Emdebian Update

Root filesystems, G Palmtop Environment and Touchscreen support.

Neil Williams

August 16, 2008

Introduction

Embedded Debian supports reducing Debian down to be suitable for much smaller systems whilst keeping the multiarchitecture support, vendor independence, social contract and huge software base.

- Emdebian 1.0 will be based on Debian 5.0 "Lenny"

Introduction

Embedded Debian supports reducing Debian down to be suitable for much smaller systems whilst keeping the multiarchitecture support, vendor independence, social contract and huge software base.

- Emdebian 1.0 will be based on Debian 5.0 "Lenny"
- Lenny already contains the build tools for Emdebian which allow Debian source packages to be cross-built and shrunk to suit embedded ARM systems.
  The Emdebian 1.0 distribution itself will contain prebuilt ARM packages sufficient to create root filesystems that can be customised for specific machines and machine variants. Kernels and kernel modules need to be provided separately. Support for armel and i386 is pending.

Introduction

Embedded Debian supports reducing Debian down to be suitable for much smaller systems whilst keeping the multiarchitecture support, vendor independence, social contract and huge software base.

- Emdebian 1.0 will be based on Debian 5.0 "Lenny"
- Lenny already contains the build tools for Emdebian which allow Debian source packages to be cross-built and shrunk to suit embedded ARM systems.
  The Emdebian 1.0 distribution itself will contain prebuilt ARM packages sufficient to create root filesystems that can be customised for specific machines and machine variants. Kernels and kernel modules need to be provided separately. Support for armel and i386 is pending.
- Full control of dependencies on small systems (some modified dependencies). Remove perl and remove or reimplement required perl scripts. Remove documentation.

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.
- To build for arm, armel, ia64, m68k, mips, mipsel, powerpc and sparc host machines.

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.
- To build for arm, armel, ia64, m68k, mips, mipsel, powerpc and sparc host machines.
- Using gcc-3.3, 3.4, 4.1, 4.2 and 4.3

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.
- To build for arm, armel, ia64, m68k, mips, mipsel, powerpc and sparc host machines.
- Using gcc-3.3, 3.4, 4.1, 4.2 and 4.3
- Also gcc-4.2 for arm on powerpc.

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.
- To build for arm, armel, ia64, m68k, mips, mipsel, powerpc and sparc host machines.
- Using gcc-3.3, 3.4, 4.1, 4.2 and 4.3
- Also gcc-4.2 for arm on powerpc.
- Based on glibc (uClibc pending).

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.
- To build for arm, armel, ia64, m68k, mips, mipsel, powerpc and sparc host machines.
- Using gcc-3.3, 3.4, 4.1, 4.2 and 4.3
- Also gcc-4.2 for arm on powerpc.
- Based on glibc (uClibc pending).
- Toolchain autobuilder.

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.
- To build for arm, armel, ia64, m68k, mips, mipsel, powerpc and sparc host machines.
- Using gcc-3.3, 3.4, 4.1, 4.2 and 4.3
- Also gcc-4.2 for arm on powerpc.
- Based on glibc (uClibc pending).
- Toolchain autobuilder.
- dpkg-cross, apt-cross and emdebian-tools in Debian Lenny.

Resources and conventions

Build = Big or buiID == Desktop
Host == Handheld

- Toolchains for installation on i386, amd64 build machines.
- To build for arm, armel, ia64, m68k, mips, mipsel, powerpc and sparc host machines.
- Using gcc-3.3, 3.4, 4.1, 4.2 and 4.3
- Also gcc-4.2 for arm on powerpc.
- Based on glibc (uClibc pending).
- Toolchain autobuilder.
- dpkg-cross, apt-cross and emdebian-tools in Debian Lenny.
- Cross-building package autobuilder.

emdebian-tools

Set of Debian packages to make cross building Debian easier, includes apt-cross, dpkg-cross and emdebian-tools.

- Installing toolchains.

emdebian-tools

Set of Debian packages to make cross building Debian easier, includes apt-cross, dpkg-cross and emdebian-tools.

- Installing toolchains.
- Downloading and installing cross-architecture libraries and headers.

emdebian-tools

Set of Debian packages to make cross building Debian easier, includes apt-cross, dpkg-cross and emdebian-tools.

- Installing toolchains.
- Downloading and installing cross-architecture libraries and headers.
- Patching Debian source packages, implementing incremental improvements.

emdebian-tools

Set of Debian packages to make cross building Debian easier, includes apt-cross, dpkg-cross and emdebian-tools.

- Installing toolchains.
- Downloading and installing cross-architecture libraries and headers.
- Patching Debian source packages, implementing incremental improvements.
- Reporting bugs to Debian with patches for crossbuild support.

emdebian-tools

Set of Debian packages to make cross building Debian easier, includes apt-cross, dpkg-cross and emdebian-tools.

- Installing toolchains.
- Downloading and installing cross-architecture libraries and headers.
- Patching Debian source packages, implementing incremental improvements.
- Reporting bugs to Debian with patches for crossbuild support.
- Cross building binary packages with support for machine:variant customisation patches.

emdebian-tools

Set of Debian packages to make cross building Debian easier, includes apt-cross, dpkg-cross and emdebian-tools.

- Installing toolchains.
- Downloading and installing cross-architecture libraries and headers.
- Patching Debian source packages, implementing incremental improvements.
- Reporting bugs to Debian with patches for crossbuild support.
- Cross building binary packages with support for machine:variant customisation patches.
- Generating customised root filesystems from binary packages.

Prebuilt packages for ARM

- Busybox based root filesystem.

Prebuilt packages for ARM

- Busybox based root filesystem.
- Binary packages to support the G Palmtop Environment based on GTK+2 and touchscreen support via tslib.

Prebuilt packages for ARM

- Busybox based root filesystem.
- Binary packages to support the G Palmtop Environment based on GTK+2 and touchscreen support via tslib.
- Cross-building autobuilder support in emdebian-tools. http://www.emdebian.org/buildd/

Prebuilt packages for ARM

- Busybox based root filesystem.
- Binary packages to support the G Palmtop Environment based on GTK+2 and touchscreen support via tslib.
- Cross-building autobuilder support in emdebian-tools. http://www.emdebian.org/buildd/
- Support for creating filesystem images that need only minimal configuration on the embedded device.

Prebuilt packages for ARM

- Busybox based root filesystem.
- Binary packages to support the G Palmtop Environment based on GTK+2 and touchscreen support via tslib.
- Cross-building autobuilder support in emdebian-tools. http://www.emdebian.org/buildd/
- Support for creating filesystem images that need only minimal configuration on the embedded device.
- Installability checks using Debian tools. Uses edos-debcheck to simulate the effects of the upload on installability.

Prebuilt packages for ARM

- Busybox based root filesystem.
- Binary packages to support the G Palmtop Environment based on GTK+2 and touchscreen support via tslib.
- Cross-building autobuilder support in emdebian-tools. http://www.emdebian.org/buildd/
- Support for creating filesystem images that need only minimal configuration on the embedded device.
- Installability checks using Debian tools. Uses edos-debcheck to simulate the effects of the upload on installability.
- Machine:variant customisation support including package selection, custom packages, custom configurations and integration with kernels and modules.

Prebuilt packages for ARM

- Busybox based root filesystem.
- Binary packages to support the G Palmtop Environment based on GTK+2 and touchscreen support via tslib.
- Cross-building autobuilder support in emdebian-tools. http://www.emdebian.org/buildd/
- Support for creating filesystem images that need only minimal configuration on the embedded device.
- Installability checks using Debian tools. Uses edos-debcheck to simulate the effects of the upload on installability.
- Machine:variant customisation support including package selection, custom packages, custom configurations and integration with kernels and modules.
- Current focus is on GPE, with more developer time any other environment could be made available.

Customising for specific machine variants

- Current method uses debootstrap, other backends are possible.

Customising for specific machine variants

- Current method uses debootstrap, other backends are possible.
- (Second stage of normal debootstrap requires a working cross compiler to work without perl.)

Customising for specific machine variants

- Current method uses debootstrap, other backends are possible.
- (Second stage of normal debootstrap requires a working cross compiler to work without perl.)
- 'Unpack' method in development to create a filesystem image without .deb files. Only needs minor configuration on the device.

Customising for specific machine variants

- Current method uses debootstrap, other backends are possible.
- (Second stage of normal debootstrap requires a working cross compiler to work without perl.)
- 'Unpack' method in development to create a filesystem image without .deb files. Only needs minor configuration on the device.
- '–machine foo' for general changes (default variant)

Customising for specific machine variants

- Current method uses debootstrap, other backends are possible.
- (Second stage of normal debootstrap requires a working cross compiler to work without perl.)
- 'Unpack' method in development to create a filesystem image without .deb files. Only needs minor configuration on the device.
- '–machine foo' for general changes (default variant)
- '–machine foo' '–variant code' for other derivations.

Customising for specific machine variants

- Current method uses debootstrap, other backends are possible.
- (Second stage of normal debootstrap requires a working cross compiler to work without perl.)
- 'Unpack' method in development to create a filesystem image without .deb files. Only needs minor configuration on the device.
- '–machine foo' for general changes (default variant)
- '–machine foo' '–variant code' for other derivations.
- Customised package selection with no fixed package sets - select anything that works.

Customising for specific machine variants

- Current method uses debootstrap, other backends are possible.
- (Second stage of normal debootstrap requires a working cross compiler to work without perl.)
- 'Unpack' method in development to create a filesystem image without .deb files. Only needs minor configuration on the device.
- '–machine foo' for general changes (default variant)
- '–machine foo' '–variant code' for other derivations.
- Customised package selection with no fixed package sets - select anything that works.
- Example full GPE GUI filesystem 25Mb compressed, about 75Mb installed. (Work is ongoing to drop below 64Mb installed.)

Emdebian 1.0 for ARM

- Embedded devices need customised kernels and kernel modules.

Emdebian 1.0 for ARM

- Embedded devices need customised kernels and kernel modules.
- Installation methods are often specific to that device or device type and might not fit with typical D-I setup.

Emdebian 1.0 for ARM

- Embedded devices need customised kernels and kernel modules.
- Installation methods are often specific to that device or device type and might not fit with typical D-I setup.
- Emdebian is investigating D-I support with a pre-built root filesystem tarball - just needs to be unpacked and run
  # dpkg –configure -a

Emdebian 1.0 for ARM

- Embedded devices need customised kernels and kernel modules.
- Installation methods are often specific to that device or device type and might not fit with typical D-I setup.
- Emdebian is investigating D-I support with a pre-built root filesystem tarball - just needs to be unpacked and run
  # dpkg –configure -a
- Example: (balloon3)
  boot flash (bootloader support on RO partition)
  mount USB stick and the root partition.
  Copy over the kernel image.
  Untar the root filesystem directly onto the root partition
  chroot into root filesystem, dpkg –configure -a
  untar kernel modules
  reboot

Mixing Debian and Emdebian

- Dependencies will differ.

Mixing Debian and Emdebian

- Dependencies will differ.
- Bandwidth considerations - much larger packages.

Mixing Debian and Emdebian

- Dependencies will differ.
- Bandwidth considerations - much larger packages.
- translations causing conflicts (until TDebs in Debian)

Mixing Debian and Emdebian

- Dependencies will differ.
- Bandwidth considerations - much larger packages.
- translations causing conflicts (until TDebs in Debian)
- Library transitions.

Things to improve in Emdebian

- Toolchain build times.

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.
- armel support.

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.
- armel support.
- Code Audit.

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.
- armel support.
- Code Audit.
- Python support (for OpenMoko).

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.
- armel support.
- Code Audit.
- Python support (for OpenMoko).
- Debian-Installer Integration - passing the prepared root filesystem to D-I to reduce the amount of work needed by the device.

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.
- armel support.
- Code Audit.
- Python support (for OpenMoko).
- Debian-Installer Integration - passing the prepared root filesystem to D-I to reduce the amount of work needed by the device.
- NMU's of old cross-building bugs.

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.
- armel support.
- Code Audit.
- Python support (for OpenMoko).
- Debian-Installer Integration - passing the prepared root filesystem to D-I to reduce the amount of work needed by the device.
- NMU's of old cross-building bugs.
- i386 support

Things to improve in Emdebian

- Toolchain build times.
- glibc & tzdata size reductions.
- armel support.
- Code Audit.
- Python support (for OpenMoko).
- Debian-Installer Integration - passing the prepared root filesystem to D-I to reduce the amount of work needed by the device.
- NMU's of old cross-building bugs.
- i386 support
- Checkpointing support in YAFFS

Using emdebian-tools

```
$ sudo apt-get install emdebian-tools subversion
$ emsetup -a arm -v -s
$ emsetup -a arm -v
$ emsource libgpewidget1
$ cd /$work/trunk/l/libgpewidget/trunk/libgpewidget-0.115/
$ emdebuild -a arm -v
$ emrecent
```

or:

```
$ embug - - prepare
$ cd ../../branches/embugG2342/
$ meld libgpewidget.debian/debian/rules
libgpewidget.emdebian/debian/rules
$ reportbug -M -b -i ../crossbuild.diff libgpewidget
```

G Palmtop Environment

- matchbox provides window manager, as in Familiar or OpenEmbedded.

G Palmtop Environment

- matchbox provides window manager, as in Familiar or OpenEmbedded.
- GPE provides a calendar, addressbook, todo, expenses, configuration tools, games, console, calculator, image viewer and text editor.

G Palmtop Environment

- matchbox provides window manager, as in Familiar or OpenEmbedded.
- GPE provides a calendar, addressbook, todo, expenses, configuration tools, games, console, calculator, image viewer and text editor.
- Support for stylus input and touchscreen control on suitable devices.

G Palmtop Environment

- matchbox provides window manager, as in Familiar or OpenEmbedded.
- GPE provides a calendar, addressbook, todo, expenses, configuration tools, games, console, calculator, image viewer and text editor.
- Support for stylus input and touchscreen control on suitable devices.
- Bluetooth and audio support included if suitable hardware is present.

Thanks

http://www.toby-churchill.com/
http://www.emdebian.org/
Many individuals : Wookey, Hector Oron, Neil Williams, Jonathan McDowell, Peter De Schrijver (p2), Nikita Y.Youshchenko, Phil Hands, Simon Richter, Phillipe de Swert, Raphael Bossek, Allen Curtis
and anyone I might have forgotten.