Sysfs

Patrick Mochel
mochel@osdl.org
Open Source Development Lab
2 November 2002

Overview

- 2.5 Feature.
- RAM-based filesystem.
- Kernel Programming Interface.
- Exports kernel objects and attributes.
- Shows object relationships.

Basics

- Mandatory Compile Option.
- Accessible via
  - mount -t sysfs sysfs /sys
- ASCII-based files.
- (Usually) One value/file.
- Use common tools: ls, grep, find.

Origins

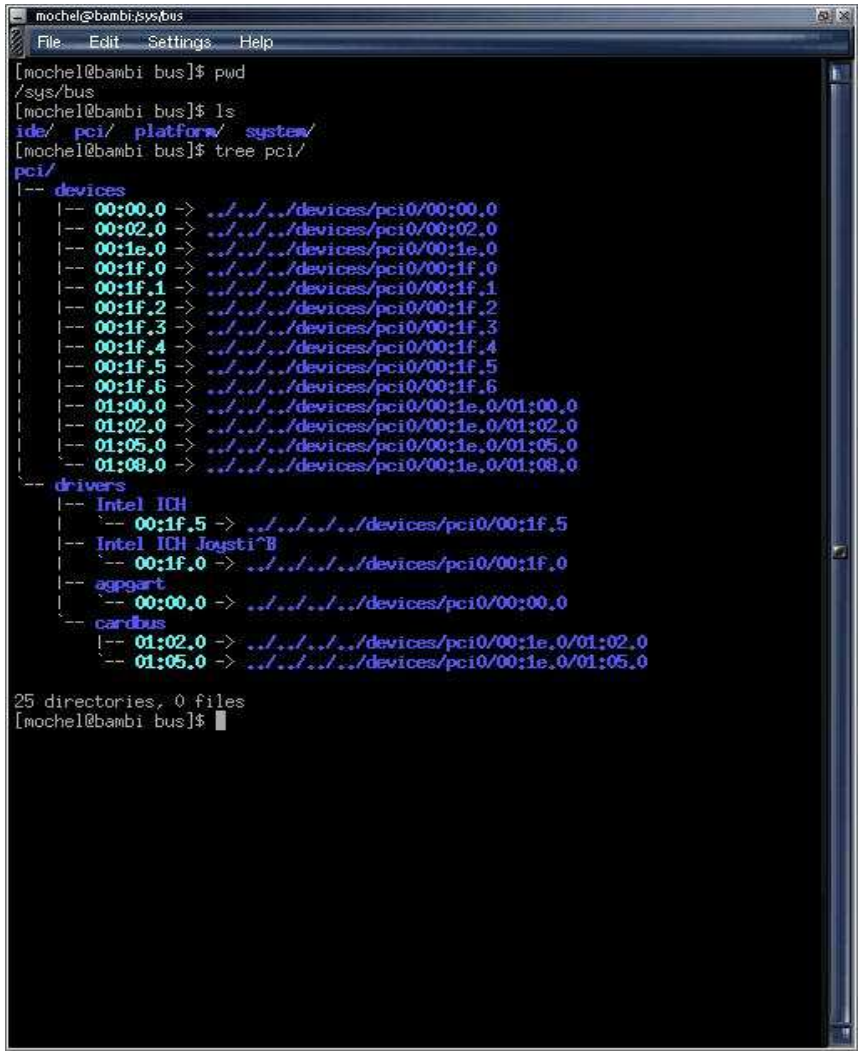
- Needed way to debug new driver model.
- Originally used procfs.
- Linus said 'procfs bad. Create own filesystem.'
- Created ddfs.
- So began The Incredibly Mutable Filesystem.

- Driver model matured.
  - Buses, drivers, classes.
- Filesystem matured.
  - Simple way to export objects & attributes.
  - (Relatively) concise, type-safe interface.

Devices

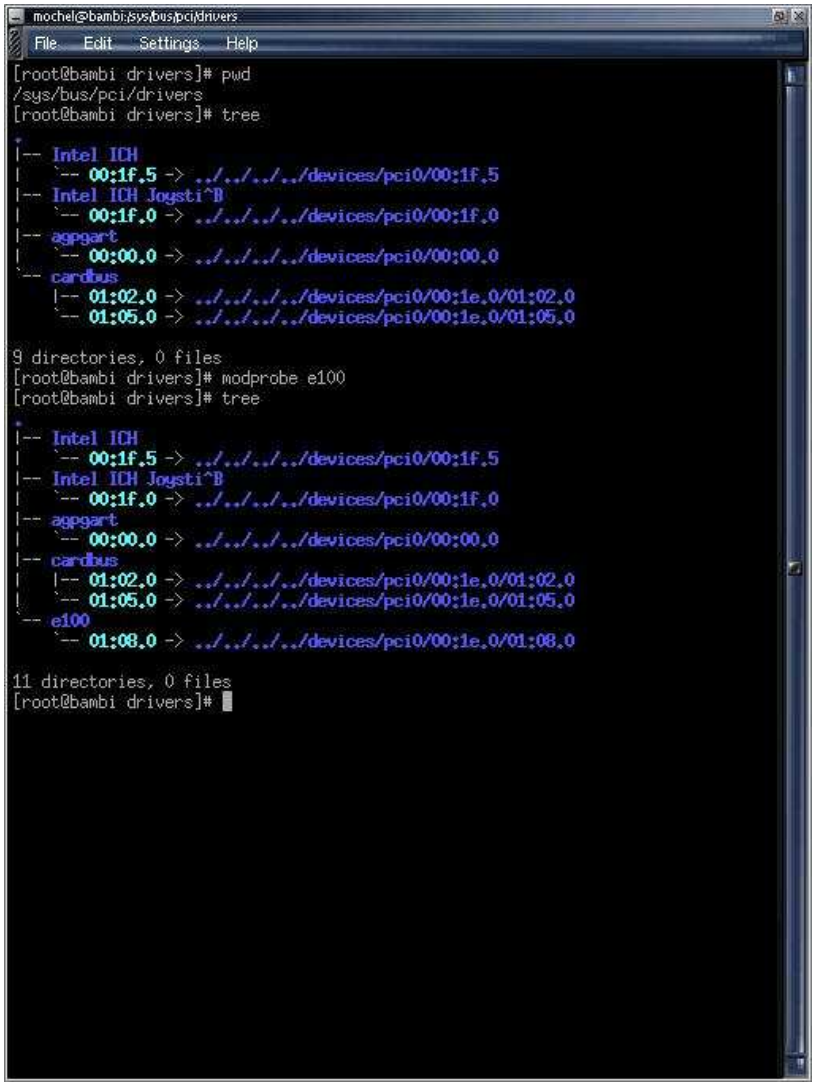
- Complete DeviceTree.
- Devices present in tree regardless of driver.
- Accurate Ancestry.
- Name is bus ID of device.
- Attributes of physical hardware.
- Attributes exported by any layer.

Buses



- Bus types exported.
- Maintains list of devices and drivers belonging to bus.
- Drivers actually children of bus type.
- Can have bus-specific attributes.
- Symlinks to point to devices' directories.

Device Drivers



- Drivers control a set of devices.
- Drivers belong to one bus type.
- Driver directories created when registered with bus.
- Symlinks created when devices are bound to drivers.

Device Classes

- Classes are types of devices.
- Maintains lists of devices and drivers.
- Drivers belong to **one** class.
- Driver symlinks created when driver registered.
- Device symlinks created when device bound to driver.

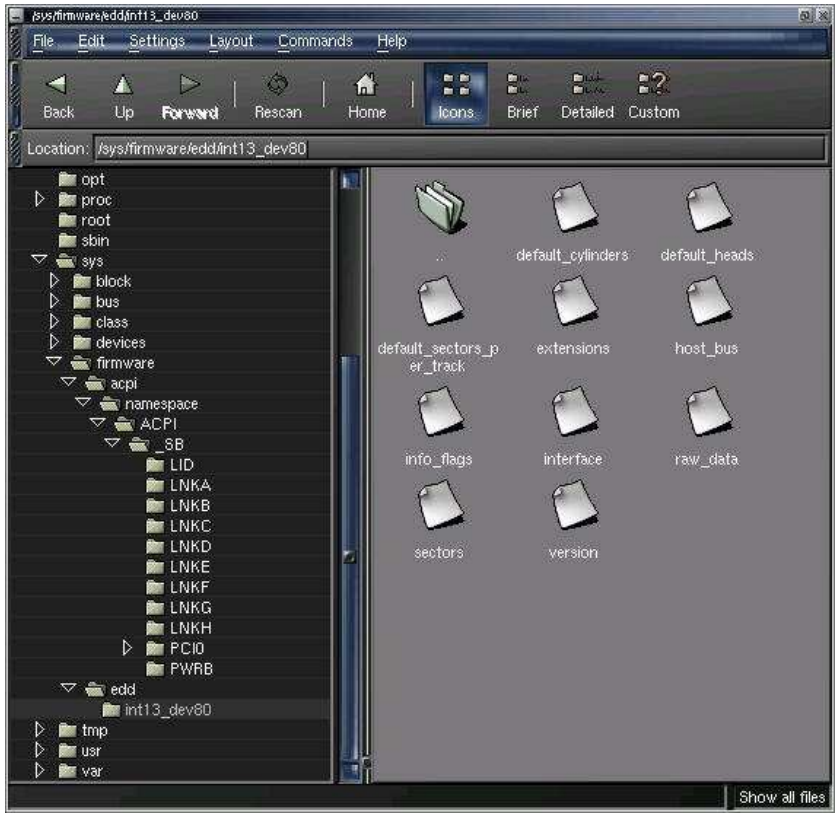
Above and Beyond The Driver Model

- Kernel Summit 2002 in Ottawa
- Presentation about unifying kernel parameters w/ Rusty Russel.
  - Boot, module, and runtime.
  - Ability to acquire boot parameters during runtime.
- Showed off driverfs and suggested doing something similar for other subsystems: modfs, vmfs, fsfs.
- Linus said 'Make them one.'

New Goals

- Make it easy to use.
- Convert procfs chaos to use sysfs.
- Unification (Mind Control).
  - Common interface to filesystem.
  - Common attribute definition.
  - Easy access to other objects in other subsystems.

Recent Conversions: firmware



- ACPI namespace.
  - System devices described in firmware hierarchy.
- EDD BIOS Interface.
  - Disk drives BIOS sees described.

Recent Conversions: block devices

- All block devices exported.
- Partitions exported.
- Attributes of partitions.
- Happens dynamically, from generic block device layer (gendisk layer).

Comparison: procfs

- Procfs
  - Anarchy wtith layout, naming and file formats.
  - Old code.
  - Bad interface.
  - No type safety.

- Sysfs
  - Infers as much info as possible: names, hierarchy.
  - ASCII files.
  - Type-safe calls.
  - Strong correlation between attributes and files.

Comparison: devfs

- Devfs
  - Unmaintainable source base.
  - Wrong level of abstraction.
  - Bad interface.
  - Old code.

- Sysfs
  - Abides by CodingStyle.
  - Maintainer easier to work with.
  - Infers most information: name, hierarchy, etc.

Overall Benefits

- Unified internal interface greatly increases code reusability.
- Single location for all kernel objects/attributes.
- Easy to use with basic tools.
- Can help solve some outstanding problems
  - Device naming.
  - Unifying boot parameters.
- Can make life happy again.
- Will end world hunger.