A Case Study: LLVM

Hal Finkel
ALCF

ATPESC 2013

A Case Study: LLVM

What is LLVM?

LLVM is an open source infrastructure for developing compilers and other software-development tools.

A Case Study: LLVM

Who develops LLVM?

Apple, Google, Intel, AMD, NVIDIA, ARM, IBM, and many other companies, academics, and individual contributors.

(Note that these companies are customers of and, competitors of, each other)

A Case Study: LLVM

How did LLVM begin?

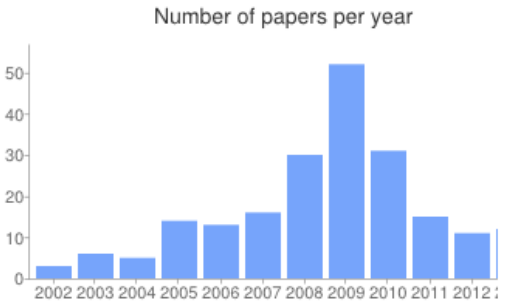
LLVM began as an academic research project at UIUC.

(Development is now driven by LLVM's numerous commercial contributors)

A Case Study: LLVM

LLVM is widely used in the research community:

2013



1. "Formal Verification of SSA Optimizations for LLVM"
   Jianzhou Zhao, Santosh Nagarakatte, Milo M K Martin, and Steve Zdancewic
   Proceedings of the 34th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI 2013), June 2013.

2. "Using Likely Invariants for Automated Software Fault Localization"
   Swarup Kumar Sahoo, John Criswell, Chase Geigle, and Vikram Adve
   Proceedings of the Eighteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2013), Mar. 2013.

3. "Safe and Automatic Live Update for Operating Systems"
   Cristiano Giuffrida, Anton Kuijsten, and Andrew S. Tanenbaum
   Proceedings of the Eighteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2013), Mar. 2013.

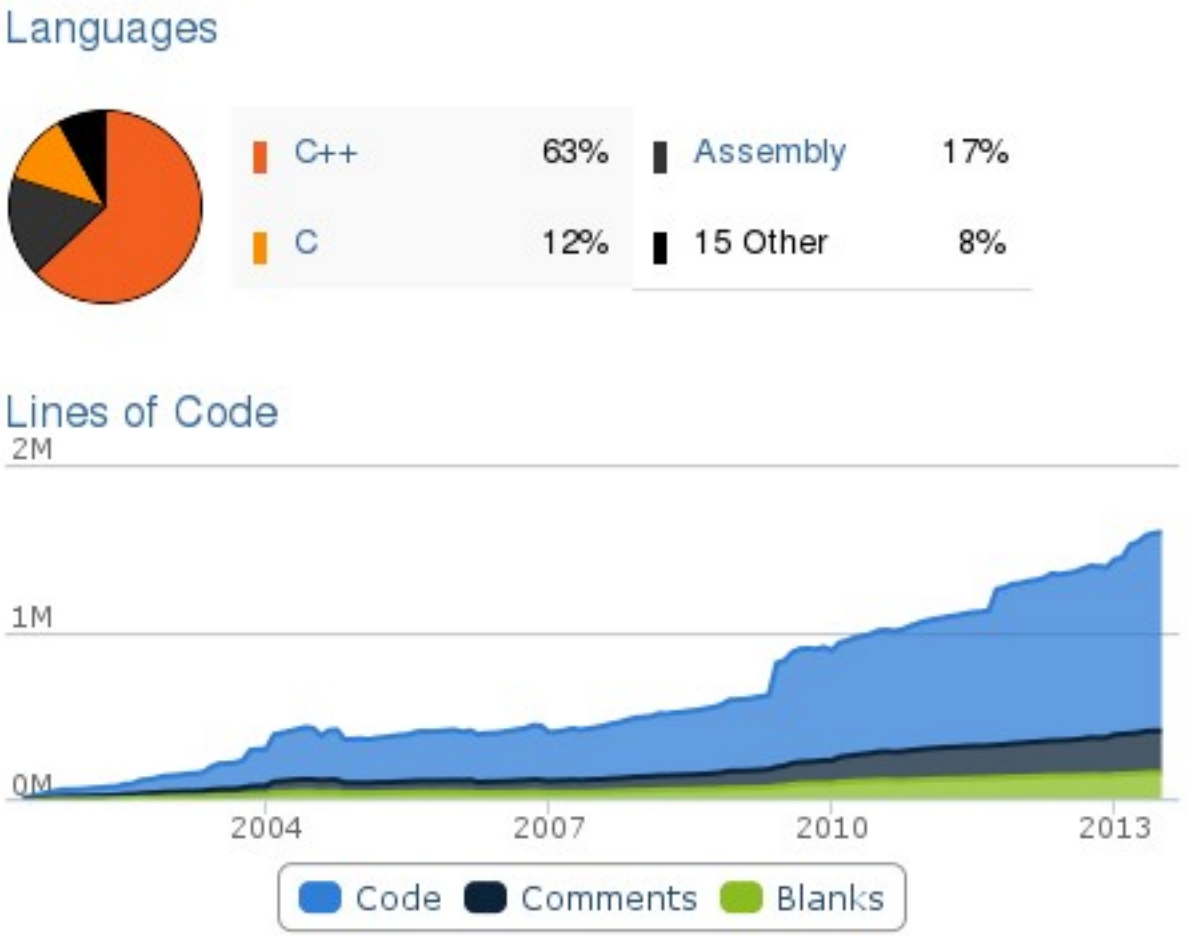
4. "Parallelizing Data Race Detection"
   Benjamin Wester, David Devecsery, Peter M. Chen, Jason Flinn, and Satish Narayanasamy
   Proceedings of the Eighteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2013), Mar. 2013.

5. "Cooperative Empirical Failure Avoidance for Multithreaded Programs"
   Brandon Lucia and Luis Ceze
   Proceedings of the Eighteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2013), Mar. 2013.

6. "ConAir: Featherweight Concurrency Bug Recovery via Single-Threaded Idempotent Execution"

A Case Study: LLVM

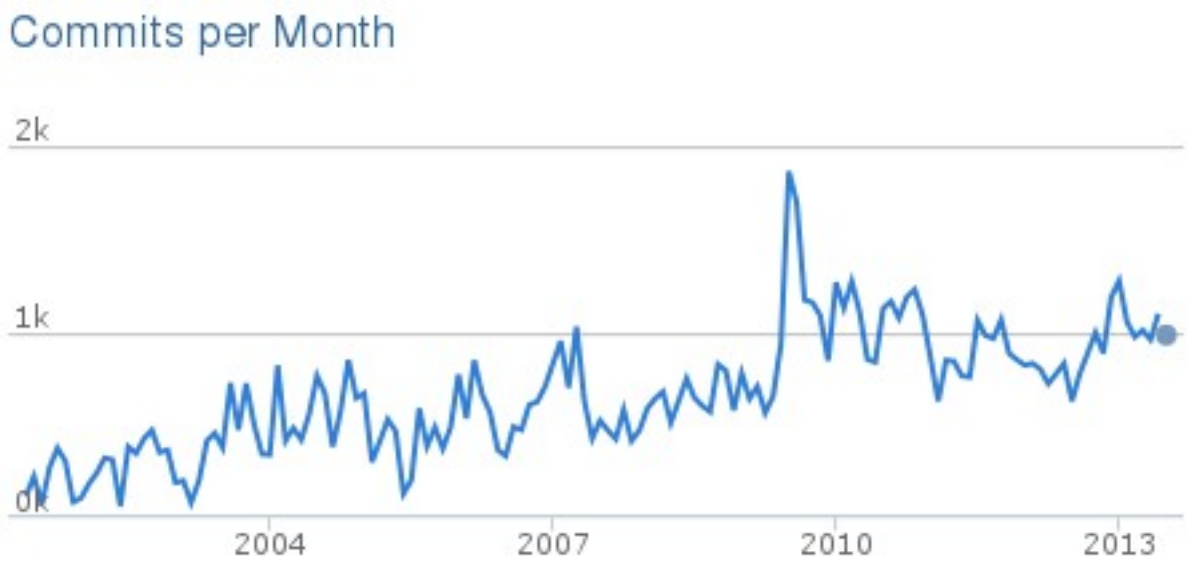
How large is LLVM?



Source: ohloh.net

A Case Study: LLVM

How active is LLVM?



Source: ohloh.net

A Case Study: LLVM

How many people commit?

Source: ohloh.net

A Case Study: LLVM

Why do competing companies contribute to the same project?

- Because they all get out more than they put in, and they know it.
- Positive ROI both in the short term (features) and in the long term (maintenance).

A Case Study: LLVM

What makes it work?

✔Quality
✔Dependability
✔Modularity
✔Community

(Let's review these in reverse order)

A Case Study: LLVM

The Community:

- Communication (Mailing lists, IRC, Meetings, Social Events, etc.)
- Culture: Maintenance sharing, being helpful/encouraging to newcomers, respecting all opinions (with no foul language), while still being firm about the rules.

A Case Study: LLVM

Modularity:

- Almost all components designed to be used as a library.
- Strict layering requirements (no cyclic dependencies).
- Well-documented file formats, interfaces and contracts.

A Case Study: LLVM

Dependability:

- Almost all changes (especially bug fixes) add a regression test.
- LLVM/Clang has a large application test suite (for functional correctness) in addition to the regression tests.
- Continuous build-and-test servers, including some under sanitizers/valgrind; supported platforms must be green.

A Case Study: LLVM

Quality:

- Almost all code is reviewed prior to being committed, for both functional and stylistic concerns. An established contributor (the code owner for major changes) must sign off before committing.
- Unless impractical, things must be done ***the right way***: quality is part of the culture.

A Case Study: LLVM

- LLVM is a compiler infrastructure, but there many associated projects:
- Clang: C/C++ frontend
- dragonegg: GCC to LLVM bridge plugin
- Clang's static analyzer
- Address/memory/thread sanatizer
- libc++ C++11 STL implementation
- polly: polyhedral loop optimizations
- And several others, plus many independent projects (ispc, pocl, gpu ocelot, halide, julia, JIT for several scripting languages, etc.)

bgclang: Clang/LLVM on the BG/Q

- Port of LLVM/Clang to the BG/Q
- Provides C99/C++03 and C++11 programming environments
- Supports the QPX vector instructions (xlc-compatible intrinsics and autovectorization)
- Automatic software prefetch in loops
- ALCF softenv keys: +mpiwrapper-bgclang and +mpiwrapper-bgclang.legacy
- Full C++11 environment (using libc++) in mpic++11
- OpenMP is coming soon.

bgclang: debugging

- MPI-specific warning messages (type matching)
- Address sanitizer: -fsanitize=address

  (especially useful because we don't have valgrind)
- Clang static analyzer

bgclang community

- http://trac.alcf.anl.gov/projects/llvm-bgq
- https://lists.alcf.anl.gov/mailman/listinfo/llvm-bgq-discuss
- IRC: llvm-ppc64 on irc.oftc.net

For more information, e-mail the llvm-bgq-discuss list.