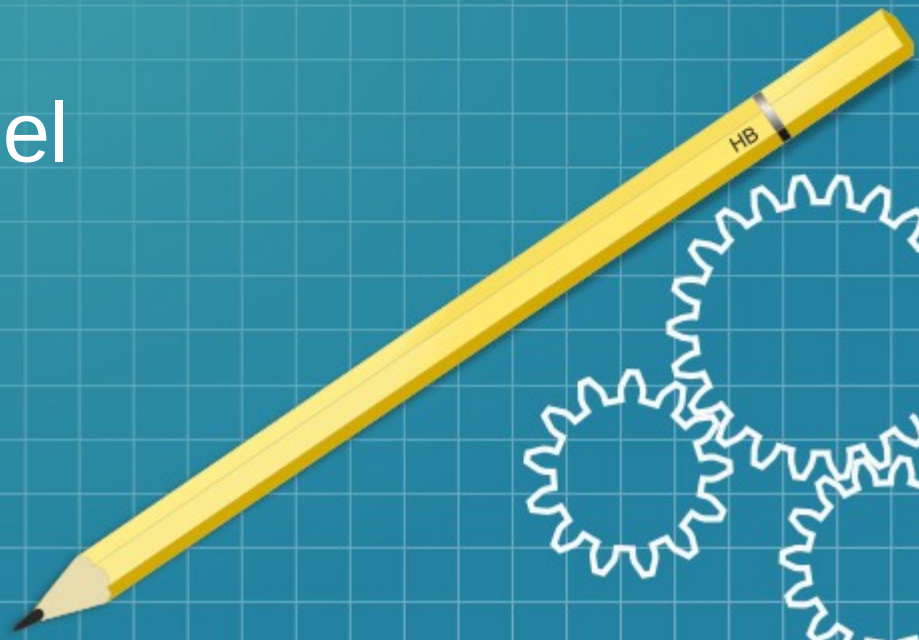
Universal Bootloader

Das U-Boot

Udo Seidel



Agenda

- Introduction
- High level
- Digging deeper
- Take home

About me

- Teacher of mathematics and physics
- PhD in experimental physics
- Started with Linux/Open Source in 1996
- With Amadeus since 2006
- Before:
  - Linux/UNIX trainer
  - Solution Engineer in HPC and CAx environment
- Now: Architecture & Technical Governance

About boot loader

- Linux Loader
- Grand Unified Boot loader
- UEFI
- Systemd-boot aka Gummiboot
- ...

Why?

- Non-x86
- Hardware constraints
- Hardware/board specifics
- Why not? ;-)

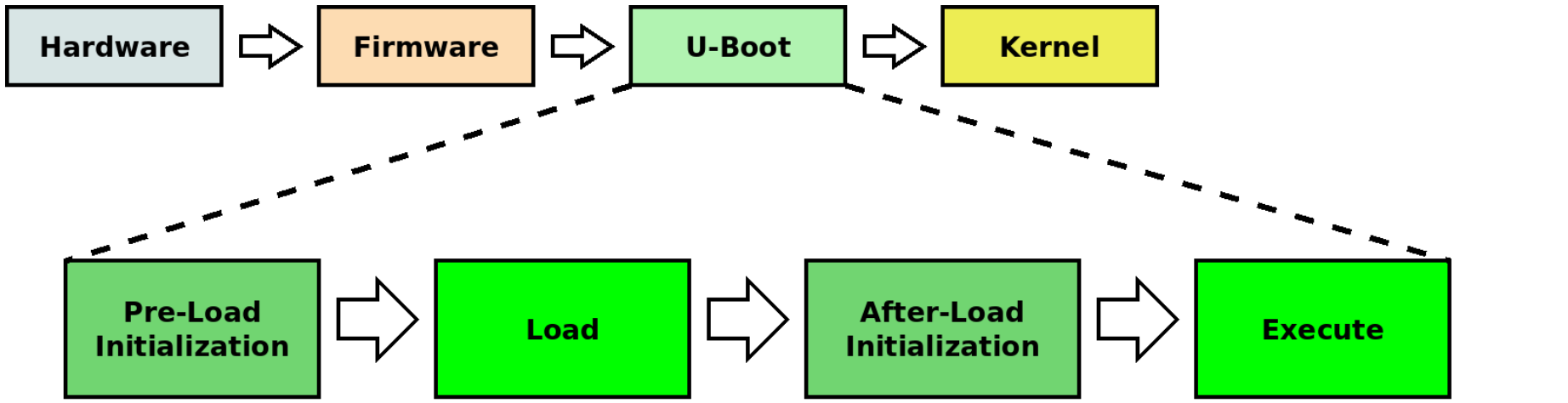
Project summary

- Initial version: 1999 (?)
- License: GPLv2
- Supported platforms: 12 (x86, MIPS, ARM, ...)
- Releases:
  - Every 2 month
  - YYYY.MM naming scheme

History

- 8xxROM for PowerPC
- New name in 2000: PPCBoot
  - Move to Sourceforge
  - 8xxROM 0.3.0 → PPCBoot 1.2
- Again new name in 2002
  - New direction
  - PPCBoot 2.0.0 → U-Boot 0.1.0

High level overview



Autoboot?
Boot configuration
Load Kernel

Usage options

- Boot loader
  - Single
  - Staged
- Boot loader & BIOS

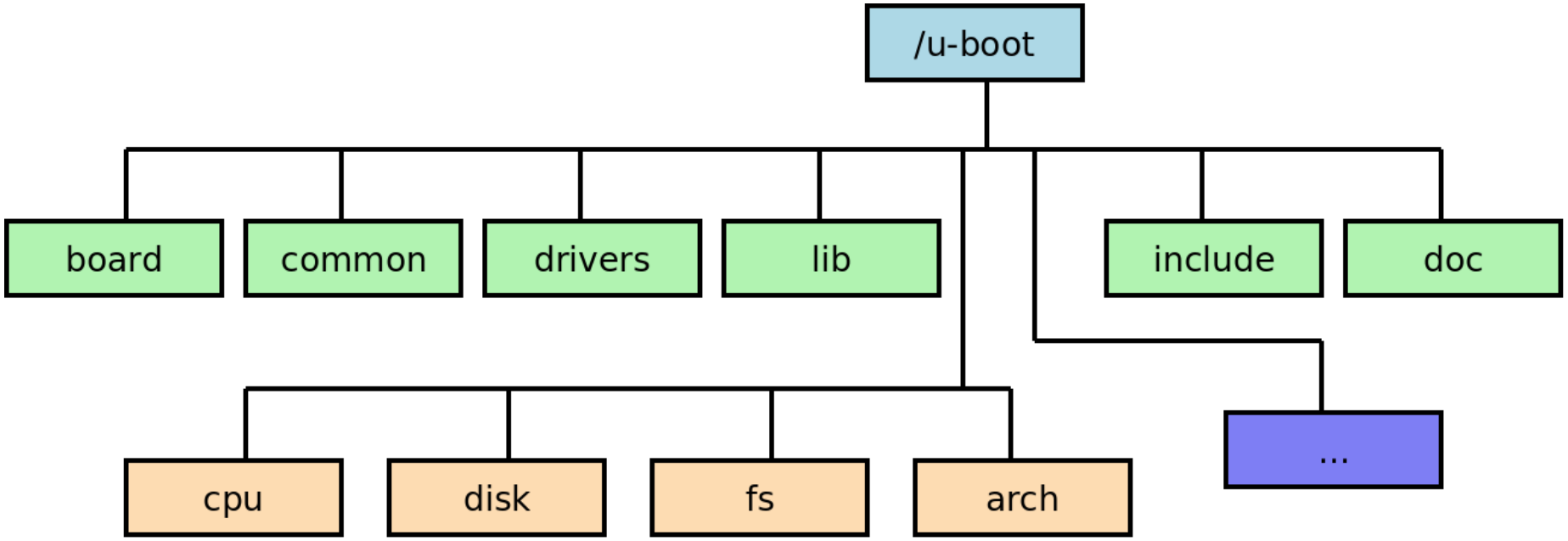
Golden design rules I

- Keep it small
- Keep it fast
- Keep it simple
- Keep it portable
- Keep it configurable

Golden design rules II

- Keep it debuggable
- Keep it usable
- Keep it maintainable
- Keep it beautiful
- Keep it open

Source code structure



Interaction

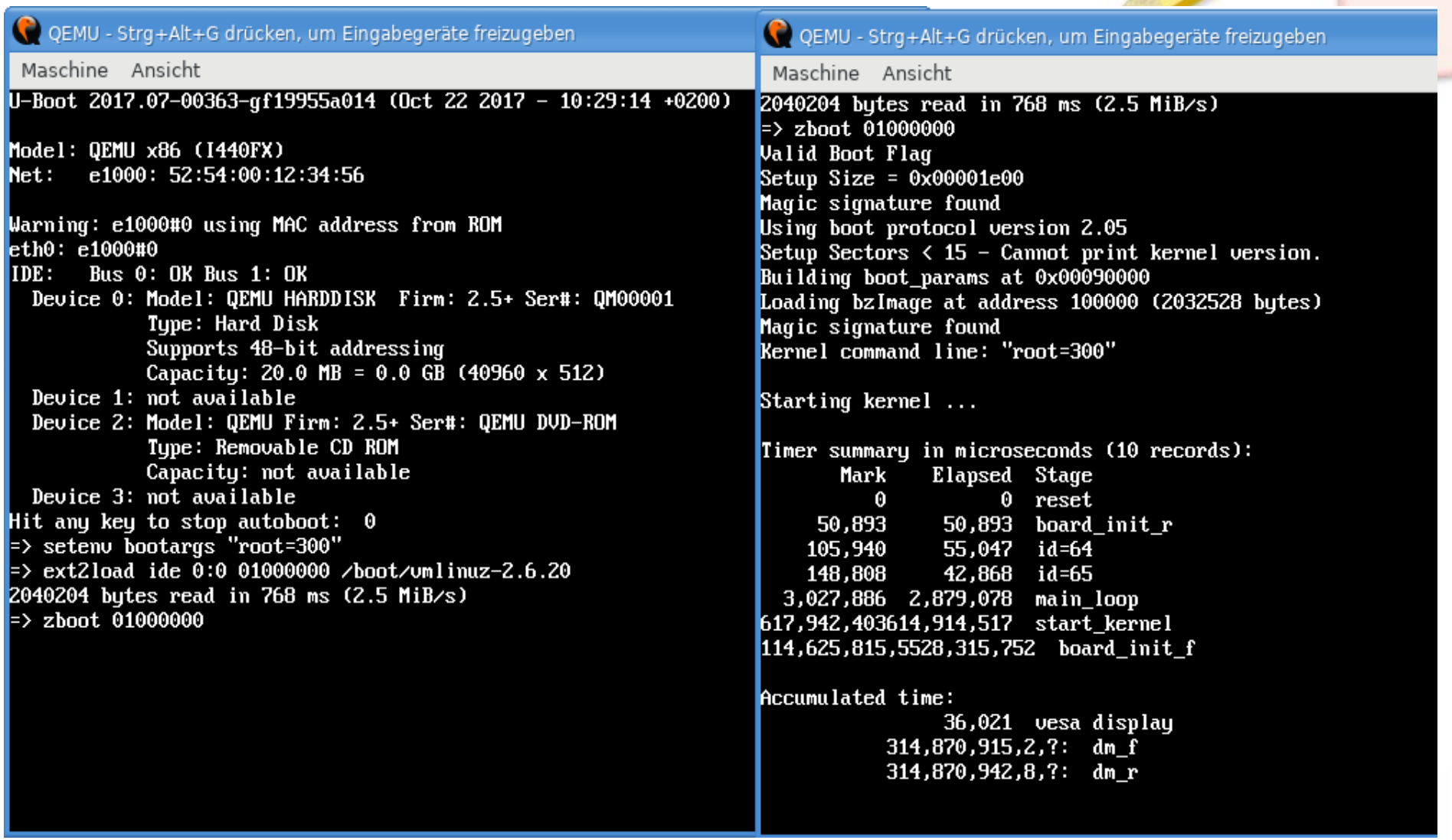
- Command line interpreter HUSH
- Bourne shell like
- Based on busybox

```
=> version
U-Boot 2017.07-00363-gf19955a014 (Jul 22 2017 - 10:29:14 +0200)

gcc (GCC) 7.2.1 20170915 (Red Hat 7.2.1-2)
GNU ld version 2.27-24.fc26
=> help
?       - alias for 'help'
base    - print or set address offset
bdinfo  - print Board Info structure
boot    - boot default, i.e., run 'bootcmd'
bootd   - boot default, i.e., run 'bootcmd'
...
usb     - USB sub-system
usbboot - boot from USB device
version - print monitor, compiler and linker version
zboot   - Boot bzImage
=>
```

Kernel

- Physical location on disk
- Filesystem „location“
  - FAT
  - ExtFS
  - CramFS
  - JFFS2
  - ...
- Network
  - TFTP
  - NFSv2/3


QEMU - Strg+Alt+G drücken, um Eingabegeräte freizugeben
Maschine Ansicht
U-Boot 2017.07-00363-gf19955a014 (Oct 22 2017 - 10:29:14 +0200)
Model: QEMU x86 (I440FX)
Net: e1000: 52:54:00:12:34:56
Warning: e1000#0 using MAC address from ROM
eth0: e1000#0
IDE: Bus 0: OK Bus 1: OK
Device 0: Model: QEMU HARDDISK Firm: 2.5+ Ser#: QM00001
Type: Hard Disk
Supports 48-bit addressing
Capacity: 20.0 MB = 0.0 GB (40960 x 512)
Device 1: not available
Device 2: Model: QEMU Firm: 2.5+ Ser#: QEMU DVD-ROM
Type: Removable CD ROM
Capacity: not available
Device 3: not available
Hit any key to stop autoboot: 0
=> setenv bootargs "root=300"
=> ext2load ide 0:0 01000000 /boot/vmlinuz-2.6.20
2040204 bytes read in 768 ms (2.5 MiB/s)
=> zboot 01000000
QEMU - Strg+Alt+G drücken, um Eingabegeräte freizugeben
Maschine Ansicht
2040204 bytes read in 768 ms (2.5 MiB/s)
=> zboot 01000000
Valid Boot Flag
Setup Size = 0x00001e00
Magic signature found
Using boot protocol version 2.05
Setup Sectors < 15 - Cannot print kernel version.
Building boot_params at 0x00090000
Loading bzImage at address 100000 (2032528 bytes)
Magic signature found
Kernel command line: "root=300"
Starting kernel ...
Timer summary in microseconds (10 records):
Mark Elapsed Stage
0 0 reset
50,893 50,893 board_init_r
105,940 55,047 id=64
148,808 42,868 id=65
3,027,886 2,879,078 main_loop
617,942,403614,914,517 start_kernel
114,625,815,5528,315,752 board_init_f
Accumulated time:
36,021 vesa display
314,870,915,2,?: dm_f
314,870,942,8,?: dm_r

udo@stderr:~/doc/itberuf/vortrag/tuebix2018

```
$ diff -Nur include/configs/x86-common.ORIG.h include/configs/x86-common.h
--- include/configs/x86-common.ORIG.h   2017-07-16 20:25:53.965377664 +0200
+++ include/configs/x86-common.h        2017-07-16 20:35:59.974703873 +0200
@@ -75,9 +75,9 @@
 #define CONFIG_CMD_ZBOOT

 #define CONFIG_BOOTARGS                    \
-       "root=/dev/sdb3 init=/sbin/init rootwait ro"
+       "root=300"
 #define CONFIG_BOOTCOMMAND      \
-       "ext2load scsi 0:3 01000000 /boot/vmlinuz; zboot 01000000"
+       "ext2load ide 0:0 01000000 /boot/vmlinuz-2.6.20; zboot 01000000"

 #if defined(CONFIG_CMD_KGDB)
 #define CONFIG_KGDB_BAUDRATE                    115200
$
$
```

First steps

- Platform?
- Virtual or physical?
- Bootloader only?

Recommended

- Virtual x86 using Qemu
- Lightweight Linux disk
- Bootloader only

Still time & passion

- Combine with coreboot

and/or

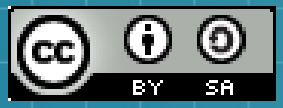
- x-compile for ARM
- Raspberry Pi 2 or 3

Take home

- Small and light
- Multi-platform & -board
- Well documented
- Easy start

Online resources

- http://www.denx.de/wiki/U-Boot/
- http://git.denx.de/?p=u-boot.git;a=summary
- http://elinux.org/RPi_U-Boot
- Internet search :-)



This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License.
It makes use of the works of Mateus Machado Luna.