The Usual Suspects: The Kernel, udev, D-Bus, HAL, NetworkManager and Friends

Timo Hönig <thoenig@suse.de>

SUSE Linux Products GmbH - R&D Mobile Devices

24th February 2008
1 Overview
- Rationale: It’s All About Events
- Event Flow

2 Analyze and Solve Problems Within the Stack
- Investigating a Use-Case
- Solving the Issue
- The Event Monitor

3 Demo
1 Overview
   - Rationale: It’s All About Events
   - Event Flow

2 Analyze and Solve Problems Within the Stack
   - Investigating a Use-Case
   - Solving the Issue
   - The Event Monitor

3 Demo

The Usual Suspects: The Kernel, udev, D-Bus, HAL, NetworkManager and Friends
Events and Signals

- We’re *not* going to talk about interfaces like `sysfs` and `procfs`.
- We’re talking about events and signals.
- Loading a module, pressing a hotkey or closing the lid of the laptop trigger events in several components.
The Components

- The Linux Kernel
- udev - Device Manager, reports uevents
- HAL - Hardware Abstraction Layer, sends signals using D-Bus
- Daemons (e.g. NetworkManager), send signals using D-Bus
- Desktop applications, receive signals using D-Bus
1 **Overview**
   - Rationale: It’s All About Events
   - Event Flow

2 **Analyze and Solve Problems Within the Stack**
   - Investigating a Use-Case
   - Solving the Issue
   - The Event Monitor

3 **Demo**

---

Timo Hönig <thoenig@suse.de>
SUSE Linux Products GmbH - R&D Mobile Devices

The Usual Suspects: The Kernel, udev, D-Bus, HAL, NetworkManager and Friends
Typical Example: Loading a Kernel Module

- Kernel module gets loaded
- Driver calls `kobject_uevent()` or `kobject_uevent_env()`
- Events get transmitted to udevd
- HAL reads events from the abstract socket `/org/freedesktop/hal/udev_event`
- HAL creates device object and emits the signal `DeviceAdded`
- Applications receive the signals from HAL via D-Bus
Overview

Event Flow

The Usual Suspects: The Kernel, udev, D-Bus, HAL, NetworkManager and Friends
Overview

1. Overview
   - Rationale: It’s All About Events
   - Event Flow

2. Analyze and Solve Problems Within the Stack
   - Investigating a Use-Case
   - Solving the Issue
   - The Event Monitor

3. Demo

Timo Hönig <thoenig@suse.de>
SUSE Linux Products GmbH - R&D Mobile Devices

The Usual Suspects: The Kernel, udev, D-Bus, HAL, NetworkManager and Friends
Use-Case: Brightness Hotkeys Don’t Work

- Hotkeys for brightness up/down don’t work
- Check one: Is there any driver support?
- Check two: Does it work when directly talking to the driver’s interface?
- Check three: Which events are being reported, which events are missing?
Investigating a Use-Case

Analysis

- Check one: The system is supported by the driver sony-laptop.
- Check two: Manually using the driver’s interface works fine.
- Check three: Input events are being reported.
- Conclusion: The input events seem to get lost somewhere in the stack.
Overview

Rationale: It’s All About Events
Event Flow

Analyze and Solve Problems Within the Stack

Investigating a Use-Case
Solving the Issue
The Event Monitor

Demo

Timo Höning <thoenig@suse.de>
SUSE Linux Products GmbH - R&D Mobile Devices

The Usual Suspects: The Kernel, udev, D-Bus, HAL, NetworkManager and Friends
Identify Event Sources and Capture Events

- Identify all relevant event sources
- Monitor all events from the identified sources
  - Receive events from udev
  - Watch messages on the D-Bus system bus
  - Check for events on device interfaces
- Track down the defective component
Overview

1. Rationale: It’s All About Events
2. Event Flow

Analyze and Solve Problems Within the Stack

2. Investigating a Use-Case
3. Solving the Issue
4. The Event Monitor

Demo
Quick Hack

- **Ingredients:** Python, python-dbus and GTK2
- **The Monitor consists of several modules which receive events**
- **Example modules:**
  - udev
  - HAL
  - NetworkManager
  - Input Layer (/dev/input/eventn)
Overview

Analyze and Solve Problems Within the Stack

The Event Monitor

Modules

- Extremely simple and small
  - udev module is 10 LoC
  - HAL module is 15 LoC
- Easily extensible
- No strings attached: Just show the events
Overview
Analyze and Solve Problems Within the Stack

The Event Monitor

Cover ’em all!

- One tool, showing all events
- Easy to use
- *No* drop-in replacement for `showkey(1)`, `lshal(1)`, `udevmonitor(8)`
The Event Monitor

Input (/dev/input/event7) (10)
Sony Vaio Keys: KEY_BRIGHTNESSDOWN

udev (102)
add: /devices/system/cpu/cpu1/cpuidle/state0, cpu,

HAL (15)
ButtonPressed: brightness-down

udev (103)
add: /devices/system/cpu/cpu1/cpuidle/state1, cpu,

udev (104)
add: /devices/system/cpu/cpu1/cpuidle/state2, cpu,
Overview
Analyze and Solve Problems Within the Stack

Kernel, udev, D-Bus, HAL, NetworkManager and Friends

Thanks for coming!

Event Monitor (Sources, GPLv2):
http://nouse.net/monitor.tar.gz