

A STROLL THROUGH THE GARDEN OF SOFTWARE CURIOSITIES

FOR PEOPLE WITH SHORT ATTENTION SPANS

Robert Schmidt|roschmidt@toh.ca



**The Ottawa
Hospital**

**L'Hôpital
d'Ottawa**

Affiliated with • Affilié à



uOttawa



DEALING WITH THE REASONABLE USER

- I want to run this thing from the web that I saw in some journal
- It has these bunch of dependencies
- None are satisfied by the OS (or we don't want them to be)

IMPORTANT VARIABLES FOR A "USER"

- Non-builder user:
 - PATH, MANPATH
 - LD_LIBRARY_PATH
 - language-specific locations for libraries (PYTHONPATH, PERL5LIB,)
- Developer user:
 - CPATH, LIBRARY_PATH



WHY THIS ISN'T EASY

- Building is slow
- users are
- RPATH in Nix, Conda and perhaps Spack

THE CONDA WAY

- conda supports relocation, this allows binaries to be moved to different user directories
- I often use conda provides a reasonable core python for use with easybuild
- conda could be used to provide os dependencies

OTHER THINGS ABOUT CONDA

- Conda supports environments and sets a pretty path
- sets just a few variables `PATH`, `CONDA_PATH` and backups

```
(root) [rob@ottbioinfo bin]$ ldd `which python`  
linux-vdso.so.1 => (0x00007ffef1fe1000)  
libpython2.7.so.1.0 => /home/rob/miniconda2/bin/../../lib/libpython2.7.so.1  
libpthread.so.0 => /lib64/libpthread.so.0 (0x00007f355f5b4000)  
libdl.so.2 => /lib64/libdl.so.2 (0x00007f355f3b0000)  
libutil.so.1 => /lib64/libutil.so.1 (0x00007f355f1ad000)  
libm.so.6 => /lib64/libm.so.6 (0x00007f355eeaa000)  
libc.so.6 => /lib64/libc.so.6 (0x00007f355eae9000)  
/lib64/ld-linux-x86-64.so.2 (0x00007f355fbd4000)
```


CONDA DYNAMIC ELF HEADERS

Dynamic section at offset 0x8e0 contains 26 entries:

| Tag | Type | Name/Value |
|--------------------|----------|-----------------------------------|
| 0x0000000000000001 | (NEEDED) | Shared library: [libpython2.7.so] |
| 0x0000000000000001 | (NEEDED) | Shared library: [libpthread.so] |
| 0x0000000000000001 | (NEEDED) | Shared library: [libdl.so.2] |
| 0x0000000000000001 | (NEEDED) | Shared library: [libutil.so.1] |
| 0x0000000000000001 | (NEEDED) | Shared library: [libm.so.6] |
| 0x0000000000000001 | (NEEDED) | Shared library: [libc.so.6] |
| 0x000000000000000f | (RPATH) | Library rpath: [\$ORIGIN/../lib] |



THE NIX WAY

- Nix prefers to install things in a static prefix using `/nix/store` allows you to install binaries since the derivation is hashed and added to the prefix, binaries names are able to be predicted and rebuilt and shared for a given set of other machines nix has some very powerful features including creating a container

SOME EXAMPLES

Trends that we start to see. Relocation support in some way is a good thing binary installs might play a role in How do we support installing in different machines where they might not be the same filesystem or paths