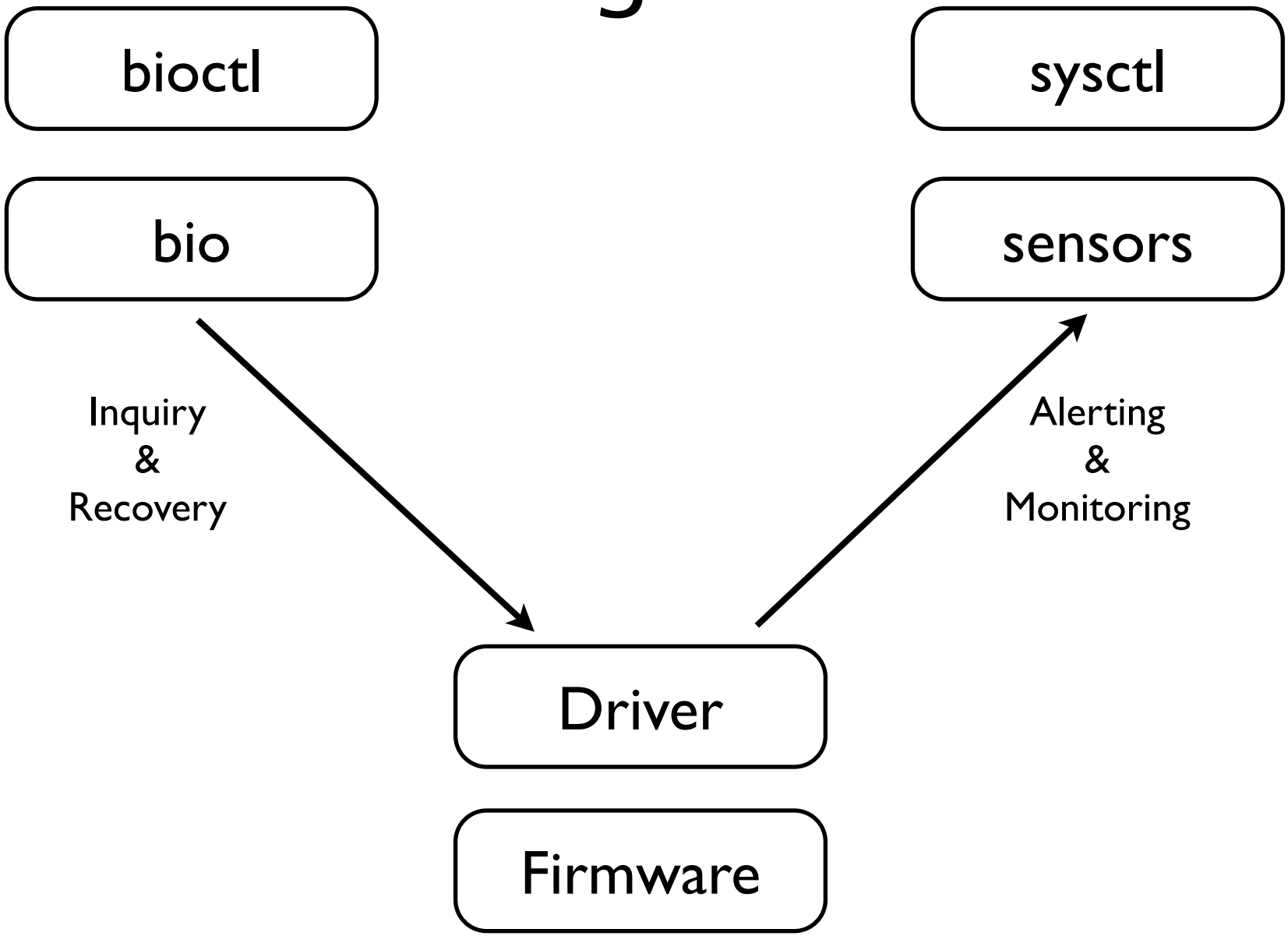
Typically these functional areas are developed by different teams resulting in large amounts of "abstraction code".

Abstraction code has no value for consumers, however it makes the overall stack large with all the inherent weaknesses.

RAID management essentials

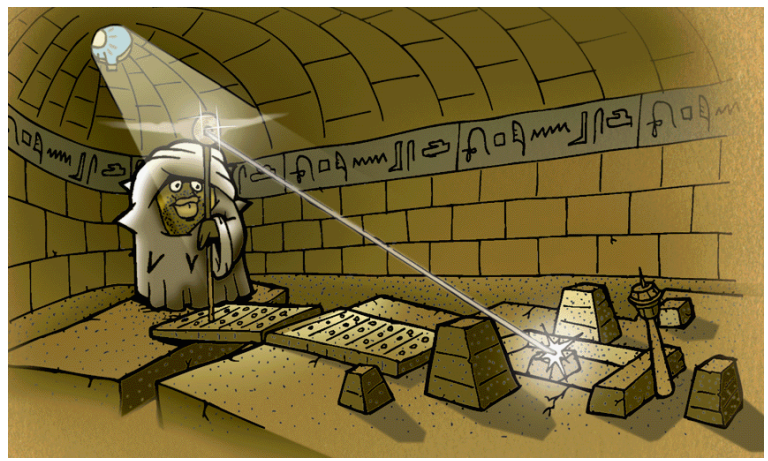
- Production machines do not need complex tool chains for RAID management. They essentially only need the following feature set:
  - Alerting
  - Monitoring
  - Inquiry
  - Recovery

OpenBSD RAID management



Bioctl

- Bioctl is the user-space RAID management tool that is used for inquiry and recovery.
- The idea behind bioctl is to be as consistent as ifconfig. There is no need for a separate tool for each RAID controller (can you say WiFi?).
- Inquiry functions:
  - Blink slot LED of a physical disk to locate it in the array
  - Display RAID setup and status
- Recovery functions:
  - Alarm management
  - Create hot-spare
  - Rebuild to hot-spare



BIO

- The bio driver provides user-land applications ioctl access to devices otherwise not found as /dev nodes.
- The /dev/bio device node operates by delegating ioctl calls to the requested device driver.
- Only drivers that have registered with the bio device can be accessed via this interface.

BIO inside drivers

- Inside the drivers is where the bio abstraction magic happens.
- In order to support bio drivers need to support some of the following primitives.
  - BIOCINQ
  - BIOCDISK
  - BIOCVOL
  - BIOCALARM
  - BIOCBLINK
  - BIOCSETSTATE

Bioctl in action

```
# bioctl ami0
Volume  Status     Size            Device
 ami0 0 Online       366372454400 sd0     RAID5
      0 Online         73403465728 0:0.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online         73403465728 0:2.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online         73403465728 0:4.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Online         73403465728 0:8.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online         73403465728 1:10.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online         73403465728 1:12.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 1 Online       366372454400 sd1     RAID5
      0 Online         73403465728 0:1.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online         73403465728 0:3.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online         73403465728 0:5.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Online         73403465728 1:9.0   ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online         73403465728 1:11.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online         73403465728 1:13.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 2 Unused         73403465728 1:14.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 3 Hot spare      73403465728 1:15.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
```

Failure Scenario

```
# bioctl ami0
Volume  Status     Size            Device
 ami0 0 Online       366372454400 sd0     RAID5
      0 Online         73403465728 0:0.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online         73403465728 0:2.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online         73403465728 0:4.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Online         73403465728 0:8.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online         73403465728 1:10.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online         73403465728 1:12.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 1 Degraded     366372454400 sd1     RAID5
      0 Online         73403465728 0:1.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online         73403465728 0:3.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online         73403465728 0:5.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Rebuild        73403465728 1:15.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online         73403465728 1:11.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online         73403465728 1:13.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 2 Unused         73403465728 1:14.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
```

Replace broken disk

```
# bioctl ami0
Volume  Status     Size            Device
 ami0 0 Online       366372454400 sd0     RAID5
      0 Online        73403465728 0:0.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online        73403465728 0:2.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online        73403465728 0:4.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Online        73403465728 0:8.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online        73403465728 1:10.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online        73403465728 1:12.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 1 Degraded     366372454400 sd1     RAID5
      0 Online        73403465728 0:1.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online        73403465728 0:3.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online        73403465728 0:5.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Rebuild       73403465728 1:15.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online        73403465728 1:11.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online        73403465728 1:13.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 2 Unused        73403465728 1:9.0   ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 3 Unused        73403465728 1:14.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
```

Create hot-spare

```
# bioctl -H 1:9 ami0
# bioctl ami0
Volume  Status     Size            Device
 ami0 0 Online       366372454400 sd0     RAID5
      0 Online         73403465728 0:0.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online         73403465728 0:2.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online         73403465728 0:4.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Online         73403465728 0:8.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online         73403465728 1:10.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online         73403465728 1:12.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 1 Degraded     366372454400 sd1     RAID5
      0 Online         73403465728 0:1.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      1 Online         73403465728 0:3.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      2 Online         73403465728 0:5.0   ses0   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      3 Rebuild        73403465728 1:15.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      4 Online         73403465728 1:11.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
      5 Online         73403465728 1:13.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 2 Hot spare      73403465728 1:9.0   ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
 ami0 3 Unused         73403465728 1:14.0  ses1   <MAXTOR  ATLAS15K2_73SCA JNZ6>
```

Other bioctl magic

- Help! I am bleeding from the ears!
  - Disable the alarm with: `# bioctl -a quiet ami0`
- Help! I can't find my disk!
  - Blink it with: `# bioctl -b 1.9 ami0`

Sysctl and sensors

- The sysctl framework provides near realtime information on the health of a RAID disk.
  - `hw.sensors.0=sd0, ami0 0, drive online, OK`
  - `hw.sensors.1=sd1, ami0 1, drive online, WARNING`
- This information in sysctl can be consumed by the sensorsd daemon to provide alerting.

SES and SAF-TE

- To make all the aforementioned functionality tick one needs a SES or SAF-TE device for one main reason:
  - SCSI does not support hot-plug support without either one of these devices. In the above example the insertion of the disk in slot 1:9 would go undetected.
- SES and SAF-TE add a lot of critical environmental readings that could indicate potential issues.

Supported hardware

- Most SCSI and SATA based LSI/AMI MegaRAID cards with varying levels of support.
- Most SAS/SATA MFI cards.
- Most SATA Areca cards.
- Basic support for MPI and CISS
- Support depends on peripheral hardware like SES, SAF-TE and type of disk.

Future

- Add bio support to other RAID cards.
- S.M.A.R.T. support for physical disks.
- New sensorsd.conf language.

Conclusion

- RAID and its associated management functionality isn't magical.
- It should be as easy and consistent as ifconfig on Ethernet. We don't need another WiFi debacle.
- Only a small core of functionality is necessary to create a useful RAID management strategy.
- Other open source projects are not helping the cause by allowing vendors to run their mostly worthless binaries on their operating systems.

Swag!

- Pick up the latest OpenBSD CD!