

Cross-compiling OpenJDK with OpenEmbedded

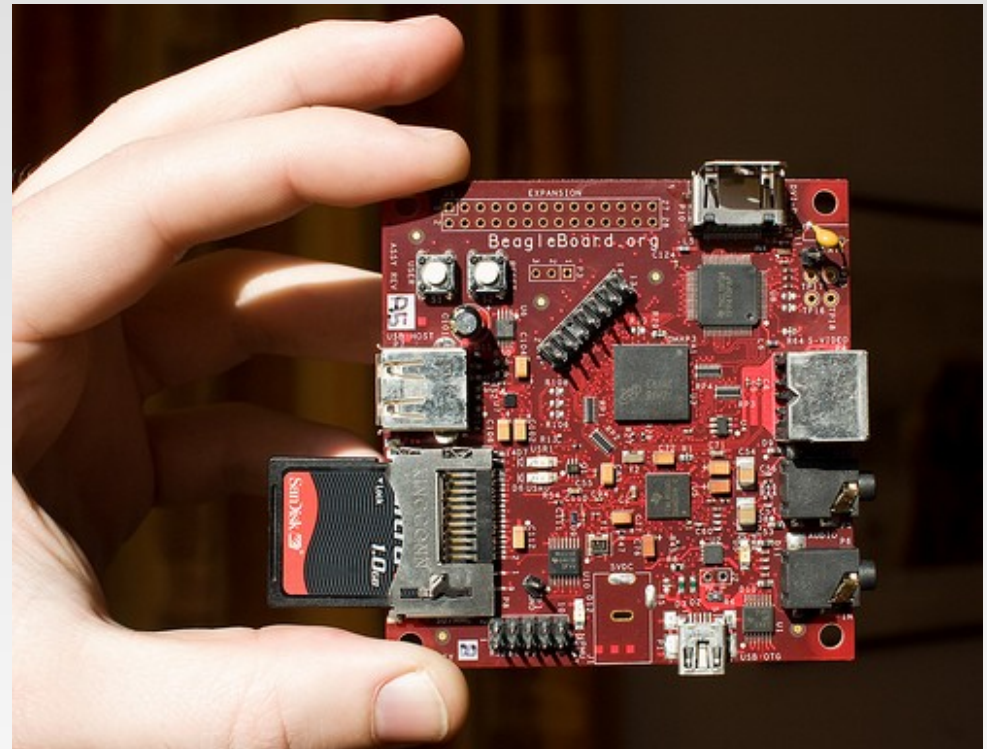
FOSDEM 2009

Robert Schuster
Jalimo + OpenEmbedded

Why?

Beagleboard specs:

- 128 MiB RAM
- SDHC slot
- ~500 MHz
- Vector FPU



Why?

- Beagle's OMAP3 is a CPU for next generation Smartphones, PDAs, ...
- other armv7a implementations (e.g. i.MX51) going to be in the same league

Why?

"We're trying to converge everything to the Java SE specification. Cell phones and TV set-top boxes are growing up,"

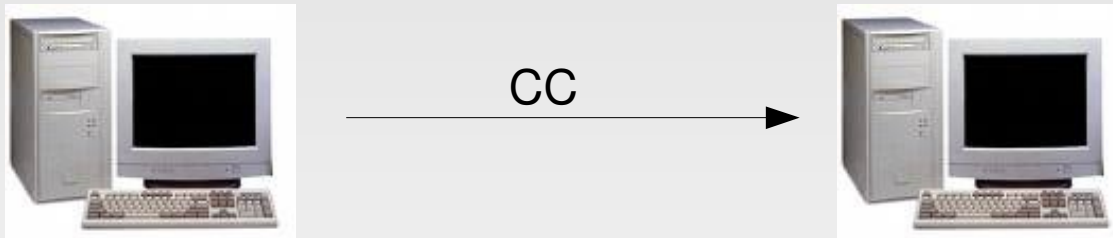
James Gosling in an CNet interview, 2007

http://news.cnet.com/8301-13580_3-9800679-39.html?part=rss&subj=news&tag=2547-1_3-0-5

Cross-compiling?

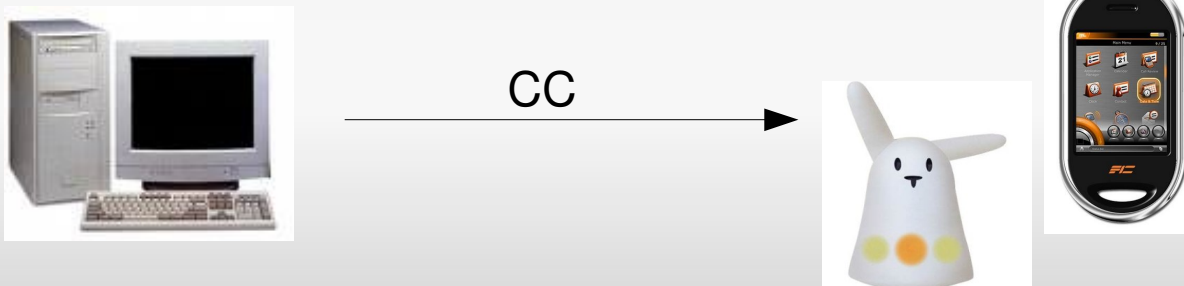
- Non-cross:

- Build for the machine the compiler is running on (build machine)



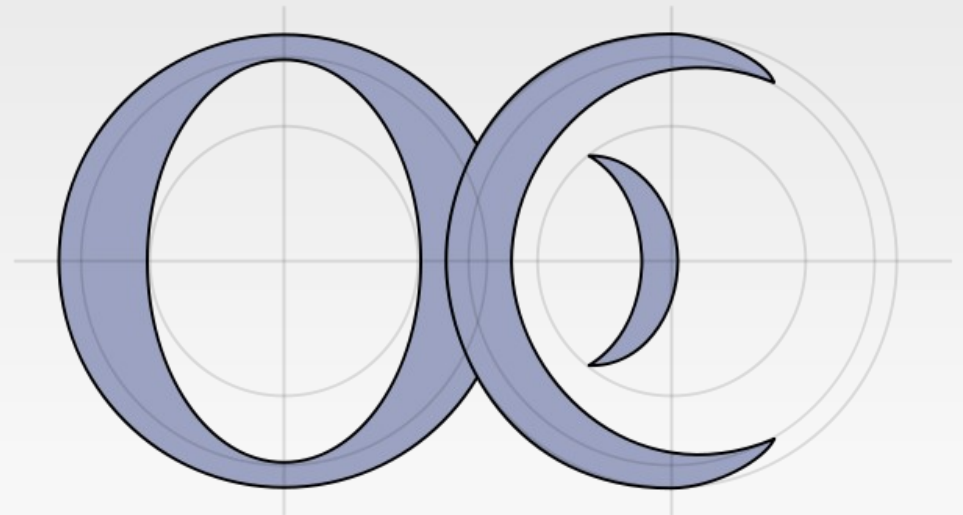
- Cross:

- Build for a different system than the compiler is running on (target machine)



OpenEmbedded

OE is a build environment that provides the necessary metadata to automate the tasks in the generation of a cross-compiled GNU/Linux distribution.



open embedded

Normal OpenJDK build

JDK 1.5 (proprietary)
builds
OpenJDK

IcedTea's main contribution

GNU Classpath/GCJ

builds

Bootstrap JDK

builds

OpenJDK

Cross-Compilation approach

GNU Classpath/GCJ (build machine)

builds

Bootstrap JDK (build machine)

builds

Bootstrap JDK (target machine)

is a linker target for

OpenJDK (target machine)

(Each box is one separate build process.)

Main build hack (target)

```
# Builds bootstrap JDK  
make icedtea-ecj
```

```
# Pull in java, javac, javah, jar and  
# sizer.32 from bootstrap JDK (build)  
ln -sf ....
```

```
# Keep jre/lib/arm directory intact  
# (contains various shared libraries)
```

```
# Builds OpenJDK  
make icedtea
```

Smaller hacks

- enforce cross-compiler usage (sed is your friend)
- disable sanity checks
- replace ``uname -m`` in makefiles with a variable
- disable building `sizer.32` binary (and use a pre-built one instead)

Smaller hacks

- enforce cross-compiler usage (sed is your friend)
- disable sanity checks
- replace ``uname -m`` in makefiles with a variable
- disable building `sizer.32` binary (and use a pre-built one instead)

And some other things I don't remember now ...

You really want all the details?

Full guide:

jalimo.org/index.php/CrossCompilingOpenJDK

Also linked from within IcedTea and

OpenEmbedded Wiki

Guide contents

- prerequisites
- different builds (Zero, Shark, Cacao)
- all patches
- build steps

Things done

- building and packaging
 - OpenJDK with Zero
 - OpenJDK with Shark*
 - OpenJDK with Cacao
- tested only for ARM

* not fully working, due to issues in LLVM

Outstanding issues

- patches need to be adjusted and pushed upstream
 - disable sanity checks, replacement of ``uname -m``
- Shark is pre-alpha
 - LLVM shows issues
- Swing is not fully usable on touchscreen-only devices yet
 - no right-click (popup menu)
 - no virtual keyboard activation (also affects AWT)
 - no mouse move possible, only drag

Demo time!

Summary

- cross-compiling OpenDK possible
 - tested on ARM
- full guide in the making
 - jalimo.org/index.php/CrossCompilingOpenJDK
- things will get easier in the future because of patch integration



Thank you
for listening
and
many many
thanks to

BugLabs

who sponsored
the work!

Copyleft

This presentation is licensed under

GNU FDL 1.2 or later (no invariant sections, cover text and all that)

or

Creative Commons Attribution Share-Alike 3.0 or later