Poky Linux & OpenEmbedded based environment

Marcin Juszkiewicz
marcin@juszkiewicz.com.pl
http://marcin.juszkiewicz.com.pl

ST-Ericsson Community Workshop 2009

Some information about me

- Bought Sharp Zaurus SL-5500 in February 2003
- Joined OpenEmbedded team in 2003
- OpenZaurus distribution maintainer in 2005-2006
- Created own company for OpenEmbedded & Poky consulting in 2006
- Became Poky Linux developer in 2007
- Worked on porting Poky for NDK-15 in 2007
- Bug Linux developer since 2008

What is Poky Linux?

Poky is build system which can be used to build single binaries, toolchains, root filesystems etc.

When it comes to software Poky's metadata contains thousands of recipes for applications, libraries, fonts etc. Target rootfs can contain minimal system but also much bigger ones with SDK integrated for building software on device itself are available.

Features

- Building applications, toolchains, SDKs, root filesystems
- Handling of cross-compilation
- Building all required tools and libraries before compilation of target package
- Independency from host distribution or architecture
- Generation of target packages with inter-packages dependencies
- Big granulation of packages (documentations, i18n goes to separate packages)

Features continues

- Simple format of package recipes and machine configs
- Ability to define own tasks inside of recipe.
- Building with device/architecture/distro defined overrides
- Optimizations for sub-architectures
- Soft-float support for ARM cpus without FPU

Getting Poky

- Fetch:
  `git clone git://git.pokylinux.org/poky`
- Add ST-Ericsson overlay

Requirements

- Fast computer with Linux (Debian or Ubuntu preferred)
- Few gigabytes of storage to make build
- Serial port (real or USB adapter)
- Network connection to NHK-15

Building Poky

- First setup everything:
  ```
  bash$ source poky-init-build-env
  ```
- Run build:
  ```
  bitbake oh-image-nomadik
  ```
- Wait few hours
- Install kernel and image on NHK-15 (both are present in `tmp/deploy/images/` directory)

Example targets

- `oh-image-nomadik` - default NHK-15 demo image
- `poky-image-sato` - default Poky X11 image
- `poky-image-sdk` - X11 image with on-board SDK
- `virtual/kernel` - kernel for device
- `meta-toolchain` - toolchain for your host

What is OpenEmbedded

- Set of metadata which describe how to build software
- Fully open (MIT license) solution of all problems with cross-compilation
- System created and maintained by passionates
- Project independent from any commercial company
- Project which allows to create single apps, whole root filesystems by one command.
- Tool which generates root filesystems in many formats
- Answer for distributions which use 200MB for "base" system

History

- OE was created in 2002 as a tool to build next version of OpenZaurus. It had to replace hacked buildroot.
- OpenZaurus 3.5.1 was released in September 2003. Big success.
- Other distributions joined: Familiar, NSLU2 team, JLime.
- 2009: Palm Pre was released with OE generated system, Garmin released devices with Poky Linux based system
- MontaVista uses OE as base for new version of MV Linux

Features

- Building applications, toolchains, SDKs, root filesystems, whole distributions
- Handling of cross-compilation
- Allows to use glibc, eglibc, uclibc, dietlibc, different gcc & binutils combos
- Building all required tools and libraries before compilation of target package
- Independency from host distribution or architecture
- Simple format of package recipes and machine configs
- Ability to define own tasks inside of recipe.

Features continues

- Generation of target packages with inter-packages dependencies
- Big granulation of packages (documentations, i18n goes to separate packages)
- Building with device/architecture/distro defined overrides
- Possibility of cooperation of many projects in one metadata
- Handling of many architectures: ARM, PowerPC, SH, x86, x86-64, MIPS
- Optimizations for sub-architectures (omap3 gets NEON, VFP, EP93xx gets Maverick support)
- Soft-float support for ARM and PowerPC cpus without FPU

Features continues again

- Many environments available: text console, OPIE, GPE, GNOME, XFCE. . .
- Many GUI toolkits available: Qt, Qt/Embedded, Gtk+, Tk, Fltk. . .
- Many server daemons available
- Lot of development tools available

Getting OpenEmbedded

Fetch:

```
git clone git://git.openembedded.org
```

There are few branches of OE:

- `org.openembedded.dev` - development, fast moving version
- `stable/2009` - stable version with small amount of changes

Building OpenEmbedded

- First setup everything
- Run build:

  ```
  bitbake x11-image
  ```
- Wait few hours
- Install kernel and image on NHK-15 (both are present in `tmp/deploy/` directory)

Example targets

- `micro-base-image` - minimal console image (BusyBox + SSH server)
- `console-image` - text console image
- `x11-image` - default X11 image with GPE
- `beagleboard-demo-image` - big X11 + E17 image with many applications preinstalled
- `virtual/kernel` - kernel for device
- `meta-toolchain` - toolchain for your host

NHK-15 support

- NHK-15 board support exists on my HDD
- Images works with ST kernel
- Waiting for official mainline kernel to be usable
- Waiting for permission to publish some of Nomadik patches

Why OE instead of Poky Linux

- Bigger set of software available
- Other X11 environments available
- Non-X11 systems are available
- No need to build software - there are repositories with packages