Adding GNU/Hurd support to GNU Guix
Porting GNU Guix/GuixSD to a new platform

Manolis Ragkousis
manolis837@gmail.com

FOSDEM
5 February 2017
What is distro bootstrapping?

Simply speaking

- Bootstrapping refers to the process of getting the distribution built “from nothing”.

But how does the first package gets built?
How is it usually done?

➢ Distros, like Debian or Arch Linux, use whatever is already present in the system.
➢ No clear notion of “bootstrap binaries”.
➢ Hard to track down the origins of a port (and reproduce it).
Guix relies on a clearly defined set of pre-built binaries

- Guile
- GCC
- Binutils
- Libc
- Bootstrap binaries

Bootstrapping is complete when we have a full tool chain that does not depend on the pre-built bootstrap tools
1) Cross-build the bootstrap binaries:

   guix build --target=i585-pc-gnu bootstrap-tarballs

2) “Inject” the bootstrap binaries in the package DAG for that platform.

   (gnu packages bootstrap)

3) Build Guix on a running OS of that platform!!

It wasn't so easy..
What's special about the Hurd

• The Hurd is a collection of servers that run on top of the Mach microkernel.

• These servers implement features that are normally implemented by the kernel.

• Through glibc the same standard interfaces known from other UNIX-like systems are provided.
Building the bootstrap binaries.

- PATH_MAX missing (acl, patch, sed, tar, etc..).
- Binaries for i686-gnu failed to run on a Hurd system. i586-gnu work.
- Linux assumptions in Guix (cross-toolchain).
- Problems with building Hurdish Glibc.
WOOHOOOO!!!
Using the binaries on the new platform

1. Update (gnu packages bootstrap) with information on the new binaries and where to download them.

2. Add rules to the gnu-system.am file on how to handle the binaries.

3. Clone Guix on the new platform.

4. Run ./configure --with-courage && make && make install

5. Begin building!

It works!!
Using Guix on GNU/Hurd

guix-daemon → pivot_root(), mount(), MS_BIND

1. Hurd utils, settrans → libhurdutil library, settrans().
2. mount() → nixMount() wrapper.
3. MS_BIND → firmlink
4. pivot_root() → fakerooot
Building GuixSD from GNU/Hurd

On Debian/Hurd:

```
  guix system disk-image config.scm
```

- Linux assumptions (ex. gnu/build, gnu/system).
- On Linux `guix system` cannot build for Hurd.
- Package build failures. Patches needed.
Current status

- Cross-build to GNU/Hurd with just:
  
  ```
  guix build --target=i586-pc-gnu foo
  ```

- 50+ patches on Guix proper.

- Branch ‘master’ can be used on a running GNU/Hurd system.
  
  ```
  guix build foo
  ```

- Currently testing guix-daemon patches.

- GNU/Hurd cross-compilation on Hydra.
Work in progress

- Finish libhurdutil library.
- Finish nix/utils/calls.(hh.cc).
- Make Guix's build system kernel agnostic.
- Package Hurd libraries.
Roadmap

- Port GuixSD
  - Isolate Linuxisms
  - Package missing dependencies.
- Port container-style features in guix-daemon
  - Sub-hurds instead of Linux name spaces
Thanks for mentoring, suggestions, code, debugging and patience..

- Ludovic Courtès, Justus Winter, Samuel Thibault my GSoC mentors for their valuable help, guidance, patience and trust!
- Everyone from Guix and Hurd for helping me in so many ways! :-)
- Extra thanks to Rene Saavedra for testing, reporting and patching Guix on Hurd non-stop!
- This work was part of my GSoC 2015 and 2016 projects so I would like to thank everyone from FSF for taking care of everything for the GNU Project participation.
Credits
