OpenMP* Support in Clang* / LLVM*

Andrey Bokhanko, Intel

What Is Clang / LLVM?

- LLVM: “Collection of modular and reusable compiler and toolchain technologies”
  - Created by Chris Lattner and Vikram Adve at UIUC
  - Developed by Apple*, Google*, Intel, IBM*, Sony*, ARM*, …
  - Interests of major players are mostly outside HPC
- Clang: “…a new C, C++ Objective C and Objective C++ front-end for LLVM compiler”
  - First to implement full C++11 (and now C++14) support
- “UIUC” BSD-style license

Copyright © 2014, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.
Optimization Notice

OpenMP In Clang: Some History

- There were a few proprietary implementations (Pathscale, Cray), but no open-source one
- "OpenMP in Clang" project
  - Started in late'12, by AMD* and Intel
  - Now ANL*, IBM*, Micron*, Pathscale*, TI* and UoHouston* are involved
- Early outlining
  - It is hard to introduce changes to LLVM IR
  - New stuff should be very general and "universal"
  - There were some proposals with late outlining, all rejected by community

Copyright © 2014, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.
Optimization Notice

clang-omp Github Repo: Home Of OpenMP-enabled Clang Compiler

- clang-omp.github.com
  - Based on clang/llvm 3.5 release
  - Also, trunk-based branch is available (maintained by Hal Finkel)
- Full OpenMP 3.1 implemented
- Partial OpenMP 4.0 implemented
- Initially developed for x86 and x86-64; ported to POWER* and ARM*
- Patches are gradually upstreamed to clang / llvm trunk
  - OpenMP 3.1 planned for llvm 3.6; OpenMP 4.0 for llvm 3.7

Copyright © 2014, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.
Optimization Notice

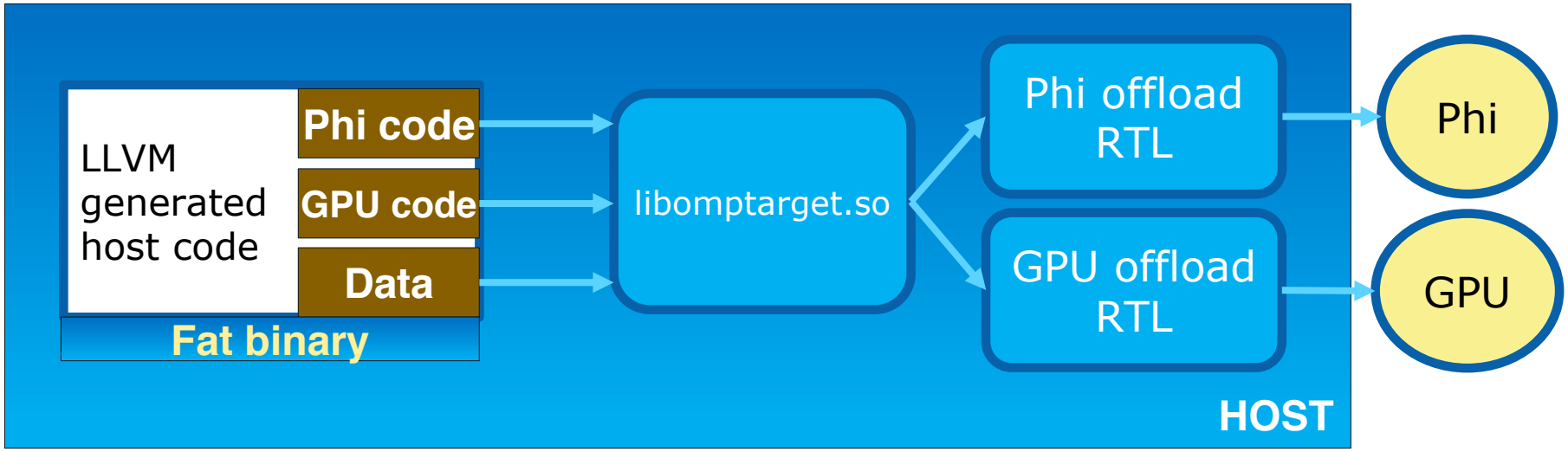
OpenMP 4.0: Supported Features

- Task enhancements, thread affinity, user-defined reductions
  - Done
- #pragma omp simd
  - Functionally complete
    - Support for “#pragma omp declare simd” and “reduction” clause missing in llvm back-end
  - Performance depends on vectorizer
  - Partially available in clang / llvm 3.5 release
    - Only basic pragma and “safelen” clause supported
  - Partially available in clang / llvm trunk
    - Support for “collapse” and “aligned” clauses added

Copyright © 2014, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.
Optimization Notice

OpenMP 4.0: Offloading

- Under development
- Plan to support x86, x86-64, POWER and ARM as hosts, multiple targets (Intel® Xeon Phi™ coprocessor, GPUs, FPGAs, ...)
- Offloading library open-sourced under LLVM license



Copyright © 2014, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.
Optimization Notice

OpenMP Runtime Library

- openmp.llvm.org
- Tried and true Intel OpenMP runtime
  - Production runtime used by icc and ifort
- Continual development / tuning since before the OpenMP language existed (>15 years)
- Highly scalable
  - Used on Intel® Xeon Phi™ coprocessor with 244 threads, large SGI* and Bull* ccNUMA SMP machines)
- Full OpenMP 4.0 support
- In addition to x86 and x86-64, ported to POWER and ARM
- Open-sourced under LLVM license

Copyright © 2014, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.

You Are Welcome!

- A lot of people already involved
- Help with development
  - Code, tests, bug reports
- Use OpenMP-enabled clang!
  - Get performance boost
  - Report your results to us
  - Influence future developments
  - Innovate on top of what we did
- clang-omp.github.com
  openmp.llvm.org

Copyright © 2014, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.
Optimization Notice

Legal Disclaimer & Optimization Notice

INFORMATION IN THIS DOCUMENT IS PROVIDED “AS IS”. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO THIS INFORMATION INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

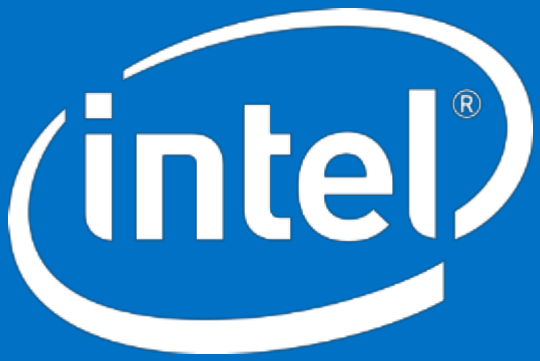
Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Copyright © 2014, Intel Corporation. All rights reserved. Intel, Pentium, Xeon, Xeon Phi, Core, VTune, Cilk, and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.

Optimization Notice

Intel’s compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804


intel®