CORAL Procurement Benchmarks

CORAL Vendor Meeting

May 31, 2013

Updated September 24, 2013

Argonne NATIONAL LABORATORY

Lawrence Livermore National Laboratory

OAK RIDGE National Laboratory

U.S. DEPARTMENT OF ENERGY

LLNL-PRE-637694
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

CORAL

Contents

- Benchmark Categories
- Marquee and Elective Benchmarks
- Benchmark Procedures
- Code Modifications

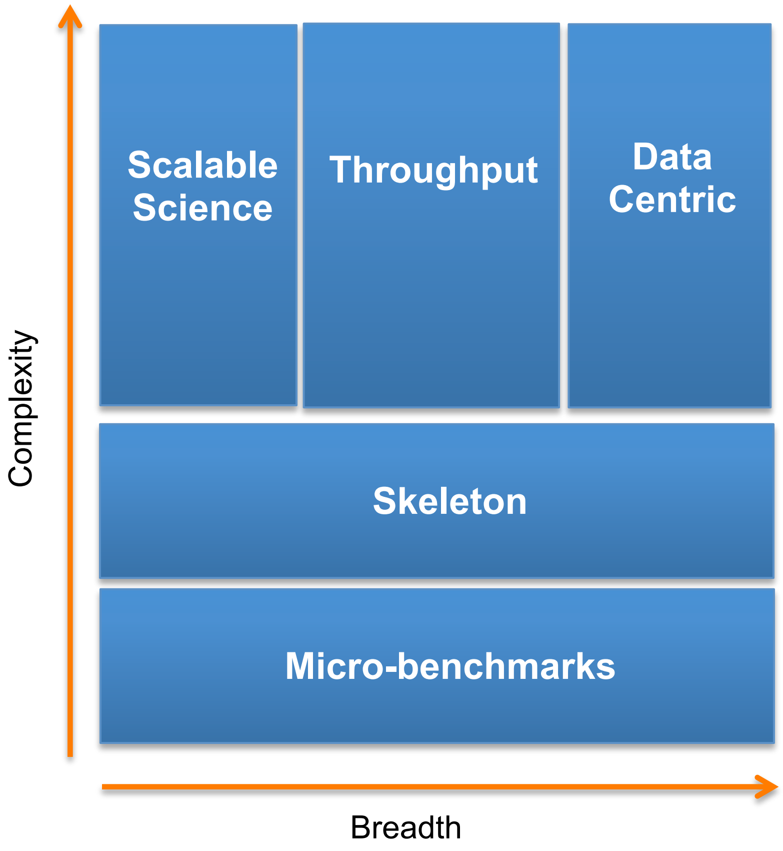
The Performance Reference Baselines are the DOE Production Systems: Sequoia, Mira, and Titan

- Single reference Figure of Merit (FOM) for each Scalable Science and Throughput app
  - Best of breed FOM of the three systems
- Target **4-8X** performance improvement for full system science runs
- Target **6-12X** performance improvement for large ensemble/throughput simulations
- CORAL systems may require disruptive application changes
- CORAL seeks to minimize the changes that are not part of standard programming models

CORAL Benchmark Categories Represent DOE Workloads and Technical Requirements

- **Scalable Science Benchmarks**
  - Expected to run at full scale of the CORAL systems
- **Throughput Benchmarks**
  - Represent large ensemble runs; may be subsets of full applications
- **Data Centric Benchmarks**
  - Represent emerging data intensive workloads
  - Integer operations, instruction throughput, indirect addressing
- **Skeleton Benchmarks**
  - Investigate various platform characteristics including network performance, threading overheads, I/O, memory, memory hierarchies, system software, and programming models
- **Micro Benchmarks**
  - Small code fragments that represent expensive compute portions of some of the scalable science and throughput applications
  - Useful for testing programming methods and performance at the node level & for emulators and simulators

Benchmark Categories



CORAL Benchmarks code and procedures are available at https://asc.llnl.gov/CORAL-benchmarks

CORAL Benchmarking Suite Uses Mini-Apps and a Few Larger Applications

	Scalable		Data	
Categories	Science	Throughput	Centric	Skeleton
Marquee (TR-1)	LSMS QBOX NEKbone HACC	CAM-SE UMT2013 AMG2013 MCB	Graph500 Int sort Hashing	CLOMP IOR CORAL MPI Memory CORAL loops
Elective (TR-2)		QMCPACK NAMD LULESH SNAP miniFE	SPECint_peak2006	Pynamic HACC I/O FTQ XSBench miniMADNESS
Elective Micro-Benchmarks (TR-3)	NEKbonemk HACCmk	UMTmk AMDmk MILCmk GFMCmk		

CORAL Benchmark Platform Stress Areas

Codes	LSMS	QBOX	NEKbone	HACC	CAM-SE	UMT 2013	AMG 2013	MCB
Floating point intensive	X	X	X	X		X		
SIMD/Vectorization			X	X				
Integer/Branch				X				X
Memory Bandwidth		X				X		
Regular (strided) memory access			X	X	X			
Irregular memory access				X			X	X
Large memory footprint						X		
Non-local P2P communication	X		X	X	X			
Small messages			X	X	X			X
Large messages					X	X		
Collective communication		X	X					
Bi-sec bandwidth		X						
Fine grained threading				X			X	

Science Domains and Algorithms Covered by CORAL Marquee Benchmarks

Codes	ASC	Engineering	Astro-physics	Chemistry	Climate	Fusion	Material Science
LSMS				X			X
QBOX	X			X			X
NEKbone						X	
HACC			X				
CAM-SE					X		
UMT2013	X		X			X	
AMG2013	X	X	X			X	
MCB	X		X			X	

Codes	Transport	FFTs	Dense Linear Algebra	Sparse Linear Algebra	Particles	Monte Carlo	Struct. Grids	Unstruct. Grids
LSMS			X			X		
QBOX		X			X			
NEKbone			X				X	
HACC		X			X			
CAM-SE		X	X	X			X	
UMT2013	X							X
AMG2013	X			X				
MCB	X				X	X		

CORAL system performance targets will be projected for both scalable science and throughput

Marquee Scalable Science Benchmarks
(each run/projected at full machine scale)

Qbox
LSMS
NEKbone
HACC

$$S_i = \text{projected } FOM_i \,/\, \text{baseline } FOM_i$$

Marquee Throughput Benchmarks
(Allow M copies of each benchmark to fill machine)

UMT2013	AMG2013	MCB	CAM-SE
UMT2013	AMG2013	MCB	CAM-SE
UMT2013	AMG2013	MCB	CAM-SE
UMT2013	AMG2013	MCB	CAM-SE
UMT2013	AMG2013	MCB	CAM-SE
UMT2013	AMG2013	MCB	CAM-SE

$$S_{scalable} = \left(\prod_{i=1}^{N_S} S_i\right)^{1/N_S}$$

$$S_{throughput} = \left(\frac{N_{TP} * M}{24}\right)\left(\prod_{i=1}^{N_{TP}} S_i\right)^{\frac{1}{N_{TP}}}$$

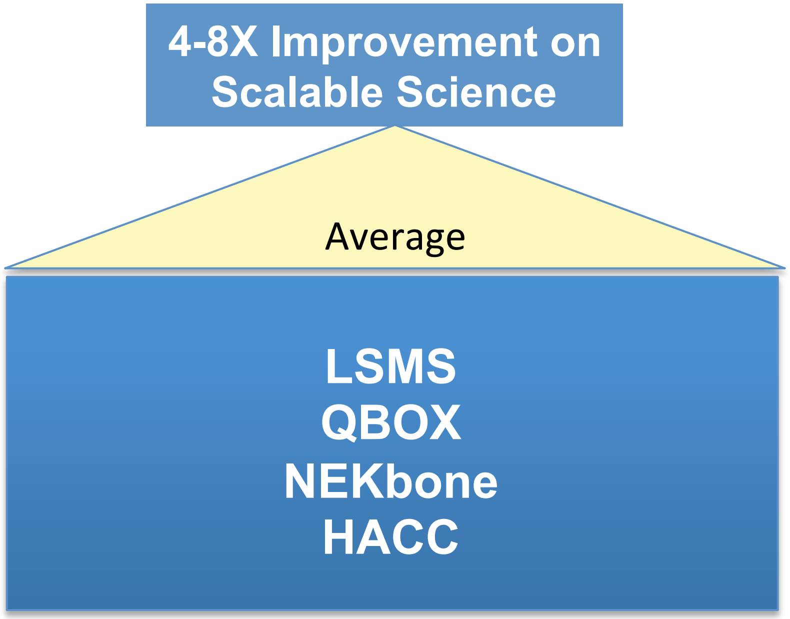
CORAL will provide:

- Only one FOM per benchmark; best obtained across reference platforms
- Throughput apps: Baseline FOM is for job running on 1/24 current systems
- Science apps: Baseline FOM is for full-system job
- All Marquee benchmarks are of equal importance
- Reference FOM's are subject to change between now and final RFP release

Offeror asked to:

- Estimate Performance, projected FOM_{App} for each Marquee Science & Throughput benchmark
- See Technical requirements for more details
- Provide raw results for Marquee Skeleton benchmarks

CORAL Sustained Performance Targets for Scalable Science and Throughput Codes Address Key Application/Workload Requirements



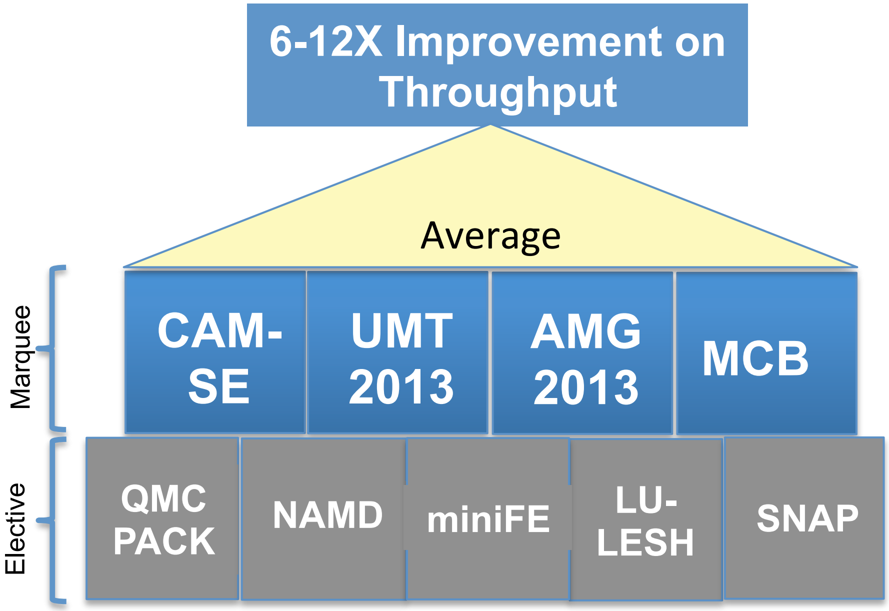
4-8X Improvement on Scalable Science

Average

LSMS
QBOX
NEKbone
HACC

Each run/projected at full machine scale (at least 90% of machine)

Recommend running problem at least 4x larger than the baseline



6-12X Improvement on Throughput

Average

Marquee

CAM-SE

UMT 2013

AMG 2013

MCB

Elective

QMC PACK

NAMD

miniFE

LU-LESH

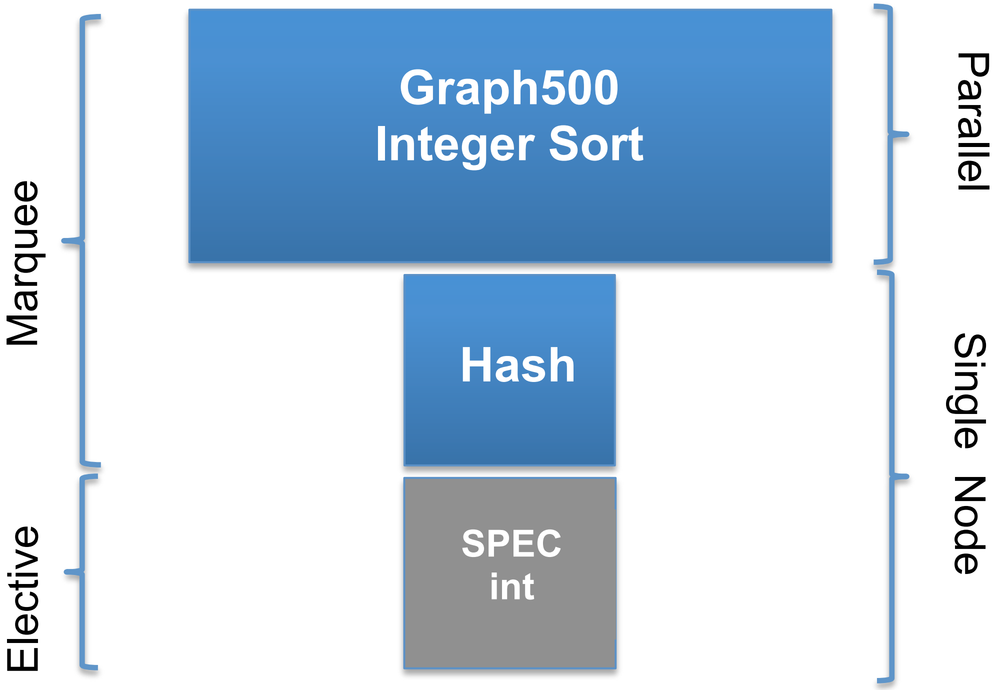
SNAP

Allow 24-144 total instances

N copies of each benchmark

Recommend running problems 2x bigger than baseline (with at least 3x faster turnaround time)

CORAL Addresses Emerging Data Centric Workloads



- ❑Both full machine and single node benchmarks
- ❑Parallel runs on 80-100% of target platform

- ❑ Stressed features
  - ❑ integer operations,
  - ❑ instruction throughput,
  - ❑ indirect addressing

- ❑ Exercised capabilities
  - ❑ interconnect
  - ❑ entire memory hierarchy
  - ❑ irregular access patterns

The CORAL Micro-Benchmark Suite

Micro-Benchmarks	TR-x	LOC	Owner	OMP/threads
NEKbonemk	3	2000	ANL	
HACCmk	3	250	ANL	X
UMTmk	3	700	LLNL	
AMGmk	3	3200	LLNL	X
MILCmk	3	5000	ANL	X
GFMCmk	3	150	ANL	X

- Small code fragments for node level tests
- Ideal for early evaluations and explorations on hardware emulators and simulators

Allowed Code Modification to CORAL Benchmarks

- **Benchmarks may be modified as necessary to get them to compile and run**
  - – Portability changes for programming models are allowed
- **A full set of benchmark runs must be reported with this "baseline" source code**
  - – Can include non-intrusive and/or portable optimizations
    - E.g. compiler flags and standard pragma-style guidance
  - – Can include anticipated changes to system software
    - E.g. MPI and OpenMP runtime improvements
- Must allocate at least 1 GB per MPI task and use threading within each task if necessary to utilize all compute resources
  - – Requirement tied directly to current CORAL codes and production usage

Allowed Code Modification to CORAL Benchmarks Cont'd

- **Offeror may also report optimized results**
  - Any and all code modifications are allowed
  - However, wholesale algorithmic changes that are strongly architecture specific have less value
  - All benchmark code modification will be documented and provided to CORAL
- **CORAL and Offeror will continue to improve the efficiency and scalability of all benchmarks between award of the contracts and delivery of the systems**
  - Emphasis on higher level optimizations as well as compiler optimization technology improvements while maintaining readable and maintainable code

CORAL Benchmark Website

- **CORAL Benchmark website contains additional information not found in the technical requirements document**
  - Benchmark summary files
  - Procedures for running CORAL benchmarks
  - Input parameters
  - Benchmark reference figures of merit (FOM)
  - Benchmark scaling data on CORAL reference systems
  - CORAL Benchmark Spreadsheet for reporting results
- **Any updates or answers to questions will be posted on the website**

https://asc.llnl.gov/CORAL-benchmarks

Please send benchmark questions to: Coral-apps@lists.llnl.gov

To receive CORAL announcements:

Subscribe to mail list by sending a message of the form:

subscribe coral-announce

by email to majordomo@lists.llnl.gov