

LLVM Binutils BoF

2019 EuroLLVM Developers' Meeting

James Henderson (SN Systems)

Jordan Rupprecht (Google)



Introduction



- LLVM binary utilities (aka binutils) include:
 - llvm-readobj/llvm-readelf
 - llvm-objdump
 - llvm-objcopy
 - llvm-cxxfilt
 - etc.
- Originally used in testing LLVM components:
 - Approximately 1715 tests in core LLVM (excluding those in the test/tools directory).
 - More in clang and lld.
- Now being used more widely.
 - See e.g. Jordan's lightning talk earlier today.

Recent Work



- Bug count:
 - 70 open (170 total) in llvm-ar, llvm-c++filt, llvm-dwarfdump, llvm-nm, llvm-objcopy/strip, llvm-objdump, llvm-ranlib, llvm-readobj, llvm-size, llvm-symbolizer.
 - 81 resolved between January 2017 and end of March 2019.
- GSOC 2018 project by Paul Semel.
 - <https://summerofcode.withgoogle.com/archive/2018/projects/5171359936151552/>
 - Added features to improve compatibility of LLVM tools with GNU.
- llvm-objcopy newly added in summer 2017, gained Mach-O and COFF support in the last 6 months.

Recent Work (2)



Tool	Commits in tools/<tool> Jan 2017 – March 2019
llvm-ar	44
llvm-cxxfilt	16
llvm-dwarfdump	25
llvm-nm	49
llvm-objcopy	288
llvm-objdump	173
llvm-readobj	277
llvm-size	12
llvm-strings	8
llvm-symbolizer	28
Total	920

Does not include commits in libraries (e.g. DebugInfo, Object, Symbolizer etc).

Command-line Compatibility



- Tools should be “drop-in replacements” with GNU tools.
 - Does anybody disagree?

- What do we mean by this?
 - Only switches that are widely used are identical?
 - All switches accepted with the same syntax?
 - All switches are semantically identical?
 - Something else?

- Is it okay to break compatibility with previous LLVM releases?
 - We did this for llvm-readelf, but now llvm-readobj and llvm-readelf aren't directly compatible...

Output Compatibility



- How different can the LLVM tool output be to GNU?
 - E.g. should we support parsers that parse GNU output?
- Should we aim to have multi-mode outputs for everything?
 - See `llvm-readobj --elf-output-style={LLVM|GNU}`
- Is it okay for our output to change between releases?

Future Goals



- Any requests for new features beyond existing GNU features?
 - For example turning llvm-objcopy into a library.
 - One binary tool to rule them all?
 - Any new switches?
 - New output formats?
 - Anything else?

- GSOC 2019 proposal for further work.

Any Other Business



- Does anybody have anything else they want to bring up?

- What do people want done sooner?

- Round table 2pm tomorrow to continue the discussion.