

openSUSE for ARM

On Your Device

ARM Booting

ARM Booting



Boot Loader

ARM Booting



Boot Loader



Kernel

ARM Booting



Boot Loader

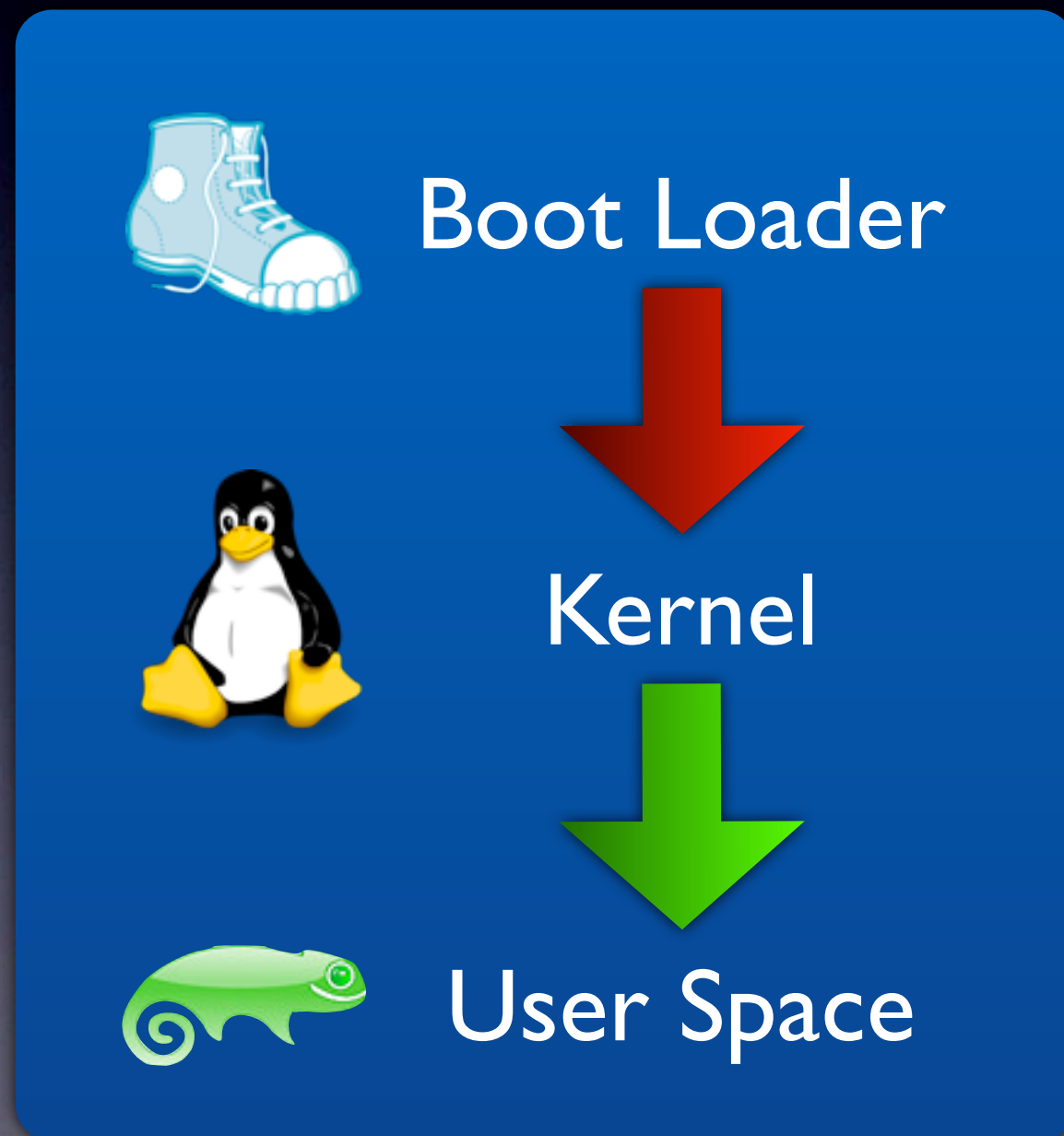


Kernel



User Space

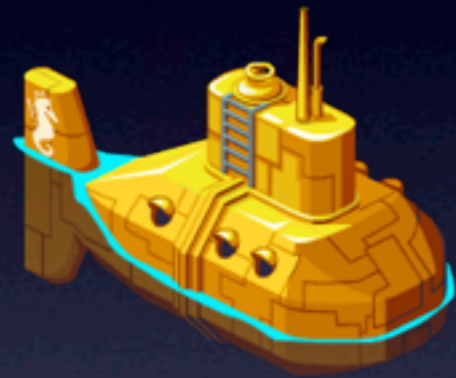
ARM Booting



Appliance



Boot Loader



U-Boot



Others

built-in external



U-Boot

- Default way of booting on ARM developer boards
- Command line
- Scripting
- GPL, so we can modify it to our needs



U-Boot

Built-in





U-Boot

Built-in





U-Boot

Built-in





U-Boot

Built-in

CuBox
** LOADER **

U-Boot 2009.08 (Jan 15 2012 - 00:43:32) Marvell version: 5.4.4 NQ

BootROM:

Version on chip: 2.33
Status: OK
Retries #: 0

Board: CuBox

SoC: 88AP510 (A1)

CPU: Marvell Sheeva (Rev 5)
CPU @ 800Mhz, L2 @ 400Mhz
DDR3 @ 400Mhz, TClock @ 166Mhz

PEX 0: interface detected no Link.

PEX 1: interface detected no Link.

DRAM: 1 GB
CS 0: base 0x00000000 size 512 MB
CS 1: base 0x20000000 size 512 MB
Addresses 12M - 0M are saved for the U-Boot usage.

SF: Detected M25P32 with page size 64 kB, total 4 MB

Streaming disabled

L2 Cache Prefetch disabled

L2 Cache ECC disabled

Modifying CPU/CORE/DDR power rails to 1.0(-2.5%) / 1.0(-5%) / 1.5(-5%)

USB 0: Host Mode

USB 1: Host Mode

Setting VPU power OFF.

Setting GPU power ON.

MMC: MV_SDHCI: 0, MV_SDHCI: 1

Net: egiga0 [PRIME]

Hit any key to stop autoboot: 0





U-Boot

Built-in



boot.scr



U-Boot

Built-in



```
printenv kerneladdr || setenv kerneladdr ${kernel_addr_r}  
printenv ramdiskaddr || setenv ramdiskaddr ${ramdisk_addr_r}  
setenv bootcmd "ext2load mmc 0:1 ${kerneladdr} boot/linux.vmx; \  
                ext2load mmc 0:1 ${ramdiskaddr} boot/initrd; \  
                bootm ${kerneladdr} ${ramdiskaddr}";
```

boot

boot.scr



U-Boot

Built-in

```
printenv kerneladdr || setenv kerneladdr ${kernel_addr_r}
printenv ramdiskaddr || setenv ramdiskaddr ${ramdisk_addr_r}
setenv bootcmd "ext2load mmc 0:1 ${kerneladdr} boot/linux.vmx; \
                ext2load mmc 0:1 ${ramdiskaddr} boot/initrd; \
                bootm ${kerneladdr} ${ramdiskaddr}";

boot
```



boot.scr



U-Boot

Built-in

```
printenv kerneladdr || setenv kerneladdr ${kernel_addr_r}  
printenv ramdiskaddr || setenv ramdiskaddr ${ramdisk_addr_r}  
setenv bootcmd "ext2load mmc 0:1 ${kerneladdr} boot/linux.vmx; \  
                ext2load mmc 0:1 ${ramdiskaddr} boot/initrd; \  
                bootm ${kerneladdr} ${ramdiskaddr}";
```

boot



boot.scr





U-Boot

Built-in

```
printenv kerneladdr || setenv kerneladdr ${kernel_addr_r}  
printenv ramdiskaddr || setenv ramdiskaddr ${ramdisk_addr_r}  
setenv bootcmd "ext2load mmc 0:1 ${kerneladdr} boot/linux.vmx; \  
                ext2load mmc 0:1 ${ramdiskaddr} boot/initrd; \  
                bootm ${kerneladdr} ${ramdiskaddr}";
```

boot



boot.scr



linux.vmx

initrd



U-Boot

Built-in



boot.scr



linux.vmx

initrd



U-Boot

Built-in



boot.scr



linux.vmx

initrd



U-Boot

Built-in

Starting kernel ...

```
Uncompressing Linux... done, booting the kernel.
[ 0.000000] Booting Linux on physical CPU 0
[ 0.000000] Initializing cgroup subsys cpuset
[ 0.000000] Initializing cgroup subsys cpu
[ 0.000000] Linux version 3.6.0-rc7-17-cubox
(abuild@build14) (gcc version 4.7.1 20120723 [gcc-4_7-branch
revision 189773] (SUSE Linux) ) #2 Mon Oct 1 02:19:54 UTC 2012
[ 0.000000] CPU: ARMv7 Processor [560f5815] revision 5
(ARMv7), cr=10c53c7d
[ 0.000000] CPU: PIPT / VIPT nonaliasing data cache, PIPT
instruction cache
[ 0.000000] Machine: SolidRun CuBox
[ 0.000000] Ignoring unrecognised tag 0x41000403
[ 0.000000] Ignoring unrecognised tag 0x41000404
[ 0.000000] Memory policy: ECC disabled, Data cache
writeback
[ 0.000000] Built 1 zonelists in Zone order, mobility
grouping on. Total pages: 260096
[ 0.000000] Kernel command line: loader=uboot disk=/dev/
disk/by-id/mmc-SU04G_0x0509b5dc resume=openSUSE-Factory-ARM-
cubox console=ttyS0,115200n8
[...]
```



boot.scr



linux.vmx

initrd



U-Boot

Built-in



boot.scr



linux.vmx

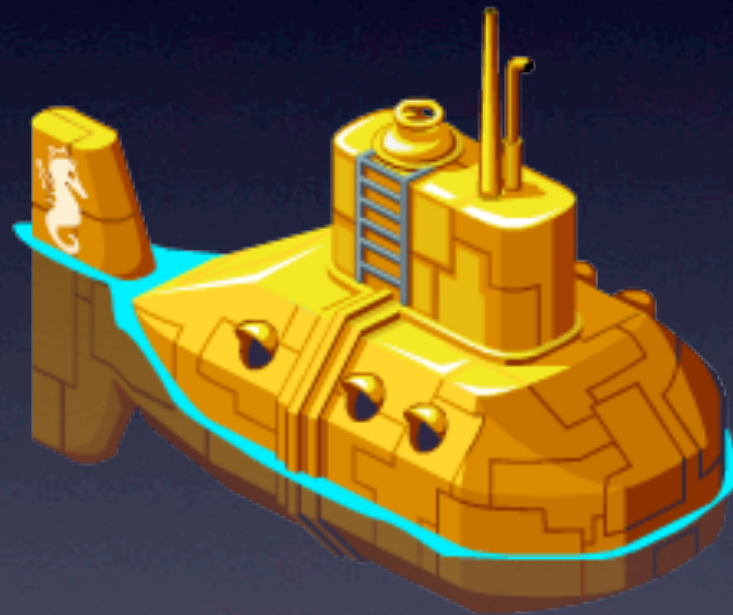
initrd

Starting kernel ...

```
Uncompressing Linux... done, booting the kernel.
[ 0.000000] Booting Linux on physical CPU 0
[ 0.000000] Initializing cgroup subsys cpuset
[ 0.000000] Initializing cgroup subsys cpu
[ 0.000000] Linux version 3.6.0-rc7-17-cubox
(abuild@build14) (gcc version 4.7.1 20120723 [gcc-4_7-branch
revision 189773] (SUSE Linux) ) #2 Mon Oct 1 02:19:54 UTC 2012
[ 0.000000] CPU: ARMv7 Processor [560f5815] revision 5
(ARMv7), cr=10c53c7d
[ 0.000000] CPU: PIPT / VIPT nonaliasing data cache, PIPT
instruction cache
[ 0.000000] Machine: SolidRun CuBox
[ 0.000000] Ignoring unrecognised tag 0x41000403
[ 0.000000] Ignoring unrecognised tag 0x41000404
[ 0.000000] Memory policy: ECC disabled, Data cache
writeback
[ 0.000000] Built 1 zonelists in Zone order, mobility
grouping on. Total pages: 260096
[ 0.000000] Kernel command line: loader=uboot disk=/dev/
disk/by-id/mmc-SU04G_0x0509b5dc resume=openSUSE-Factory-ARM-
cubox console=ttyS0,115200n8
[...]
```




U-Boot

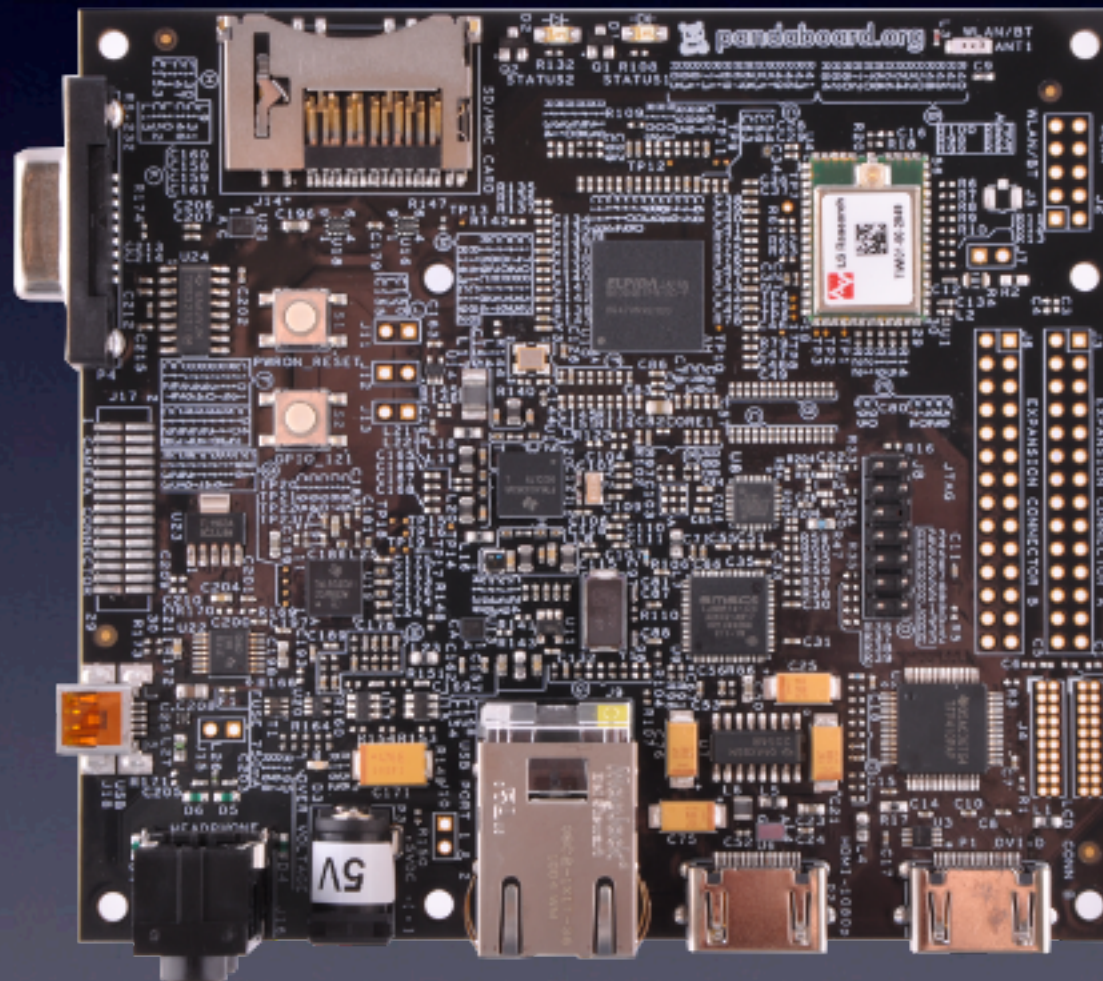


External



U-Boot

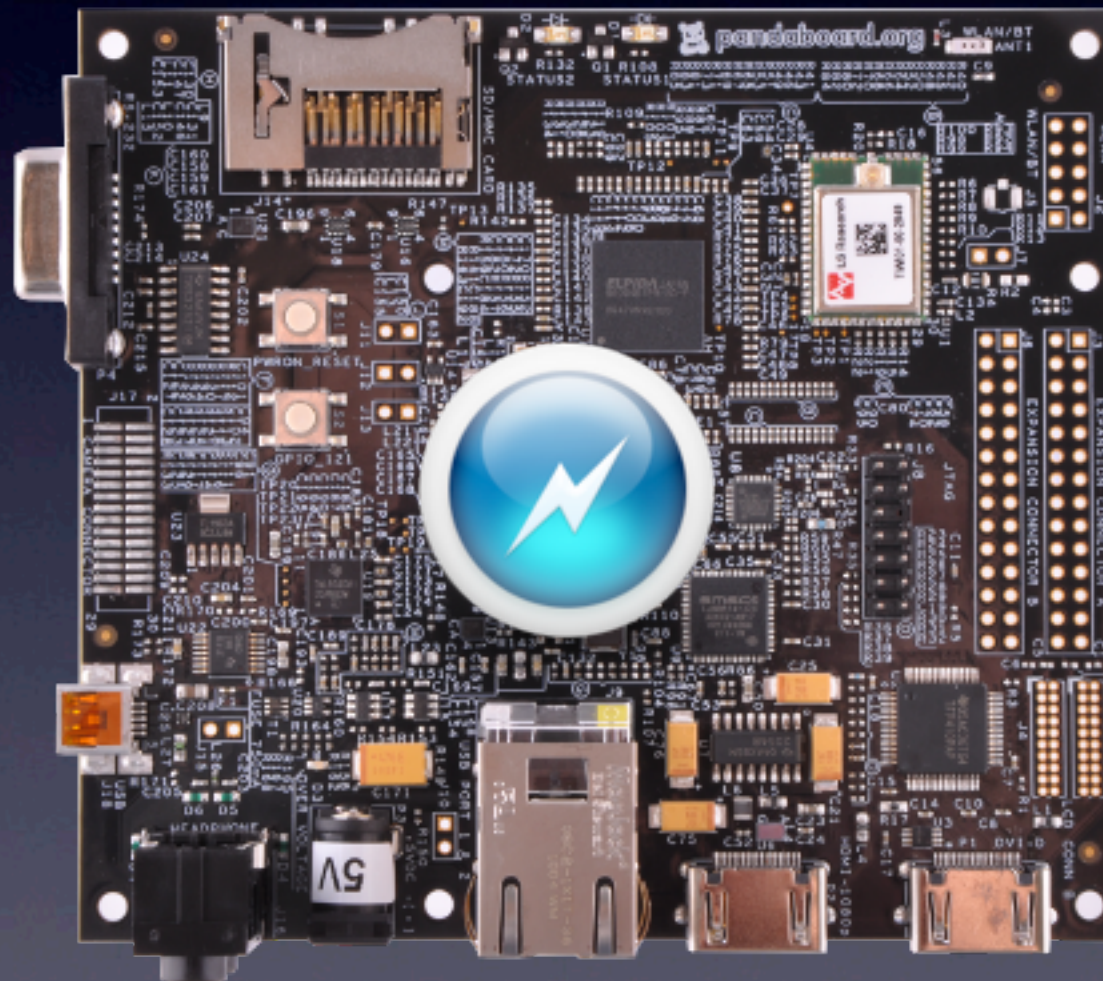
External





U-Boot

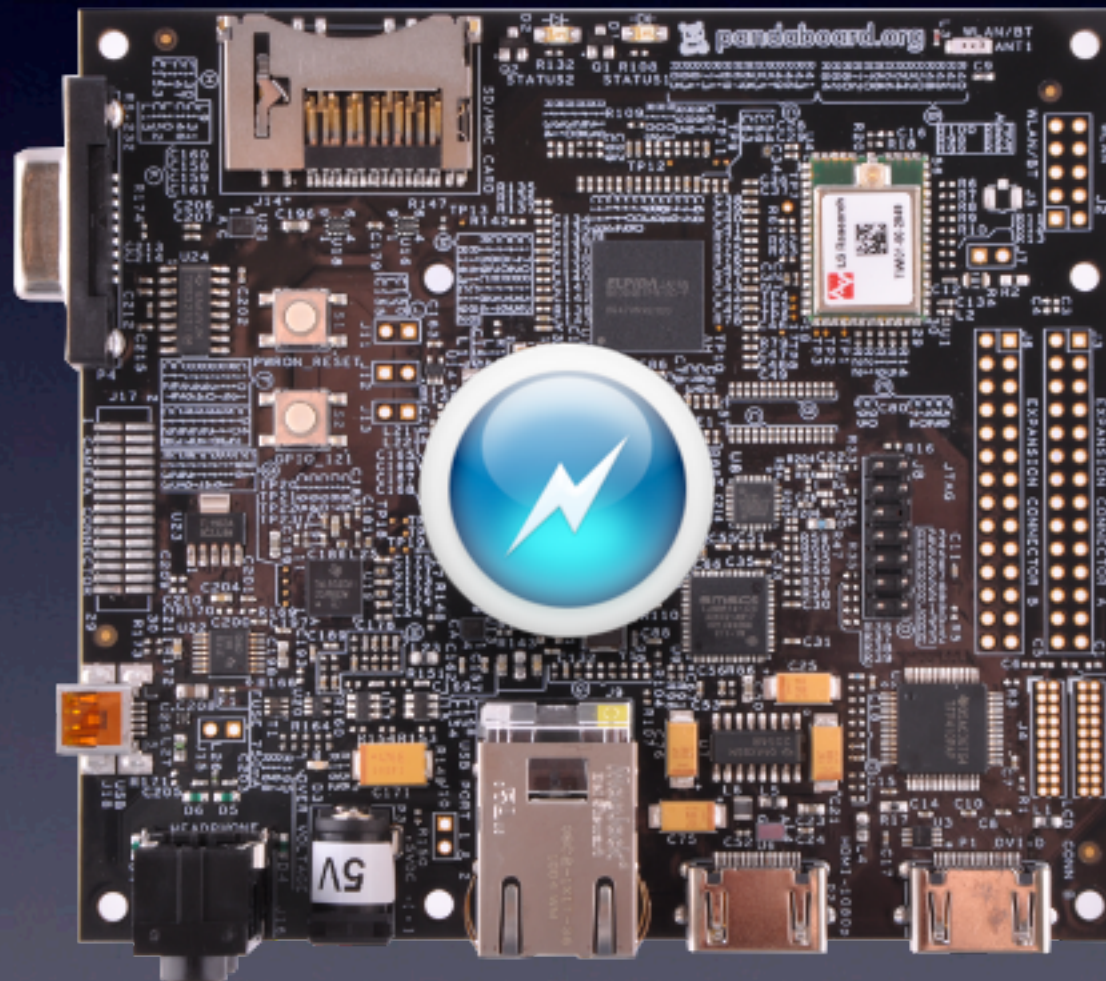
External





U-Boot

External





U-Boot

External





U-Boot

External



SPL



U-Boot

External

SPL





U-Boot

External

SPL



/boot



U-Boot

External



U-Boot 2012.04.01 (Oct 18 2012 - 10:24:59) for ORIGIN

CPU: S5PC210@1000MHz

Board: ORIGIN

DRAM: 1 GiB

WARNING: Caches not enabled

MMC: SAMSUNG SD/MMC: 0

*** Warning - bad CRC, using default environment

In: serial

Out: serial

Err: serial

Hit any key to stop autoboot: 0



U-Boot

External



- Rest of boot identical to built-in u-boot



Other bootloaders





Other bootloaders



- Some OSes have special bootloaders
- Load specific format straight on
- Bad for openSUSE
- Best bet: Get u-boot running



Boot Loader



Kernel



User Space



Boot Loader



Kernel



User Space



Boot Loader

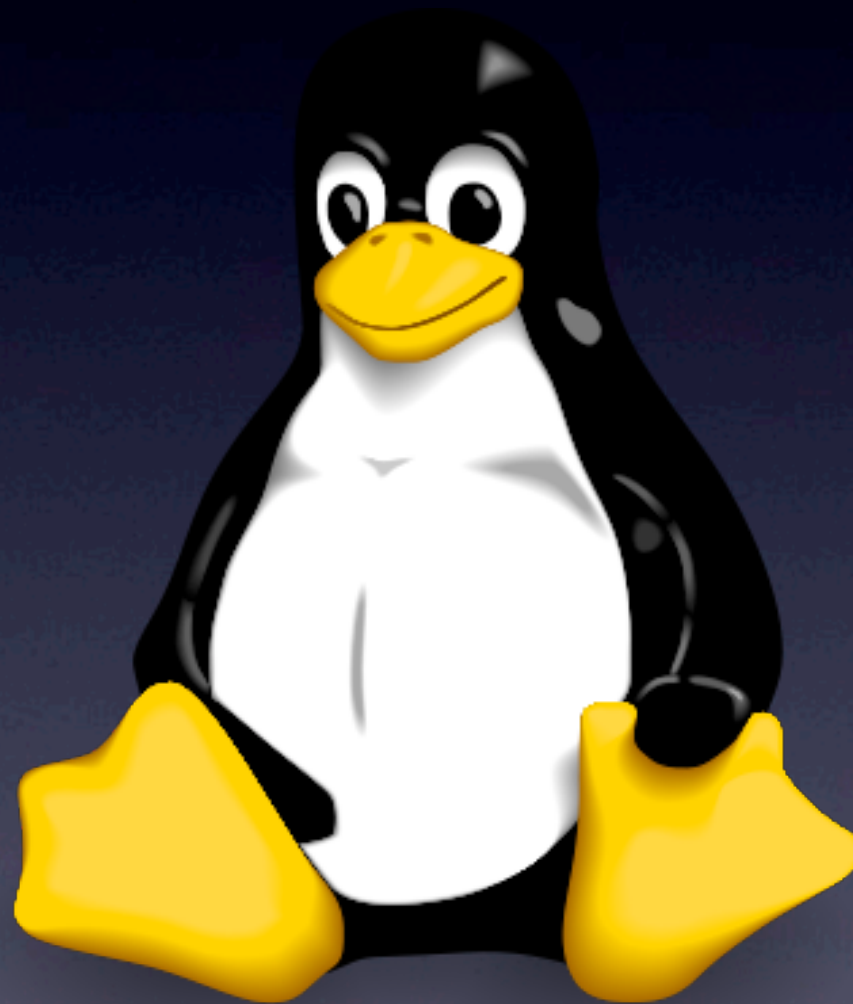


Kernel



User Space

Kernel





Kernel



Upstream



Kernel



Upstream



Downstream



Kernel



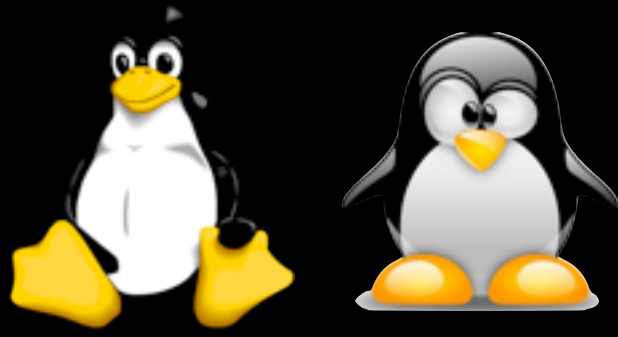
Upstream



Downstream

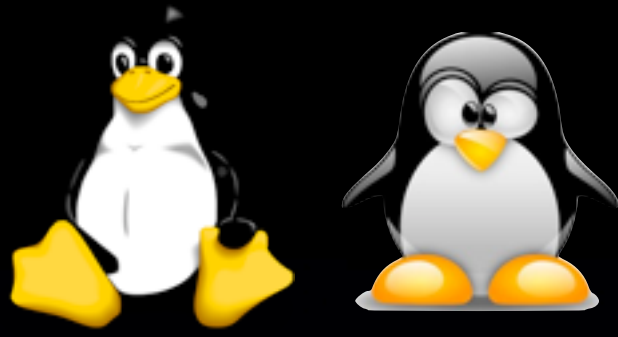


Proprietary



Upstream Kernel

- Code is in Linus' git tree
- Automatically pulled in by openSUSE
- Doesn't support all devices usually
- Different kernels have different targets
 - Getting unified in 'single zImage' effort



Upstream Kernel

- OMAP3-5
 - Beagle Board
 - Panda Board
- Highbank
- i.MX 51,53,6 (no gfx)
- Samsung Exynos 4,5
- ...



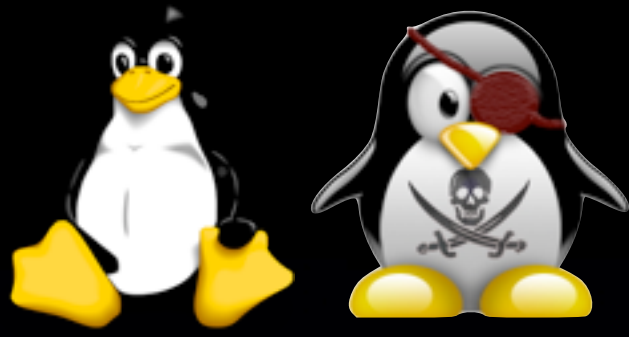
Kernel



Kernel

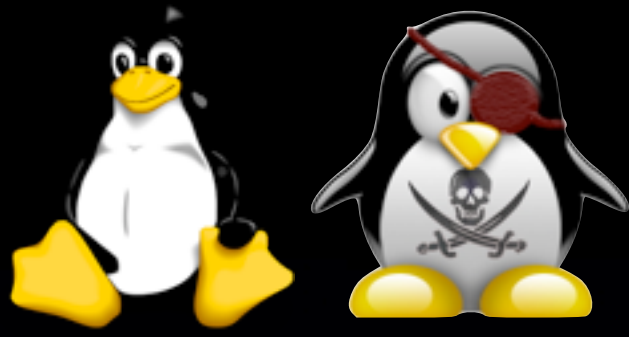


Downstream



Downstream Kernel

- Some kernel tree on the internet
- Open Source, but not mainline
- Not picked up by openSUSE
- “Normal” case in the ARM world
- Supports a board/SoC completely



Downstream Kernel

- Raspberry PI
- Efika MX (2.6.31)
- NVidia Tegra AC100



Kernel



Kernel

Proprietary



Kernel



Proprietary



Proprietary Kernel

- Can not be downloaded anywhere
- Does not adhere to the GPL
- Ask / sue your device manufacturer



Boot Loader



Kernel



User Space



Boot Loader



Kernel



User Space



Boot Loader



Kernel



User Space





SUSE Rootfs





SUSE Rootfs



SPL



SUSE Rootfs



SPL





SUSE Rootfs



SPL





SUSE Rootfs



http://download.opensuse.org/repositories/openSUSE:/12.2:/ARM/images/openSUSE-12.2-ARM-JeOS.armv7l-*.tbz



SUSE Rootfs

- Download .tbz
- Extract to a partition
- Boot from it!



Starting Command Scheduler...

Started Command Scheduler

[OK]

Starting LSB: Configure the remote-fs depending network interfaces...

Starting OpenSSH Daemon...

Starting /etc/init.d/boot.local Compatibility...

Started /etc/init.d/boot.local Compatibility

[OK]

Starting Wait for Plymouth Boot Screen to Quit...

Starting Terminate Plymouth Boot Screen...

Welcome to openSUSE 12.3 "Mantis" - Kernel 3.6.0-rc7-17-cubox (ttyS0).

linux login:



Done, are we?



openSUSE way

- Packaging
- Appliance building



OBS

- Create your own repo in your home directory for your target
- copy meta prj and meta prjconf from openSUSE:12.2:Contrib:* to your project
- Adjust them for your target



U-boot

- Boot from ext2/3, no FAT
- Provide \$kerneladdr and \$ramdiskaddr
- Default openSUSE boot.scr on SD card
- Package it as rpm to /boot/u-boot.bin



Kernel



Upstream



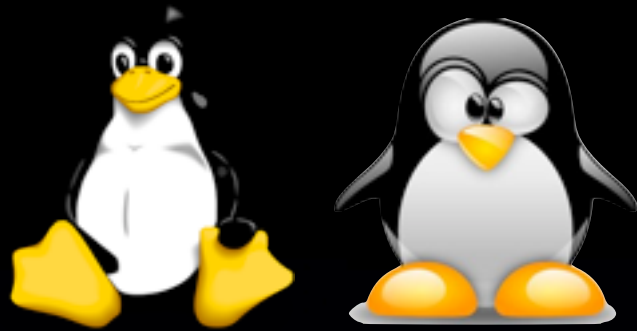
Downstream



Kernel

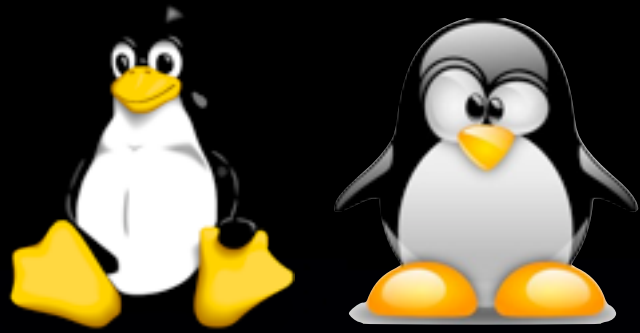


Upstream



Upstream

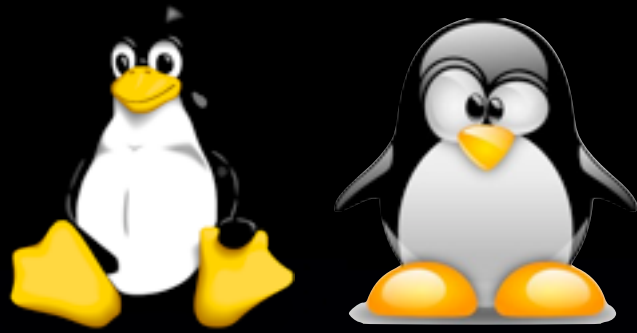
- Fetch openSUSE kernel git
- Add config for your SoC / board
- Update your config for openSUSE kernel
- Package it up as rpm source
- Build it



Upstream

- Fetch openSUSE kernel git

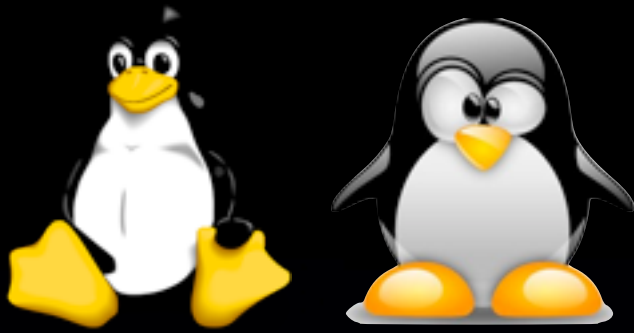
```
$ git clone git://kernel.opensuse.org/kernel-source.git
Initialized empty Git repository in /dev/shm/kernel-source/.git/
remote: Counting objects: 213374, done.
remote: Compressing objects: 100% (69607/69607), done.
remote: Total 213374 (delta 144761), reused 208009 (delta 141278)
Receiving objects: 100% (213374/213374), 861.77 MiB | 11600 KiB/
s, done.
Resolving deltas: 100% (144761/144761), done.
```



Upstream

- Add config for your SoC / board

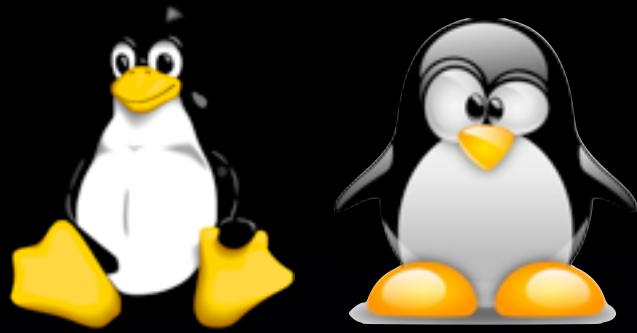
```
$ cd kernel-source  
$ cp /working_kernel/.config config/armv7hl/<your target>  
$ vi config.conf  
<add your target to +armv7hl list>
```

Upstream

- Update your config for openSUSE kernel

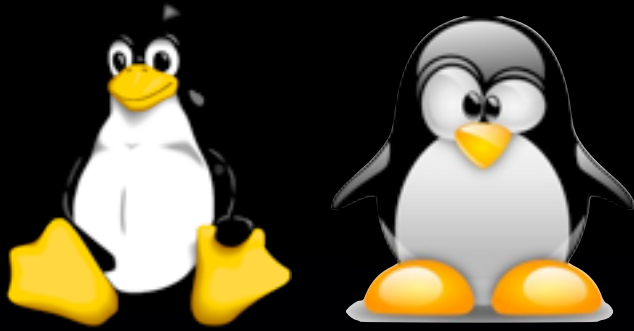
```
$ ./scripts/sequence-patch.sh
./scripts/sequence-patch.sh
WARNING: You should run ./scripts/install-git-hooks to enable
pre-commit checks.
SCRATCH_AREA not defined (defaulting to "tmp")
Creating tree in /dev/shm/kernel-source/tmp/linux-3.6-master
Cleaning up from previous run (background)
Linking from /dev/shm/kernel-source/tmp/linux-3.6.orig
[ patches.kernel.org/patch-3.6.1 ]
[ patches.kernel.org/patch-3.6.1-2 ]
[...]
$ cd tmp/linux-3.6-master/
$ ./run-oldconfig.sh --arch armv7hl --flavor <your target>
```



Upstream

- Package it up as rpm source

```
$ ./scripts/tar-up.sh
WARNING: You should run ./scripts/install-git-hooks to enable
pre-commit checks.
linux-3.6.tar.bz2
config.tar.bz2
patches.apparmor.tar.bz2
patches.arch.tar.bz2
patches.drivers.tar.bz2
patches.fixes.tar.bz2
patches.kernel.org.tar.bz2
patches.rpmify.tar.bz2
[...]
```



Upstream

- Build it

```
$ cd kernel-source
$ osc build --alternative-project openSUSE:12.2:ARM standard
armv7l kernel-<target>.spec
WARNING: package is not existing on server yet
WARNING: source service from package or project will not be
executed. This may not be the same build as on server!
Building kernel-omap2plus.spec for standard/armv7l
Getting buildinfo from server and store to /dev/shm/kernel-
source/kernel-source/_buildinfo-standard-armv7l.xml
Getting buildconfig from server and store to /dev/shm/kernel-
source/kernel-source/_buildconfig-standard-armv7l
Updating cache of required packages
[...]
```




Kernel



Upstream



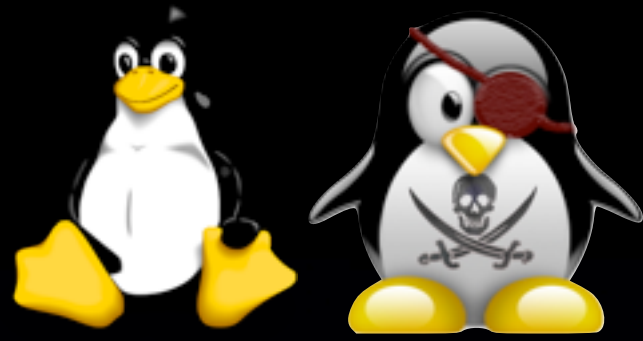
Downstream



Kernel

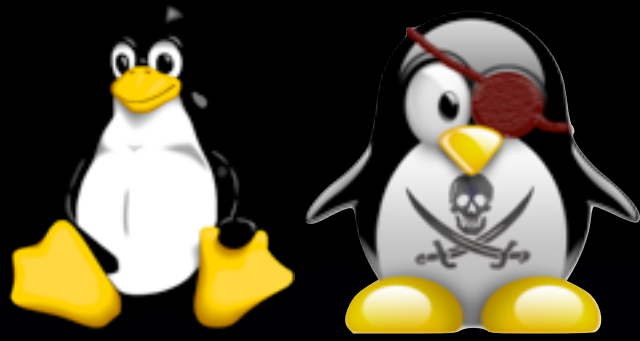


Downstream



Downstream

- Good luck :)

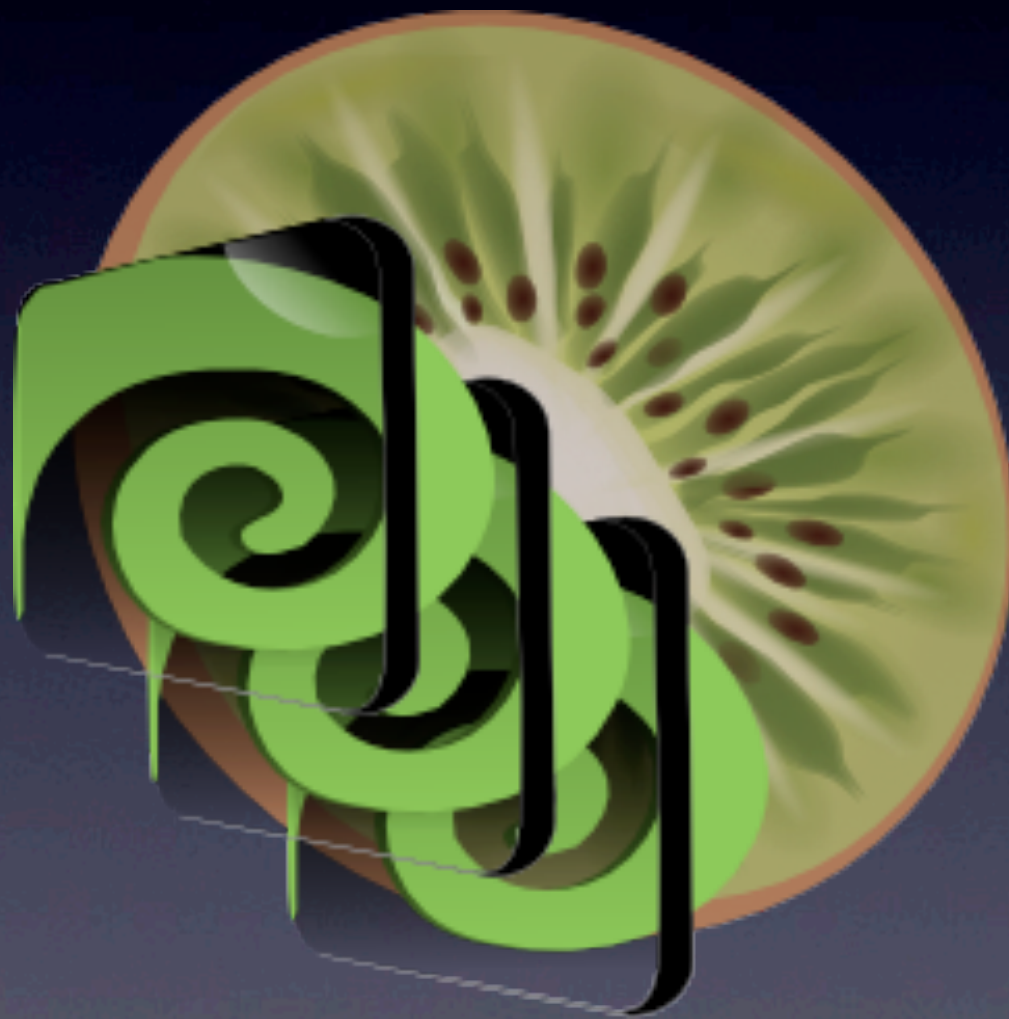


Downstream

- Check out the Raspberry Pi kernel image

[https://build.opensuse.org/package/show?
package=kernel-
raspberrypi&project=openSUSE:Factory:ARM:C
ontrib:RaspberryPi](https://build.opensuse.org/package/show?package=kernel-raspberrypi&project=openSUSE:Factory:ARM:Contrib:RaspberryPi)

Kiwi





Kiwi

- Easy framework to create your own image
- Create JeOS description in your repo
- Change config according to your target
- Fine tune U-boot setup scripts



Kiwi

- Create JeOS description in your repo

```
$ osc copypac openSUSE:12.2:ARM:Contrib:Highbank JeOS-highbank  
<your repo> JeOS-<target>  
$ osc co <your repo> JeOS-<target>  
$ cd <your repo>/JeOS-<target>  
$ mv JeOS-highbank.kiwi JeOS-<target>.kiwi  
$ vi JeOS-<target>.kiwi
```



Kiwi

- Change config according to your target

```
<image schemaversion="5.5" name="openSUSE-12.2-ARM-JeOS-highbank">
  <preferences>
    <type image="oem" filesystem="ext3" boot="oemboot/suse-12.2"
bootloader="uboot" kernelcmdline="console=ttyAMA0" editbootconfig="uboot-
image-setup" editbootinstall="uboot-image-install">
    </type>
  </preferences>
  <repository type="rpm-md">
    <source path="obs://openSUSE:12.2:ARM/standard"/>
  </repository>
  <repository type="rpm-md">
    <source path="obs://openSUSE:12.2:ARM:Contrib:Highbank/standard" /
>
  </repository>
  <packages type="bootstrap">
    <package name="kernel-highbank" bootinclude="true"/>
  </packages>
  [...]
</image>
```



Kiwi

- Build it

```
$ osc build *.kiwi images armv7l
Building JeOS-panda.kiwi for images/armv7l
Overriding config value for packagecachedir='/var/tmp/osbuild-packagecache'
with '/studio/arm/pkgs/'
Getting buildinfo from server and store to /studio/arm/openSUSE:12.2:ARM/
JeOS/.osc/_buildinfo-images-armv7l.xml
Getting buildconfig from server and store to /studio/arm/openSUSE:12.2:ARM/
JeOS/.osc/_buildconfig-images-armv7l
Updating cache of required packages
0.0% cache miss. 1440/1440 dependencies cached.
[...]
```


Boot up and debug!