SHIRAKUMO

Porting SBCL to the Nintendo Switch

Yukari Hafner, Charles Zhang

https://shirakumo.org

The Device

- CPU: ARM 4 Cortex-A57 64-bit
- OS: “Horizon OS”, proprietary micro-kernel
- SDK: C++, proprietary version of Clang

Immediate Challenges

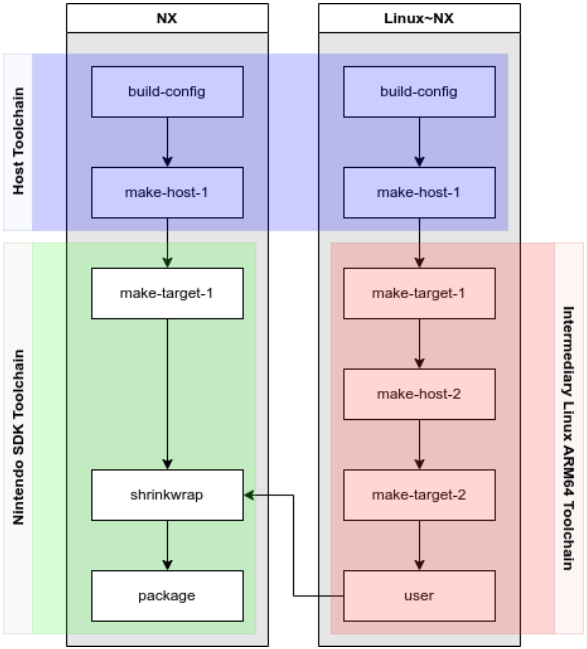
- Everything is proprietary and under NDA
  ⇒ Scarce public information
- The OS is not BSD or even fully POSIX
  ⇒ Need new OS abstractions
- There are no inter-thread signals
  ⇒ Can't use usual GC tricks
- We are not allowed to create executable pages
  ⇒ No compilation at runtime

Basic Ideas

- Everything is proprietary and under NDA
  ⇒ Only publicise our own interfaces
- The OS is not BSD or even fully POSIX
  ⇒ Write C(++) shim libraries for access
- There are no inter-thread signals
  ⇒ Use safepoints
- We are not allowed to create executable pages
  ⇒ Compile everything on linux and shrinkwrap

A Standard SBCL Build

- build-config
  ⇒ Gather system info
- make-host-1
  ⇒ Emit C headers and support files
- make-target-1
  ⇒ Compile the C runtime on the target
- make-host-2
  ⇒ Cross-compile the compiler on the host
- make-target-2
  ⇒ Use the compiler from the host to incrementally compile the rest on the target


NX
Linux~NX
Host Toolchain
build-config
make-host-1
build-config
make-host-1
Nintendo SDK Toolchain
make-target-1
shrinkwrap
package
make-target-1
make-host-2
make-target-2
user
Intermediary Linux ARM64 Toolchain

Relocation

Garbage Collection

Live Demo

Further Work

- Optimising CLOS dispatch ahead of time
  ⇒ Christophe? 🥺
- Optimising Trial and Kandria
  ⇒ Lots of profiling work that can be done on PC
- Porting to the Nintendo Switch 2
  ⇒ As soon as plebians like us get access from almighty Nintendo

SHIRAKUMO

Thank you!

Consider supporting our work:

https://patreon.com/shinmera