

lld

Friday, April 13,
2012

The LLVM Linker

What is lld?

- A system linker
 - ▣ Produce final libraries and executables, no other tools or runtime required
 - ▣ Understands platform ABI

What is lld?

- A system linker
- Modular and embeddable
 - ▣ Designed as a set of libraries
 - ▣ As with all LLVM/Clang tools, the main executable code is quite short.
 - ▣ Can be embedded into other tools such as custom language compilers.

What is lld?

- A system linker
- Modular and embeddable
- Portable (both host and target)
 - ▣ Currently tested on Mac, Linux, and Windows
 - ▣ Target specific code is isolated for both OS and CPU architecture

What is lld?

- A system linker
- Modular and embeddable
- Portable (both host and target)
- Compatible
 - ▣ Driver to support multiple styles (gnu-lld, lld64, link.exe)
 - ▣ Supports ELF, COFF (including MinGW), Mach-O, and LLVM IR
 - ▣ Support for a restricted subset of linker scripts is planned

What is lld?

- A system linker
- Modular and embeddable
- Portable (both host and target)
- Compatible
- Extensible
 - ▣ Both LLVM IR and atom graph passes can be added

Why a new linker?

□ Performance

- ▣ Linking is a speed bump in the way of fast iteration
- ▣ gold may be fast, but gnu-ld and link.exe are not, and gold is ELF only

Why a new linker?

- Performance
- Portability
 - ▣ gold and gnu-ld are UNIX-oriented, link.exe is Windows only, and ld64 is Darwin only.

Why a new linker?

- Performance
- Portability
- Reliable cross linking
 - ▣ No single linker can currently be used to reliably cross link

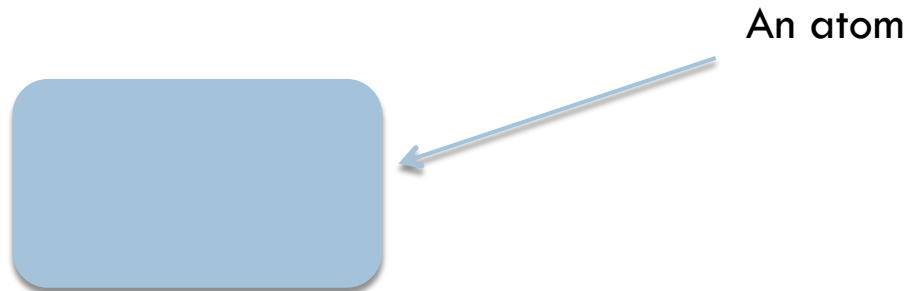
What makes lld different?

□ Atom Model

- ▣ An atom is an indivisible chunk of code or data
- ▣ It has a set of attributes such as type, name, and scope
- ▣ It has a list of references to other atoms
- ▣ References represent relocations and other relationships such as grouping.

Atom Model

- An atom is an indivisible chunk of code or data



Atom Model

- It has a set of attributes such as type, name, and scope

name: main
type: code
scope: global

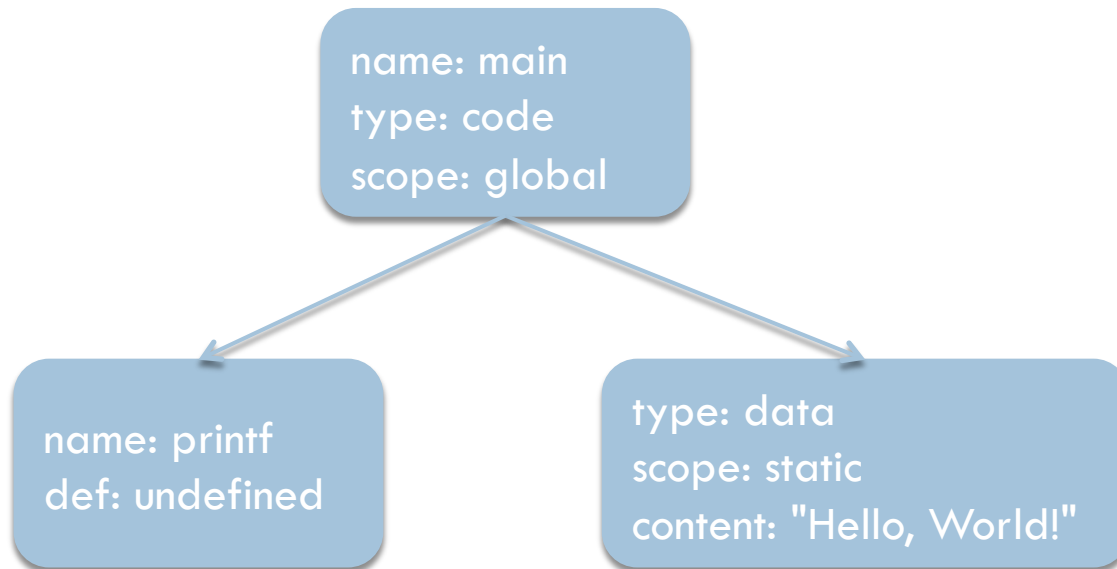
type: data
scope: static
content: "Hello, World!"

But not all atoms have names



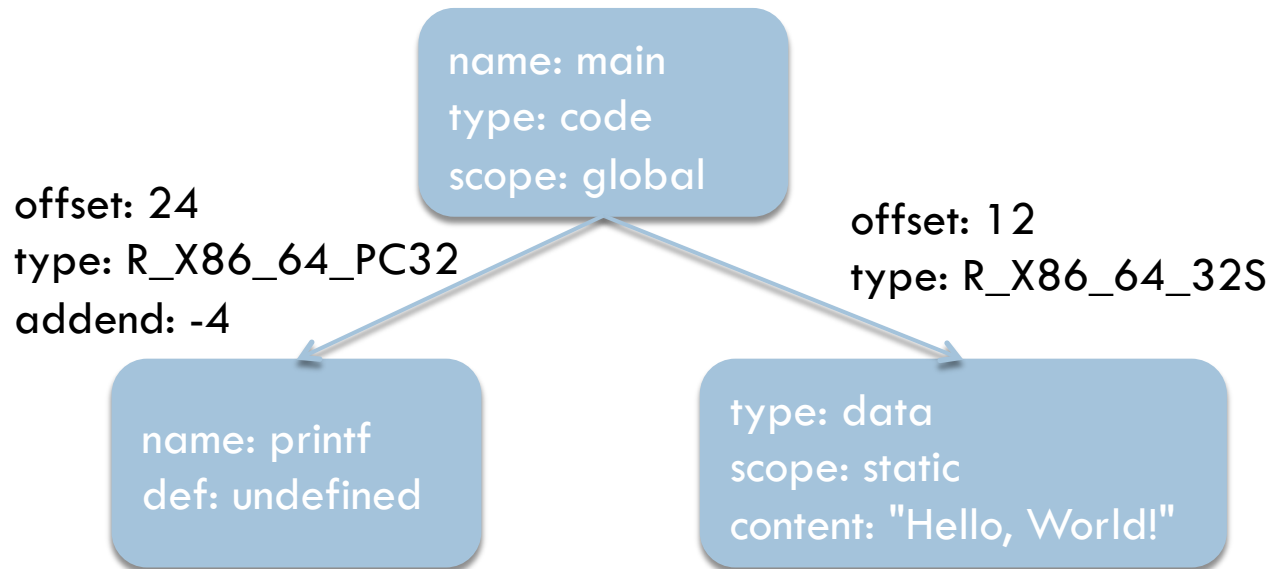
Atom Model

- It has a list of references to other atoms



Atom Model

- References represent relocations and other relationships such as grouping.



What makes lld different?

- Atom Model
- LLVM IR and atom graph passes
 - ▣ In addition to the standard LLVM passes, other passes can run at pre and post LLVM IR link phases
 - ▣ lld passes can be run over the atom graph at different stages of the link
 - Branch island generation
 - Order file
 - C++ Open Multi-Methods...

What makes lld different?

- Atom Model
- LLVM IR and atom graph passes
- Uniform text, binary, and in memory representation
 - ▣ lld supports a human-readable YAML input and output format to aid in testing and debugging

Uniform Representation

```
# RUN: lld-core %s 2>&1 | FileCheck %s
```

```
# CHECK: duplicate symbol
```

```
---
```

```
atoms:
```

```
- name:      _foo  
  scope:    global  
  type:     data
```

```
---
```

```
atoms:
```

```
- name:      _foo  
  scope:    global  
  type:     code
```

```
...
```

What makes lld different?

- Atom Model
- LLVM IR and atom graph passes
- Uniform text, binary, and in memory representation
 - ▣ lld supports a human readable YAML input and output format to aid in testing and debugging
 - ▣ lld also adds a binary representation of the in memory atom model
 - The purpose of this is purely for speed
 - The goal is for LLVM to eventually produce these natively

What makes lld different?

- Atom Model
- LLVM IR and atom graph passes
- Uniform text, binary, and in memory representation
- C++11
 - ▣ lld has been developed using C++11 from the beginning
 - ▣ Dogfooding Clang and libc++
 - ▣ Makes it easier to write faster code
 - ▣ C++11 atomics and memory model for multithreading

Project Status

- Atom graph and resolving work with YAML as input and output
- Ild native binary reading and writing
- COFF and Mach-O are already in progress
 - ▣ The Mach-O writer can produce a Hello World executable from multiple YAML files

Patches Welcome

- Open Projects

- ▣ http://lld.llvm.org/open_projects.html

Questions?

Friday, April 13,
2012

bigcheese@bigcheese.com