Porting LLVM to a new OS

Kai Nacke

31 January 2016

LLVM devroom @ FOSDEM‘16
Porting LLVM

• There are two possible goals
  – Run LLVM tools on OS
  – Generate code for OS / CPU architecture

• Mission is to run LLVM on previously unsupported OS and adding code generation for the OS

• Adding a new CPU architecture is a major task
  – Not considered here
A very brief look on AIX

- OS for mission critical tasks

- A UNIX OS, supports System V and BSD APIs

- Runs on POWER architecture
  - Already a target for LLVM

- Good software support
  - Native toolchain
  - Major Open Source Software available
Toolchain

- Prerequisites for LLVM are available
  - gcc, cmake, gmake, ...
  - Use this toolchain

- GNU tools use as and ld from OS
  - Expect different command line options

- Not every package works out of the box
  - Python 2.7.x is missing => compile yourself
  - cmake 3.x had problems => use cmake 2.8.x
Compile LLVM

• First compiler run identifies code problems
  – Missing endian definitions
  – Wrong path handling code choosen

• Linking fails because of unsupported options
  – Updates to the cmake modules necessary

• Still problems with ThreadPool code
  – Needs more investigation
Running LLVM on AIX

- All LLVM tools are compiled and run

- Results of test suite are similar to Linux/PPC

- Still no code generation for AIX
Code Generation

• LLVM misses code generation for AIX

• You can’t use the Linux/PPC ELF-based tools
  – Binary format is XCOFF
  – Textual assembler is different

• Idea is to tweak assembler generation and use external assembler to create object file
XCOFF

• XCOFF is an extended COFF format
• Basically it adds the TOC concept to COFF
• Major differences to COFF
  – No PE header
  – Smallest addressable unit is *csect*
  – A csect always has a storage class associated
• Assembler text uses `.csect`
Partial Class Hierarchy

- AsmPrinter
  - PPCAsmPrinter
    - PPCAIxAsmPrinter
  - MCSection
    - MCSectionCOFF
    - MCOBJECTFileINFO
      - Triple
  - MCAsmInfo
    - MCAsmInfoCOFF
    - MCAsmInfoXCOFF
    - new
    - changed
    - UsesAIXSectionDirective
    - usesAIXSectionDirective()
Implementation

• Outputs `.section` as `.csect`
  – Required lot of changes

• Makes storage class part of section name
  – Only a hack

• Use raw text output for missing op‘s

• Relocation syntax requires more work
Summary / Outlook

• First patches submitted
• Work on relocation syntax required
  – Needed for “Hello World” application
• Working approach

• Next step is dumping of XCOFF files