Adopting LLVM Binary Utilities in Toolchains

Jordan Rupprecht
rupprecht@google.com

Google

Why?

1. Toolchains are expensive!
   a. Already build one for Clang + LLD
   b. Code, build, test, deploy, and debug miscompiles just once!
2. LLVM everywhere
   a. Native support for LLVM bitcode embedded anywhere

Testing

1. Plug it in and see what breaks!
   a. Great for catching obvious failures
   b. Not great for subtle issues

```
# This fails, as expected, and
# complains loudly.
$ llvm-readelf --xyz foo

# This "succeeds" but used to have
# bug and create a bad object file.
# It manifests as strange runtime
# failures.
$ llvm-objcopy --localize-hidden foo
```

Testing

1. Plug it in and see what breaks!
2. Use another binutils test suite
   a. Free test coverage!
   b. But noisy results.

```
$ objcopy -B foo ...
objcopy: architecture foo
unknown

$ llvm-objcopy -B foo ...
llvm-objcopy: error: Invalid
architecture: 'foo'
```

Testing

1. Plug it in and see what breaks!
2. Use another binutils test suite
3. Manual testing
    a. :(

Everything is broken!

1. Flags and formatting

```
Different alias meanings
$ readelf -s => --symbols
$ llvm-readelf -s =>  --sections

$ size foo | cut -f 2
   text\t    data\t   bss\t ...
  23616\t    1472\t  1336\t ...
$ llvm-size foo | cut -f 2
   text     data      bss ...
  23616     1472     1336 ...
```

Everything is broken!

1. Flags and formatting
2. Broken features

- llvm-ar: doesn't create thin archives correctly (in kernel builds)
- llvm-size: miscalculated sizes (!)
- llvm-symbolizer (addr2line): requires stdin input
- llvm-objcopy/llvm-strip: lots of catching up to do
- ...

Everything is broken!

1. Flags and formatting
2. Broken features
3. Incompatibilities we *want*

```
# GNU ar uses -M and silently ignores rcs
$ ar rcs -M < mri_script.txt
$ llvm-ar rcs -M < mri_script.txt
llvm-ar: error: Cannot mix -M and other
options

# GNU accepts any non-ambiguous prefix
$ readelf --seg foo
$ llvm-readelf --segments foo

# -s removes more in llvm-strip; we need
# an additional llvm flag, --keep-section
$ strip -s foo
$ llvm-strip -s --keep-section=.bar foo
```

Conclusion

Everything is now ... less broken?

Try it out! Contribute patches or file bugs!

Switched!

ar, c++filt (llvm-cxxfilt), nm, size, strings

Very soon!

addr2line (llvm-symbolizer), objdump, readelf

Great progress, but not yet

objcopy, strip

Thanks!

Jordan Rupprecht
rupprecht@google.com

Questions? Suggestions?
Come to the BoF at 4:15pm today, or the round table at 2pm tomorrow.