Debian Linux support for ARM MPCore and Open Platforms

Wookey
(Aleph1 / Balloonz / Toby Churchill Ltd)
`wookey@wookware.org`

ARM Developers Conference
& Design Pavillion 2007,
Santa Clara

What is this talk about?

- Debian and its ports, infrastructure and mechanisms
- EABI changes
- Porting choices and process
- Current Status

What is this talk about?

- Debian and its ports, infrastructure and mechanisms
- EABI changes
- Porting choices and process
- Current Status

What is this talk about?

- Debian and its ports, infrastructure and mechanisms
- EABI changes
- Porting choices and process
- Current Status

What is this talk about?

- Debian and its ports, infrastructure and mechanisms
- EABI changes
- Porting choices and process
- Current Status

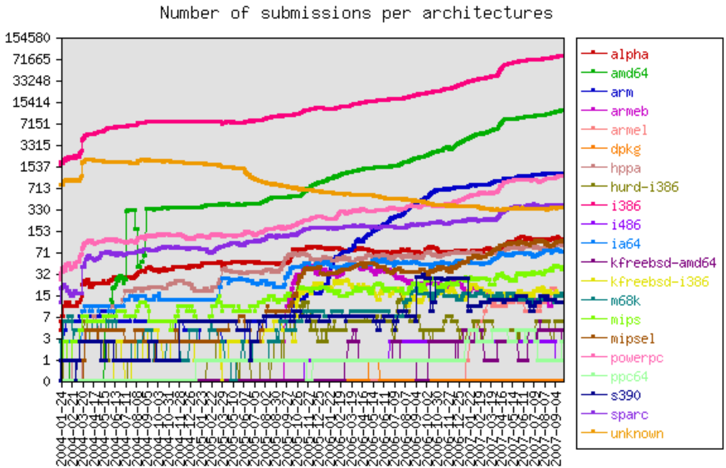
Debian - The Universal OS

- Handhelds to Mainframes
- Release architectures:
    - alpha, amd64, arm, hppa, i386, ia64, mips, mipsel, powerpc, s390, sparc
- Non released official architectures:
    - hurd-386, m68k
- debian-installer, emdebian
- Unofficial builds adds:
    - armeb, kfreebsd-amd64, kfreebsd-i386, m32r
- External projects:
    - nexenta (solaris kernel)

Some Statistics

- The unstable distribution has
  - 10,783 source packages
  - 18,317 binary (arm) packages
  - 2,042,254 files
  - (+316 source, 356 binary in contrib/non-free)
- 96% of i386 packages build for arm
- 99% of suitable packages
- about 1000 developers
- 13GB per architecture

ARM port popularity



All non-x86 downloads total about 5%

Historical Context

- ARM kernel port created in 1998
- Used GCC’s C calling convention for arm
- Userspace to kernel syscall interface designed to pass 5 or more arguments efficiently (via registers). Similar to RISCOS conventions, without condition codes to indicate errors.
- Floating point was done with FP instruction set. Executed by FPU if present, emulated if not.

Supported machines

Debian-arm port started in 2000

- Netwinder: 2000
- RiscPC, Cats: 2001
- Lart, Bast: 2003
- Iyonix, Manga: 2004
- NSLU2: 2005
- Thecus, Allnet: 2006
- Versatile: 2007

Many others without debian-installer support

Architecture Release Criteria

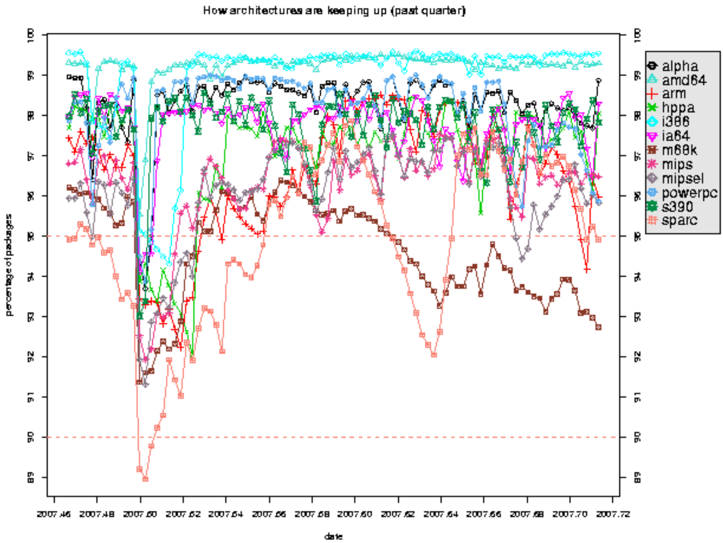
- Available in the market without NDA
- 50 users, 5 developers
- Working installer
- Upstream and debian porter support, esp toolchain
- 95% of archive built
- Must keep up with 2 buildds (relaxed for arm and m68k)
- Veto by release team, security team.

Process

Everything is autobuilt, except uploaded package.

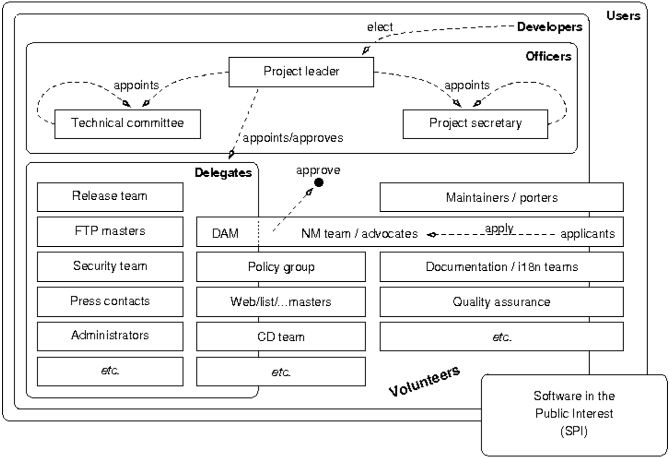
- DD builds, signs, uploads
- Buildd network processes:
    - Checks dependencies
    - Hands out for build
    - Classified as 'built-OK', 'maybe-successful', 'failed'.
- Packages migrate when:
    - All arches are done
    - No release-critical bugs filed
    - Ready for 10 days in unstable
    - Dependencies satisfied
- Build failure on any release arch will block.

Architectures - percent-built



Dealing with Debian

- (Lack of) structure can be hard



Dealing with Debian - 2

- Find the responsible people
- Enable them with hardware/time/information
- Solutions must be general

Main EABI changes

- Structure packing
  - Old ABI had minimum structure packing size of 4 bytes
  - EABI has no minimum - packing determined by type sizes
- Argument alignment
  - 8-byte stack alignment at public function entry points (Old ABI was 4-bytes)
  - 64-bit data types (e.g. long long) are 8-byte aligned (Old ABI was 4-bytes)
- Enums
  - EABI allows enums to have variable type size (-mabi=aapcs)
  - Not used on GNU/Linux - they remain as 4-bytes. (-mabi=aapcs-linux)
- Floating point
  - Mixed-endian LE format goes away
  - Can mix GCC softfloat and FPU hardfloat/emulation

Thumb interworking

EABI allows thumb/arm mixing at function level granularity
Not supported on v3 or v4

- Current GCC:
  - `-march=armv4:  mov pc,lr`
    v4 onwards, only interworking-safe from v7
  - `-march=armv4t:  bx lr`
    v4t onwards, interworking-safe
- Modified GCC:
  - `tst lr, #1; moveq pc, lr; bx lr`
    v4 onwards, interworking on v4t onwards. extra instructions
  - `ldm/ldr:`
    v4 onwards, interworking on v5t onwards.

Debian maximises device coverage, not speed

New kernel syscall convention

Example: long ftruncate64(unsigned int fd, loff_t length) (syscall number 194):

- legacy ABI:
  - - put fd into r0
  - - put length into r1-r2
  - - use "`swi #(0x900000 + 194)`" to call the kernel
- Better on von Neuman architecture - already in cache
- EABI:
  - - put fd into r0
  - - put length into r2-r3 (skipping over r1)
  - - put 194 into r7
  - - use "`swi 0`" to call the kernel
- Better on Harvard architecture - doesn't pollute data cache

New syscalls (2)

- Changed in kernel 2.6.15 - mainline 2.6.16
- Kernel supports old syscalls (no speed gain)
- glibc 2.3.6 used old syscalls - 2.4 uses new

Why does Debian care?

Pros

- Most arm weirdness removed (FP formats, packing, C++ exceptions)
- Hard/soft float interworking (soft-float *much* faster)
- Standardisation across toolchains, debuggers
- Thumb interworking
- More efficient syscall convention
- Interchangeable binaries (PalmOS, GNU/Linux, Symbian OS)

Cons

- Almost total incompatibility with existing port

Debian port to eabi

- Worth changing to:
  - Avoid obsolescence
  - Fix the FP problem
  - Build stuff that never worked
- Lesser gains:
  - Binary compatibility (can use commerical debuggers)
- Incompatibility with existing port a problem...

How to make the change?

4 possible schemes:

- Rename all library packages
- New architecture
- ABI: field in control file
- Conflicting libc packages

Rename all library packages

Standard debian ABI change model

1. libfoo renamed to libfooeabi and uploaded
2. packages changed to depend on libfooeabi
3. wait till nothing depends on libfoo
4. remove libfoo
5. next libfoo abi change rename libfooeabi back to libfoo
6. leaves a few oddly-named packages: zlib1g, libfam102c

Rename all library packages-2

Pros:

- Can do apt-get dist-upgrade

Cons:

- Every single library package needs to be renamed
- Will take a long time, during which unstable will be broken for all arches (6months for C++, so 2yrs?)
- Not popular due to large hassle for other arches
- Will lose v3, (and maybe v4) support.

New architecture

Pros:

- Fits with gcc approach
- Does not affect non-arm arches
- Can keep ‘arm’ for v3 (and maybe v4) machines
- Can be done relatively quickly as no interaction with other arches/releases

Cons:

- Current arm users don’t have easy upgrade path
- Need archive space for new arch

ABI: field in control file

Suggested as part of multiarch proposal

Pros:

- Reflects ABI correctly, would help other transitions too
- Technically best?

Cons:

- No existing implementation
- No consensus on including it yet
- Questions over resolving dependencies and how it fits into archive
- Will lose v3 (and maybe v4) support

Conflicting libc packages

Make a libc6-eabi-dev depending on eabi and ld-linux.so.3, that conflicts with libc6.

Pros:

- Only have to change glibc (and rebuild everything)
- Does not affect other arches

Cons:

- Most of port will be uninstallable for a very long time
- apt-get dist-upgrade still won't work due to huge number of conflicts
- Will lose v3, (and maybe v4) support

A decision was made

- ‘New architecture’ won.
- But what to call it?
- best name already taken
- ‘arm-gnueabi’, ‘armeabi’, ‘earm’, ‘newarm’?
- Discussion online and Extremedura April 2006
- Called ‘armel’
- ‘armeb’ status somewhat confused

A decision was made

- ‘New architecture’ won.
- But what to call it?
- best name already taken
- ‘arm-gnueabi’, ‘armeabi’, ‘earm’, ‘newarm’?
- Discussion online and Extremedura April 2006
- Called ‘armel’
- ‘armeb’ status somewhat confused

Debian port process - in principle

1. Get working eabi toolchain
2. Get working eabi kernel
3. Get working eabi Rootfs
4. Patch/build armel from debian sources
5. Debootstrap Buildd
6. Publish build logs
7. File bugs/patches for issues

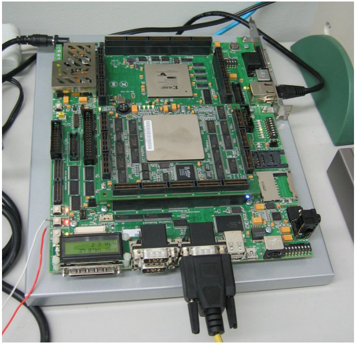
Bootstrapping Debian is hard

- Not designed to be built from scratch
- Not cross-buildable
- No docs for a reason!
- Circular dependencies
  - qt/doxygen
  - ldap/kerberos
  - gettext wants java
  - Doc-building: groff, tetex, dvi, ps2html
  - Patches needed to simplify
- 29 essential packages
- 124 base and required packages
- 16 build-essential packages
- 1000-odd build dependencies

Bootstrapping mechanisms

- Scratchbox+crocodile.
- Maemo - old syscalls/glibc
- OpenEmbedded angstrom.

Builds using mpcore board, QEMU and, later, thecus.



Port process - in practice

- 3-stages needed
  1. Bodge a working eabi rootfs to build in
  2. Build armel packages - tainted but adequate
     Quite a small set of patches needed - 40-odd
  3. Debootstrap armel packages
     Rebuild kosher packages

Typical patches

```
-Architecture: alpha amd64 arm hppa i386 ia64 powerpc ppc64 sparc
+Architecture: alpha amd64 arm armeb armel hppa i386 ia64 powerpc ppc64 sparc
```

```
-  java-gcj-compat-dev [!m68k !mips !mipsel !arm !hurd-i386],
+  java-gcj-compat-dev [!m68k !mips !mipsel !arm !armel !hurd-i386],
```

```
-#if defined(__arm) || defined(__arm__) || defined(__arm26__) \
-    || defined(__arm32__)
+#if (defined(__arm) || defined(__arm__) || defined(__arm26__) \
+    || defined(__arm32__)) && !defined(__ARM_EABI__) && !defined(__ARMEB__)
 #define IEEE_ARM
```

Timeline

- Code sourcery 1st cross-tools q3 2005 GCC v3.4.4
- 2005: Early Linux adopters (montavista, nokia) - shimmed glibc
- Kernel syscalls changed during 2.6.15 (early 2006)
- Debian port started q1 2006 - all new
- Aleph One and Code sourcery gcc4.1 cross-tools q1 2006
- Angstrom OE EABI Aug 2006
- ADS/Lennert Buytenhek working port Jan 2007 (v4t build)
- DD-signed (Riku Voipio) buildd announced April 2007 (v4t build)

Current armel Status

- Port now working and useable - 90% built
- 92 Bugs filed - *eabi* tag
- 2 Buildds working (thecus n2100) - unofficial
- Installer base support
- Repository at http://ftp.gnuab.org/
- Used on mpcore, NSLU2, Thecus, versatile(QEMU), Balloon, ADS boards

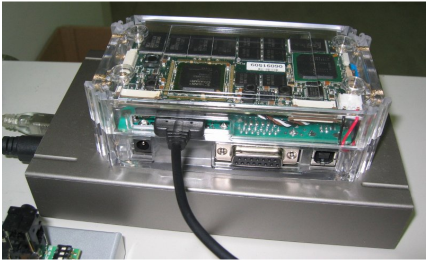
  The last 10% requires some work

Package Status

- Working:
  - Toolchain
  - C, C++, glibc
  - Java (last week)
  - perl
  - python
  - apt, dpkg, db
- Broken:
  - Fortran (g77)
  - Haskell
  - Objective-C
  - Mono
  - ocaml
  - Iceweasel/Firefox

Remaining Issues

- Need to qualify for Lenny (95% built, etc)
- v4 support in gcc
- Porter machine



- How long to keep ‘arm’ going?
- Transitions from arm->armel need support/testing

Thanks

- ARM corp
- ADS
- Nokia
- Paul Brook (Codesourcery)
- Lennert Buytenhek
- Riku Voipio
- Joey Hess
- Martin Guy