Reproducible builds in ports

Baptiste Daroussin
bapt@FreeBSD.org

gandi.net

FreeBSD

FOSDEM 2016
Brussels
January 31, 2016

A shared effort

http://reproducible-builds.org/

archlinux

Guix

coreboot

Tor

OpenWrt
Wireless Freedom

NetBSD

FreeBSD

debian

FreeBSD

What is that?

basically: Rebuild twice, same result

What is it useful for

Obvious:

- Creating binary patches: delta packages, freebsd-update
- Verifiable path from source to binary
- Reduce package repository storage size (dedup)
- Reduce the bandwith when mirroring

What is it useful for

Obvious:

- Creating binary patches: delta packages, freebsd-update
- Verifiable path from source to binary
- Reduce package repository storage size (dedup)
- Reduce the bandwith when mirroring

But not only: more accurate exp-run!

- better track changing toolchain components
- better track what packages are inpacted by a change in headers/macros: bump portrevision
- better track what packages uses static libraries: improved packaging Q/A

Hey that's easy... hum... maybe not

Issues:

- timestamp

Hey that's easy... hum... maybe not

Issues:

- timestamp
- hostname

Hey that's easy... hum... maybe not

Issues:

- timestamp
- hostname
- username

Hey that's easy... hum... maybe not

Issues:

- timestamp
- hostname
- username
- threaded producers

Hey that's easy... hum... maybe not

Issues:

- timestamp
- hostname
- username
- threaded producers
- optimisations

Hey that's easy... hum... maybe not

Issues:

- timestamp
- hostname
- username
- threaded producers
- optimisations
- toolchain

Hey that's easy... hum... maybe not

Issues:

- timestamp
- hostname
- username
- threaded producers
- optimisations
- toolchain
- build path

Upstream solutions

For timestamps:

- Stop storing build timestamps when possible

Upstream solutions

For timestamps:

- Stop storing build timestamps when possible
- If impossible: support SOURCE_DATE_EPOCH

Upstream solutions

For timestamps:

- Stop storing build timestamps when possible
- If impossible: support SOURCE_DATE_EPOCH

For hostname:

- Stop storing build hostname

Upstream solutions

For timestamps:

- Stop storing build timestamps when possible
- If impossible: support SOURCE_DATE_EPOCH

For hostname:

- Stop storing build hostname
- Provide a mechnaism to override (ex: env var)

Upstream solutions

For timestamps:

- Stop storing build timestamps when possible
- If impossible: support SOURCE_DATE_EPOCH

For hostname:

- Stop storing build hostname
- Provide a mechnaism to override (ex: env var)

For username:

- Stop storing build username

Upstream solutions

For timestamps:

- Stop storing build timestamps when possible
- If impossible: support SOURCE_DATE_EPOCH

For hostname:

- Stop storing build hostname
- Provide a mechnaism to override (ex: env var)

For username:

- Stop storing build username
- Provide a mechnaism to override (ex: env var)

Upstream solutions

For timestamps:

- Stop storing build timestamps when possible
- If impossible: support SOURCE_DATE_EPOCH

For hostname:

- Stop storing build hostname
- Provide a mechnaism to override (ex: env var)

For username:

- Stop storing build username
- Provide a mechnaism to override (ex: env var)

For threaded consumers

- Allow the build system to control the number of threads

Upstream solutions

For toolchain/optimisations

- Ensure the build system respect CFLAGS, CC, CXX, LD, etc.

Upstream solutions

For toolchain/optimisations

- Ensure the build system respect CFLAGS, CC, CXX, LD, etc.

For build path

- Stop storing the build path

Upstream solutions

For toolchain/optimisations

- Ensure the build system respect CFLAGS, CC, CXX, LD, etc.

For build path

- Stop storing the build path
- Really don't store it!

Upstream solutions

For toolchain/optimisations

- Ensure the build system respect CFLAGS, CC, CXX, LD, etc.

For build path

- Stop storing the build path
- Really don't store it!

Otherthing to really not store:

- Host CPU identifiers/type

Upstream solutions

For toolchain/optimisations

- Ensure the build system respect CFLAGS, CC, CXX, LD, etc.

For build path

- Stop storing the build path
- Really don't store it!

Otherthing to really not store:

- Host CPU identifiers/type
- Locales of the builder

Upstream solutions

For toolchain/optimisations

- Ensure the build system respect CFLAGS, CC, CXX, LD, etc.

For build path

- Stop storing the build path
- Really don't store it!

Otherthing to really not store:

- Host CPU identifiers/type
- Locales of the builder
- Timezone of the builder

FreeBSD base

- Under our control

FreeBSD base

- Under our control
- Almost done (needed by freebsd-update(8))

FreeBSD base

- Under our control
- Almost done (needed by freebsd-update(8))
- Still issues:

FreeBSD base

- Under our control
- Almost done (needed by freebsd-update(8))
- Still issues:
  - loaders: contains build time/date

FreeBSD base

- Under our control
- Almost done (needed by freebsd-update(8))
- Still issues:
  - loaders: contains build time/date
  - kernel by default (fixed via exporting some env variables)

FreeBSD base

- Under our control
- Almost done (needed by freebsd-update(8))
- Still issues:
  - loaders: contains build time/date
  - kernel by default (fixed via exporting some env variables)
  - aacraid/acpi_asus modules

FreeBSD base

- Under our control
- Almost done (needed by freebsd-update(8))
- Still issues:
  - loaders: contains build time/date
  - kernel by default (fixed via exporting some env variables)
  - aacraid/acpi_asus modules
  - sqlite db file (mandoc) - should not be part of the build system

Right but what about reality: ports

Right but what about reality: ports

- It's a wild world

Right but what about reality: ports

- It's a wild world
- currently not reproducible

Right but what about reality: ports

- It's a wild world
- currently not reproducible

use poudrière

Poudrière

Controled environment:

- Always build with the same user

Poudrière

Controled environment:

- Always build with the same user
- Use jails which allows:
  - Controlling the hostname
  - Controlling the kernel name/reldate (not used yet because too recent)

Poudrière

Controled environment:

- Always build with the same user
- Use jails which allows:
  - Controlling the hostname
  - Controlling the kernel name/reldate (not used yet because too recent)
- Controls the kernel name/reldate

Poudrière

Controled environment:

- Always build with the same user
- Use jails which allows:
  - Controlling the hostname
  - Controlling the kernel name/reldate (not used yet because too recent)
- Controls the kernel name/reldate
- Controls the build path

Poudrière

Controled environment:

- Always build with the same user
- Use jails which allows:
  - Controlling the hostname
  - Controlling the kernel name/reldate (not used yet because too recent)
- Controls the kernel name/reldate
- Controls the build path
- Define a reproducible build env

Then why not there yet?

Then why not there yet?

- Timestamps:
  - pkg 1.6.3+ (add support for SOURCE_DATE_EPOCH)
  - ports not exposing SOURCE_DATE_EPOCH: D4385

Then why not there yet?

- Timestamps:
  - pkg 1.6.3+ (add support for SOURCE_DATE_EPOCH)
  - ports not exposing SOURCE_DATE_EPOCH: D4385
- More ports framework improvements needed:
  - some bytecodes: pyc, elc, zwc, ...
  - automatic detection of non reproducibility
  - diffoscope (just landed in the ports tree r407589)

Then why not there yet?

- Timestamps:
  - pkg 1.6.3+ (add support for SOURCE_DATE_EPOCH)
  - ports not exposing SOURCE_DATE_EPOCH: D4385
- More ports framework improvements needed:
  - some bytecodes: pyc, elc, zwc, ...
  - automatic detection of non reproducibility
  - diffoscope (just landed in the ports tree r407589)
- Lots of ports needs patching:
  - Use gzip -n
  - Fix unpredictable tool output
  - Add support for SOURCE_DATE_EPOCH
  - Fix other corner case unpredicability issues

SOURCE_DATE_EPOCH

Allows a third party to provide a predictable timestamp to the build system/packager when needed

Ports tree support:

- timestamp should not be updated too often
- should be more recent that the distfiles
- added by make makesum to distinfo
- fallback to the date when the support was added

Questions?

gandi.net

Thanks

FreeBSD