



CANONICAL

BIOS, Linux and Firmware Test Suite in-between

Alex Hung <alex.hung@canonical.com>



What is BIOS?

What is BIOS?



- **Basic Input/Output System (BIOS)**
- For most people, BIOS is
 - A flashing screen with riddles that shows up when they hit the power button

```
Phoenix - AwardBIOS v6.00PG, An Energy Star Ally
Copyright (C) 1984-2008, Phoenix Technologies, LTD
K987PV-PLUS-PRO

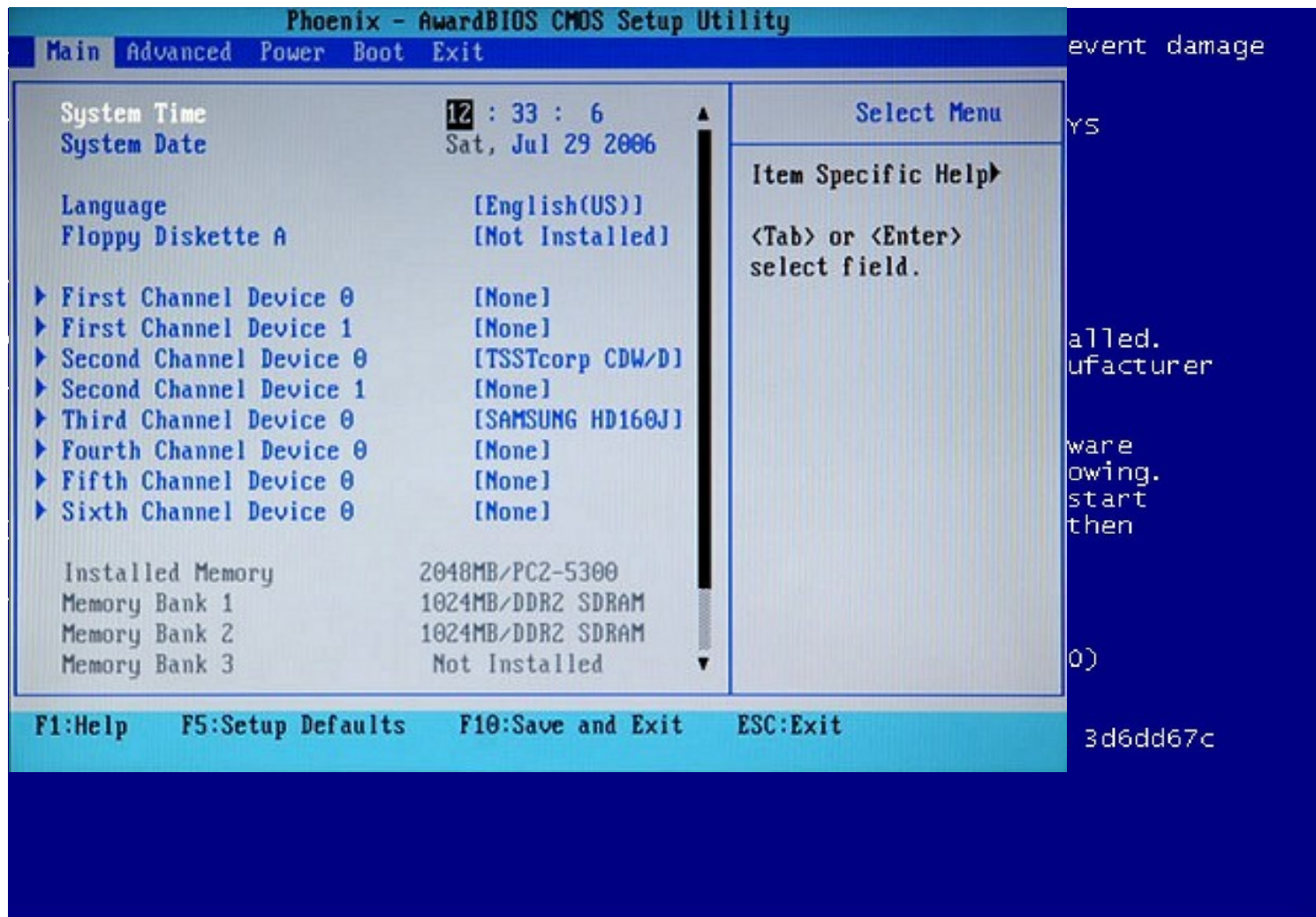
Main Processor: AMD Athlon(tm) XP 3200+
CPU Temperature Protection is ON
Memory Testing : 4096000K OK
Memory Clock is: 166MHz (DDR333)
Detecting IDE drives ...

Press DEL to enter SETUP
06/18/2007-KT440-1523-8E6LY7299C-00
```

What is BIOS?



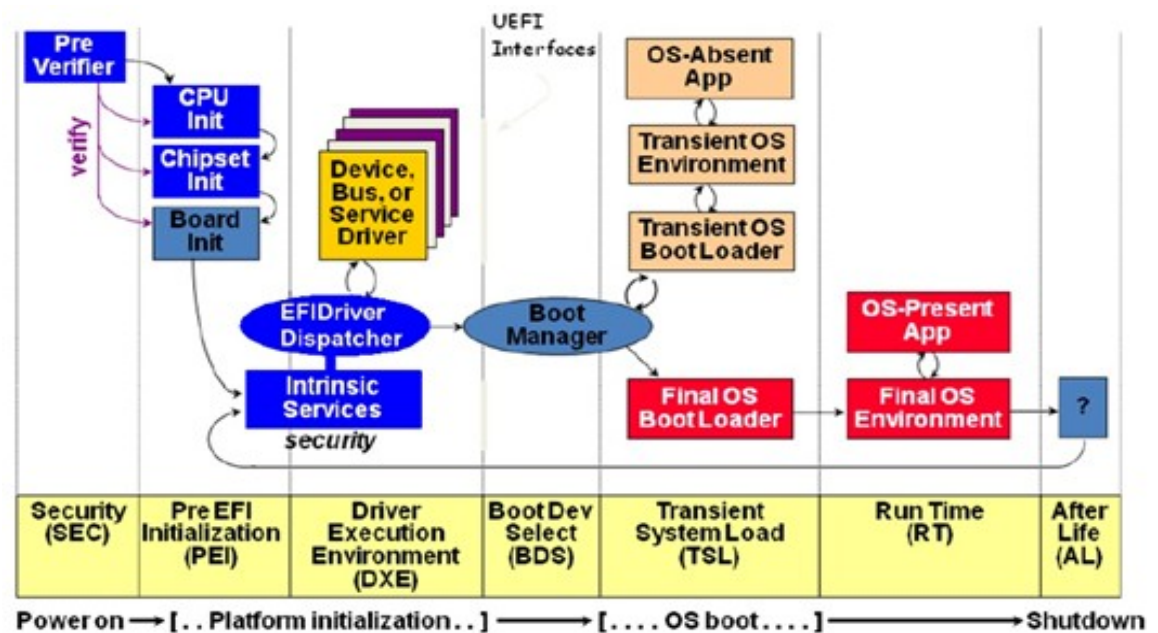
- And a blue screen when you press setup hotkey



BIOS is more than a blue screen...



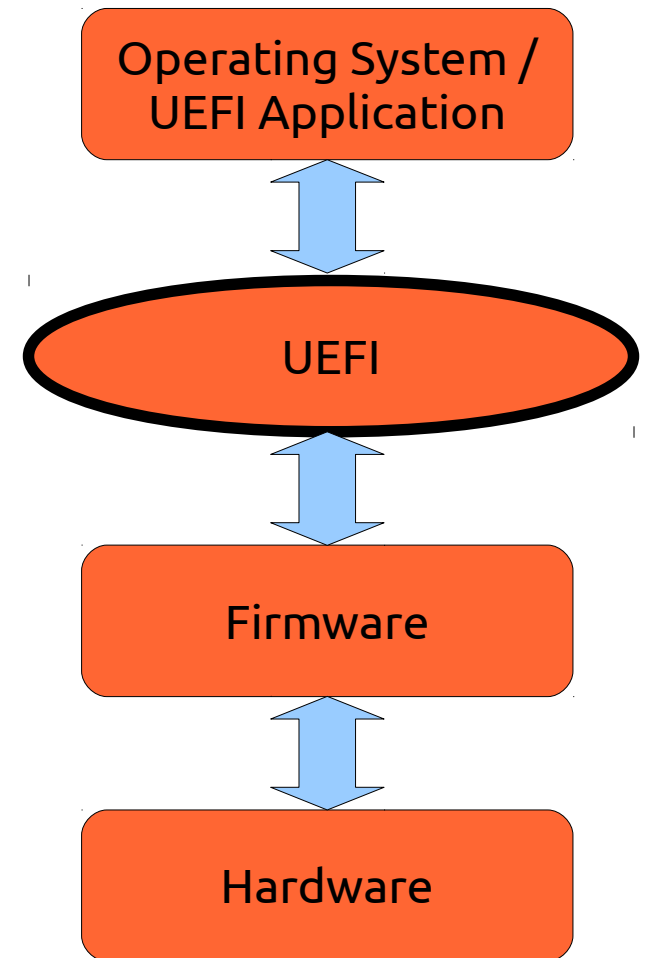
- Hardware initialization
- Hardware self-test (POST - Power-on-Self-Test)
- Customization for different OEM
- Standardized APIs (UEFI)
 - Boot-time Services
 - Runtime Services
- Power Management (ACPI)
 - Hardware description
 - Event handles (hotkeys)
- Multi-OS supports



UEFI - Unified Extensible Firmware Interface



- UEFI intends to replace the “legacy BIOS”.
- UEFI is an interface between the operating system (OS) and the system firmware (BIOS).
- UEFI provides boot-time and runtime services - a standard environment for booting an OS.
- UEFI is a pure interface specification. Common implementation includes:
 - ✓ Intel EDK2 (Tiano)
 - ✓ AMI's Aptio
 - ✓ Phoenix SecureCore
 - ✓ Insyde H20



UEFI's Advantages



- Faster Boot Time
- Support for Larger Disk Size
- Protected Mode (real mode in legacy BIOS)
- Standardized Services
- UEFI Shell (DOS replacement)
- Enhanced Security (Secure Boot)

Advanced Configuration and Power Interface (ACPI)



- ACPI was developed to establish industry interfaces enabling robust OS-directed motherboard device configuration and power management.
- ACPI is an interface specification comprised of both **software** and **hardware** element.
- In ACPI, the BIOS provides the OS with methods for **directly** controlling hardware.
 - ACPI uses ACPI Machine Language (AML) for implementing power event handlers.

What kind of information is defined in ACPI?



Power States

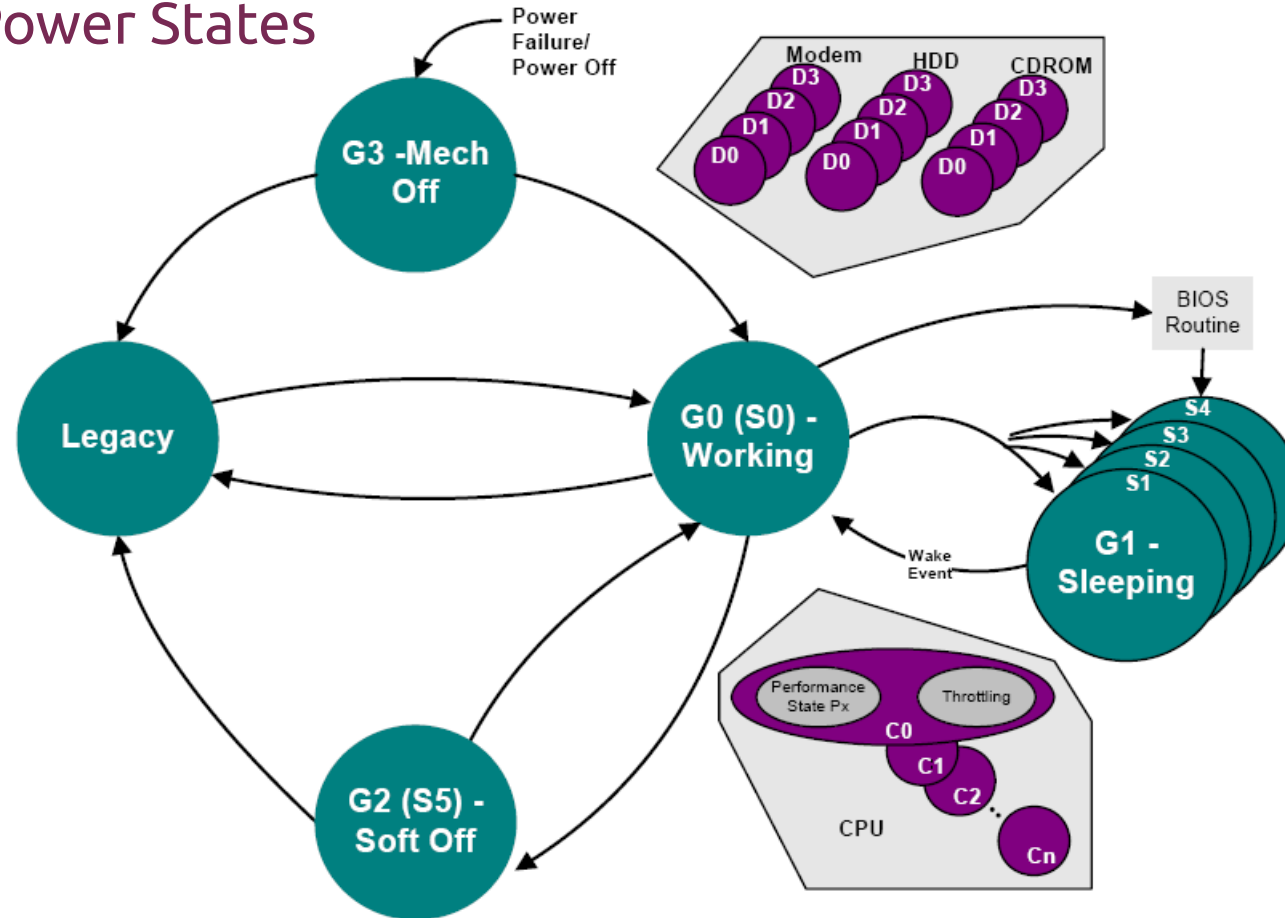


Figure 3-1 Global System Power States and Transitions



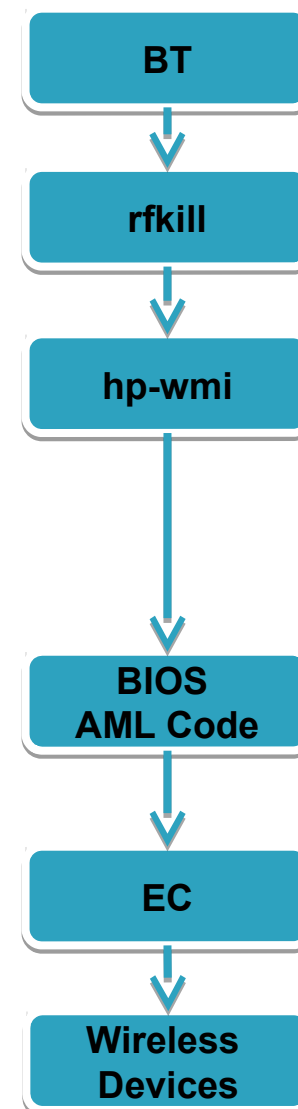
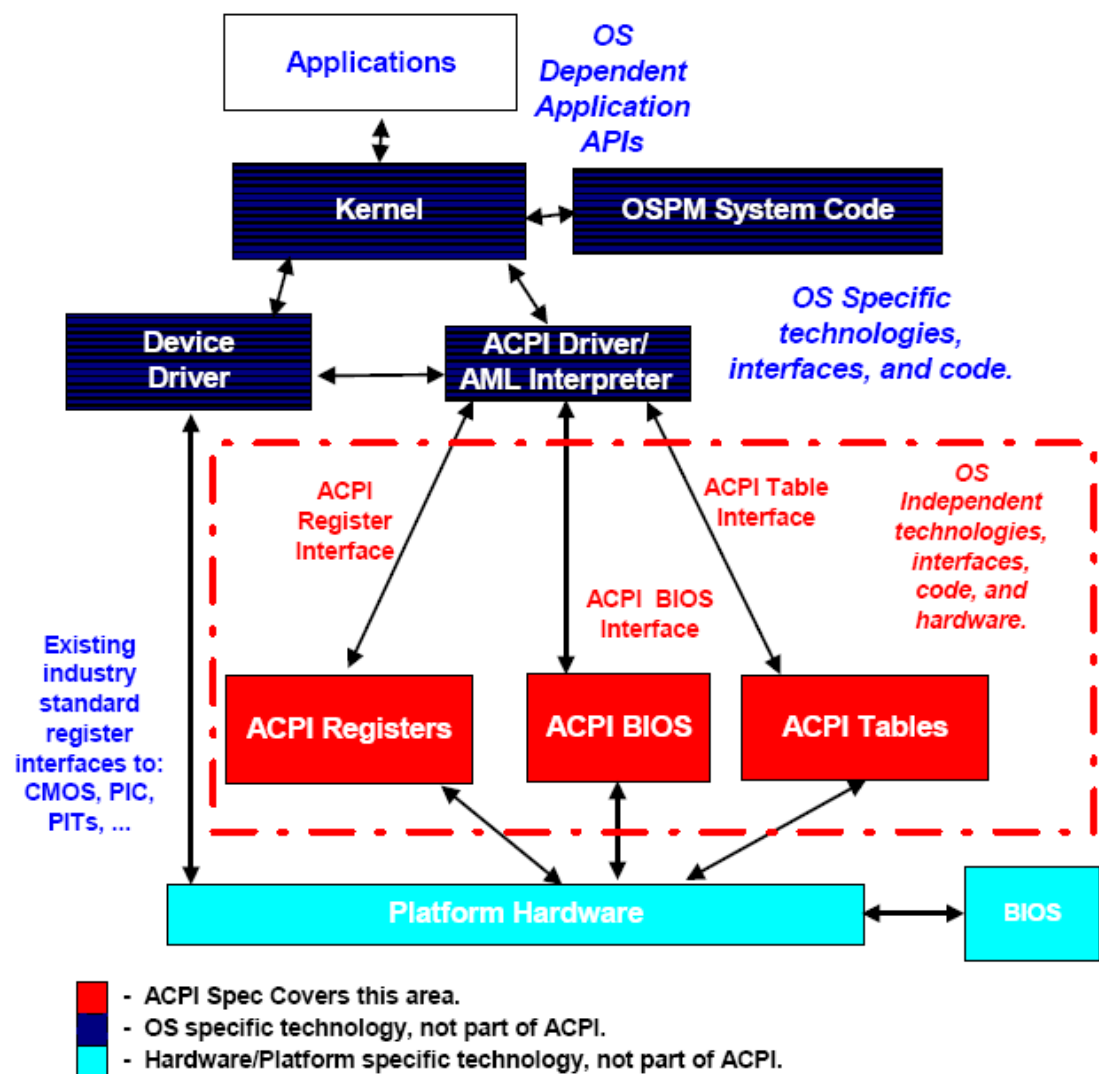
How does BIOS interact with Linux Kernel?

Interaction between Linux Kernel and BIOS



- UEFI runtime services
- ACPI registers, tables and control methods

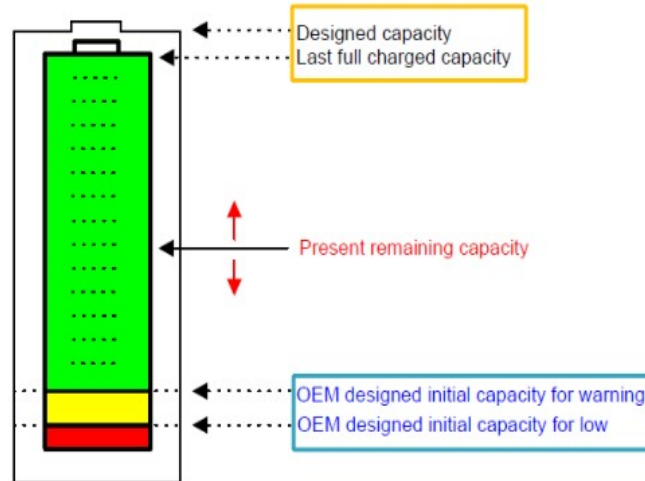
Advanced Configuration and Power Interface (ACPI)



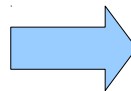
Exampe: ACPI Control Method Battery



```
Package {  
    Power Unit  
    Design Capacity  
    Last Full Charge Capacity  
    Battery Technology  
    Design Voltage  
    Design Capacity of Warning  
    Design Capacity of Low  
    Battery Capacity Granularity 1  
    Battery Capacity Granularity 2  
    Model Number  
    Serial Number  
    Battery Type  
    OEM Information  
}
```



```
Device (BAT0) {  
    Name (_HID, EISAID ("PNPOCOA"))  
    Name (_UID, 0x00)  
    ...  
}
```

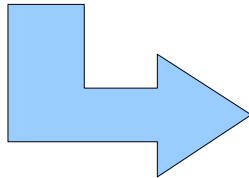


```
alexhung@fire: /sys/class/power_supply/BAT0  
alexhung@fire:/sys/class/power_supply/BAT0$ ls  
alarm          current_now    power          technology  
capacity       cycle_count    present        type  
charge_full    device         serial_number  uevent  
charge_full_design manufacturer    status         voltage_min_design  
charge_now     model_name     subsystem      voltage_now  
alexhung@fire:/sys/class/power_supply/BAT0$
```

Exampe: ACPI Control Method Battery (cont'd)



```
alexhung@fire: /sys/class/power_supply/BAT0
alexhung@fire:/sys/class/power_supply/BAT0$ ls
alarm          current_now    power          technology
capacity       cycle_count    present        type
charge_full    device         serial_number  uevent
charge_full_design  manufacturer  status         voltage_min_design
charge_now      model_name     subsystem      voltage_now
alexhung@fire:/sys/class/power_supply/BAT0$
```



Power Statistics

AC adapter
Laptop battery
Processor

Details History Statistics

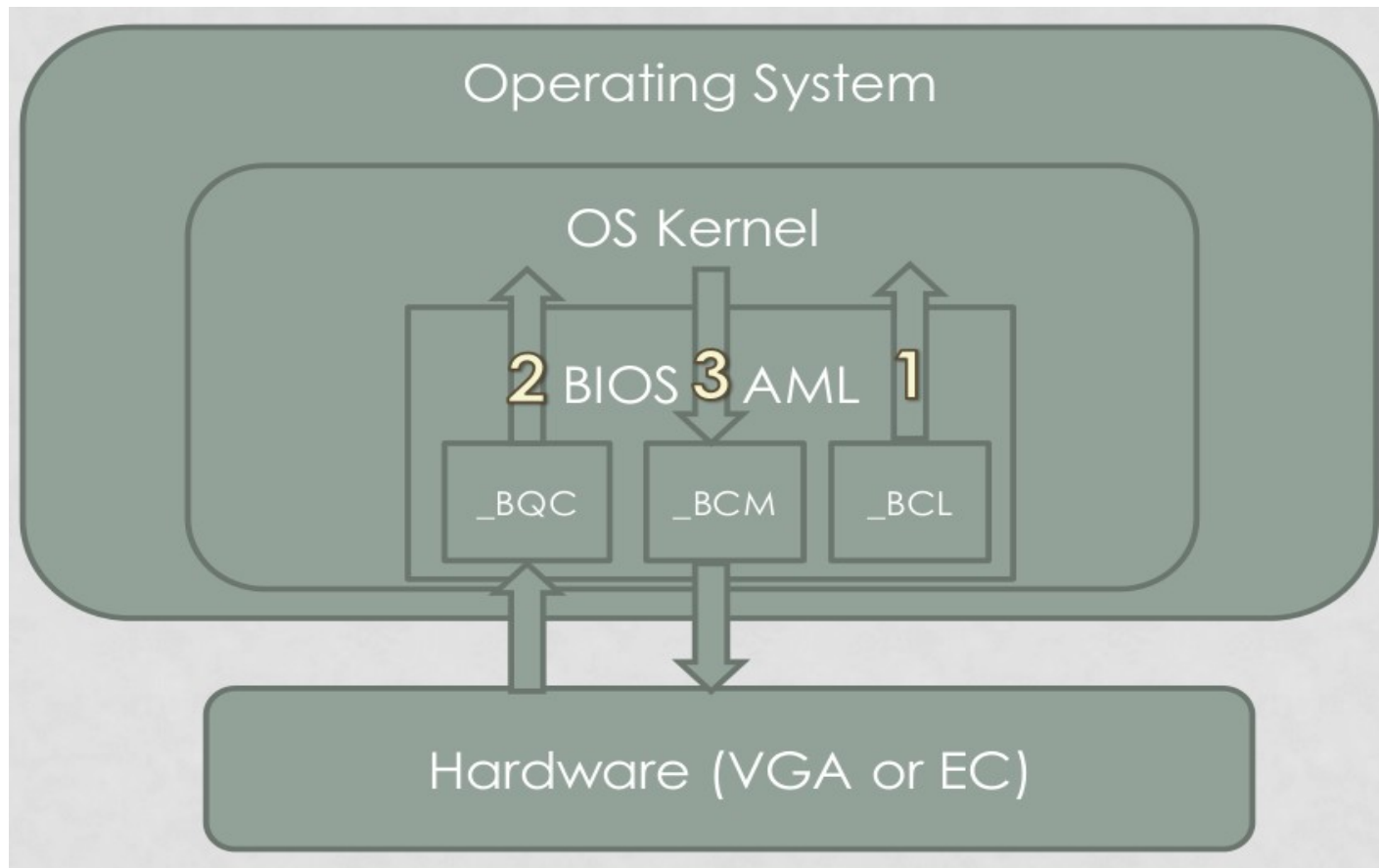
Attribute	Value
Device	battery_BAT0
Type	Laptop battery
Vendor	Hewlett-Packard
Model	Primary
Serial number	08197 2012/11/23
Supply	Yes
Refreshed	8 seconds
Present	Yes
Rechargeable	Yes
State	Discharging
Energy	35.5 Wh
Energy when empty	0.0 Wh
Energy when full	38.2 Wh
Energy (design)	38.2 Wh
Rate	10.7 W
Voltage	16.2 V
Time to full	0 seconds
Time to empty	3.3 hours
Percentage	92.0%
Capacity	100.0%
Technology	Lithium Ion

Close

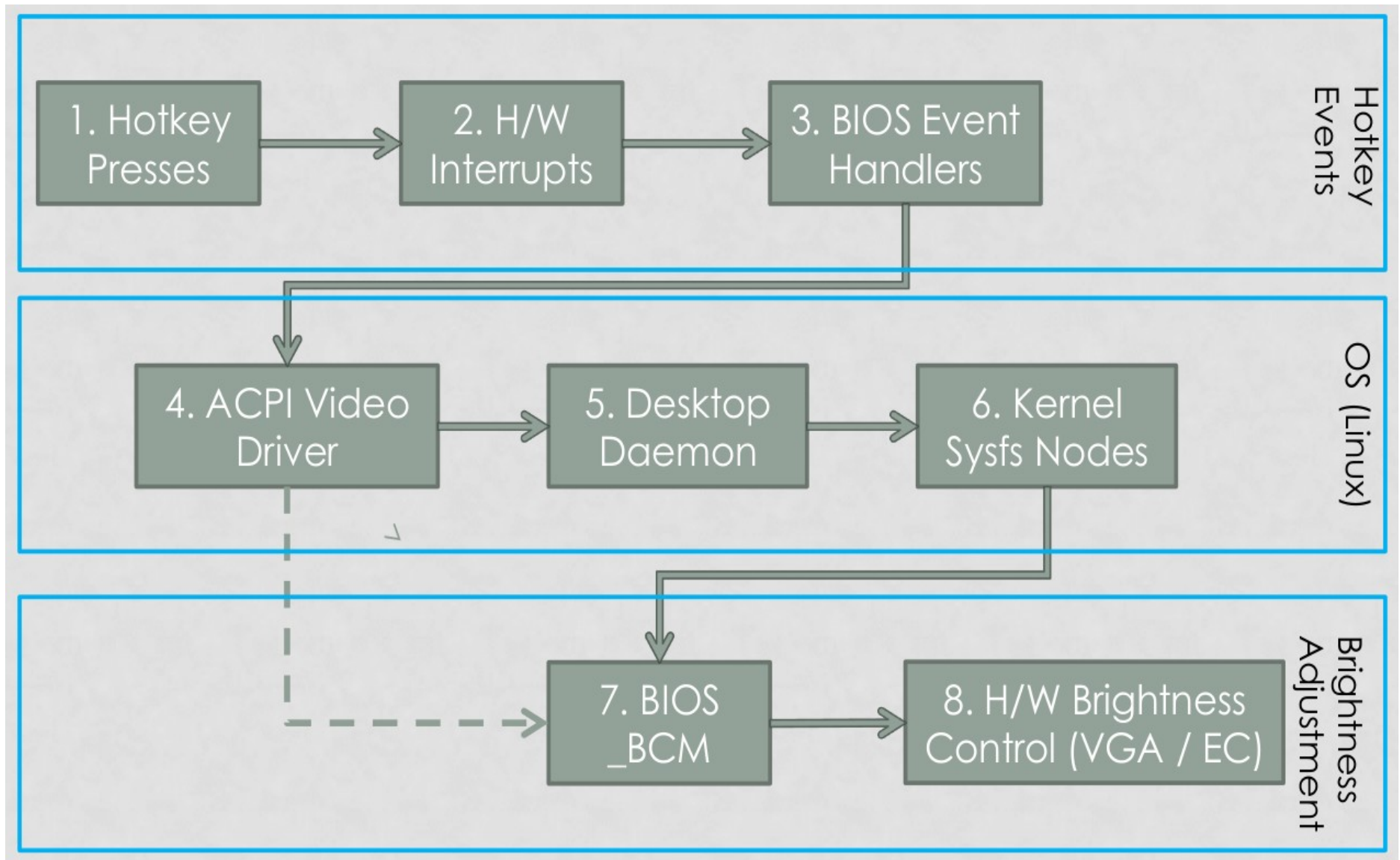
Example: ACPI Brightness



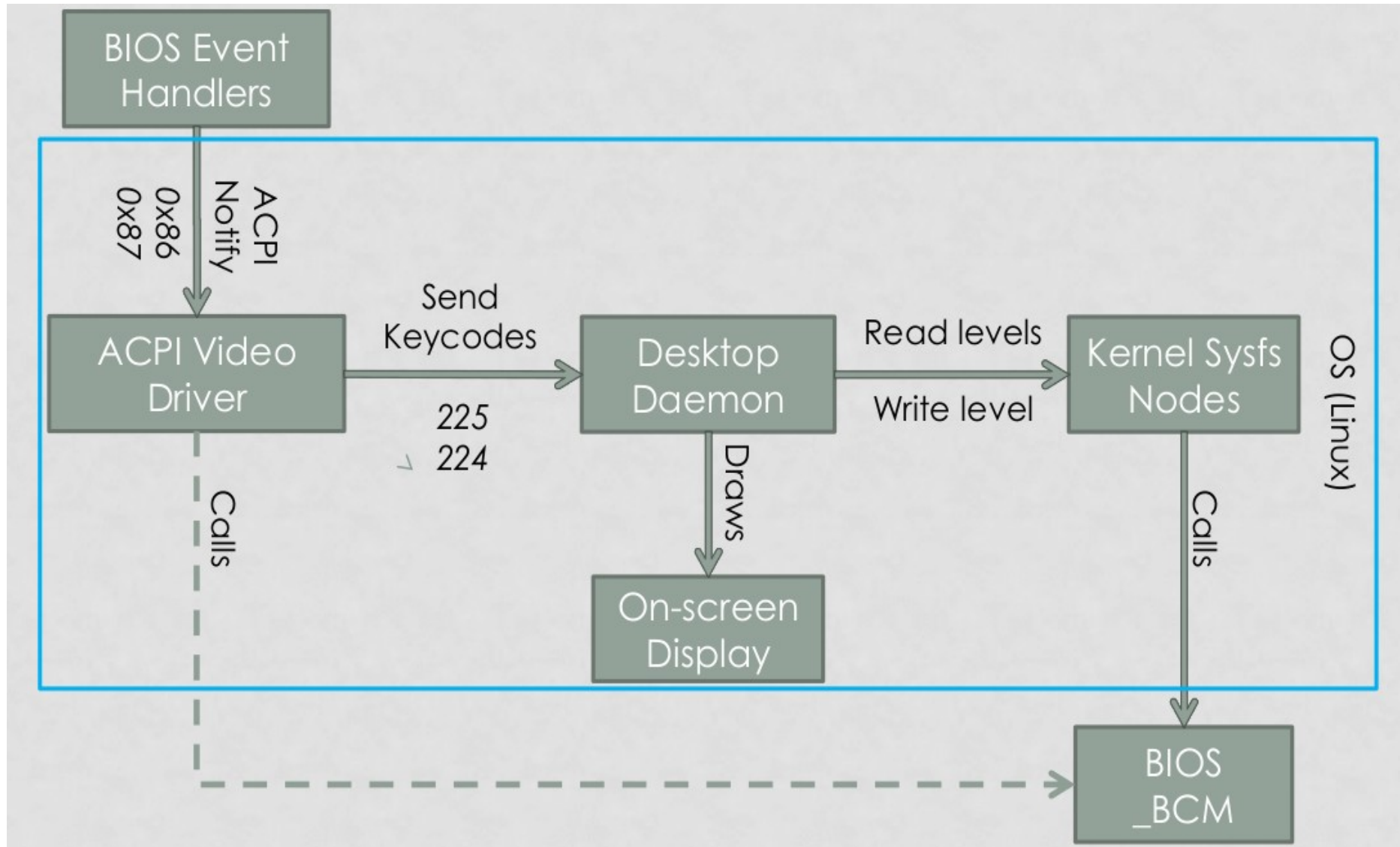
- `_BCL`: Query List of Brightness Control Levels Supported
- `_BCM`: Set the Brightness Level
- `_BQC`: Brightness Query Current level



Exampe: ACPI Brightness Hotkey



Example: ACPI Brightness Hotkey (cont'd)



More Examples



- AC Adapters (cat /sys/class/power_supply/AC/online)
- CPU's power management (C States and P States)
- Interrupt routing table
- Memory allocation table
- Thermal management (fan, critical shutdown and so on)
- Wakeup from sleep (cat /proc/acpi/wakeup)
- and many others

These are not BIOS-related (at least not often):

Audio distortion / no audio

VGA corruption

PCI devices

HDD-related

USB devices (slow bluetooth speed)



- PC market is shrinking
 - Nightmare (BIOS) is going away...
- WRONG!!!
- ARM is adopting UEFI and ACPI (which merged to UEFI Forum)
 - “Moving ACPI into the UEFI Forum is probably one of the biggest changes in the computing industry since the formation of the UEFI Forum in 2005”



Firmware Test Suite (FWTS)



What is FWTS?

FWTS is a Linux tool that automates firmware checking. It aims to detect bugs and to get firmware fixed.

What does FWTS test?

fwts comprises large set of fifty tests that are designed to exercise and test different aspects of a machine's firmware – including ACPI, UEFI, hardware configuration, power-saving and so on.

- ✓ ACPI
- ✓ CPU Configuration
- ✓ Legacy/UEFI BIOS Interface
- ✓ PCIe/PCI Configuration
- ✓ Power Management
- ✓ System Configuration
- ✓ Others



Installation

- `sudo apt-get install fwts`
- Development (stable) release:
 - `sudo add-apt-repository ppa:firmware-testing-team/ppa-fwts-stable`
- Development (unstable) release:
 - `sudo add-apt-repository ppa:firmware-testing-team/ppa-fwts-devel`

Source code

- Source code: `git clone git://kernel.ubuntu.com/hwe/fwts.git`



FWTS examples

Show tests

```
fwts --show-tests / fwts --show-tests-full
```

Scan ACPI Methods

```
sudo fwts method
```

Execute UEFI tests (runtime services)

```
sudo fwts uefirtmisc uefirttime uefirtvariable
```

Scan CPU configuration

```
sudo fwts msr mtrr nx virt
```

Scan kernel log

```
sudo fwts klog
```

Run all batch tests

```
sudo fwts
```

FWTS Example: ACPI Control Method Battery



```
alexhung@fire: ~  
alexhung@fire:~$ sudo fwts battery  
[sudo] password for alexhung:  
Running 1 tests, results appended to results.log  
==== Please PLUG IN the AC power of the machine ====  
Press <Enter> to continue  
==== Please now UNPLUG the AC power of the machine ====  
==== Waiting to see if battery 'BAT0' discharges ====  
==== Please wait 30 seconds while the battery is discharged a little ====  
==== Please now PLUG IN the AC power of the machine ====  
==== Waiting to see if battery 'BAT0' charges ====  
Waiting 070/120
```


FWTS Example: CPU Scaling



```
alexhung@fire: ~
alexhung@fire:~$ sudo fwts cpufreq
Running 1 tests, results appended to results.log
Test: CPU frequency scaling tests.
    CPU P-State Checks.                                1 passed
alexhung@fire:~$
```

```
alexhung@fire: ~
CPU 0: 12 CPU frequency steps supported.
Frequency | Relative Speed | Bogo loops
-----+-----+-----
2.25 GHz | 100.0 % | 292276
2.25 GHz | 70.5 % | 206010
2.15 GHz | 67.2 % | 196389
2.05 GHz | 63.6 % | 185994
1.95 GHz | 60.6 % | 177251
1.85 GHz | 57.3 % | 167401
1.75 GHz | 53.9 % | 157478
1.65 GHz | 50.6 % | 147925
1500 MHz | 47.3 % | 138152
1400 MHz | 44.2 % | 129313
1300 MHz | 41.0 % | 119935
1200 MHz | 37.4 % | 109370

CPU 1: 12 CPU frequency steps supported.
Frequency | Relative Speed | Bogo loops
-----+-----+-----
2.25 GHz | 100.0 % | 287466
2.25 GHz | 72.9 % | 209427
2.15 GHz | 69.6 % | 200155
2.05 GHz | 66.3 % | 190667

153,1 49%
```

Release Schedule



- fwts is released monthly
 - New versioning: Year.Month.Minor_version: 13.06,13.07 and 13.08, 13.09(.01)
- Release notes are available online
 - 13.09.01 @ <https://wiki.ubuntu.com/FirmwareTestSuite/ReleaseNotes/13.09.01>
 - 1x.0y @ <https://wiki.ubuntu.com/FirmwareTestSuite/ReleaseNotes/1x.0y.00>

CANONICAL



Online Resource for fwts

Information for fwts



Architecture Supported:

x86 PC – full support

ARM – limited features





Looking for more information?

fwts:

URL: <https://wiki.ubuntu.com/Kernel/Reference/fwts>

Mailing list: fwts-devel@lists.ubuntu.com





CANONICAL

Thank you