

# OpenEmbedded musings about buildsystems

Holger Hans Peter Freyther

February 15, 2007

Outline

What is  
OpenEmbedded

Past  
Present

The problems

No world domination  
for you  
Problems in depth

The future

Wishlist  
Sane systems

## What is OpenEmbedded

Past

Present

## The problems

No world domination for you

Problems in depth

## The future

Wishlist

Sane systems

OpenEmbedded  
musings about  
buildsystems

Holger Hans  
Peter Freyther

### Outline

What is  
OpenEmbedded

Past

Present

The problems

No world domination  
for you

Problems in depth

The future

Wishlist

Sane systems

# Let there be determinism

- ▶ Founded by Chris Larson and others to leave the age of buildroot behind
- ▶ Created to be flexible. Support various architectures and configurations
- ▶ Make it possible to just create one package, one toolchain or complete distributions
- ▶ Self contained and deterministic, helps to obey the GPL

## Success is near

- ▶ OpenEmbedded is used to bootstrap various Debian ports
- ▶ OpenEmbedded is used to create community distributions
- ▶ OpenEmbedded is used at research facilities
- ▶ OpenEmbedded is used for Consumer Electronic
- ▶ OpenEmbedded is a collaboration point for system and application developers
- ▶ OpenEmbedded is used in your company?!

# The future

- ▶ Add new architectures and machines
- ▶ Add and update packages
- ▶ world domination

OpenEmbedded  
musings about  
buildsystems

Holger Hans  
Peter Freyther

Outline

What is  
OpenEmbedded

Past  
Present

The problems

No world domination  
for you  
Problems in depth

The future

Wishlist  
Sane systems

# Your buildsystem was just too ugly

- ▶ Convenience
- ▶ Cleverness
- ▶ Laziness
- ▶ Wrong assumptions (stupidity)

OpenEmbedded  
musings about  
buildsystems

Holger Hans  
Peter Freyther

Outline

What is  
OpenEmbedded

Past

Present

The problems

No world domination  
for you

Problems in depth

The future

Wishlist

Sane systems

# Convenience the lovely enemy

- ▶ You hardcode the compiler, please use `$(CC)` ...
- ▶ I installed this helper library to `/usr/local` lets change `$(CFLAGS)`
- ▶ you paste `libtool` in your `autoconf` verbatim
- ▶ Bad examples are: `asterisk`, `mad-fcl`, ...

# Cleverness - keep it stupid simple

- ▶ I could compile and execute this test and decide which algorithm performs best
- ▶ If you use autoconf please honor the cross variable and provide a way to select the algorithm from the outside



# Laziness

- ▶ Compile the host/native application with the target flags
- ▶ Reinvent the wheel to detect libraries
- ▶ Bad examples are: fontconfig, poppler

# Wrong assumptions

- ▶ Hardcode paths and filenames, simply assume they are present
- ▶ Everybody wants to install to /usr
- ▶ Let us hardcode pkgconfig files
- ▶ Bad examples include glibc

# To take part in the world domination your buildsystem should

- ▶ be something you don't create in the last minute
- ▶ not hardcode paths, or simply add them
- ▶ make it possible to enable/disable features from the outside

OpenEmbedded  
musings about  
buildsystems

Holger Hans  
Peter Freyther

Outline

What is  
OpenEmbedded

Past  
Present

The problems

No world domination  
for you  
Problems in depth

The future

Wishlist  
Sane systems

# Sane systems include

- ▶ (GNU)Makefiles
- ▶ autoconf, automake
- ▶ cmake
- ▶ qmake

OpenEmbedded  
musings about  
buildsystems

Holger Hans  
Peter Freyther

Outline

What is  
OpenEmbedded

Past  
Present

The problems

No world domination  
for you  
Problems in depth

The future

Wishlist  
Sane systems