Providing a Robust Tools Landscape for CORAL Machines

4th Workshop on Extreme Scale Programming Tools @ SC15 in Austin, Texas

November 16, 2015

Dong H. Ahn	Lawrence Livermore National Lab
Michael Brim	Oak Ridge National Lab
Scott Parker	Argonne National Lab
Gregory Watson	IBM
Wesley Bland	Intel

CORAL: Collaboration of ORNL, ANL, LLNL

Objective - Procure 3 leadership computers to be sited at Argonne, Oak Ridge and Lawrence Livermore in 2017.

Current DOE Leadership Computers

Titan (ORNL) 2012 - 2017

Sequoia (LLNL) 2012 - 2017

Mira (ANL) 2012 - 2017

Leadership Computers - RFP requests >100 PF, 2 GiB/core main memory, local NVRAM, and science performance 4x-8x Titan or Sequoia

Approach

- Competitive process - one RFP (issued by LLNL) leading to 2 R&D contracts and 3 computer procurement contracts
- For risk reduction and to meet a broad set of requirements, **2 architectural paths will be selected** and **Oak Ridge and Argonne must choose different architectures**
- Once Selected, Multi-year Lab-Awardee relationship to co-design computers
- **Both R&D contracts jointly managed by the 3 Labs**
- Each lab manages and negotiates its own computer procurement contract, and may exercise options to meet their specific needs

OAK RIDGE National Laboratory

Argonne NATIONAL LABORATORY

Lawrence Livermore National Laboratory

CORAL Innovations and Value

IBM, Mellanox, and NVIDIA Awarded $325M U.S. Department of Energy's CORAL Contracts

- Scalable system solution – scale up, scale down – to address a wide range of application domains
  - Modular, flexible, cost-effective,
  - Directly leverages OpenPOWER partnerships and IBM's Power system roadmap
  - Air and water cooling
- Heterogeneous compute elements with POWER processor, Volta GPU coupled with NVLINK
- Enhanced Mellanox interconnect
- Open source LLVM compiler, XL compiler, PGI compiler
  - Support for high level, open, programming models
- System Resource Manager coupled with Platform Computing LSF
- Spectrum Scale – GPFS

CORAL Centers of Excellence

- Department of Energy has funded two Centers of Excellence
  - Sierra machine at Lawrence Livermore National Laboratory
  - Summit machine at Oak Ridge National Laboratory
- In place now through early science application deployments

Sierra

Summit

CORAL System Configuration

System	> 100 PF peak; at least 5x Titan/Sequoia application performance
Power	~ 10 MW
Nodes	~ 3,400; > 40 TF peak per node
CPUs	Multiple IBM POWER9™
GPUs	Multiple NVIDIA® Volta™
On-chip Interconnect	NVLINK
Memory	Coherent; > 800 GB (HBM + DDR2) per node; local NVRAM
Interconnect	Dual-rail Mellanox® EDR-IB; full non-blocking fat-tree
I/O Bandwidth	1 TB/s
Disk Capacity	120 PB

CORAL Software Stack

Function	Software
Operating system	Linux
System management	PCM, xCAT, GPFS
Resource management	IBM Platform LSF
Programming models	CUDA, MPI, OpenMP, OpenACC, GA/ARMCI, CCI, GASNet, OpenSHMEM
Compilers	LLVM, IBM XL, PGI
Communication Libraries	MRNet
Math Libraries	ESSL, FFTW, ScaLAPACK, PETSc, Trilinos
I/O Libraries	netCDF, HDF5
MPI Library	Open MPI

CORAL Programming Tools

Activity	Supported Tools
Debugging	pdb, TotalView, DDT
Stack traceback	STAT
Profiling	IBM PE DE, Open|SpeedShop, TAU, HPCToolkit, VAMPIR, nvprof
Tracing	IBM PE DE, Score-P, Open|SpeedShop
Hardware performance monitoring	IBM PE DE, PAPI, nvprof
Memory/thread checking	Valgrind
Build	GNU Autotools, GNU Make, CMake

Two Upcoming Argonne Systems

Theta

- Arriving in 2016
- Greater than 8.5 Petaflops peak performance
- Cray XC platform
- Self hosted 2nd generation Xeon Phi processor

Aurora

- Arriving in 2018
- Greater than 180 Petaflops peak performance
- Cray Shasta platform
- Self hosted 3rd generation Xeon Phi processor

Common Architectural Features

- **Xeon Phi Processors**
  - Theta: 2nd Generation Knight Landing (KNL)
    - Intel® "Silvermont" enhanced for HPC
    - 60+ processor cores with 2 cores per tile connected in a 2D mesh network
    - 1-MB cache-coherent L2 cache shared between 2 cores in a tile
    - AVX-512 vector pipelines with a hardware vector length of 512 bits
    - Two vector units
  - Aurora: 3rd Generation Knights Hill (KNH)
- **Heterogeneous Memory**
  - High speed memory
    - On socket
    - High bandwidth – 100's of GB/s
    - Limited capacity – 10's of GB
  - Capacity memory
    - Higher capacity, 100's GB
    - Limited bandwidth
  - Node local non-volatile memory
- **Network**
  - Dragonfly topology

Software Stack

- **Cray software stack with Intel improvements**
  - Compilers: Intel, Cray, and GNU
  - Languages: C, C++, Fortran, Coarray Fortran, UPC, Chapel
  - Programming Models: MPI, OpenMP*, Intel TBB, Cilk Plus, SHMEM
- **Communication: optimized for high performance and scalability**
  - Multiple MPI options: MPICH3, Intel® MPI, Cray MPI
- **Performance libraries**
  - Intel® Math Kernel Library
  - Cray Scientific & Math Libraries
  - BLAS, ScaLAPACK, FFTW, PETSc, Trilinos
- **Application analysis tools**
  - Intel® Parallel Studio XE
  - Cray Performance Analysis Suite
- **System and Infrastructure: focused on scalability and reliability**
  - Multiple OS environment, including low-jitter Linux
  - Lustre* distributed file system with efficient user-space I/O offload
  - Scheduling and resource management: Cobalt & Cray ALPS

Theta Details

- **System**
  - Cray XC supercomputing platform
  - > 8.5 PF peak
  - >2,500 single socket nodes
- **Processor**
  - 2nd Generation Intel® Xeon Phi™ processors (Knights Landing)
  - >60 cores with four hardware threads per core
- **Memory**
  - Up to 16 Gigabytes High Bandwidth Memory per compute node
  - HBM bandwidth >400 Gigabytes/sec
  - 192 Gigabytes DDR4 using 6 channels per compute node
- **Network**
  - Cray Aries high speed Dragonfly topology interconnect
- **Storage**
  - 128 Gigabytes SSD per compute node
  - 10 PB Lustre fie system
  - 200 Gigabytes/s bandwidth

Aurora Details

- **System**
  - Cray Shasta next generation supercomputing platform
  - > 180 PF peak
  - >50,000 nodes
- **Processor**
  - 3rd Generation Intel® Xeon Phi™ processor (Knights Hill)
- **Memory**
  - Total memory >7 PetaBytes
  - Total HBM bandwidth >30 PB/s
- **Network**
  - 2nd Generation Intel® Omni-Path Architecture with silicon photonics
  - Dragonfly topology
  - Aggregate node link bandwidth >2.5 PB/s
  - Bi-sectional bandwidth >500 TB/s
- **Storage**
  - Intel® SSDs, 2nd Generation Intel® Omni-Path Architecture
  - Intel Lustre* File System with > 150 Petabytes
  - >1 TeraByte/s file system throughput

CORAL System Programming Models

Global Arrays

Cray SHMEM, UPC

Fortran 2008 Co-arrays

POSIX Threads

LLNL RAJA (C++11)

Cray Co-arrays (Fortran, C++)

C++11 Threads

Charm++

Fortran 2008

C11

C++11

Python

DOE/SC ASCR: Two Tracks for Future Large Systems

Tianhe-2 (NUDT): TH-IVB-FEP
Intel Xeon E5-2692 12 C 2.2 GHz
TH Express-2
Intel Xeon Phi 31S1P

Titan (Cray): Cray XK7
AMD Opteron 6274 16C 2.2 GHz
Cray Gemini
NVIDIA K20x

Sequoia (IBM): BlueGene/Q
Power BQC 16C 1.6 GHz

K computer (Fujitsu)
SPARC64 VIIIfx 2.0 GHz
Tofu

Mira (IBM): BlueGene/Q
PowerPC A2 16C 1.6 GHz

Piz Daint (Cray): Cray XC30
Intel Xeon E5-2670 8C 2.6 GHz
Cray Aries
NVIDIA K20x

Edison (Cray): Cray XC30
Intel Xeon E%-2695v2 12C 2.4 GHz
Aries

Many Core

- 10's of thousands of nodes with millions of cores
- Homogeneous cores
- Multiple levels of memory – on package, DDR, and non-volatile
- Unlike prior generations, future products are likely to be self hosted

Hybrid Multi-Core

- CPU / GPU Hybrid systems
- Likely to have multiple CPUs and GPUs per node
- Small number of very fat nodes
- Expect data movement issues to be much easier than previous systems – coherent shared memory within a node
- Multiple levels of memory – on package, DDR, and non-volatile

Cori at NERSC

- Intel/Cray
- Self-hosted many-core system
  - Intel Xeon Phi Knights Landing (KNL)
  - 16GB HBM, 64-128 GB DDR4
  - Cray Aries interconnect
- 9,300 single-socket nodes
- Target delivery date: 2016

Summit at OLCF

- IBM/NVIDIA/Mellanox
- Hybrid CPU/GPU system
  - IBM POWER9 + NVIDIA Volta
  - More than 512 GB coherent memory per node
  - EDR Infiniband interconnect
- 3,400+ multi-socket nodes
- Target delivery date: 2017

Aurora at ALCF

- Intel/Cray
- Self-hosted many-core system
  - Intel Xeon Phi 3rd Gen
  - HBM + Persistent memory
  - Intel OmniPath 2nd Gen interconnect
- 50,000+ single-socket nodes
- Target delivery date: 2018

DOE/SC ASCR: Importance of Performance Portability

Application portability among NERSC, ALCF and OLCF architectures is a critical concern to DOE's Office of Science Advanced Scientific Computing Research program

- Application developers target wide range of architectures
- Maintaining multiple code version is difficult
- Porting to different architectures is time-consuming
- Many Principal Investigators have allocations on multiple resources
- Applications far outlive any computer system

Primary task is exposing parallelism and data locality

Challenge is to find the right abstraction:

- MPI + X (X=OpenMP, OpenACC)
- PGAS + X
- DSL
- …

DOE/SC ASCR: Application Readiness Programs

NESAP at NERSC

NERSC Exascale Science Application Program

- Call for Proposals – June 2014
- 20 Projects selected
- Partner with Application Readiness Team and Intel IPCC
- 8 Postdoctoral Fellows

Criteria

- An application's computing usage within the DOE Office of Science
- Representation among all 6 Offices of Science
- Ability for application to produce scientific advancements
- Ability for code development and optimizations to be transferred to the broader community through libraries, algorithms, kernels or community codes
- Resources available from the application team to match NERSC/Vendor resources

CAAR at OLCF

Center for Accelerated Application Readiness

- Call for Proposals – November 2014
- 13 Projects selected
- Partner with Scientific Computing group and IBM/NVIDIA Center of Excellence
- 8 Postdoctoral Associates

Criteria

- Anticipated impact on the science and engineering fields
- Importance to the user programs of the OLCF
- Feasibility to achieve scalable performance on Summit
- Anticipated opportunity to achieve performance portability for other architectures
- Algorithmic and scientific diversity of the suite of CAAR applications.
- Optimizations incorporated in master repository
- Size of the application's user base

ESP at ALCF

Early Science Program

- Call for Proposals
- 10 Projects to be selected
- Partner with Catalyst group and ALCF Vendor Center of Excellence
- Postdoctoral Appointee per project

Criteria

- Science Impact
- Computational Readiness
- Proposed science problem of appropriate scale to exercise capability of new machine
- Confidence code will be ready in time
- Project code team appropriate
- Willing partner with ALCF & vendor
- Diversity of science and numerical methods
- Samples spectrum of ALCF production apps

Synergy Between Application Readiness Programs

- Application Developer Team involvement
  - Knowledge of the application
  - Work on application in development "moving target"
  - Optimizations included in application release
- Early Science Project
  - Demonstration of application on real problems at scale
  - Shake-down on the new system hardware and software
  - Large-scale science project is strong incentive to participate
- Vendor technical support is crucial
  - Programming environment often not mature
  - Best source of information on new hardware features
- Access to multiple resources, including early hardware
- Joint training activities

- Portability is a critical concern
- Experience benefits other developers and users
  - Coverage of scientific domains
  - Coverage of algorithmic methods and programming models
- Persistent culture of application readiness
  - More computational ready applications available
  - Experience of science liaisons and catalysts for user programs
  - Synergy with libraries and tools projects

- ***Which capabilities will be available to enable performance portability?***
- ***How can we use current software development tools for the architectures that will be delivered in 2-4 years?***

Preparing Tools for Aurora and Theta

- **ALCF-3 Early Tools project**
  - Modeled on successful ALCF-2 Early Tools project, ported 15 popular 3rd party packages to BG/Q
  - Access to early hardware
  - Training and information on Intel Xeon Phi and software environment
  - Share code porting & tuning experience with community through technical reports and community workshop presentations
- **Early Tools efforts is part of the larger Early Science Program**
  - **Theta phase (Q1 CY2015 – CY2017)**
    - Six science projects (3 preselected, 3 from Call For Proposals)
    - Funded postdoc for 4 projects
  - **Aurora phase (Q3 CY2016 – CY2019)**
    - Ten projects (all from a Call For Proposals)
    - Funded postdocs for all 10 projects
- **ALCF Intel-Cray Center of Excellence**
  - Resource for Theta software
  - Support from Intel and Cray staff

HPC Open-Source Tools Enabling for Aurora

Enable open source profiling and analysis tools for HPC to run well on Intel's newest and upcoming high-end server platforms.

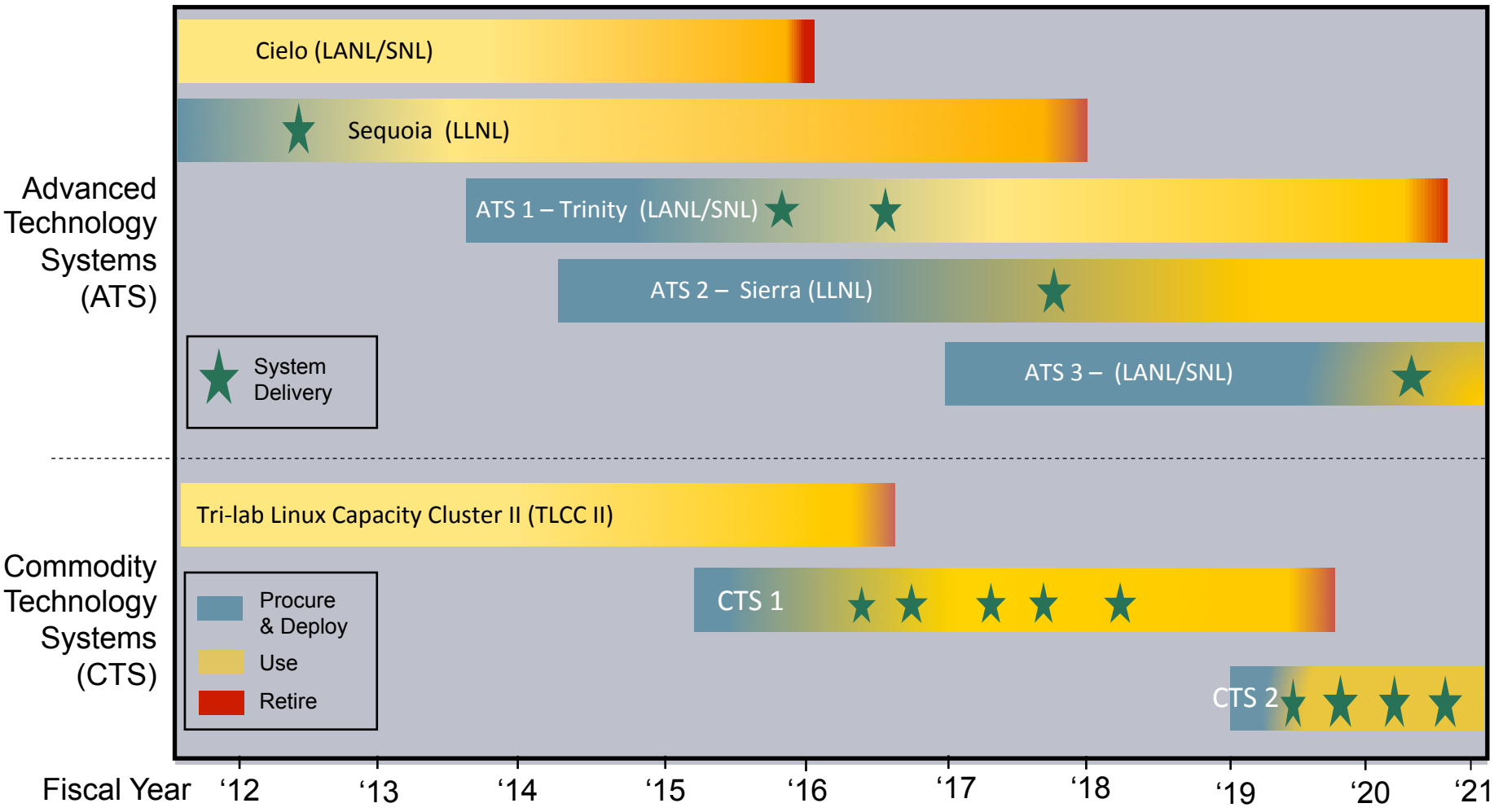
Important Goal: Develop relationships with institutions and tool owners

- Contribute patches to ensure tool coverage, quality, and performance on Intel platforms
- Demonstrate a path for all tools on the new platforms via Intel and GNU compilers

Be involved with OpenHPC to make tools available to the community

HPC Open-Source Tools Enabling for Aurora
Current Sample of Tools and Status Overview

	Tool	Description	Status
Low-level	Dyninst (U. Wisc)	Dynamic binary instrumentation tool	GNU and Intel compilations, Test suite completed, Minor change to CMake configuration (***patch accepted***); ***patch for test*** *suite; early KNL tests* Versions 8.2.1 (Jun. '15), 9.0.2 (Sep. '15), 9.0.3 (Oct '15) Ported examples and investigation into failed tests from test suite
	PAPI (U. Tenn-Knoxville)	Interface to sample CPU and off-core performance events	GNU and Intel compilations, Test suite completed, ***Patch accepted*** to enable off-core events in Component tests. (Jul. '15) Versions 5.4.1, 5.4.2
High-level	TAU (U. Oregon)	Profiling and tracing tool for parallel applications, supporting both MPI and OpenMP	Intel Compilation with Intel MPI and Intel C/C++/Fortran compilers, Dyninst, and PAPI. Many suite examples tested (Jul. '15) MPICH and GNU installations (Oct. '15) Version: 2.24.1
	Score-P (Vi-HPS)	Provides a common interface for high-level tools	**In Progress** TAU + Score-P (Intel, GCC compilers, Dyninst, PAPI, Intel MPI, MPICH) Version 1.3
	Open|SpeedShop (Krell)	Dynamic Instrumentation tool for Linux: profiling, event tracing for MPI and OpenMP programs. Incorporates Dyninst and PAPI	**In Progress** Intel, GNU, Intel MPI, MPICH, Dyninst, PAPI Version 2.2
	HPCToolKit (Rice U.)	Lightweight sampling measurement tool for HPC; supports PAPI	GNU and Intel compilations with Intel MPI and PAPI. Tested with compute pi and Intel pre-built 64-bit HPL. (Aug. '15) Version 5.4.0
	Darshan (Argonne Nat. Lab)	IO monitoring tool	Intel and GCC compilers, Intel MPI, MPICH Tested with IOR benchmark (Sep. '15) Version: 2.3.1
Independent	Valgrind Base	framework for constructing dynamic analysis tools; includes suite of tools including a debugger, and error detection for memory and pthreads.	**In Progress** GNU and Intel compilations; test suite passes under GNU; In progress under Intel.16.0.* compiler; Version: 3.11.0
	Valgrind's memcheck	Detects memory errors: stack, heap, memory leaks, and MPI distributed memory. For C and C++.	**In Progress** test suite passes under GNU; In progress under Intel.16.0.* compiler; Version: 3.11.0
	Valgrind's helgrind	Pthreads error detection: synchronization, incorrect use of pthreads API, potential deadlocks, data races. C, C++, Fortran	**In Progress** test suite passes under GNU; In progress under Intel.16.0.* compiler; Version: 3.11.0



ASC Platform Strategy includes application code transition for all platforms

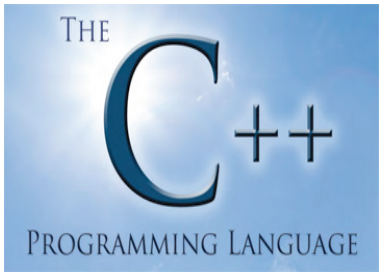
LLNL's programming challenge is significant

- ASC applications are complex and long lived
  - Several existing 1M+ LOC applications
  - ~10 years from application start to complete mission acceptance
- Long application lifetimes prioritize desired properties
  - Maintainability
  - Portability of performance as well as functionality
- Architectural challenges are looming
  - Accelerators, in the form of GPUs for Sierra
  - Complex memory hierarchies: The real exascale challenge

We need a solution today that will work on Sierra and long into the future

LLNL will use RAJA on top of MPI + OpenMP

- RAJA
  - Build on new features of C++ standard
  - High-level abstractions reduce unsustainable recoding



- MPI
  - Use a proven solution for portable internode parallelism
  - Use new features as necessary

- OpenMP
  - Actively extend a portable intranode model to address key issues
  - Support complex node architectures
  - Has/will have critical features to support RAJA

Flexible abstractions built on top of accessible, low-level models provide hope

RAJA is a C++ abstraction layer that encapsulates architecture-specific concerns

C-style for-loop

```
double* x ; double*y ;
double a ;
// …
for ( int i = begin; i < end; ++i )  {
    y[ i ] += a * x[ i ] ; }
```

RAJA-style loop

```
Real_ptr x;   Real_ptr y ;
Real_type a ;
// …
forall< exec_policy >( IndexSet, [&] (Index_type i)  {
    y[ i ] += a * x[ i ] ; }  );
```

- Data type encapsulation: Hides compiler directives, data attributes, etc.
- Traversal template & execution policy: Encapsulate platform-specific scheduling
- Index set: Encapsulates iteration space partitioning & data placement
- C++ lambda function: Captures unmodified loop body

RAJA applies to a wide range of loops including ones with indirect addressing

OpenMP 4.0 device constructs were essential for LLNL's Sierra decision

- Use `target` directive to offload a region

```
#pragma omp target [clause [[,] clause] …]
```

- Clauses support data movement and conditional offloading
  - `device` supports offload to a device other than default
  - `map` ensures variables accessible on device (may share host memory)
  - `if` supports running on host if amount of work is small

- Other constructs support device data environment
  - `target data` places `map` variables in device data environment
  - `target update` ensures variables are consistent on host and device

These constructs not only support accelerators but also memory placement

Providing a rich and most useful tools ecosystem is nontrivial

- Take a look at one class of tools: debugger
  - Ported to little endian POWER
  - DWARF qualified for all compilers
  - Integrated with OpenMP awareness
  - Being able to debug multiple processes driving a GPU
- MANY tools must work with MANY other system elements
  - Compilers, OpenMP runtimes, CUDA debug interface …
- Co-design is the key
  - Stakeholders actively involved in tools interface design
  - Info sharing; give and take; and quick progress towards common goals

CORAL/IBM tools working group facilitates co-design activities

- Provide effective tool set by facilitating co-design
- Currently consists of over 35 stakeholders and growing
- Regularly scheduled bi-weekly concalls
- The strategy to meet this purpose:
  - Identified representative tools and their dependent elements
  - Analyzed gaps and co-design needs to fill
  - Spun off co-design sub-teams (8 identified)

Co-design with OpenMP runtime implementers and NVIDIA tools interface teams is critical

- 5 of 8 teams to support OpenMP and GPU programming
- Many technical interlocks start to happen and await us …
  - OpenMP debug interface (OMPD)
  - OpenMP performance tools interface (OMPT)
  - OMPT extension to support OpenMP race detectors
  - Identify/provide meaningful hardware counter events through early user feedback and tool use

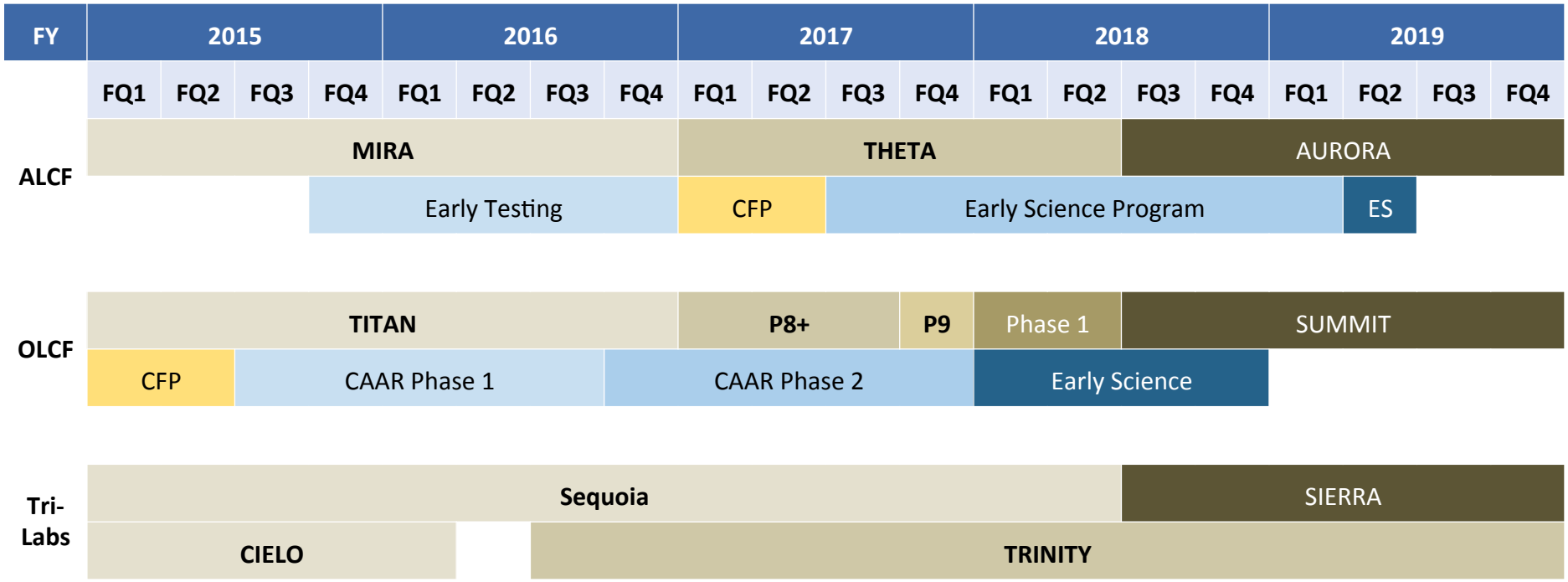
ORNL – Tool Engagement Activities

- Working Groups
  - CORAL/IBM working group on Open Source Tools
  - OpenPOWER Foundation Academia Discussion Group
  - 2015 Scalable Tools Workshop working group on "Tools for CORAL"
    - 2016 workshop slated for early August
- Coordination with existing tool partners for debugging and performance analysis
  - strategic funding to develop critical features
- Evaluation of vendor and open source tools, and related support infrastructure, on current generation OpenPOWER (IBM/NVIDIA) testbed
  - provide feedback to developers on issues/deficiencies/opportunities

ORNL – Tool Engagement Opportunities in CAAR

- CAAR program to identify and invest in scientific applications that have the potential to deliver early, high-impact science accomplishments
  - modeled after successful program for transition to hybrid computing on Titan
  - project selection criteria involved willingness to utilize tools and provide use cases
- Joint development effort among application teams, OLCF staff, and IBM/NVIDIA Center of Excellence
  - provide porting **expertise**
  - improve the software development environment
    - scientific libraries, compilers, **debuggers**, **correctness/performance analysis tools**
- Early access to testbeds and Summit
  - measure readiness for production, and select projects that will receive "early science" computational resources
  - provides valuable feedback to OLCF on potential system issues/problems before final system acceptance

System Timelines (Tentative)



CORAL Tools - Points of Contact

- CORAL Leads for Tool Activities
  - ANL: Scott Parker [sparker <at> anl.gov]
  - LLNL: Dong H. Ahn [ahn1 <at> llnl.gov]
  - ORNL: Mike Brim [brimmj <at> ornl.gov]

- Public Mailing Lists
  - coral-tools-announce <at> email.ornl.gov
    - subscribe to receive announcements of newly released or otherwise interesting information related to CORAL tools
  - coral-tools-discuss <at> email.ornl.gov
    - non-NDA discussion forum for CORAL tool development and support
    - post tool use cases and functionality regression tests

CORAL Tools – Vendor Contacts

- Aurora
  - Intel:
    - Rashawn Knapp [rashawn.l.knapp <at> intel.com]
    - Preeti Suman [psuman <at> intel.com]
    - Rashawn Knapp [tmineeva <at> intel.com]
- Sierra & Summit
  - IBM
    - Serban Maerean [serban <at> us.ibm.com]
  - Mellanox
    - Scot Schultz [scots <at> mellanox.com]
    - Gilad Shainer [shainer <at> mellanox.com]
  - NVIDIA
    - Duncan Poole [dpoole <at> nvidia.com]
    - Rafael Campana [rcampana <at> nvidia.com]

CORAL Tools – Focus Areas & Wish List

- OpenMP 4.x (with target)
  - OMPT support
  - OMPD support
- Analysis of Deep Memory Hierarchies
- Analysis of Intra-node Parallelism
  - unprecedented levels of threading and concurrency
- Advanced methods for utilizing hardware counters to produce performance insights

- ***Join us and help lead the way!!***

CORAL Tools – Feedback Welcome

- Questions?

- Suggestions!

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.

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