Embedded Linux and the mainline kernel

David Woodhouse

Kernel Recipes

September 2012

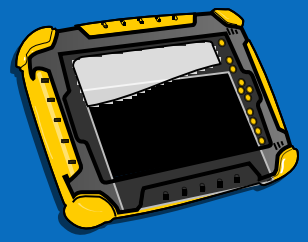
Open Source Technology Center

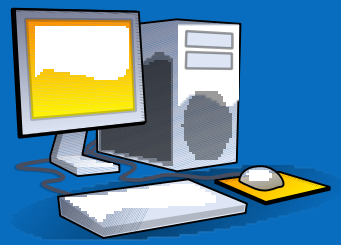
Software and Services Group

Where does Linux run?




Embedded control device... phone...
PDA... Internet tablet... router...
media device... netbook... laptop...
desktop... server... supercomputer...




“Embedded”...?

- Control equipment
- Phones
- PDAs
- “Internet Tablets”
- Routers
- Televisions
- VCR / PVR / DVD / Media
- Netbooks (?)

“Embedded”...?

- Headless?
- Handheld?
- Power source?
- Physical size?
- Limited RAM?
- Storage?
- Other...

Embedded needs

- Power management
- Fast startup
- Headless operation
- Uncluttered user interfaces
- Solid state storage

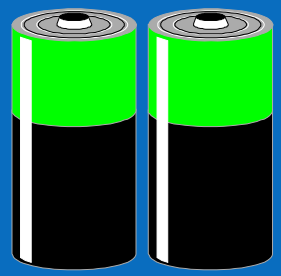
Embedded needs

- Power management
- Fast startup
- Headless operation
- Uncluttered user interfaces
- Solid state storage

Other users need these features too!

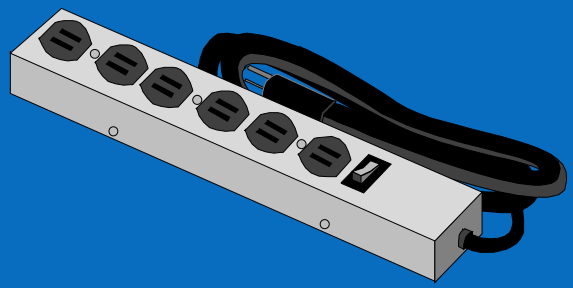
Power Management

- Battery life



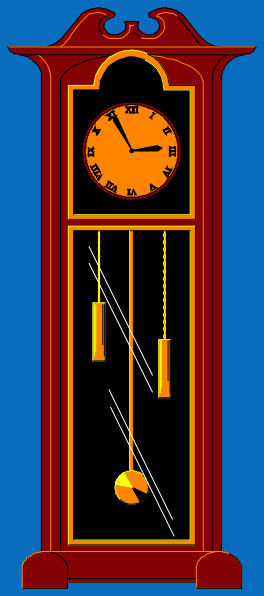
Power Management

- Battery life
- Cost of power consumption
- Heat output



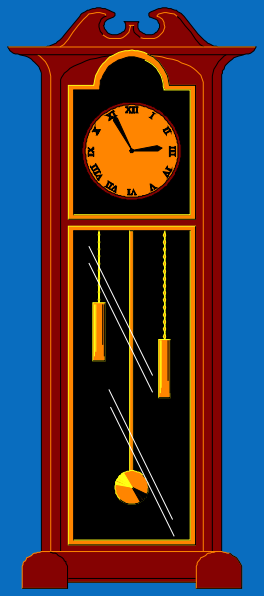
Tickless operation

- Power savings



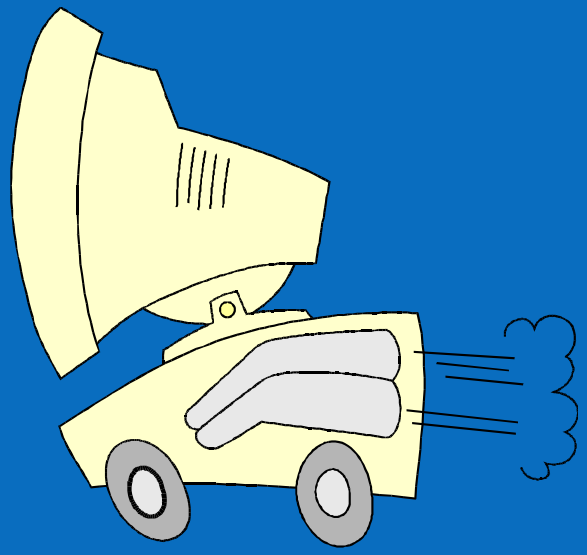
Tickless operation

- Power savings
- Scalability for virtualisation



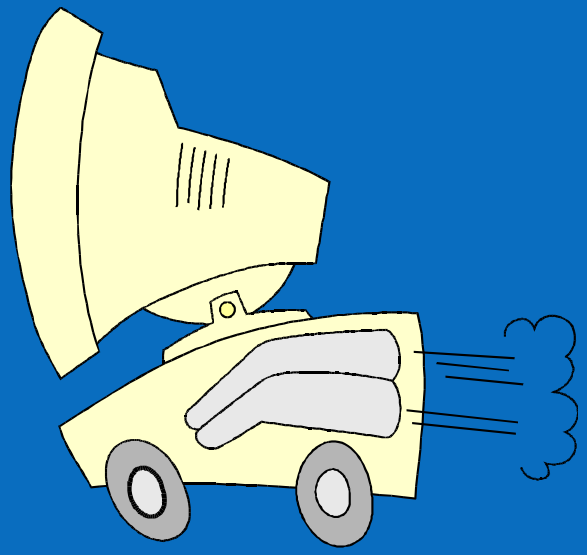
Fast boot

- Hard limits for mobile telephones
- User experience for consumer electronics



Fast boot

- Hard limits for mobile telephones
- User experience for consumer electronics
- Server availability



User interfaces

- Ease of use for consumer equipment

User interfaces

- Ease of use for consumer equipment
- ... and for everyone else:
  - OLPC / Sugar
  - Netbooks
  - Simple desktop environments

Solid state storage

- FLASH storage in “embedded” devices

Solid state storage

- FLASH storage in “embedded” devices
- Solid State Disk

Others...

- Execute in place (XIP)
  - From FLASH for embedded systems
  - Shared file system data under virtualisation
- DMA API usage
  - For cache coherency on embedded systems (ARM, some PPC)
  - For IOMMU on larger systems

We are not so special!

Community impressions

- “Enterprise” Linux
- “Embedded” Linux

Community impressions

- “Enterprise” Linux
- “Embedded” Linux
  - Working with old code
  - Not working with upstream
  - Inclined towards “special” one-off hacks
  - Irrelevant to the general case

Community impressions

- “Enterprise” Linux
- “Embedded” Linux
  - Working with old code
  - Not working with upstream
  - Inclined towards “special” one-off hacks
  - Irrelevant to the general case

We must prove them wrong!

“Embedded” success stories

- Tickless
- Preemptive kernel
- Power management
- Suspend to RAM
- Solid state storage
- Squashfs

Working with the community

- Find generic points of interest
- Publish early and often
  - In git trees
  - Separate trees for separate development efforts
  - Also send patches for review
- Solicit and respond to feedback
- Work with upstream maintainers
- ***BE PART OF THE COMMUNITY!***

Staying close to upstream

- Advantages
  - Easier for product updates and new products
  - Easy to use fixes and new features
  - External contributions
  - Code review and testing
- Costs
  - Writing acceptable code can be hard and takes time
  - Upstream kernel is a fast-moving target
  - Releasing information may be difficult

Tips on contributing code

- Avoid hacking around problems
- Avoid overengineering
- Care about locking
- Coding Style
- Submit patches carefully

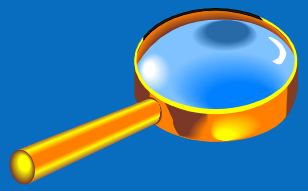
Coding Style

- Simple, short functions
- Avoid:
  - typedefs
  - StudlyCaps
  - Hungarian Notation
- Read `Documentation/CodingStyle`

http://www.linux.or.jp/JF/JFdocs/kernel-docs-2.6/CodingStyle.html

Submitting Patches

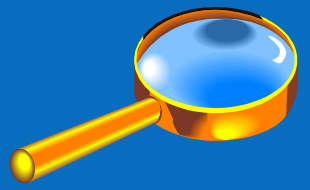
- Read the patch before you send it



```
--- a/drivers/net/wireless/libertas/assoc.c
+++ b/drivers/net/wireless/libertas/assoc.c
@@ -13,6 +13,7 @@

 static const u8 bssid_any[ETH_ALEN] = { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF };
 static const u8 bssid_off[ETH_ALEN] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char
*str);

 static void print_assoc_req(const char * extra, struct assoc_request * assoc_req)
 {
@@ -211,9 +212,17 @@ static int assoc_helper_channel(wlan_private *priv,
 {
        wlan_adapter *adapter = priv->adapter;
        int ret = 0;
+       int restart_mesh = 0;

        lbs_deb_enter(LBS_DEB_ASSOC);

+       if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
+       {
+               libertas_mesh_config(priv,0,adapter->curbssparams.channel,NULL);
+               restart_mesh = 1;
+
+       }
+
        ret = update_channel(priv);
        if (ret < 0) {
                lbs_deb_assoc("ASSOC: channel: error getting channel.");
@@ -225,11 +234,13 @@ static int assoc_helper_channel(wlan_private *priv,
        lbs_deb_assoc("ASSOC: channel: %d -> %d\n",
               adapter->curbssparams.channel, assoc_req->channel);

+
        ret = libertas_prepare_and_send_command(priv, CMD_802_11_RF_CHANNEL,
                                CMD_OPT_802_11_RF_CHANNEL_SET,
```

Submitting Patches

- Read the patch before you send it



```
--- a/drivers/net/wireless/libertas/assoc.c
+++ b/drivers/net/wireless/libertas/assoc.c
@@ -13,6 +13,7 @@

 static const u8 bssid_any[ETH_ALEN] = { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF };
 static const u8 bssid_off[ETH_ALEN] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char
 *str);

 static void print_assoc_req(const char * extra, struct assoc_request * assoc_req)
 {
@@ -211,9 +212,17 @@ static int assoc_helper_channel(wlan_private *priv,
 {
        wlan_adapter *adapter = priv->adapter;
        int ret = 0;
+       int restart_mesh = 0;

        lbs_deb_enter(LBS_DEB_ASSOC);

+       if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
+       {
+               libertas_mesh_config(priv,0,adapter->curbssparams.channel,NULL);
+               restart_mesh = 1;
+
+       }
+
        ret = update_channel(priv);
        if (ret < 0) {
                lbs_deb_assoc("ASSOC: channel: error getting channel.");
@@ -225,11 +234,13 @@ static int assoc_helper_channel(wlan_private *priv,
        lbs_deb_assoc("ASSOC: channel: %d -> %d\n",
               adapter->curbssparams.channel, assoc_req->channel);

+

        ret = libertas_prepare_and_send_command(priv, CMD_802_11_RF_CHANNEL,
                                CMD_OPT_802_11_RF_CHANNEL_SET,
```

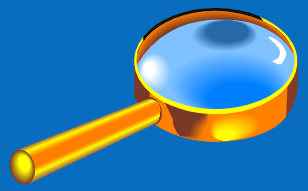
Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`

```
$ scripts/checkpatch.pl 0002-MESH-START-STOP_IOCTLS.patch
WARNING: line over 80 characters
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);

ERROR: "foo * bar" should be "foo *bar"
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);

ERROR: need space after that ',' (ctx:VxV)
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);
                                                                            ^

WARNING: externs should be avoided in .c files
#7: FILE: drivers/net/wireless/libertas/assoc.c:16:
+extern int libertas_mesh_config(wlan_private * priv, int action,int channel,char *str);

ERROR: That open brace { should be on the previous line
#19: FILE: drivers/net/wireless/libertas/assoc.c:219:
+       if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
+       {
ERROR: need a space before the open parenthesis '('
#19: FILE: drivers/net/wireless/libertas/assoc.c:219:
+       if(adapter->mesh_connect_status == LIBERTAS_CONNECTED)
```

Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`
- Send patches to yourself first



```
$ patch -p1 < saved-email.txt
patching file drivers/net/wireless/libertas/assoc.c
Hunk #1 FAILED at 13.
Hunk #2 FAILED at 212.
Hunk #3 FAILED at 234.
patch: **** malformed patch at line 71: adapter->meshid);
```

Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`
- Send patches to yourself first
- Read `Documentation/SubmittingPatches`

  or `Documentation/ja_JP/SubmittingPatches`

Submitting Patches

- Read the patch before you send it
- Use `scripts/checkpatch.pl`
- Send patches to yourself first
- Read `Documentation/SubmittingPatches`
- Include appropriate description

```
commit 4ac9137858e08a19f29feac4e1f4df7c268b0ba5
Author: Jan Blunck <jblunck@suse.de>
Date:   Thu Feb 14 19:34:32 2008 -0800

  Embed a struct path into struct nameidata instead of nd->{dentry,mnt}

  This is the central patch of a cleanup series. In most cases there is no good
  reason why someone would want to use a dentry for itself. This series reflects
  that fact and embeds a struct path into nameidata.

  Together with the other patches of this series
  - it enforced the correct order of getting/releasing the reference count on
    <dentry,vfsmount> pairs
  - it prepares the VFS for stacking support since it is essential to have a
    struct path in every place where the stack can be traversed
```

What's next for “Embedded” Linux?

- Solid state storage
  - More work on SSDs
  - Flash file system development (UBI, logfs, btrfs)
- Better power management
- More real time development
- What do ***you*** need?

Questions?