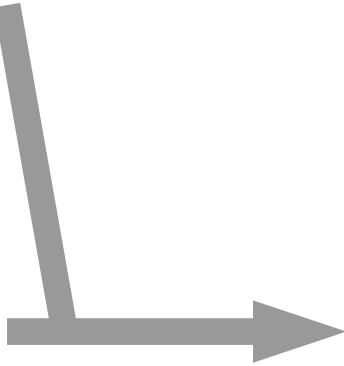


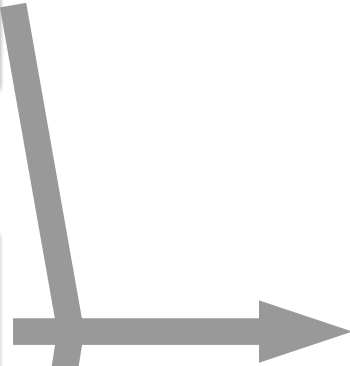
# FreeBSD Tool Chain

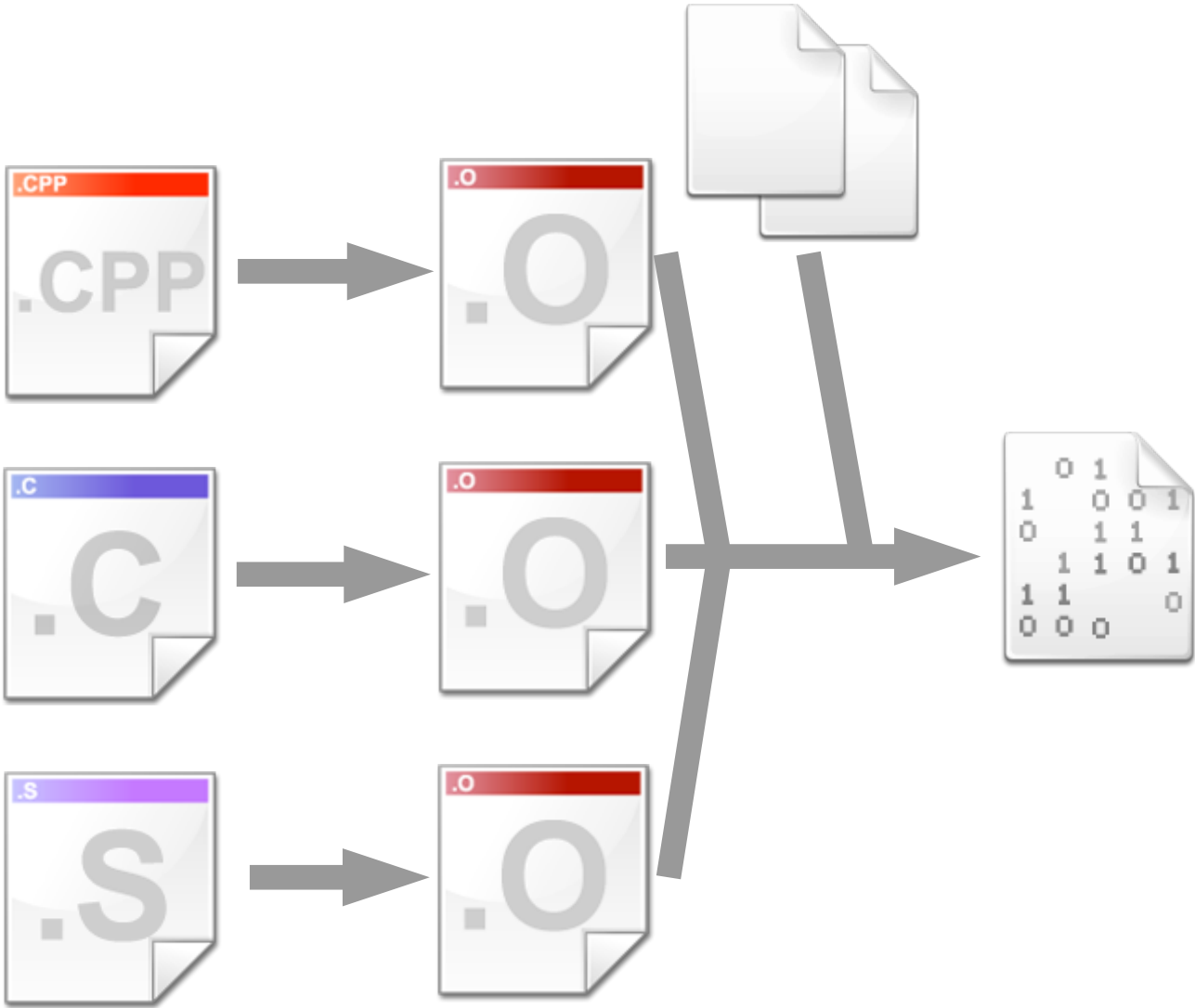
Ed Maste  
BSDCan 2017

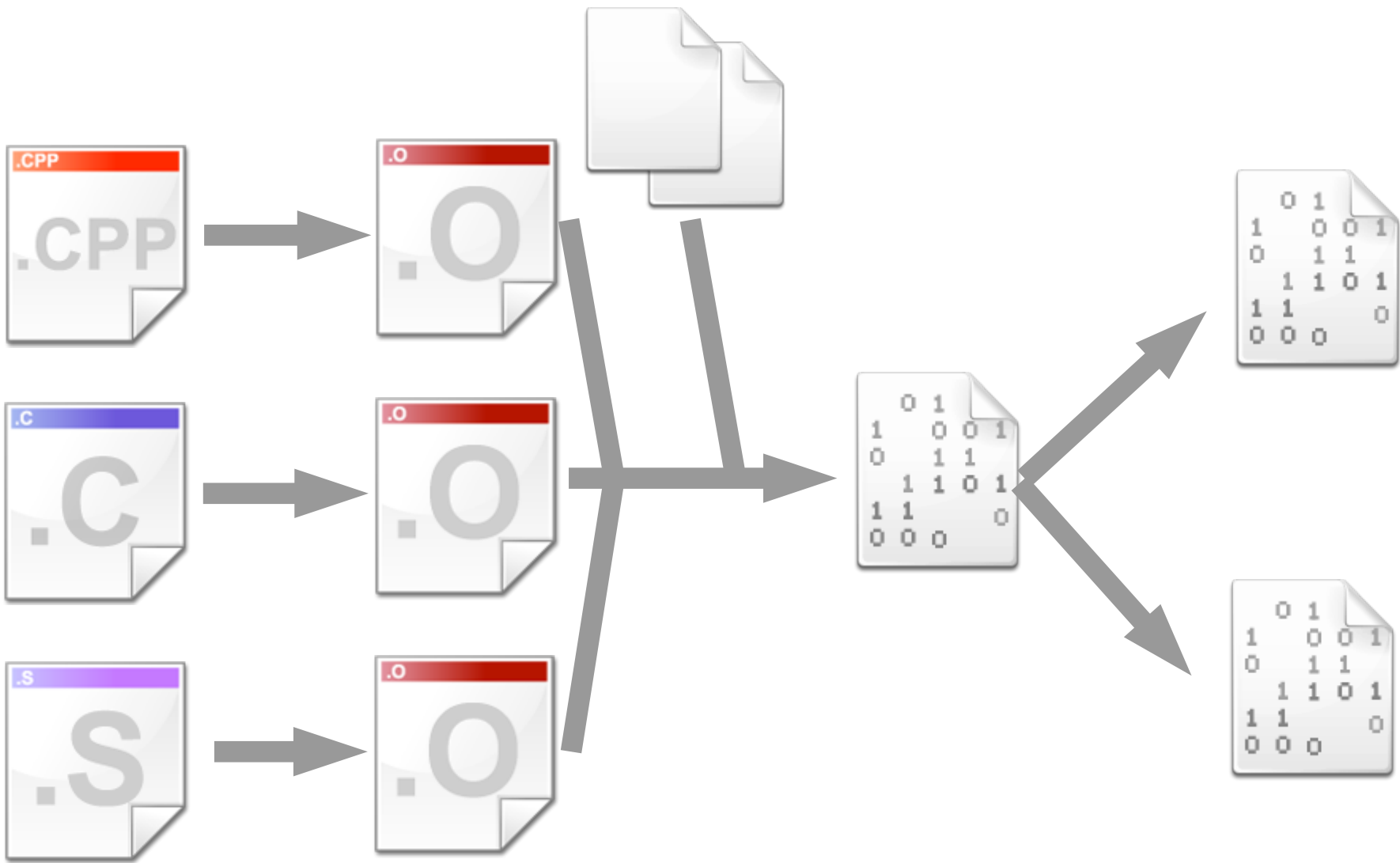


```
  0 1
1  0 0 1
0  1 1
  1 1 0 1
1 1      0
0 0 0
```

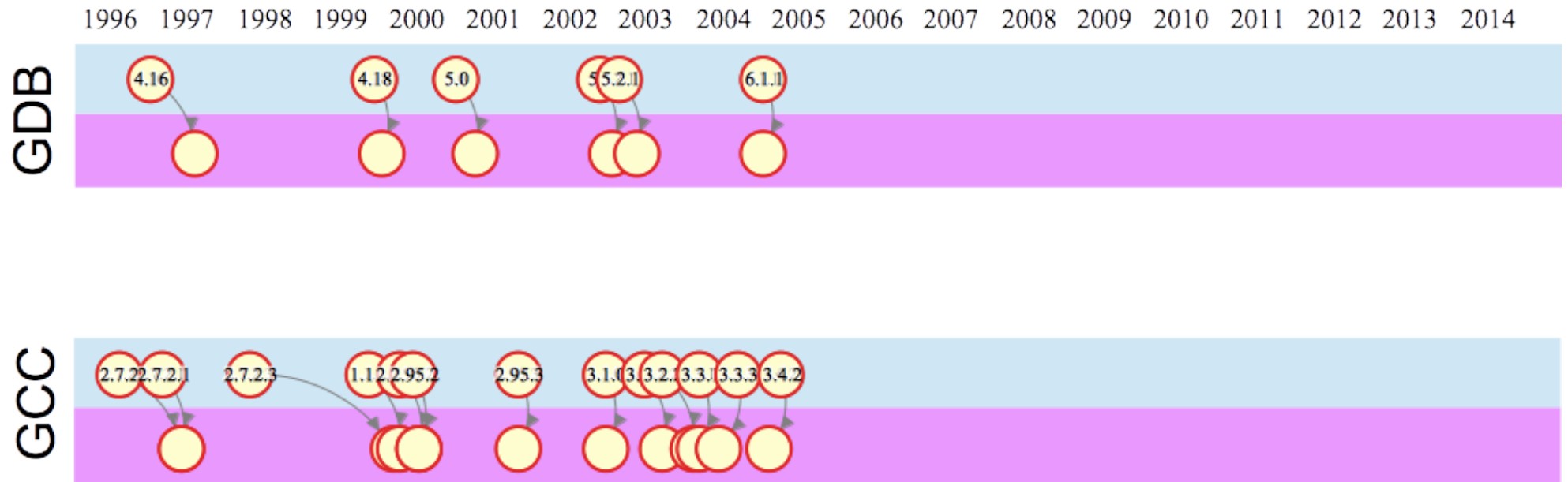




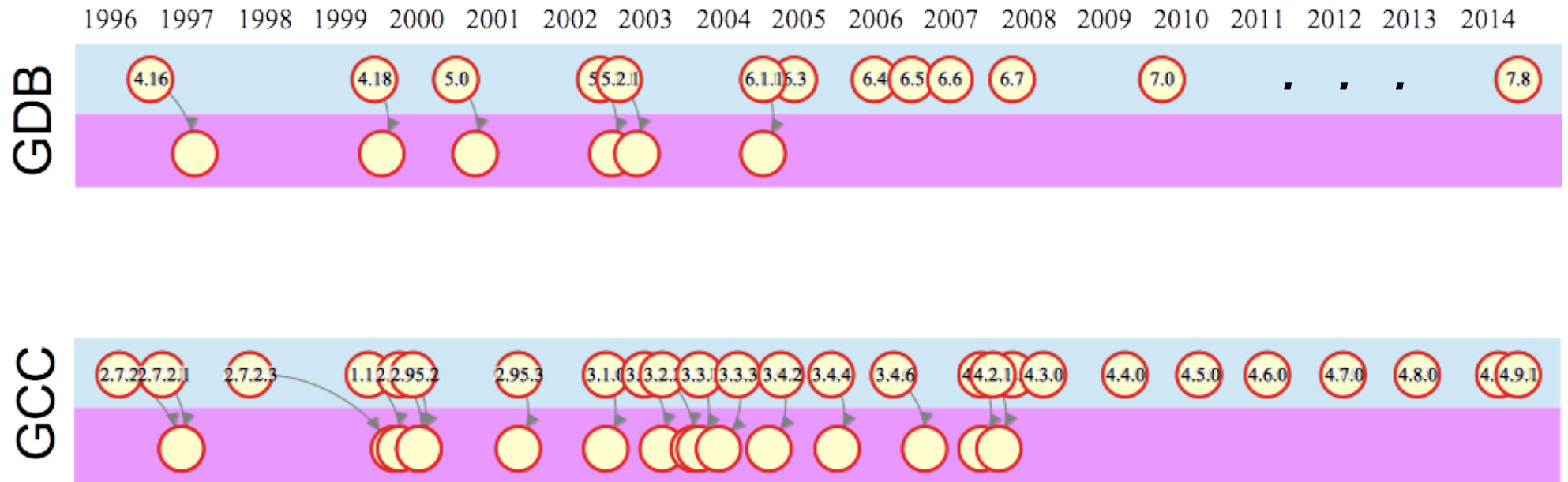




# GNU Tool Chain

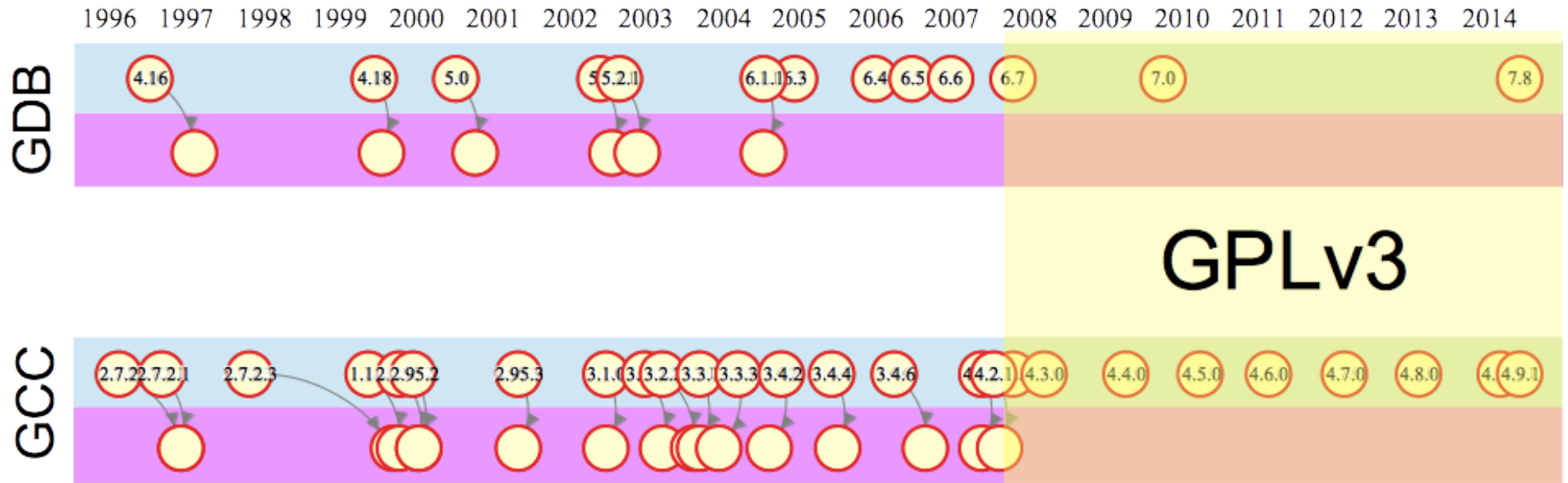


# GNU Tool Chain





# GNU Tool Chain



# Architecture Dependency

## C/C++ Compiler

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	GCC 4.2.1	Clang 3.3	Clang 3.4.1	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
amd64	GCC 4.2.1	Clang 3.3	Clang 3.4.1	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
armv6	GCC 4.2.1	Clang 3.3	Clang 3.4.1	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
arm64	-	-	-	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
mips	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 6.x	Clang 6+
powerpc	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 6.x	Clang 6+
sparc64	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 6.x	GCC
riscv64	-	-	-	-	-	GCC 6.x	Clang 6+

In-tree GCC  
In-tree Clang  
Ports GCC

## Assembler

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	Clang 5	Clang
amd64	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	Clang 5	Clang
armv6	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	Clang 5	Clang
arm64	-	-	-	gas 2.25	Clang 4.0.0	Clang 5	Clang
mips	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.27	Clang 5+
powerpc	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.27	Clang 5+
sparc64	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.27	gas
riscv64	-	-	-	-	-	gas 2.27	Clang 6+

In-tree GNU as  
In-tree Clang Integrated as  
Ports Binutils GNU as

## Linker

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	LLD 5+	LLD 5+
amd64	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	LLD 5	LLD
armv6	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	LLD 5+
arm64	-	-	-	BFD 2.25	LLD 4.0.0	LLD 5	LLD
mips	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.27	LLD 5+
powerpc	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.27	LLD 5+
sparc64	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.27	BFD 2.27
riscv64	-	-	-	-	-	BFD 2.27	LLD 6+

In-tree GNU BFD ld  
In-tree LLD  
Ports GNU BFD ld

## Debugger

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 7.12	LLDB 5+ GDB 8
amd64	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	LLDB 3.8.0	LLDB 4.0.0	LLDB 4.0.0	LLDB 5 GDB 8
armv6	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	LLDB 5 GDB 8
arm64	-	-	-	LLDB 3.8.0	LLDB 4.0.0	LLDB 4.0.0	LLDB 5 GDB 8
mips	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 7.12	LLDB 5 GDB 8
powerpc	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 7.12	LLDB 5 GDB 8
sparc64	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 8
riscv64	-	-	-	-	-	?	LLDB 6+ GDB 8+

In-tree GNU GDB  
In-tree LLDB  
Ports GNU GDB

## Binary Utilities

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	BU 2.17.50	BU 2.17.50	BU 2.17.50	ElfTC 3477	ElfTC 3520	ElfTC 3520	ElfTC
amd64	BU 2.17.50	BU 2.17.50	BU 2.17.50	ElfTC 3477	ElfTC 3520	ElfTC 3520	ElfTC
armv6	BU 2.17.50	BU 2.17.50	BU 2.17.50	ElfTC 3477	ElfTC 3520	ElfTC 3520	ElfTC
arm64	-	-	-	ElfTC 3477	ElfTC 3520	ElfTC 3520	ElfTC
mips	BU 2.17.50	BU 2.17.50	BU 2.17.50	ElfTC 3477	ElfTC 3520	ElfTC 3520	ElfTC
powerpc	BU 2.17.50	BU 2.17.50	BU 2.17.50	ElfTC 3477	ElfTC 3520	ElfTC 3520	ElfTC
sparc64	BU 2.17.50	BU 2.17.50	BU 2.17.50	ElfTC 3477	ElfTC 3520	ElfTC 3520	ElfTC
riscv64	-	-	-	-	-	ElfTC 3520	ElfTC

In-tree GNU Binutils  
In-tree ELF Tool Chain



In-tree Outdated Tools

In-tree Maintained Tools

Ports Tools

# C/C++ Compiler

## C/C++ Compiler

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	GCC 4.2.1	Clang 3.3	Clang 3.4.1	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
amd64	GCC 4.2.1	Clang 3.3	Clang 3.4.1	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
armv6	GCC 4.2.1	Clang 3.3	Clang 3.4.1	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
arm64	-	-	-	Clang 3.8.0	Clang 4.0.0	Clang 5	Clang
mips	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 6.x	Clang 6+
powerpc	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 6.x	Clang 6+
sparc64	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 4.2.1	GCC 6.x	GCC
riscv64	-	-	-	-	-	GCC 6.x	Clang 6+

In-tree GCC

In-tree Clang

Ports GCC

# Clang Upgrades

- Upgrade through each new version from 3.3 to 4.0, with 5.0 now in progress.
- New compilers bring new warnings and lower tolerance for broken source
- clang####-import project branches used for staging imports

# Clang Upgrades - Ports

- “exp-run” test build of the ports tree against src import branch
- [PR219139](#) exp-run for projects/clang500-import

New failures on amd64:

```
+ {"origin"=>"archivers/xmill", "phase"=>"build", "errortype"=>"clang"}
+ {"origin"=>"audio/mac", "phase"=>"build", "errortype"=>"clang"}
+ {"origin"=>"audio/mous", "phase"=>"build", "errortype"=>"clang"}
+ {"origin"=>"cad/openvsp", "phase"=>"build", "errortype"=>"bad_C++_code"}
+ {"origin"=>"devel/codeblocks", "phase"=>"build", "errortype"=>"bad_C++_code"}
+ {"origin"=>"devel/cpprestsdk", "phase"=>"build", "errortype"=>"clang_werror"}
```

...

# Clang Upgrades - Ports

In file included from src/mongo/db/db.cpp:47:

src/mongo/db/client.h:263:59: error: ordered comparison between pointer and zero ('mongo::Client \*' and 'int')

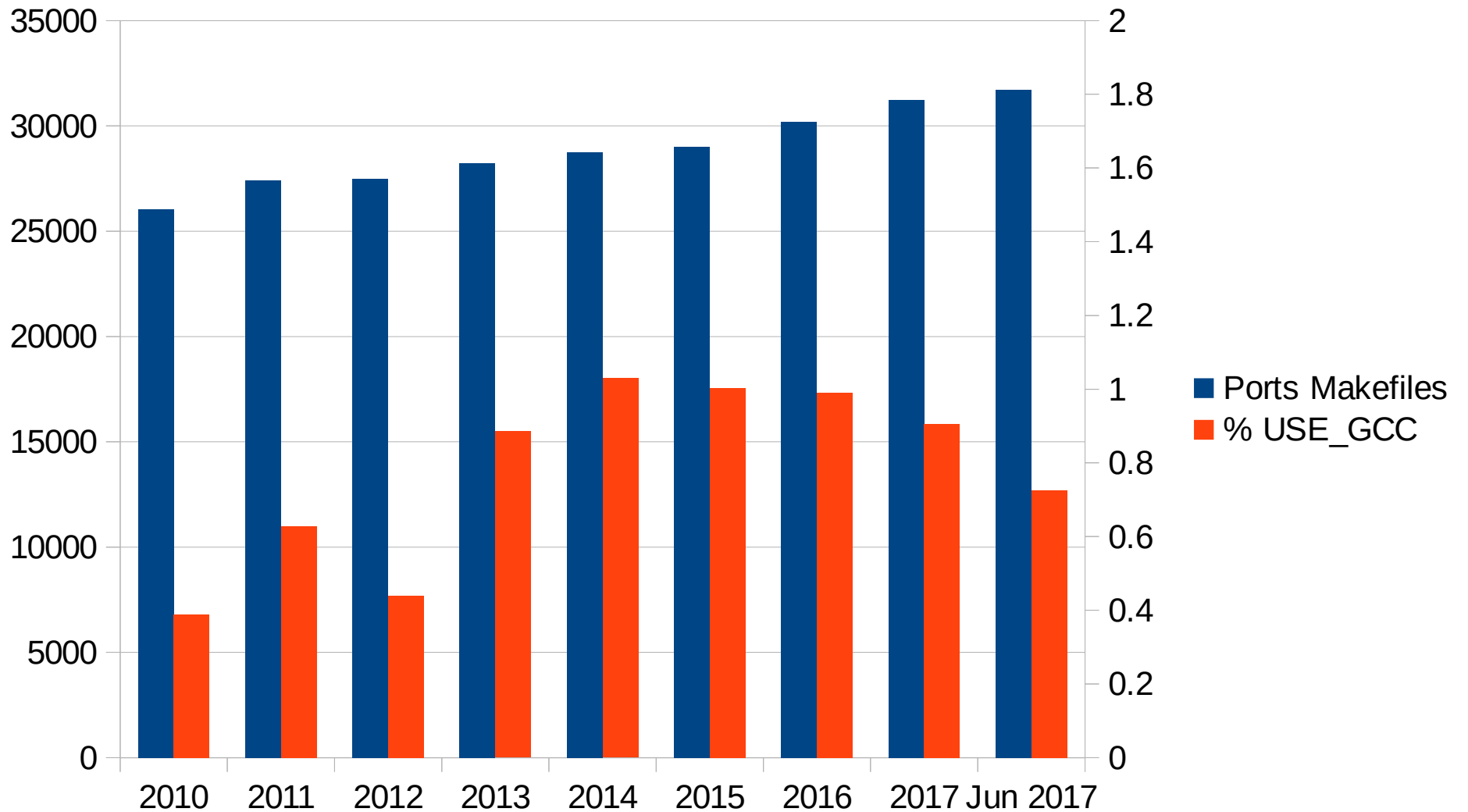
```
inline bool haveClient() { return currentClient.get() > 0; }  
                    ~~~~~ ^ ~
```

# Clang Upgrades - Ports

```
Assertion failed: (isa<X>(Val) && "cast<Ty>() argument of
incompatible type!"), function cast, file
/poudriere/jails/headamd64PR219139/usr/src/contrib/llvm/include/ll
vm/Support/Casting.h, line 254.
cc: error: unable to execute command: Abort trap (core dumped)
cc: error: clang frontend command failed due to signal (use -v to
see invocation)
FreeBSD clang version 5.0.0 (trunk 303291) (based on LLVM
5.0.0svn)
Target: x86_64-unknown-freebsd12.0
Thread model: posix
InstalledDir: /usr/bin
cc: note: diagnostic msg: PLEASE submit a bug report to
https://bugs.freebsd.org/submit/ and include the crash backtrace,
preprocessed source, and associated run script.
```



# Clang in Ports



# Assembler

## Assembler

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	Clang 5	Clang
amd64	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	Clang 5	Clang
armv6	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	Clang 5	Clang
arm64	-	-	-	gas 2.25	Clang 4.0.0	Clang 5	Clang
mips	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.27	Clang 5+
powerpc	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.27	Clang 5+
sparc64	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.17.50	gas 2.27	gas
riscv64	-	-	-	-	-	gas 2.27	Clang 6+

In-tree GNU as

In-tree Clang Integrated as

Ports Binutils GNU as

# GNU Assembler

- Very little actually requires `/usr/bin/as`
- [PR205250](#) exp-run with no `as`
- amd64 world + kernel built fine
- 11 new port failures (and 200 skipped)

# GNU Assembler

- .macro complicates GNU as replacement
- e.g. `sys/crypto/skein/amd64/skein_block_asm.s`

```
#  
# Input:  reg  
# Output: <reg> <<< RC_BlkJSize_roundNum_mixNum, BlkJSize=256/512/1024  
#  
.macro RotL64    reg, BLK_SIZE, ROUND_NUM, MIX_NUM  
_RCNT_ = RC_\BLK_SIZE&_\ROUND_NUM&_\MIX_NUM  
    .if _RCNT_ #is there anything to do?  
        rolq    $_RCNT_, %\reg  
    .endif  
.endm
```

# Linker

## Linker

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	LLD 5+	LLD 5+
amd64	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	LLD 5	LLD
armv6	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	LLD 5+
arm64	-	-	-	BFD 2.25	LLD 4.0.0	LLD 5	LLD
mips	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.27	LLD 5+
powerpc	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.27	LLD 5+
sparc64	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.17.50	BFD 2.27	BFD 2.27
riscv64	-	-	-	-	-	BFD 2.27	LLD 6+

In-tree GNU BFD ld

In-tree LLD

Ports GNU BFD ld

# LLVM's LLD Linker Progress

- May 2015: start new COFF implementation
- July 2015: new ELF
- End 2015: LLD could form part of self-hosting FreeBSD/amd64 toolchain (build Clang/LLVM/LLD with Clang/LLVM/LLD)

# LLD 2016 Progress

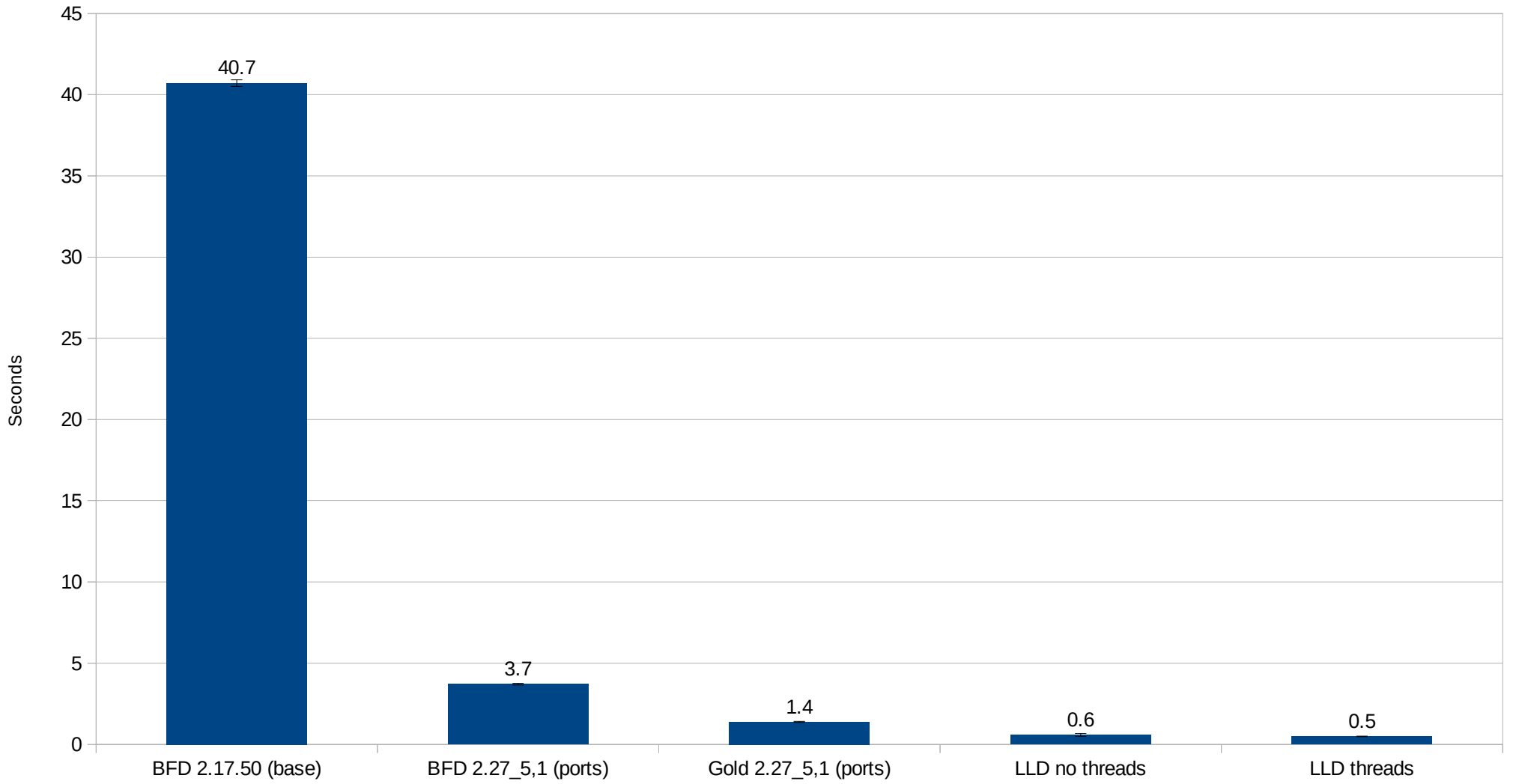
- FreeBSD work: Rui Ueyama, Rafael Espindola, George Rimar, Davide Italiano, Ed Maste
- Relocatable output
- Library search paths
- Linker scripts (arithmetic expressions)
- Symbol versioning
- Misc commandline options

# LLD

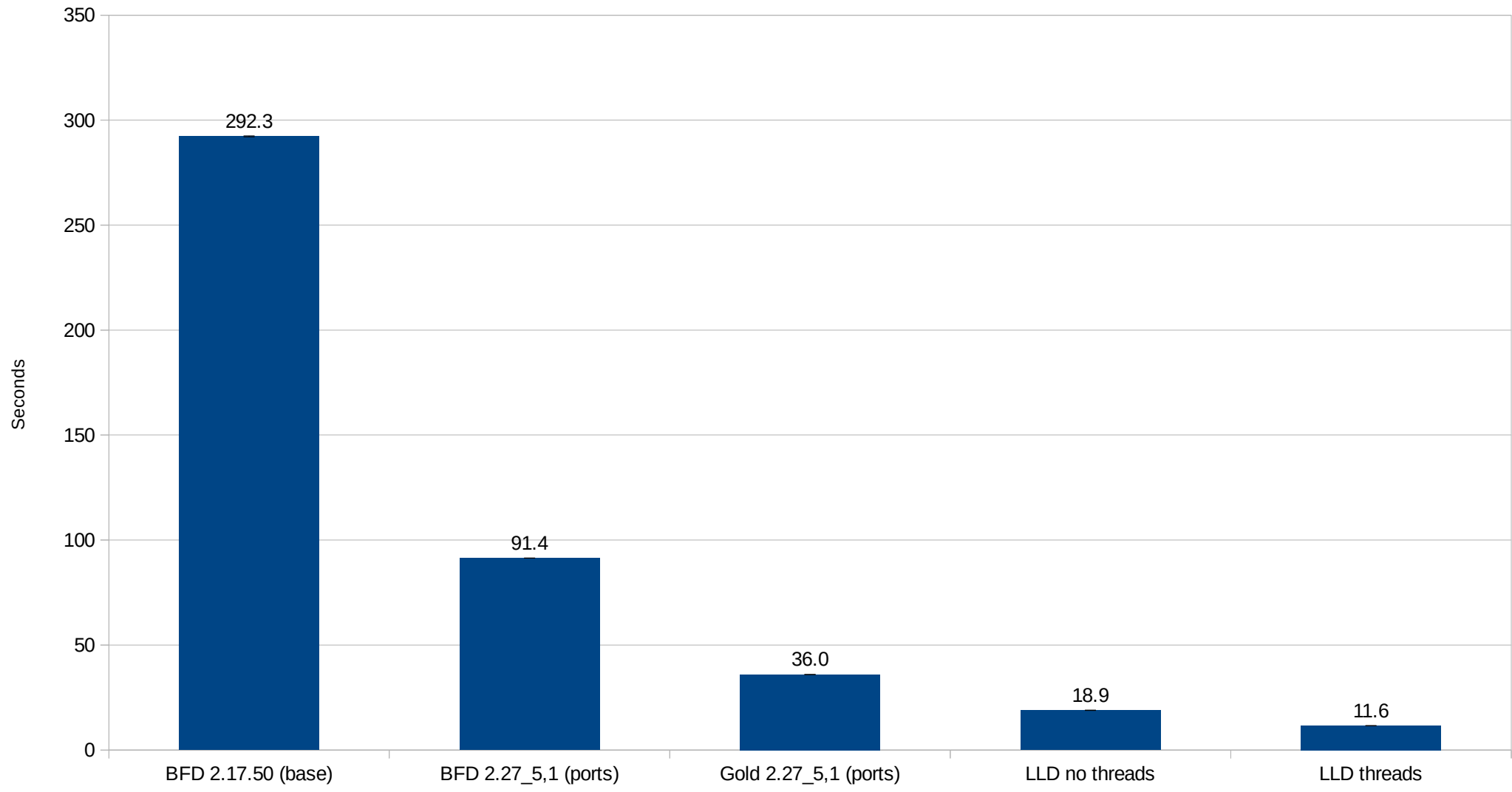
- [PR214864](#) exp-run with LLD as /usr/bin/ld
- amd64 world + kernel built fine
- 11 new port failures (and 200 skipped)



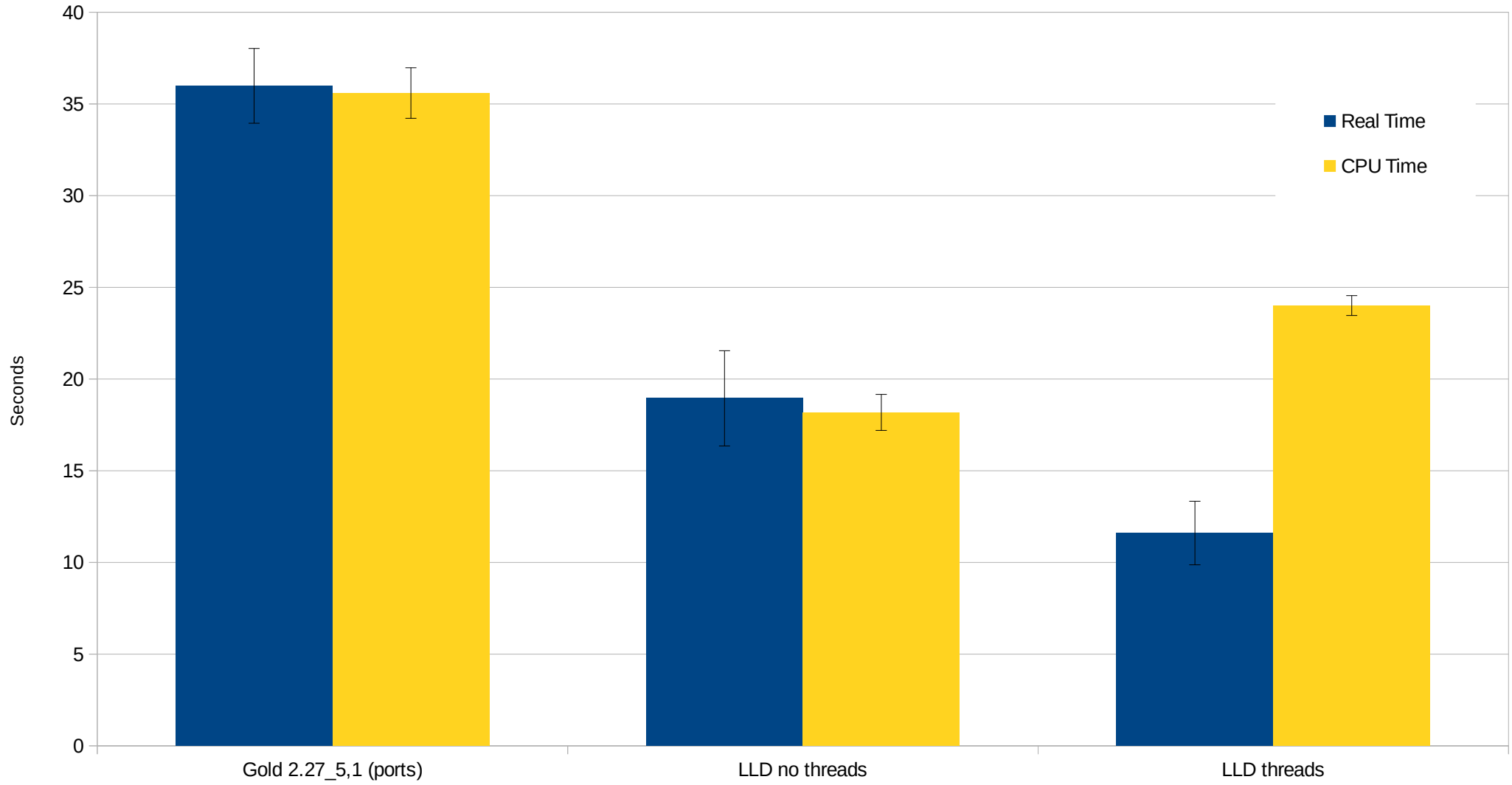
# Real Link Time, Clang Release Build



### Real Link Time, Clang Debug Build



Real & CPU Link Time, Clang Debug Build



# Linker Next Steps

- Enable build-id
- Incorporate LTO build infrastructure
- Fix arm and i386

# Binary Utilities

## Binary Utilities (other than elfdump, ar, ld)

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
All except below	BU 2.17.50	BU 2.17.50	BU 2.17.50	ElfTC 3477	ElfTC 3520	ElfTC	ElfTC/LLVM
arm64	-	-	-	ElfTC 3477 bfd 2.25	ElfTC 3520	ElfTC	ElfTC/LLVM
riscv64	-	-	-	-	-	ElfTC bfd 2.27	ElfTC/LLVM

In-tree GNU Binutils
In-tree ELF Tool Chain
Ports Binutils

# ELF Tool Chain

BSD licensed ELF toolchain

Brought to you by: [jkoshy](#), [kaiwang27](#)

Search Wiki

Create Page

Wiki Home

Browse Pages

Browse Labels

Subscribe to wiki

Formatting Help

Home

Edit

Authors:  

## The ELF Tool Chain Project

### Quick Links

- [SF.Net project overview.](#)
- [Mailing Lists.](#)
- [Source Tree.](#)
- [Wiki Index.](#)
- [Source analysis by Ohloh.Net.](#)
- [Commit log search, by FreshBSD.org](#)

### News

- **2016/02/18:** [Release 0.7.1](#) is now available for [download](#).
- **2012/09/24:** [Release 0.6.1](#) is now available for [download](#).
- **2011/11/16:** [Release 0.5.1](#) is now available for [download](#).

# ELF Tool Chain: POSIX/Native Tools

Tool	ELF TC	FreeBSD
ar	Released	Bespoke
as	Planning	BFD
brandelf	Released	Bespoke
c++filt	Released	ELF Tool Chain
elfdump	Released	Bespoke
isa	Devel	N/A
ld	Alpha	BFD / LLD
nm	Released	ELF Tool Chain
mcs	Released	N/A
ranlib	Released	Bespoke
size	Released	ELF Tool Chain
strings	Released	ELF Tool Chain
strip	Released	ELF Tool Chain

# ELF Tool Chain: GNU Tools

<b>Tool</b>	<b>ELF TC</b>	<b>FreeBSD</b>
addr2line	Released	ELF Tool Chain
elfcopy	Released	ELF Tool Chain
findtextrel	Released	N/A
objdump	Planning	GNU
readelf	Released	ELF Tool Chain



# ELF Tool Chain: Libraries

<b>Library</b>	<b>ELF TC</b>	<b>FreeBSD</b>
libdwarf	Released	ELF Tool Chain
libelf	Released	ELF Tool Chain
libelftc	Released	ELF Tool Chain
libmc	Planning	N/A
libpe	Alpha	ELF Tool Chain

# Binary Utilities

Bespoke

ar/ranlib, brandelf, elfdump

ELF Tool Chain

addr2line, c++filt, elfcopy/objcopy, nm, size, strings, strip, readelf

TBD?

objdump

# Device Tree Compiler

- Device Tree Source files (.dts) compiled to Device Tree Blob (.dtb)
- libfdt is dual GPL/BSD licensed, dtc is GPL
- David Chisnall's bespoke C++11 implementation at <https://svn.freebsd.org/base/head/usr.bin/dtc>

# Debugger

## Debugger

Arch	9.x	10.0	10.x	11.0	11.x	Soon	Later
i386	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 7.12	LLDB 5+ GDB 8
amd64	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	LLDB 3.8.0 GDB 6.1.1	LLDB 4.0.0 GDB 6.1.1	LLDB 4.0.0 GDB 7.12	LLDB 5 GDB 8
armv6	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	LLDB 5 GDB 8
arm64	-	-	-	LLDB 3.8.0	LLDB 4.0.0	LLDB 4.0.0	LLDB 5 GDB 8
mips	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 7.12	LLDB 5 GDB 8
powerpc	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 7.12	LLDB 5 GDB 8
sparc64	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 6.1.1	GDB 8
riscv64	-	-	-	-	-	?	LLDB 6+ GDB 8+

In-tree GNU GDB

In-tree LLDB

Ports GNU GDB

lldb

Python

IDE



LLDB APIs

LLDB Core

Breakpoint

Expression Parser

Commands

Host Abstraction

Interpreter

Script Interpreter



Symbol

Target

Data Formatters

Utility

Plugins

ABI

Operating System

Disassembler

Platform

Dynamic Loader

Process

Instruction

Symbol File

JITLoader

Symbol Vendor

Language Runtime

System Runtime

Object Container

Unwind Assembly

Object File

LLVM or Clang-provided components

# Debugger Next Steps

# Runtime Libraries

- `libc`, `libc++`, `libgcc`, `libunwind`, sanitizers

# Cast

Andrew Turner – andrew@ – all things arm64

Dimitry Andric – dim@ – Clang/LLVM, Binutils, compiler-rt

Ed Schouten – ed@ – compiler-rt

Ed Maste – emaste@ – LLD, LLDB, ELF Tool Chain

John Baldwin – jhb@ – GDB ports

Justin Hibbits – jhibbits@ – all things powerpc

Marius Strobl – marius@ – sparc64

Pedro Giffuni – pfg@ – misc Binutils and GCC

David Chisnall – theraven@ – C++ runtimes



Questions?

# Acknowledgements

“Crystal Clear” icon set from Everaldo Coelho  
(<http://www.iconarchive.com/show/crystal-clear-icons-by-everaldo.html>)