Barebox

The Barebox Bootloader

Barebox (formerly known as u-boot-v2) is a bootloader that inherits the best of U-Boot and the Linux kernel. The size and look-and-feel of u-boot, with driver model and lots of design concepts from the kernel.

You can find the latest of our timed releases in the download section. Check the repository for the most recent developments. Feel free to subscribe to the mailing list. Note also the slides from the ELCE09 talk by Sascha Hauer, creator of barebox.

If you search for a kernel hacker friendly bootloader, read on!

Last change: Thu May 3 09:54:22 2012
http://www.barebox.org
BIOS / Bootloader

- Barebox / Redboot / U-Boot / Linux Kernel
  - Responsible for hardware bring up
    - Basic clock
    - Memory init
  - Load the kernel in memory if needed
  - Jump in the kernel
Advanced Bootloader

• Barebox
  • Failsafe update
  • Net boot
  • Security
  • User interaction
  • Modules
  • Menu
  • Applications (up coming)
History

- **2007 / u-boot-v2-rc1**: Forked from U-Boot, as a technology study under the “U-Boot-v2” name

- **2009 / barebox-2009.12.0**: Renamed to barebox, with its own infrastructure

- **2013 / barebox-2013.03.0**: 49 releases up to now

  - Timed releases: about once per month

  - Maintenance releases: on demand
Development Resources

• Website
  http://www.barebox.org

• GIT Server
  http://git.pengutronix.de/?p=barebox.git

• next branch
  accumulates new features

• master branch
  next is merged into master after release

• Mailing List
  http://lists.infradead.org/mailman/listinfo/barebox/
Code analysis

- C: 90%
- Other: 10%

Lines of Code

- Code
- Comments
- Blanks
Code analysis

Codebase History

Lines of Code

Zoom: 1yr 3yr 5yr 10yr All

500k

250k

2008 2009 2010 2011 2012

2002 2004 2006

2008 2010 2012
Development speed

Commit History:

2008 364
2009 583
2010 690
2011 970
2012 1979
2013 439 (until now)
## Development speed

<table>
<thead>
<tr>
<th>30 Day Summary</th>
<th>12 Month Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Feb 2 2013 — Mar 4 2013</strong></td>
<td><strong>Mar 4 2012 — Mar 4 2013</strong></td>
</tr>
<tr>
<td>188 Commits</td>
<td>2244 Commits</td>
</tr>
<tr>
<td>16 Contributors</td>
<td>Up +1016 (82%) from previous 12 months</td>
</tr>
<tr>
<td>including 1 new contributor</td>
<td>62 Contributors</td>
</tr>
<tr>
<td></td>
<td>Up +14 (29%) from previous 12 months</td>
</tr>
</tbody>
</table>
CPU Architectures

- Supported Hardware:
  - arm at91, ep93, i.MX, netX, nomadik, nvidia, omap, pxa, samsung, mxs, versatile, calxeda, vexpress, zynq
  - Blackfin
  - mips
  - openrisc
  - ppc mpc5xxx, mpc85xxx
  - sandbox linux
  - x86 bios based
All Features on One Slide

- Build system: Kconfig, Kbuild
- Boot media: linux16, nand, ubi, sd, spiflash
- Data Transport: DFU, Kermit, X/Y/Z-Modem, tftp, nfs
- Graphics: Framebuffer, splash screen
- Filesystem: cd, ls, cp, saveenv/loadenv, mount, partitions
- Tools: crc, edit, gpio, unlzo
- User interaction: login, menu, password, application
- Drivers: i2c, mfd, flash, serial, spi, usb host+device
- Modules: insmod, lsmod
- Memory: meminfo, memtest, md, mw
- Network: ipv4, dhcp, netconsole, tftp, rarp, ping, nfs, dns
- Binfmt, complete, hush common env
Menu

- This kind of menu is very useful when you do not have a keyboard or a serial console attached to your board to allow you to interact with barebox

For the developer part,
The framework introduce two API

1. **C** that allow you to create menu, submenu, entry and complex menu action

2. **Command** that allow you as the C API to create menu, submenu, entry and complex menu action but this time the actions will be store in a function and then be evaluated and executed at runtime.
Menu example

Welcome on Barebox Boot Sequence

- 1: Boot
  2: boot (default)
  3: boot from nand
  4: boot from nfs (kernel nfs)
  5: boot from nfs (kernel tftp)
- 6: Command
  7: shell
  8: update
  9: reset

Auto Select in 5
Applications
Applications

- Based on syscalls
- Binary ABI
- Libc
- Curses (menu, panel, form)
Let hack
Questions?