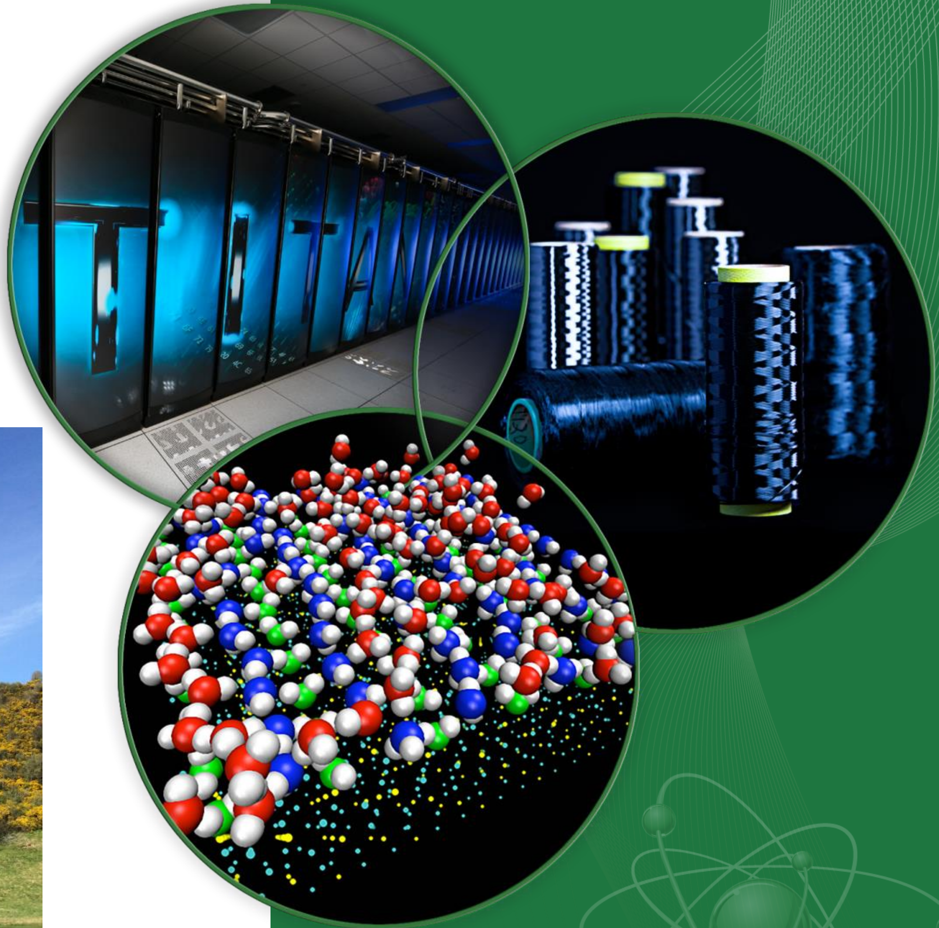
Exploring Emerging Technologies in the Extreme Scale HPC Co-Design Space with Holistic Performance Modeling

Jeffrey S. Vetter

Jeremy Meredith



Presented to
Exascale Applications and Software Conference (EASC)

EPCC/University of Edinburgh

22 Apr 2015

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Georgia Tech College of Computing
Computational Science and Engineering

http://ft.ornl.gov ♦ vetter@computer.org

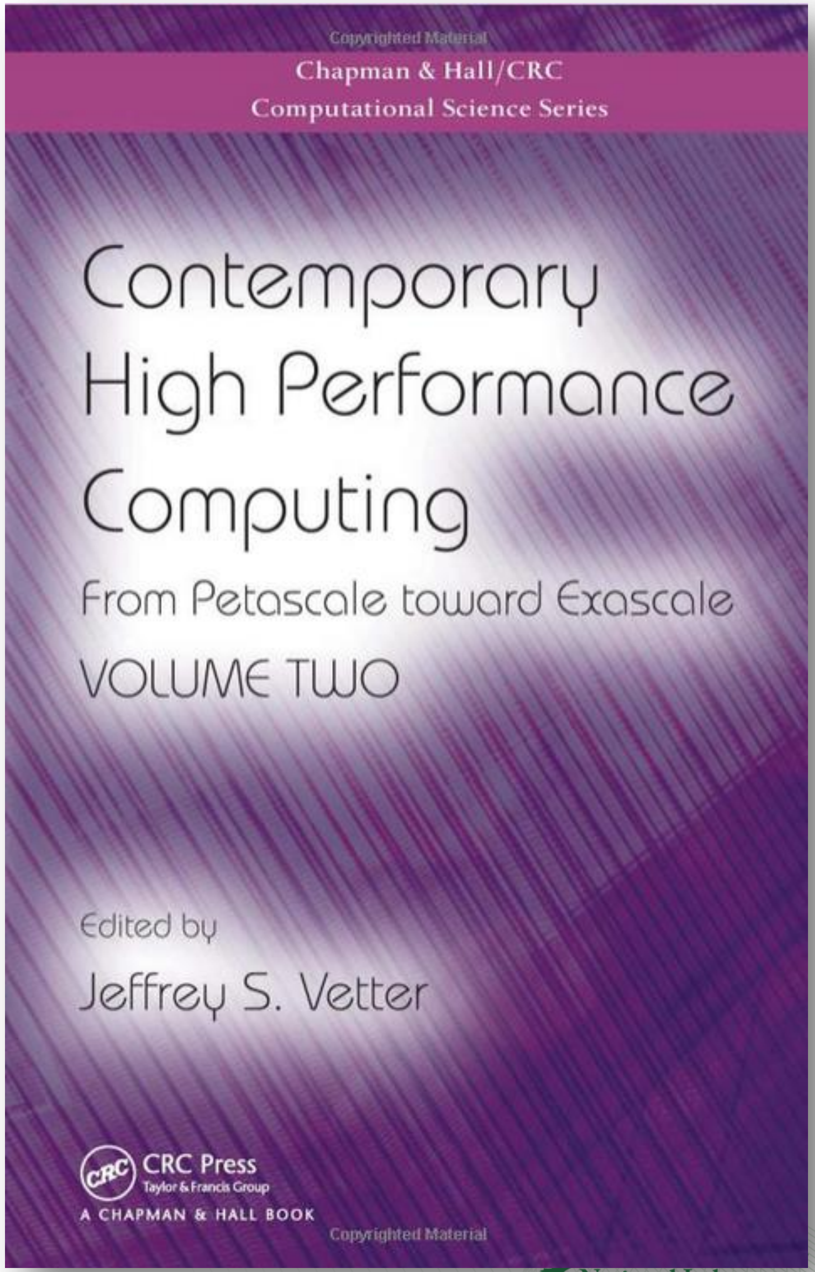
ORNL is managed by UT-Battelle
for the US Department of Energy

OAK RIDGE National Laboratory


*****WHISKY LIST*****
WHISKY REGIONS & TYPE OF WHISKY
L – Lowland
C – Campbell Town
H – Highlands
IS – Islands
IN – India
IR – Irish
I – Islay
J – Japanese
S – Speyside
B – Blend
NZ – New Zealand
W – Welsh
WHISKI BAR · RESTAURANT
LOC
*** MALTS ***
Age
ABV
£

ARCHER in new book

- Vol 2:
  - EPCC - ARCHER
  - NERSC
  - NREL
  - NCAR
  - ZIB
  - RIKEN
  - KTH Royal Institute of Technology



http://j.mp/chpc2015

Overview

- Our community has major challenges in HPC as we move to extreme scale
  - Power, Performance, Resilience, Productivity
  - Major shifts in architectures, software, applications
    - Not just HPC: Most uncertainty in two decades
- New technologies emerging to address some of these challenges
  - Heterogeneous computing
  - Nonvolatile memory
- Consequently, we now have critical situations in
  - Portable programming models
  - Performance prediction for procurement, optimization, etc
- Aspen is a tool we have developed for performance prediction


Surveying the HPC Landscape:
Today and Tomorrow

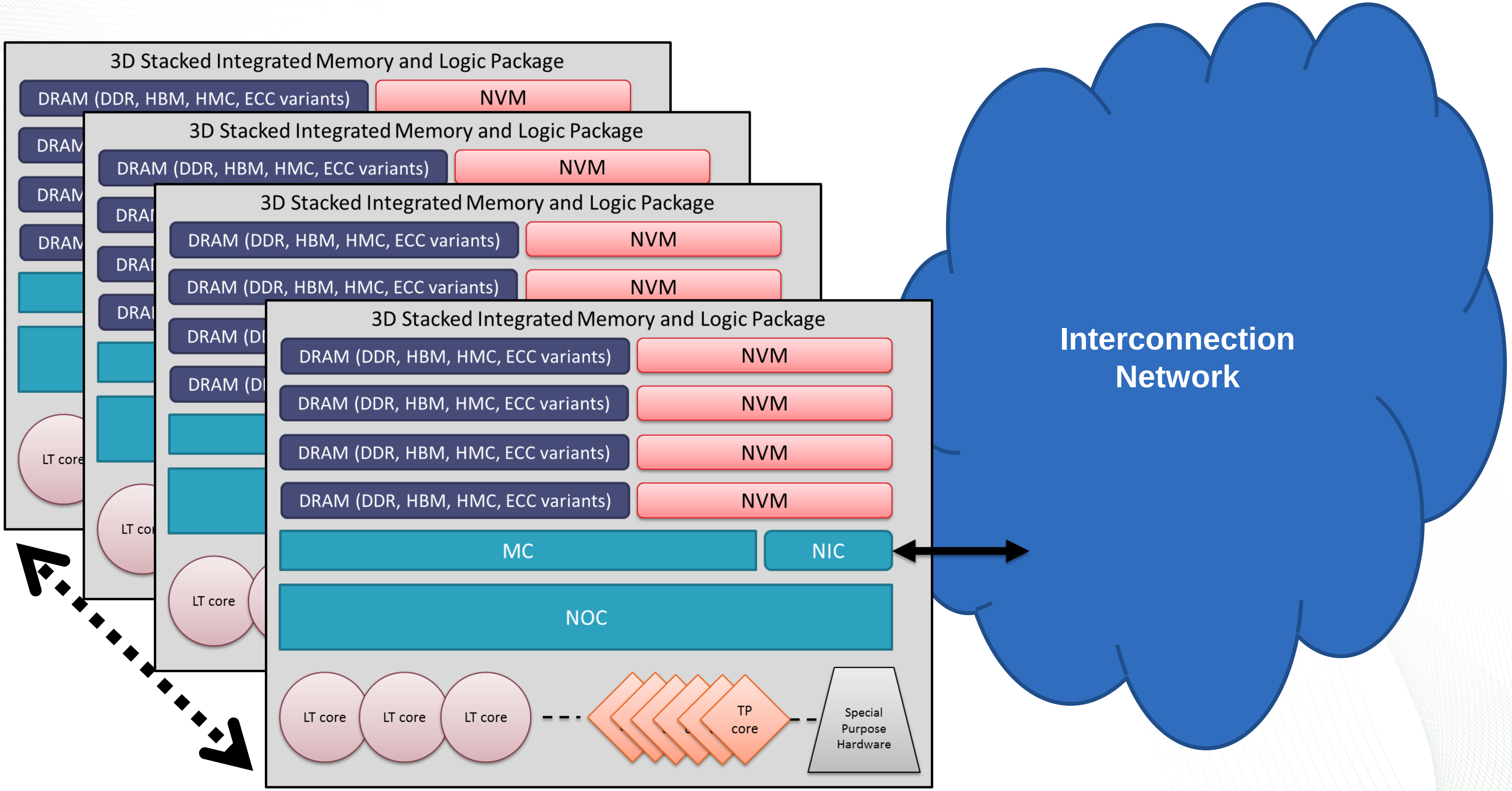
Notional Exascale Architecture Targets
(From Exascale Arch Report 2009)

System attributes	2001	2010	"2015"		"2018"	
System peak	10 Tera	2 Peta	200 Petaflop/sec		1 Exaflop/sec	
Power	~0.8 MW	6 MW	15 MW		20 MW	
System memory	0.006 PB	0.3 PB	5 PB		32-64 PB	
Node performance	0.024 TF	0.125 TF	0.5 TF	7 TF	1 TF	10 TF
Node memory BW		25 GB/s	0.1 TB/sec	1 TB/sec	0.4 TB/sec	4 TB/sec
Node concurrency	16	12	O(100)	O(1,000)	O(1,000)	O(10,000)
System size (nodes)	416	18,700	50,000	5,000	1,000,000	100,000
Total Node Interconnect BW		1.5 GB/s	150 GB/sec	1 TB/sec	250 GB/sec	2 TB/sec
MTTI		day	O(1 day)		O(1 day)	

Parallel I/O ??

http://science.energy.gov/ascr/news-and-resources/workshops-and-conferences/grand-challenges/

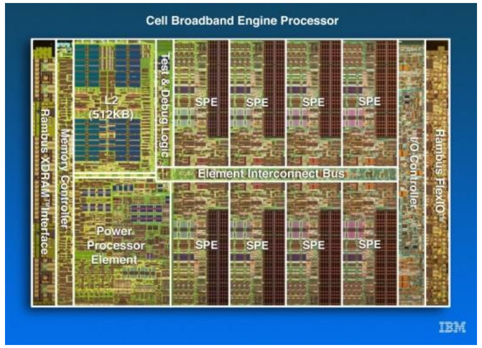
Notional Future Architecture

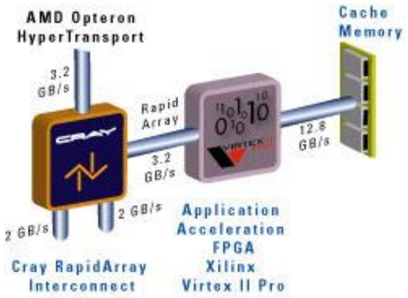


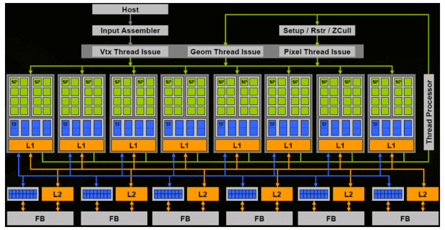
Earlier Experimental Computing Systems

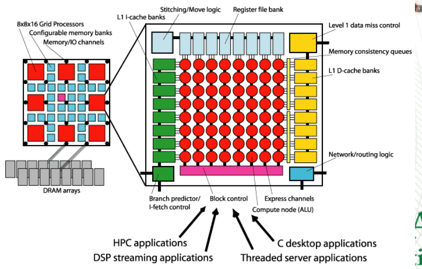
- The past decade has started the trend away from traditional ‘simple’ architectures
- Mainly driven by facilities costs and successful (sometimes heroic) application examples
- Examples
  - Cell, GPUs, FPGAs, SoCs, etc
- Many open questions
  - Understand technology challenges
  - Evaluate and prepare applications
  - Recognize, prepare, enhance programming models

Popular architectures since ~2004



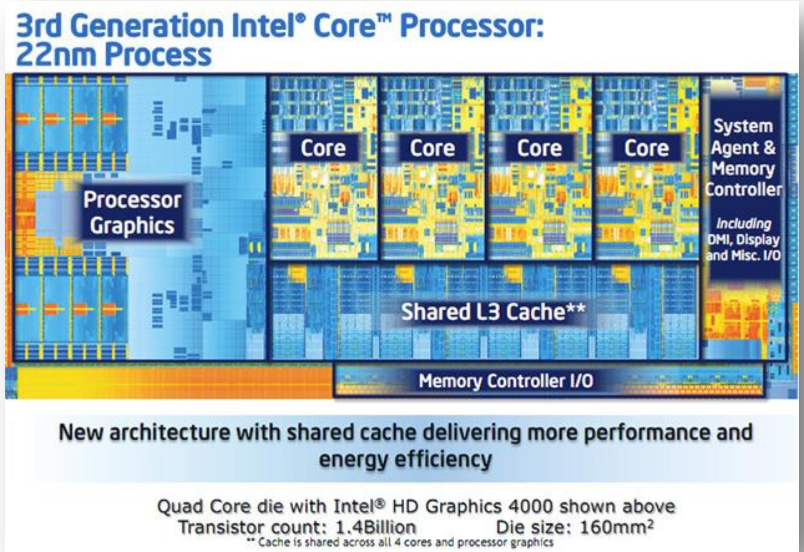


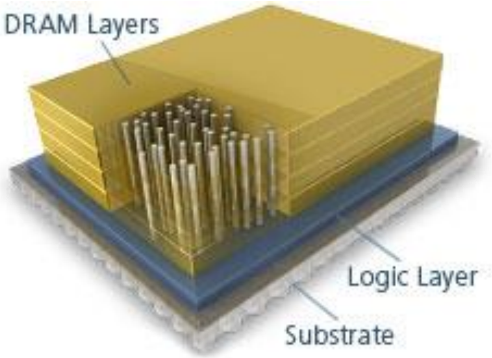




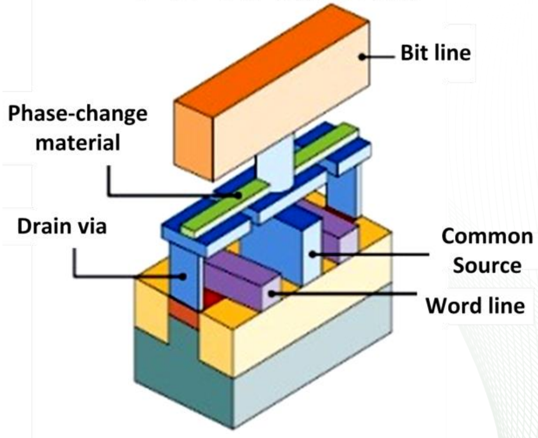
Emerging Computing Architectures – Future

- Heterogeneous processing
  - Latency tolerant cores
  - Throughput cores
  - Special purpose hardware (e.g., AES, MPEG, RND)
  - Fused, configurable memory
- Memory
  - 2.5D and 3D Stacking
  - HMC, HBM, WIDEIO2, LPDDR4, etc
  - New devices (PCRAM, ReRAM)
- Interconnects
  - Collective offload
  - Scalable topologies
- Storage
  - Active storage
  - Non-traditional storage architectures (key-value stores)
- Improving performance and programmability in face of increasing complexity
  - Power, resilience





PC-RAM Cell



HPC (mobile, enterprise, embedded) computer design is more fluid now than in the past two decades.

Recent announcements

Nvidia and IBM create GPU interconnect for faster supercomputing

"NVLink" shares up to 80GB of data per second between CPUs and GPUs.

by Jon Brodkin - Ma

Nvidia CEO Jen-H

Nvidia

Nvidia and IBM h
units, letting GP
The fatter pipe v
compared to 16

It Begins: AMD Announces Its First ARM Based Server SoC, 64-bit/8-core Opteron A1100

123 Comments

by Anand Lal Shimpi on January 28, 2014 6:35 PM EST

+ Add Comme

Posted in CPUs IT Computing Enterprise enterprise CPUs AMD Opteron Opteron A1100 ARM

"SEATTLE" 64-BIT ARM SERVER PROCESSOR

FIRST 28NM AR

Indus
prove
- The
- Serv
CPU
- 2-4x
impr
- 8 co
- Base
- Exte
CPU
- Serv
- Lega
- Stora
SAMPLI

Around 15 mon
2014. Less thar
A1100: a 64-bit

The Opteron A1
talking about ha
away entirely, b
bets going on. E
process at Glob

Nvidia Jetson TK1 mini supercomputer is up for pre-order

Will ship on 15 May

By Lee Bell

Fri May 02 2014, 11:

computers attempt
self-driving cars.

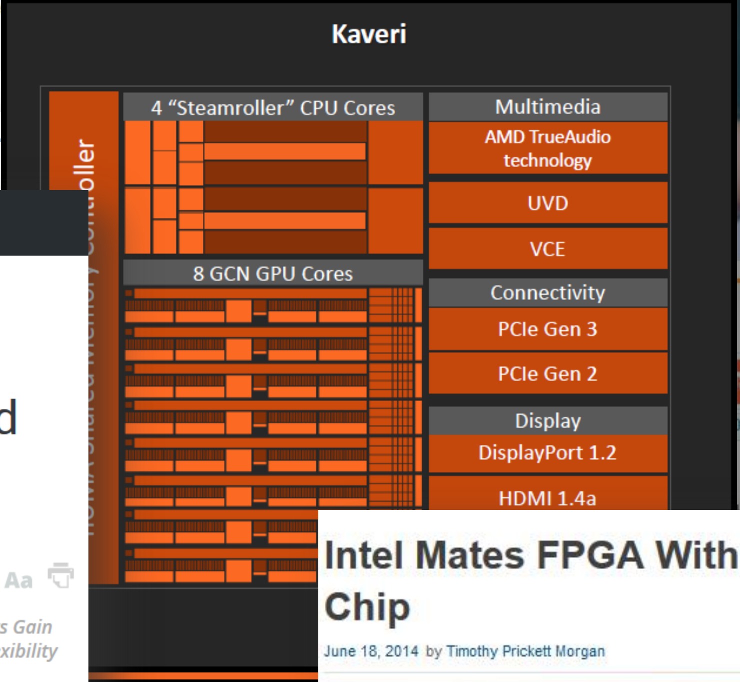
Speaking at the GP
Hsun Huang descri
can run, but at a sl

With a total perforr
Raspberry Pi board
in the US - a numb
launched at CES in

"The Jetson TK1 als
comes with a whole
it could be classifyi

Parameters are loa
recognises objects,

A-SERIES REDEFINES COMPUTE



lying AMD's "Kaveri" APU for Heter

MarketWatch

PRESS RELEASE

Altera and IBM Unveil FPGA-accelerated POWER Systems with Coherent Shared Memory

By

Published: Nov 17, 2014 8:00 a.m. ET

f 8 | 13 | Aa

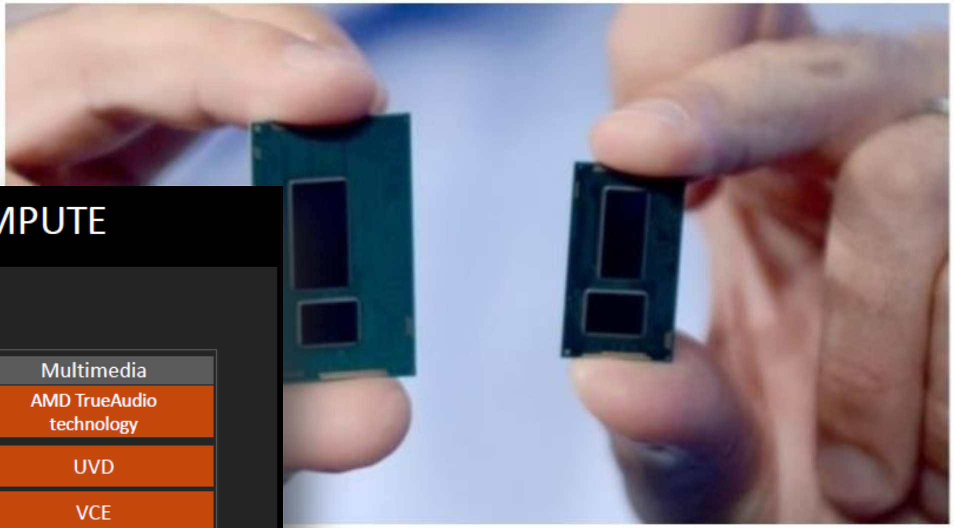
POWER8 Systems that Leverage Reprogrammable FPGA Accelerators Gain Significant Improvements in System Performance, Efficiency and Flexibility

NEW ORLEANS, Nov. 17, 2014 /PRNewswire/ -- **SuperComputing 2014** -- Altera Corporation ALTR, +0.00% and IBM IBM, +0.00% today unveiled the industry's first FPGA-based acceleration platform that coherently connects an FPGA to a POWER8 CPU leveraging IBM's Coherent Accelerator Processor Interface (CAPI). The reconfigurable hardware accelerator features shared virtual memory between the FPGA and processor which significantly improves system performance, efficiency and flexibility in high-performance computing (HPC) and data center applications. Altera and IBM are presenting several POWER8 systems that are coherently accelerated using FPGAs at SuperComputing 2014.

Working together through the OpenPOWER Foundation, Altera and IBM are

Intel's 14nm Broadwell GPU takes shape, indicates major improvements over Haswell

By Sebastian Anthony on November 5, 2013 at 10:21 am | 16 Comments



18 | 2 reddit

Ahead of its 2014 launch, Intel has started open-sourcing the Linux driver for Broadwell's GPU. Broadwell is the 14nm die shrink of Intel's microarchitecture, and while the CPU side of things isn't expected to change much, Broadwell's GPU looks ... es (Iris) GPU ... a modified ... GPU, and the

Intel Mates FPGA With Future Xeon Server Chip

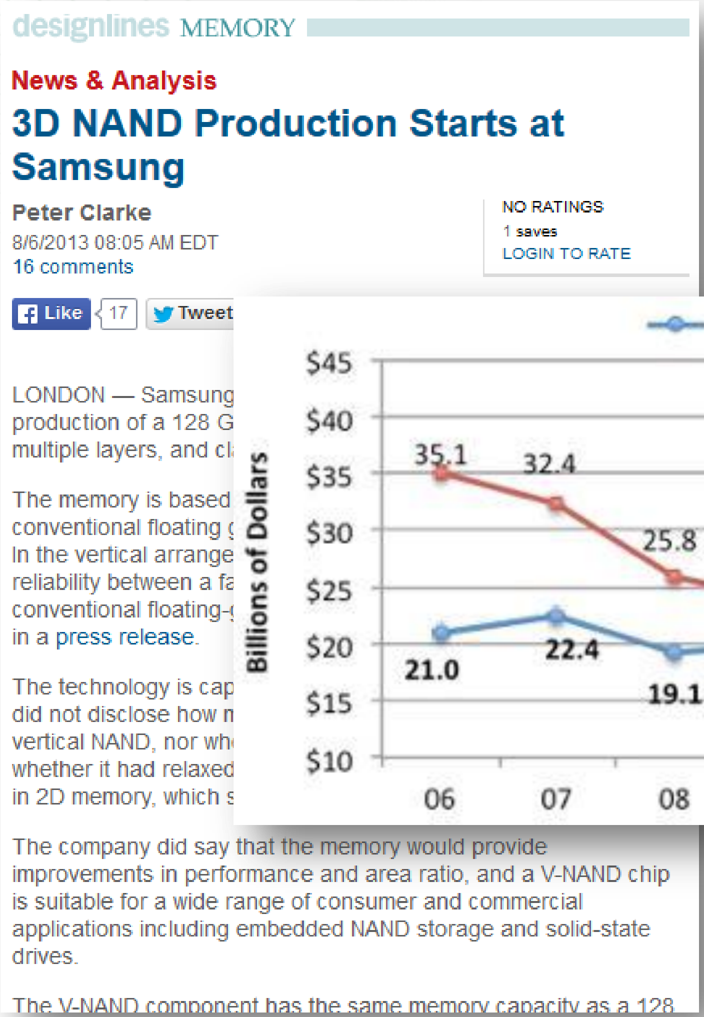
June 18, 2014 by Timothy Prickett Morgan

intel inside

Intel is taking field programmable gate arrays seriously as a means of accelerating applications and has crafted a hybrid chip that marries an FPGA to a Xeon E5 processor and puts them in the same processor socket.

Oak Ridge National Laboratory

NVRAM Technology Continues to Improve – Driven by Market Forces


designlines MEMORY

News & Analysis

3D NAND Production Starts at Samsung

Peter Clarke
8/6/2013 08:05 AM EDT
16 comments

NO RATINGS
1 saves
LOGIN TO RATE

Like 17 Tweet

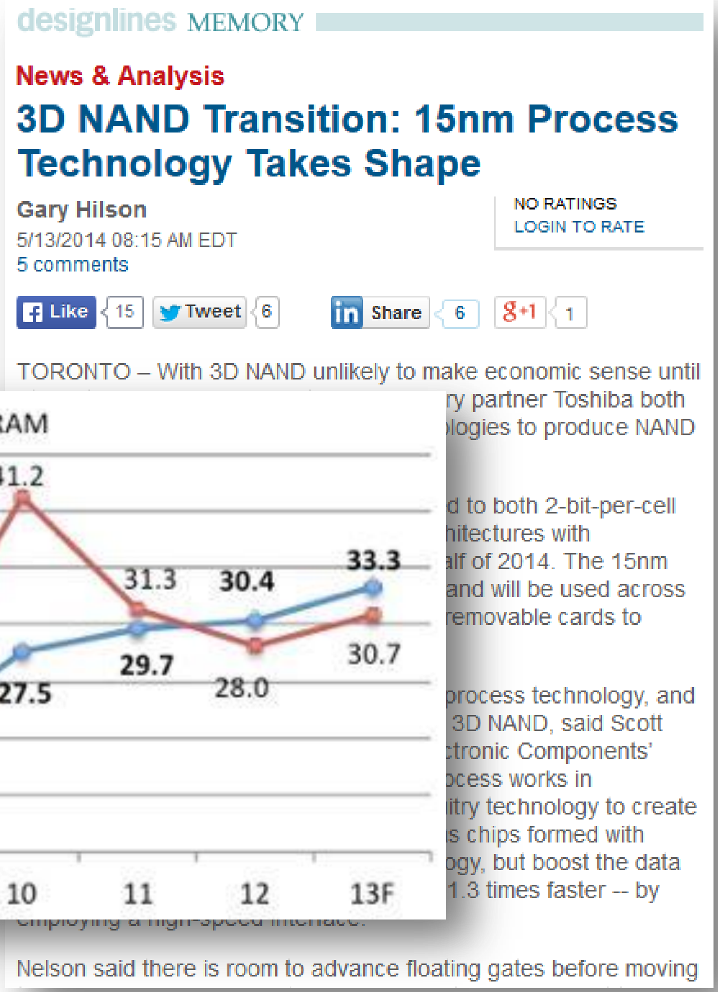
LONDON — Samsung
production of a 128 G
multiple layers, and cl

The memory is based
conventional floating g
In the vertical arrange
reliability between a fa
conventional floating-g
in a press release.

The technology is cap
did not disclose how m
vertical NAND, nor wh
whether it had relaxed
in 2D memory, which s

The company did say that the memory would provide improvements in performance and area ratio, and a V-NAND chip is suitable for a wide range of consumer and commercial applications including embedded NAND storage and solid-state drives.

The V-NAND component has the same memory capacity as a 128


designlines MEMORY

News & Analysis

3D NAND Transition: 15nm Process Technology Takes Shape

Gary Hilson
5/13/2014 08:15 AM EDT
5 comments

NO RATINGS
LOGIN TO RATE

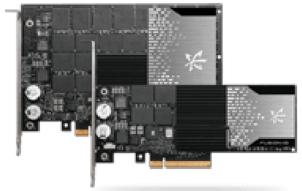
Like 15 Tweet 6 Share 6 g+1 1

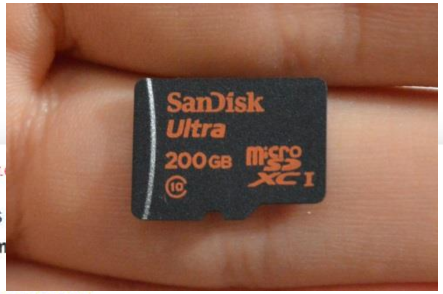
TORONTO – With 3D NAND unlikely to make economic sense until
ry partner Toshiba both
ologies to produce NAND

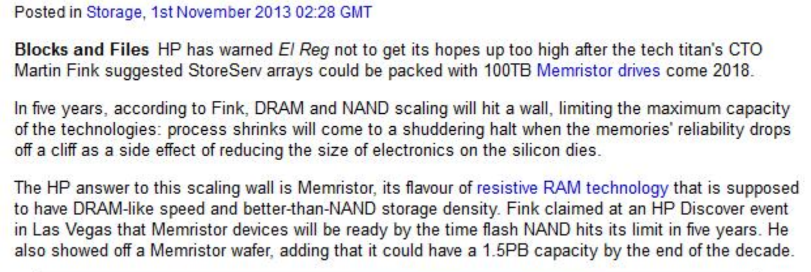
d to both 2-bit-per-cell
hitectures with
alf of 2014. The 15nm
and will be used across
removable cards to

process technology, and
3D NAND, said Scott
tronic Components'
ocess works in
itry technology to create
s chips formed with
ogy, but boost the data
1.3 times faster -- by
employing a high-speed interface.

Nelson said there is room to advance floating gates before moving






Original URL: http://www.theregister.

HP 100TB Memristor drives

Universal memory slow in com

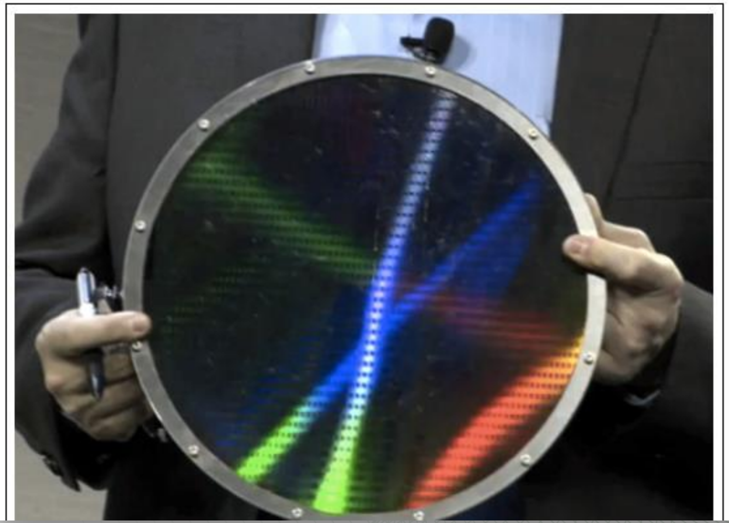
By Chris Mellor

Posted in Storage, 1st November 2013 02:28 GMT

Blocks and Files HP has warned *El Reg* not to get its hopes up too high after the tech titan's CTO Martin Fink suggested StoreServ arrays could be packed with 100TB Memristor drives come 2018.

In five years, according to Fink, DRAM and NAND scaling will hit a wall, limiting the maximum capacity of the technologies: process shrinks will come to a shuddering halt when the memories' reliability drops off a cliff as a side effect of reducing the size of electronics on the silicon dies.

The HP answer to this scaling wall is Memristor, its flavour of resistive RAM technology that is supposed to have DRAM-like speed and better-than-NAND storage density. Fink claimed at an HP Discover event in Las Vegas that Memristor devices will be ready by the time flash NAND hits its limit in five years. He also showed off a Memristor wafer, adding that it could have a 1.5PB capacity by the end of the decade.



http://www.eetasia.com/STATIC/ARTICLE_IMAGES/201212/EEOL_2012DEC28_STOR_MFG_NT_01.jpg

Comparison of emerging memory technologies

	SRAM	DRAM	eDRAM	2D NAND Flash	3D NAND Flash	PCRAM	STTRAM	2D ReRAM	3D ReRAM
Data Retention	N	N	N	Y	Y	Y	Y	Y	Y
Cell Size (F^2)	50-200	4-6	19-26	2-5	<1	4-10	8-40	4	<1
Minimum F demonstrated (nm)	14	25	22	16	64	20	28	27	24
Read Time (ns)	< 1	30	5	10^4	10^4	10-50	3-10	10-50	10-50
Write Time (ns)	< 1	50	5	10^5	10^5	100-300	3-10	10-50	10-50
Number of Rewrites	10^{16}	10^{16}	10^{16}	10^4-10^5	10^4-10^5	10^8-10^{10}	10^{15}	10^8-10^{12}	10^8-10^{12}
Read Power	Low	Low	Low	High	High	Low	Medium	Medium	Medium
Write Power	Low	Low	Low	High	High	High	Medium	Medium	Medium
Power (other than R/W)	Leakage	Refresh	Refresh	None	None	None	None	Sneak	Sneak
Maturity									

http://ft.ornl.gov/trac/blackcomb

OAK RIDGE
National Laboratory

Thinking back to 2009 projections, where is DOE in 2015?

System attributes	Today		CORAL	
Name	**TITAN**	**MIRA**	**Summit**	**Aurora**
System peak (PF)	27	10	150	180
Peak Power (MW)	9	4.8	10	13
Total system memory	710TB	768TB	2 PB DDR4 + HBM + 2.7 PB persistent memory	>7 PB High Bandwidth On-Package Memory, local Memory and Persistent Memory
Node performance (TF)	1.452	0.204	> 40	> 17 times Mira
Node processors	AMD Opteron Nvidia Kepler	64-bit PowerPC A2	Multiple IBM Power9 CPUs & multiple Nvidia Voltas GPUS	Intel Xeon Phi processors (codenamed Knights Hill)
System size (nodes)	18,688 nodes	49,152	>3,400 nodes	>50,000 nodes
System Interconnect	Gemini	5D Torus	Dual Rail EDR-IB	2nd generation Intel Omni-Path Architecture
File System	32 PB 1 TB/s, Lustre®	26 PB 300 GB/s GPFS™	120 PB 1 TB/s GPFS™	150 PB >1 TB/s Lustre®

Some ratios will be challenging to mitigate

System attributes	2001	2010	2014	"2015"		est 2018	Ratio of Summit to Titan	"2018"	
Name	Seaborg3	Jaguar	Titan			SUMMIT			
System peak	10 Tera	2 Peta	27	200		136	5.04	1 Exaflop/sec	
Power (MW)	0.8	6	9	15		10	1.11	20	
Node main memory (GB)			38			512	13.47		
System memory (PB)	0.006	0.3	0.7106	5		1.7408	2.45	32-64	
Node Persistent Memory (GB)						800			
System Persistent Memory (PB)						2.72	∞		
Node performance (TF)	0.024	0.125	1.4	0.5	7	40	28.57	1	10
Node memory BW		25 GB/s		0.1 TB/sec	1 TB/sec			0.4 TB/sec	4 TB/sec
Node concurrency	16	12		O(100)	O(1,000)	*POWER9s + *VOLTAs		O(1,000)	O(10,000)
System size (nodes)	416	18700	18700	50000	5000	3400	0.18	1000000	100000
Total Node Interconnect BW		1.5 GB/s		150 GB/sec	1 TB/sec			250 GB/sec	2 TB/sec
injection bandwidth per node (GB/s)			6.4			23	3.59		
File system capacity (PB)			32			120	3.75		
File system bandwidth (TB/s)			1			1	1.00		
MTTI		day		O(1 day)				O(1 day)	

Observations about these trends

- Aside from all the interesting technical questions for computer scientists…


DANGER
HALT!
DEADLY FORCE
IS AUTHORIZED
BEYOND THIS POINT
10 Code of Federal Regulations 1047 and Department of Energy Manual 470.4-3

Observations about these trends (2)

1. For the success of HPC, we need to be very careful at this point
2. Complexity is everyone’s enemy!

3. Performance portable programming models should be mandatory on all current and future architectures
   1. Increasingly, apps teams are spending time porting to new architectures rather than doing science
4. Performance prediction techniques and tools are critical
   1. Previously, a poor (procurement, optimization, facility) decision could cost 30%; now it could be 10x!
5. And then there is power consumption, reliability, etc

Holistic Performance Modeling for Extreme-Scale HPC

Prediction Techniques Ranked

	Speed	Ease	Flexibility	Accuracy	Scalability
Ad-hoc Analytical Models	1	3	2	4	1
Structured Analytical Models	1	2	1	4	1
Simulation – Functional	3	2	2	3	3
Simulation – Cycle Accurate	4	2	2	2	4
Hardware Emulation (FPGA)	3	3	3	2	3
Similar hardware measurement	2	1	4	2	2
Node Prototype	2	1	4	1	4
Prototype at Scale	2	1	4	1	2
Final System	-	-	-	-	-

OAK RIDGE National Laboratory

Prediction Techniques Ranked

	Speed	Ease	Flexibility	Accuracy	Scalability
Ad-hoc Analytical Models	1	3	2	4	1
Structured Analytical Models	1	2	1	4	1
Aspen	1	1	1	4	1
Simulation – Functional	3	2	2	3	3
Simulation – Cycle Accurate	4	2	2	2	4
Hardware Emulation (FPGA)	3	3	3	2	3
Similar hardware measurement	2	1	4	2	2
Node Prototype	2	1	4	1	4
Prototype at Scale	2	1	4	1	2
Final System	-	-	-	-	-

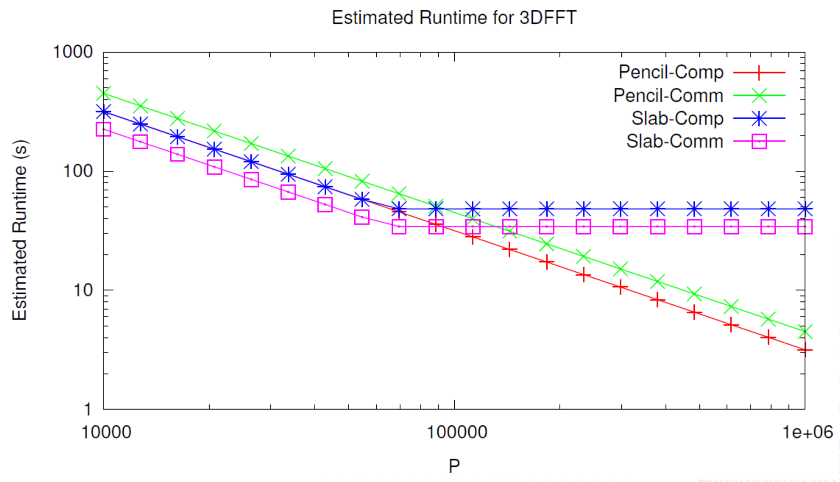
OAK RIDGE National Laboratory

Aspen – Design Goals

- Abstract Scalable Performance Engineering Notation
  - Create a deployable, extensible, and highly semantic representation for analytical performance models
  - Design and implement a new language for analytical performance modeling
  - Use the language to create machine-independent models for important applications and kernels
- Models are composable

```
kernel localFFT {
  exposes parallelism [n^2]
  requires flops [5 * n * log2(n)] as dp,
      complex, simd
  requires loads [a * n * max(1, log(n)/
      log(Z)) * wordSize] from fftVolume
}
```

Listing 2. Aspen statements for the local 1D FFTs



K. Spafford and J.S. Vetter, “Aspen: A Domain Specific Language for Performance Modeling,” in *SC12: ACM/IEEE International Conference for High Performance Computing, Networking, Storage, and Analysis, 2012*

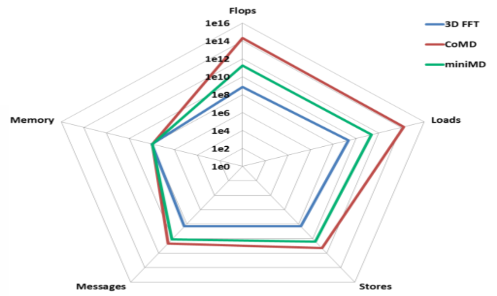
Aspen Design Flow

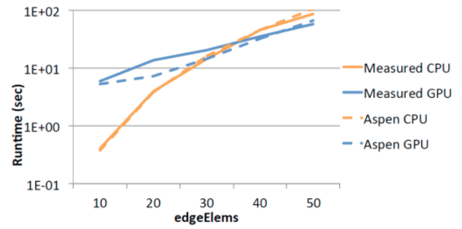
Source code

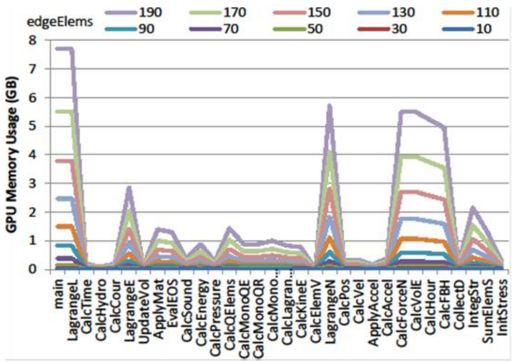
```
static inline
void CalcMonotonicQGradientsForElems(Index_t p_nodelist[T_NUMELEM8],
        Real_t p_x[T_NUMNODE], Real_t p_y[T_NUMNODE], Real_t p_z[T_NUMNODE],
        Real_t p_xd[T_NUMNODE], Real_t p_yd[T_NUMNODE],Real_t p_zd[T_NUMNODE],
        Real_t p_volo[T_NUMELEM], Real_t p_vnew[T_NUMELEM],
        Real_t p_delx_zeta[T_NUMELEM], Real_t p_delv_zeta[T_NUMELEM],
        Real_t p_delx_xi[T_NUMELEM], Real_t p_delv_xi[T_NUMELEM],
        Real_t p_delx_eta[T_NUMELEM], Real_t p_delv_eta[T_NUMELEM])
{
   Index_t i;
   Index_t numElem = m_numElem;
#pragma acc parallel loop independent present(p_vnew, p_nodelist, p_x, p_y, p_z, p_xd, \
p_yd, p_zd, p_volo, p_delx_xi, p_delx_eta, p_delx_zeta, p_delv_xi, p_delv_eta,\
p_delv_zeta)
   for (i = 0 ; i < numElem ; ++i ) {
      const Real_t ptiny = 1.e-36 ;
      Real_t ax,ay,az ;
      Real_t dxv,dyv,dzv ;

      const Index_t *elemToNode = &p_nodelist[8*i];
      Index_t n0 = elemToNode[0] ;
      Index_t n1 = elemToNode[1] ;
      Index_t n2 = elemToNode[2] ;
      Index_t n3 = elemToNode[3] ;
      Index_t n4 = elemToNode[4] ;
      Index_t n5 = elemToNode[5] ;
      Index_t n6 = elemToNode[6] ;
      Index_t n7 = elemToNode[7] ;

      Real_t x0 = p_x[n0] ;
```

Aspen code

```
kernel CalcMonotonicQGradients {
  execute [numElems]
  {
    loads [8 * indexWordSize] from nodelist
    // Load and cache position and velocity.
    loads/caching [8 * wordSize] from x
    loads/caching [8 * wordSize] from y
    loads/caching [8 * wordSize] from z

    loads/caching [8 * wordSize] from xvel
    loads/caching [8 * wordSize] from yvel
    loads/caching [8 * wordSize] from zvel

    loads [wordSize] from volo
    loads [wordSize] from vnew
    // dx, dy, etc.
    flops [90] as dp, simd
    // delvk delxk
    flops [9 + 8 + 3 + 30 + 5] as dp, simd
    stores [wordSize] to delv_xeta
    // delxi delvi
    flops [9 + 8 + 3 + 30 + 5] as dp, simd
    stores [wordSize] to delx_xi
    // delxj and delvj
    flops [9 + 8 + 3 + 30 + 5] as dp, simd
    stores [wordSize] to delv_eta
  }
}
```







Creation

- Manual for future applications
- Static analysis via compilers
- Historical
- Empirical

Representation in Aspen
- **Modular**
- **Sharable**
- **Composable**
- **Reflects prog structure**

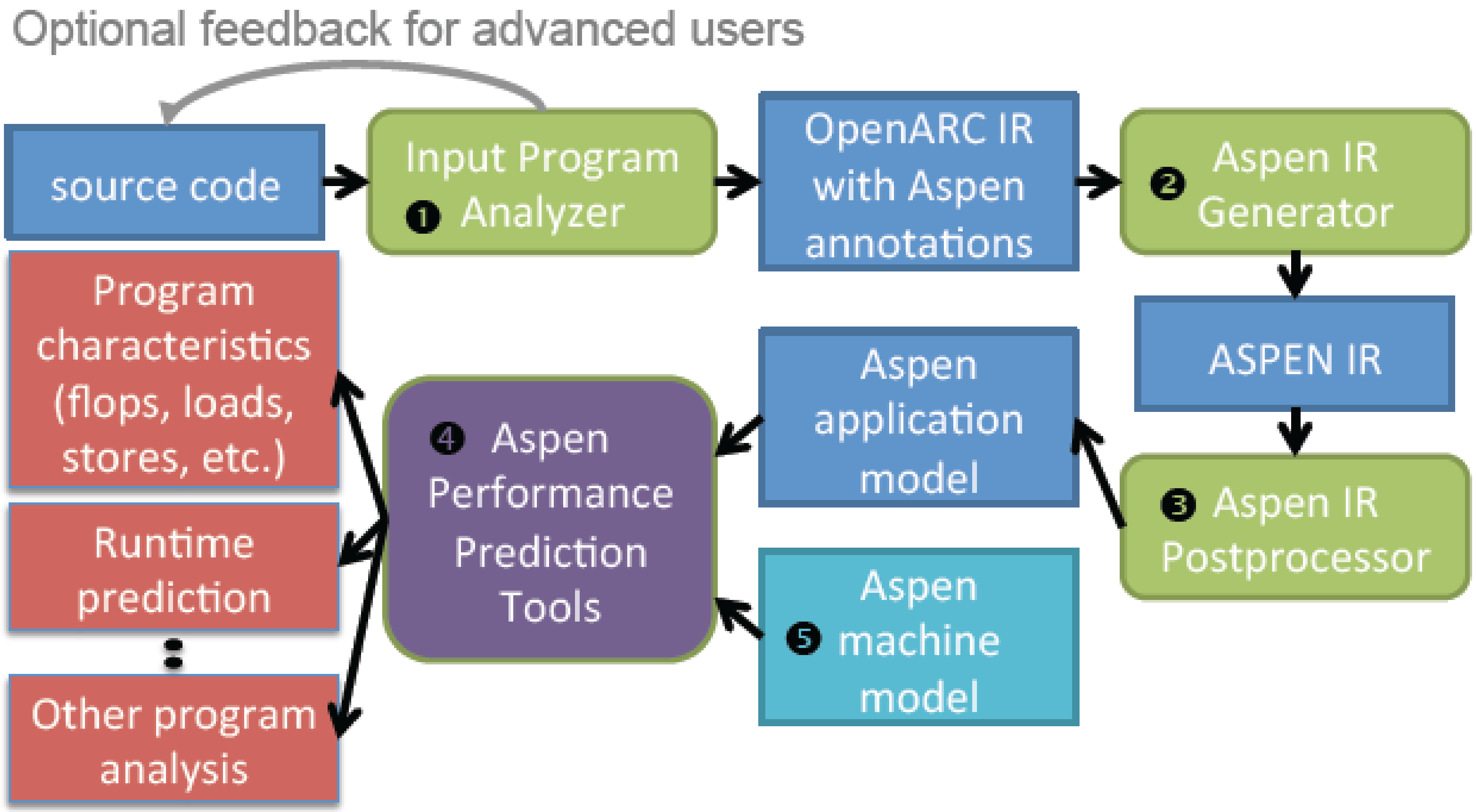
Existing models for MD, UHPC CP 1, Lulesh, 3D FFT, CoMD, VPFFT, …

Use

- Interactive tools for graphs, queries
- Design space optimization
- Drive simulators
- Feedback to runtime systems

K. Spafford and J.S. Vetter, “Aspen: A Domain Specific Language for Performance Modeling,” in *SC12: ACM/IEEE International Conference for High Performance Computing, Networking, Storage, and Analysis, 2012*

Creating Aspen Models



S. Lee, J.S. Meredith, and J.S. Vetter, "COMPASS: A Framework for Automated Performance Modeling and Prediction," in *ACM International Conference on Supercomputing (ICS)*. *Newport Beach, California: ACM, 2015, 10.1145/2751205.2751220.*

Simple MM example generated from COMPASS

```
int N = 1024;
void matmul(float *a, float *b, float *c){ int i, j, k ;
#pragma acc kernels loop gang copyout(a[0:(N*N)]) \
copyin(b[0:(N*N)],c[0:(N*N)])
  for (i=0; i<N; i++){
#pragma acc loop worker
    for (j=0; j<N; j++) { float sum = 0.0 ;
      for (k=0; k<N; k++) {sum+=b[i*N+k]*c[k*N+j];}
      a[i*N+j] = sum; }
  } //end of i loop
} //end of matmul()
int main() {
  int i; float *A = (float*) malloc(N*N*sizeof(float));
  float *B = (float*) malloc(N*N*sizeof(float));
  float *C = (float*) malloc(N*N*sizeof(float));
  for (i = 0; i <  N*N; i++)
  {  A[i] = 0.0F; B[i] = (float) i; C[i] = 1.0F; }
#pragma aspen modelregion label(MM)
  matmul(A,B,C);
  free(A); free(B); free(C); return 0;
} //end of main()
```

```
model MM {
  param floatS = 4; param N = 1024
  data A as Array((N*N), floatS)
  data B as Array((N*N), floatS)
  data C as Array((N*N), floatS)
  kernel matmul {
    execute matmul2_intracommIN
    {  intracomm [floatS*(N*N)] to C as copyin
      intracomm [floatS*(N*N)] to B as copyin }
    map matmul2 [N] {
      map matmul3 [N] {
        iterate [N] {
          execute matmul5
          {  loads [floatS] from B as stride(1)
            loads [floatS] from C; flops [2] as sp, simd }
        } //end of iterate
        execute matmul6 { stores [floatS] to A as stride(1) }
      } // end of map matmul3
    } //end of map matmul2
    execute matmul2_intracommOUT
    { intracomm [floatS*(N*N)] to A as copyout }
  } //end of kernel matmul
  kernel main { matmul() }
} //end of model MM
```

LULESH in Aspen

branch: master ▾ aspen / models / lulesh / **lulesh.aspen**

jsmeredith on Sep 20, 2013 adding models

1 contributor

336 lines (288 sloc) | 9.213 kb

Raw | Blame | History

```
//
// lulesh.aspen
//
// An ASPEN application model for the LULESH 1.01 challenge problem.  Based
// on the CUDA version of the source code found at:
//  https://computation.llnl.gov/casc/ShockHydro/
//
param nTimeSteps = 1495

// Information about domain
param edgeElems = 45
param edgeNodes = edgeElems + 1

param numElems = edgeElems^3
param numNodes = edgeNodes^3

// Double precision
param wordSize = 8

// Element data
data mNodeList as Array(numElems, wordSize)
data mMatElemList as Array(numElems, wordSize)
data mNodeList as Array(8 * numElems, wordSize) //  8 nodes per element
data mlxim as Array(numElems, wordSize)
data mlxip as Array(numElems, wordSize)
data mletam as Array(numElems, wordSize)
data mletap as Array(numElems, wordSize)
data mzetam as Array(numElems, wordSize)
data mzetap as Array(numElems, wordSize)
data melemBC as Array(numElems, wordSize)
data mE as Array(numElems, wordSize)
data mP as Array(numElems, wordSize)
```

```
kernel CalcMonotonicQGradients {
  execute [numElems]
  {
    loads [8 * indexWordSize] from nodelist
    // Load and cache position and velocity.
    loads/caching [8 * wordSize] from x
    loads/caching [8 * wordSize] from y
    loads/caching [8 * wordSize] from z

    loads/caching [8 * wordSize] from xvel
    loads/caching [8 * wordSize] from yvel
    loads/caching [8 * wordSize] from zvel

    loads [wordSize] from volo
    loads [wordSize] from vnew
    // dx, dy, etc.
    flops [90] as dp, simd
    // delvk delxk
    flops [9 + 8 + 3 + 30 + 5] as dp, simd
    stores [wordSize] to delv_xeta
    // delxi delvi
    flops [9 + 8 + 3 + 30 + 5] as dp, simd
    stores [wordSize] to delx_xi
    // delxj and delvj
    flops [9 + 8 + 3 + 30 + 5] as dp, simd
    stores [wordSize] to delv_eta
  }
}
```

LULESH – runtime optimizations

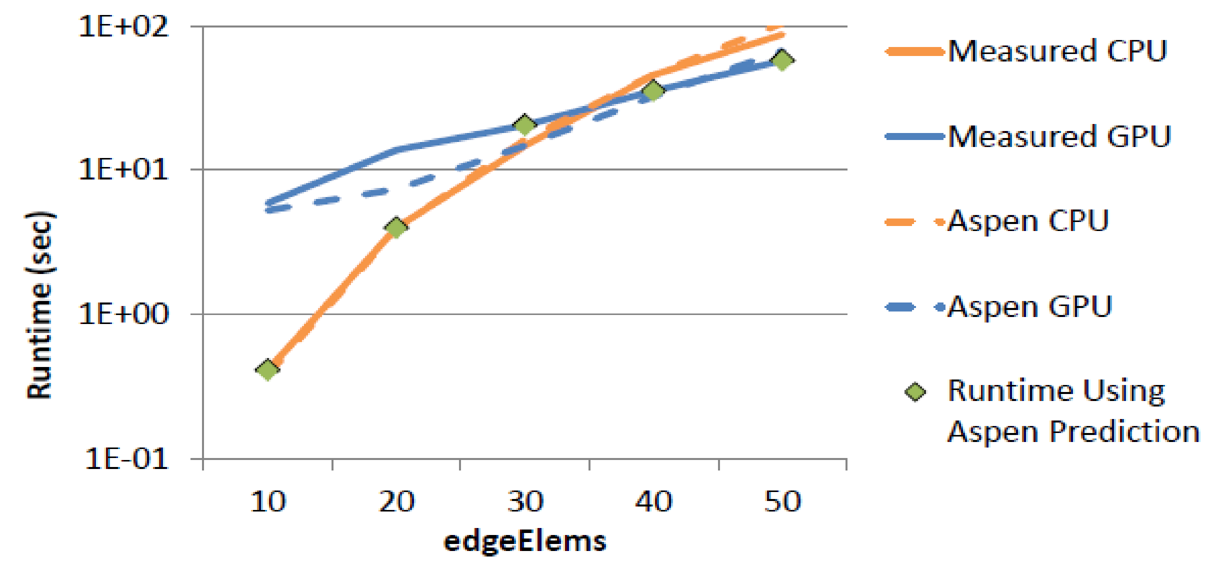


Fig. 7: Measured and predicted runtime of the entire LULESH program on CPU and GPU, including measured runtimes using the automatically predicted optimal target device at each size.

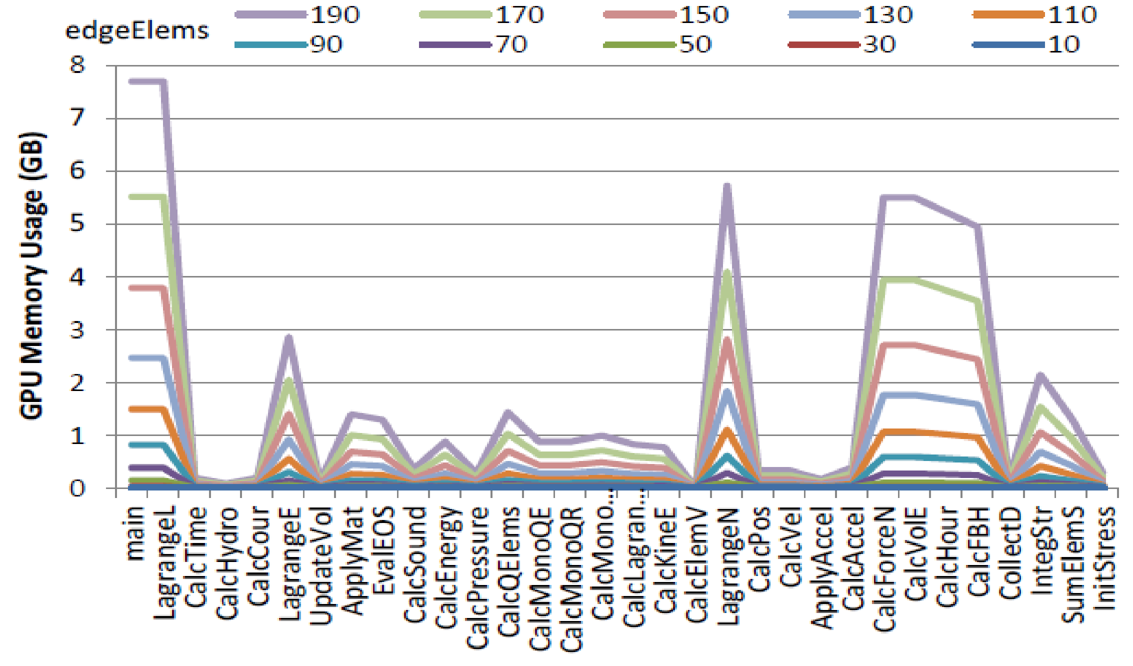


Fig. 8: GPU Memory Usage of each Function in LULESH, where the memory usage of a function is inclusive; value for a parent function includes data accessed by its child functions in the call graph.

3DFFT

```
// Dimension of cubic 3D Volume
param n = 8192
param a = 6.3
param wordSize = 16 // Double Complex Words
param dataPerProc = (n^3 * wordSize) / P
data fftVolume [n^3 * wordSize]
```

```
control pencil {
  localFFT -> transpose // in X
  exchange
  localFFT -> transpose // in Y
  exchange
  localFFT -> transpose // in Z
}
```

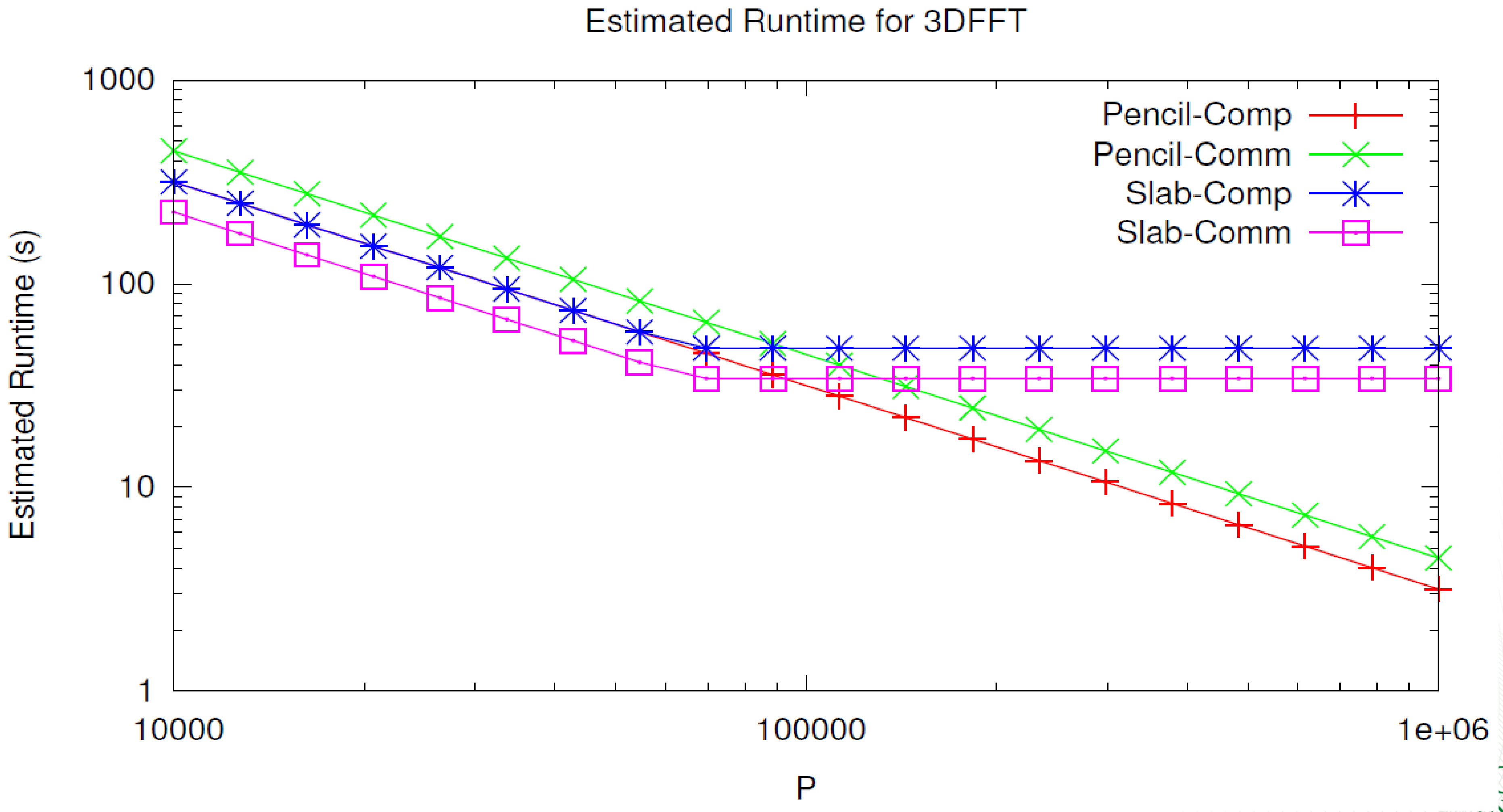
```
control slab {
  localFFT -> transpose // in X
  localFFT -> transpose // in Y
  exchange
  localFFT -> transpose // in Z
}
```

```
kernel localFFT {
  exposes parallelism [n^2]
  requires flops [5 * n * log2(n)] as dp, simd
  requires loads [a * (n*wordSize) * max(1, log(n*wordSize)/log(Z))]
    from fftVolume
}
```

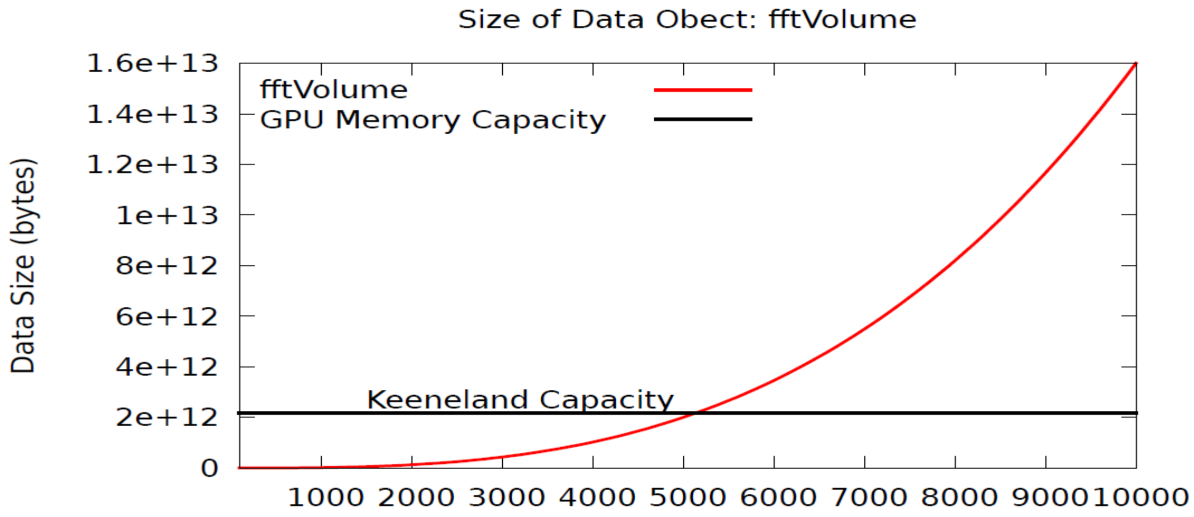
```
kernel exchange {
  exposes parallelism [P]
  requires messages [(n^3 * wordSize) / P] as
      allToAll
}
```

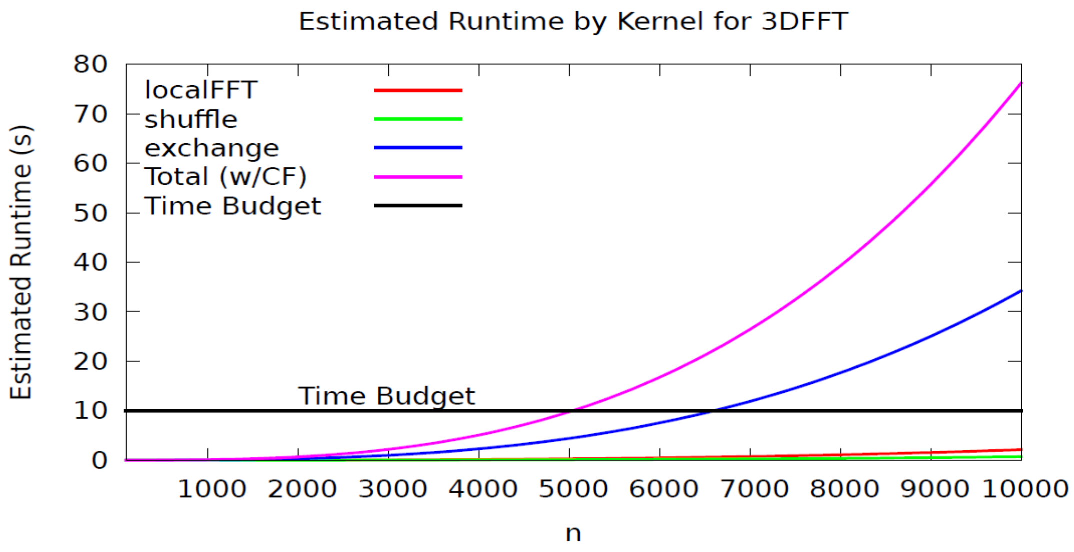
3DFFT: Slab vs. Pencil Tradeoff
Ideal Parallelism

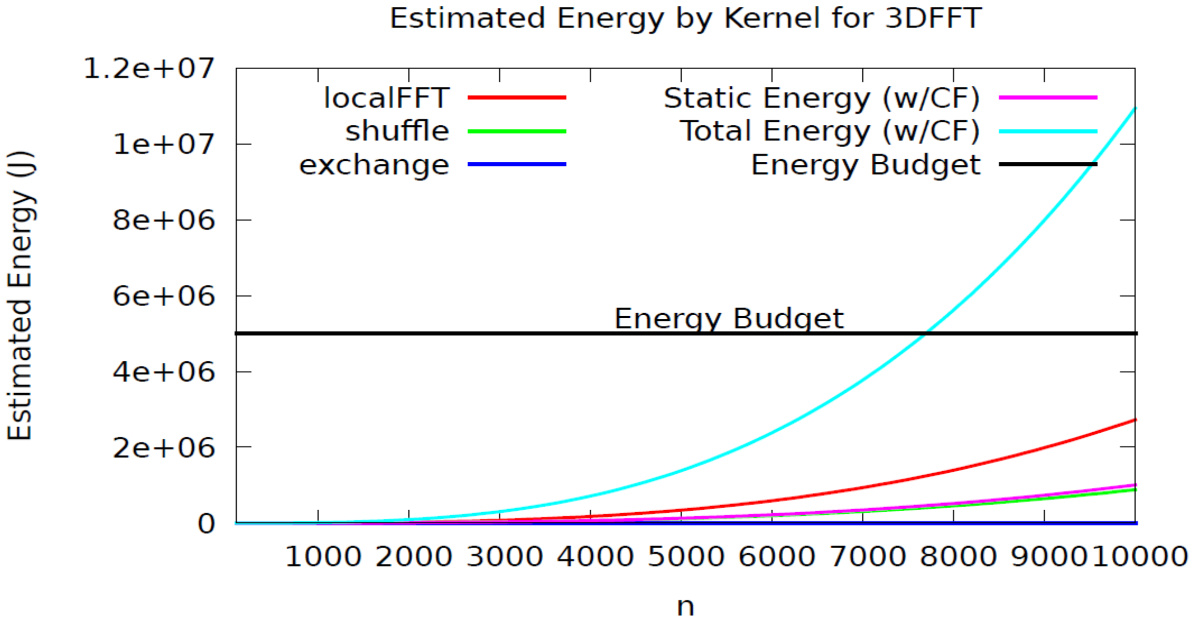
- Insights become obvious with Aspen



Design Space Exploration

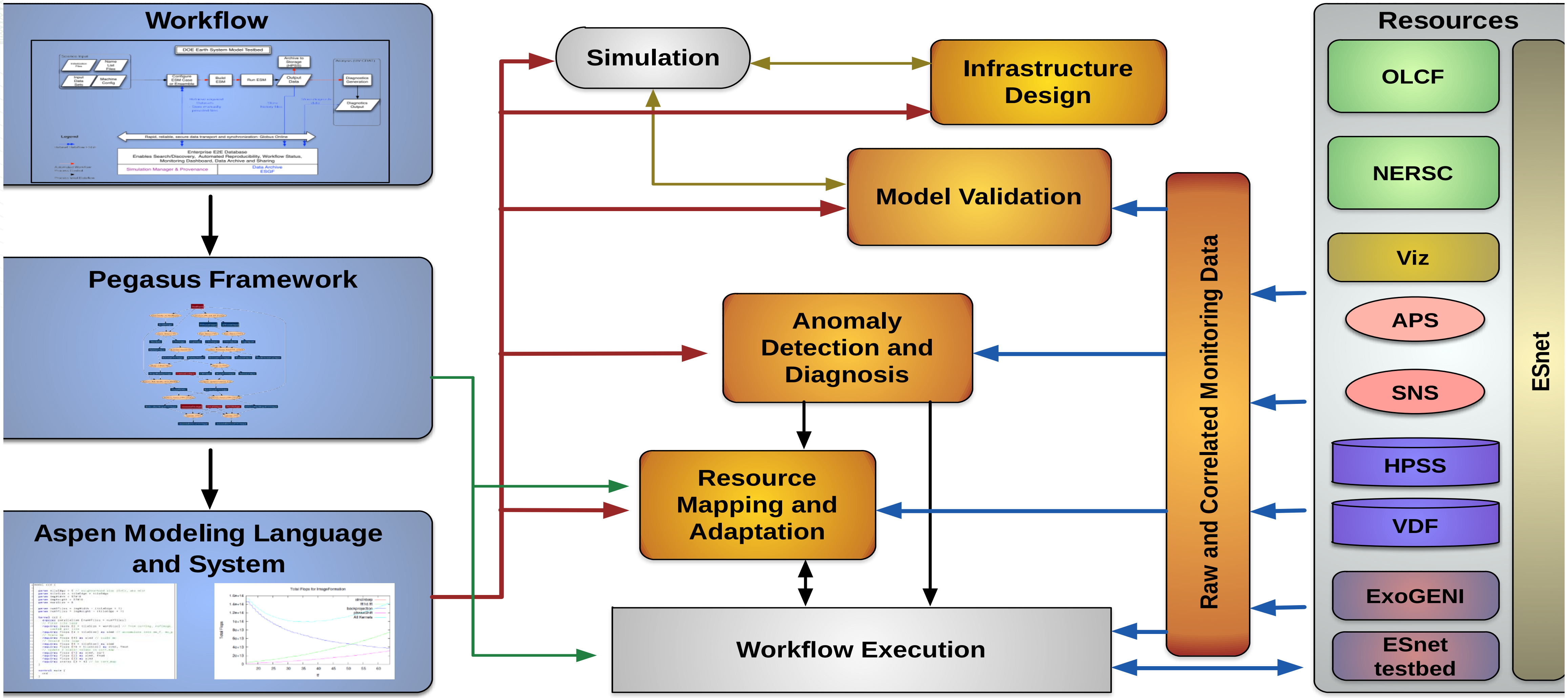






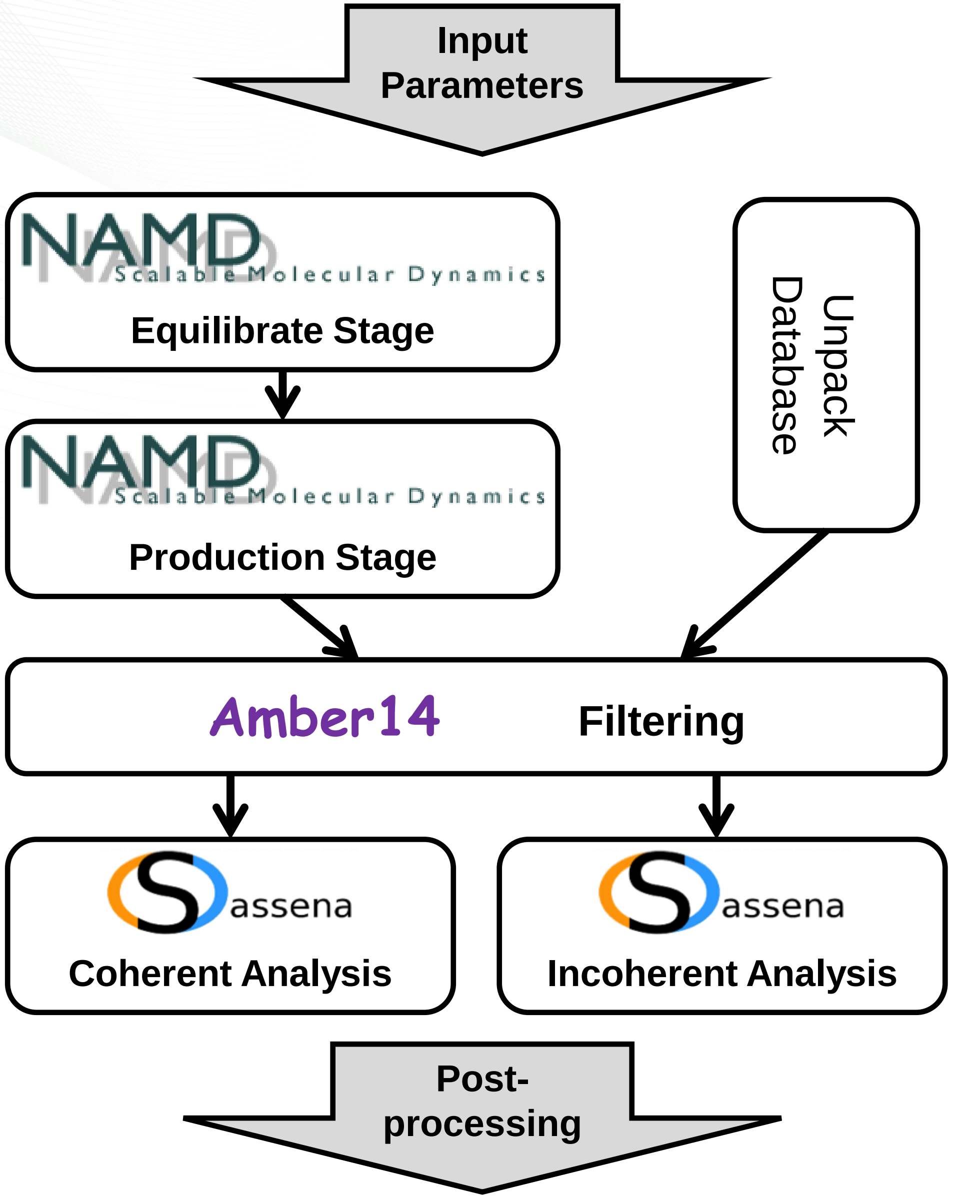
- n is approximately 5000

PANORAMA Overview



E. Deelman, C. Carothers *et al.*, *"PANORAMA: An Approach to Performance Modeling and Diagnosis of Extreme Scale Workflows," International Journal of High Performance Computing Applications, (to appear), 2015,*

Spallation Neutron Source Workflow




```
kernel main
{
    par {
        seq {
            call namd_eq_200()
            call namd_prod_200()
        }
        seq {
            call namd_eq_290()
            call amd_prod_290()
        }
        call unpack_database()
    }
    par {
        call amber_ptraj_200()
        call amber_ptraj_290()
    }
    par {
        call sassena_coh_200()
        call sassena_coh_290()
        call sassena_inc_200()
        call sassena_inc_290()
    }
}
```

Summary

- Our community has major challenges in HPC as we move to extreme scale
  - Power, Performance, Resilience, Productivity
  - Major shifts in architectures, software, applications
    - Not just HPC: Most uncertainty in two decades
- New technologies emerging to address some of these challenges
  - Heterogeneous computing
  - Nonvolatile memory
- Consequently, we now have critical situations in
  - Portable programming models
  - Performance prediction for procurement, optimization, etc
- Aspen is a tool we have developed for performance prediction

Acknowledgements

- Contributors and Sponsors
  - Future Technologies Group: http://ft.ornl.gov
  - US Department of Energy Office of Science
    - DOE Vancouver Project: https://ft.ornl.gov/trac/vancouver
    - DOE Blackcomb Project: https://ft.ornl.gov/trac/blackcomb
    - DOE ExMatEx Codesign Center: http://codesign.lanl.gov
    - DOE Cesar Codesign Center: http://cesar.mcs.anl.gov/
    - DOE Exascale Efforts: http://science.energy.gov/ascr/research/computer-science/
  - Scalable Heterogeneous Computing Benchmark team: http://bit.ly/shocmarx
  - US National Science Foundation Keeneland Project: http://keeneland.gatech.edu
  - US DARPA
  - NVIDIA CUDA Center of Excellence