INFORMATION TECHNOLOGY LABORATORY

Explore Exascale Big Data Analytics and Systems

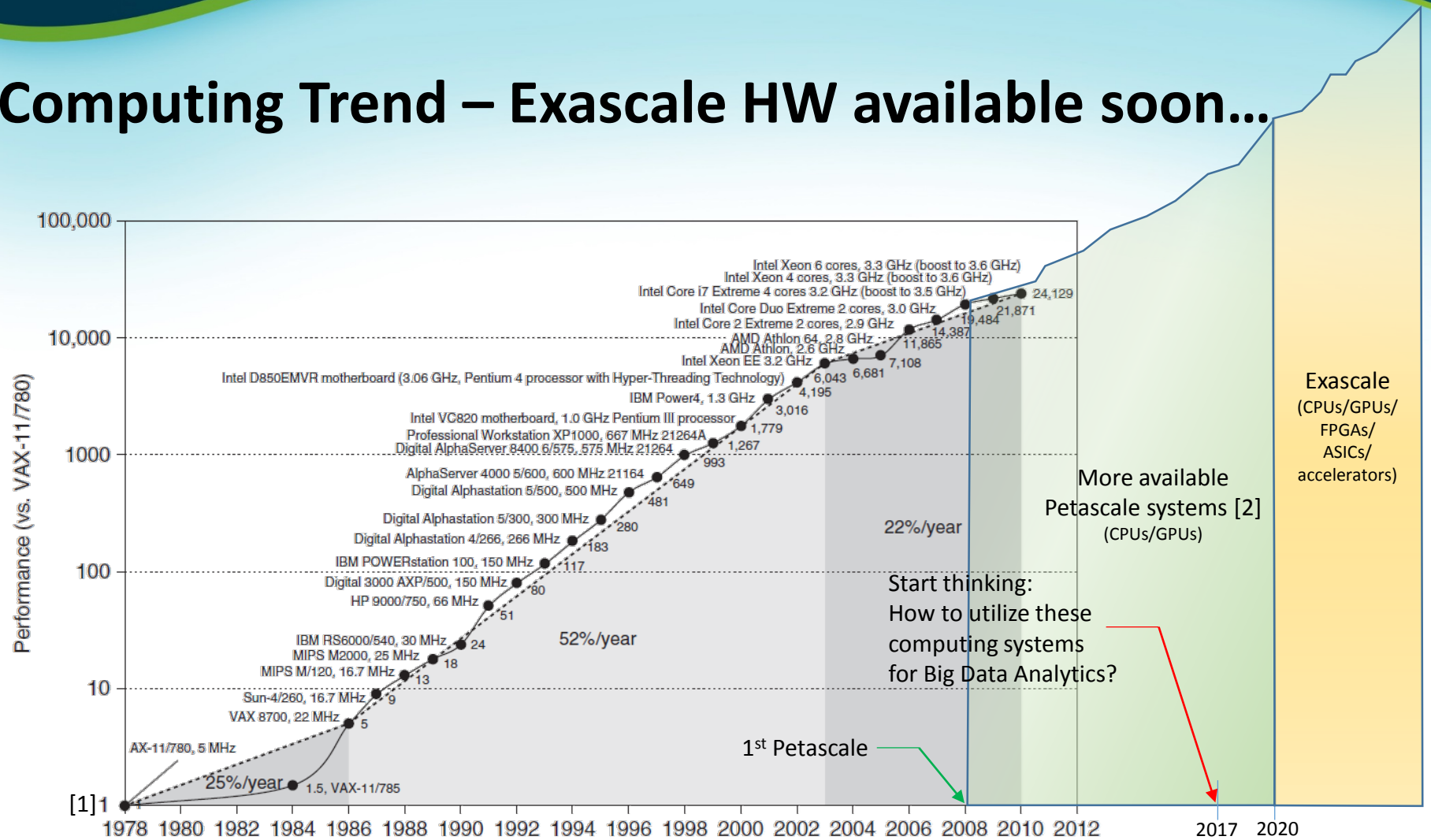
Wo Chang
Digital Data Advisor
wchang@nist.gov

June 2, 2017

Agenda

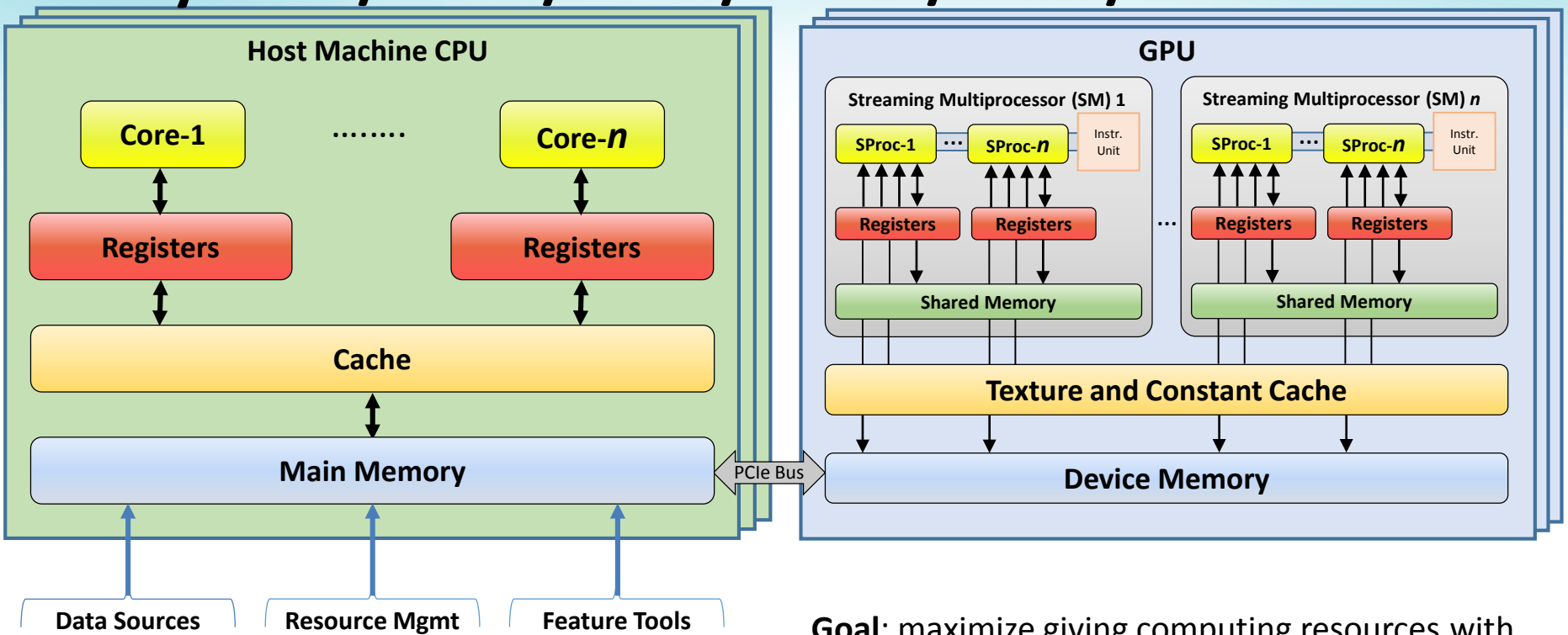
- Computing Trend – Exascale HW available soon…
- Computing Trend – Current HPC and Big Data Stacks
- Exascale Big Data Analytics Opportunities and Challenges
- Collective Synergy to Push Exascale Big Data Analytics and Systems
  - IEEE Big Data initiative
  - NIST Big Data Reference Architecture for Analytics and Beyond
  - IEEE Big Data Governance and Metadata Management (BDGMM)
  - IEEE Future Big Data Analytics and Systems Direction

Computing Trend – Exascale HW available soon…



[1] Computer Architecture A quantitative Approach (5th edition) by John L. Hennessy and David A. Patterson
[2] Top500 Supercomputing Listing: https://www.top500.org/lists/2016/11/
FPGAs=Field Programmable Gate Arrays
ASICs=Application Specific Integrated Circuits

Many CPUs/Cores/GPUs/FPGAs/ASICs/accelerators



Data Sources | Resource Mgmt | Feature Tools

Audiovisual Content | High-/Low-level Job Scheduler | Feature extraction Libraries

Goal: maximize giving computing resources with proper load balancing between tasks (CPUs vs GPUs, etc.) via intelligence scheduler.
Targets: Legacy HPC apps + BD analytics
Applications: simulation, modeling, streaming, social media, IoT, etc.

Computing Trend – Current HPC and Big Data Stacks

High-Performance Computing (HPC)		Big Data
Fortran /Parallel C/MP/MPI Applications *High Performance*	Programming Model	Java, Python, etc. Applications *Scalable and Simple to Use*
SLURM/Torque *Resource Management and Scheduler*	Resource Management	YARN *Resource Management and Scheduler* *(More resilient of hardware failures)*
Lustre Parallel Distributed File Systems *Remote Storage*	File System	HDFS, SPARK *Local Storage*
Compute & Memory Focused *High Performance Components* *(Sever, Storage, Switch Fabric, Infrastructure)*	Hardware	Scalable Computing and Storage Focused *Standard Server Components* *(Sever, Storage, Switch Fabric, Infrastructure)*

Exascale Big Data Analytics Challenges and Opportunities [1]

Differences	HPC	Big Data
Interconnect Hardware	RDMA (Remote Direct Memory Access) via Infiniband and OmniPath	Conventional hardware for horizontal scaling
Programming Language	C, C++, etc. required recompile with different OS	JVM, Python, etc. for portability between OS
Computing	Large computation loads	Large and complex datasets (order of terabytes/exabytes)
Filesystems	Mostly NSF	Distributed file systems between cluster nodes (e.g., HDFS)
Storage	Not much, mainly on computing	Very high demand
Fault Tolerance	Needs to enforce to handle system failures and soft errors	Built-in
Execution Control Flow	MPI directly execute on target machines; much better control	Spark uses descriptive API managed by Spark driver and submit job to cluster nodes for execution
Scalability	Mostly vertical	Mostly horizontal
Others…	…	…

Common Goals	HPC	Big Data
Optimize code for performance, energy, and reliability	YES	YES
Reduce data in motion with dynamic tasks scheduler	YES	YES
Others…	…	…

Questions:

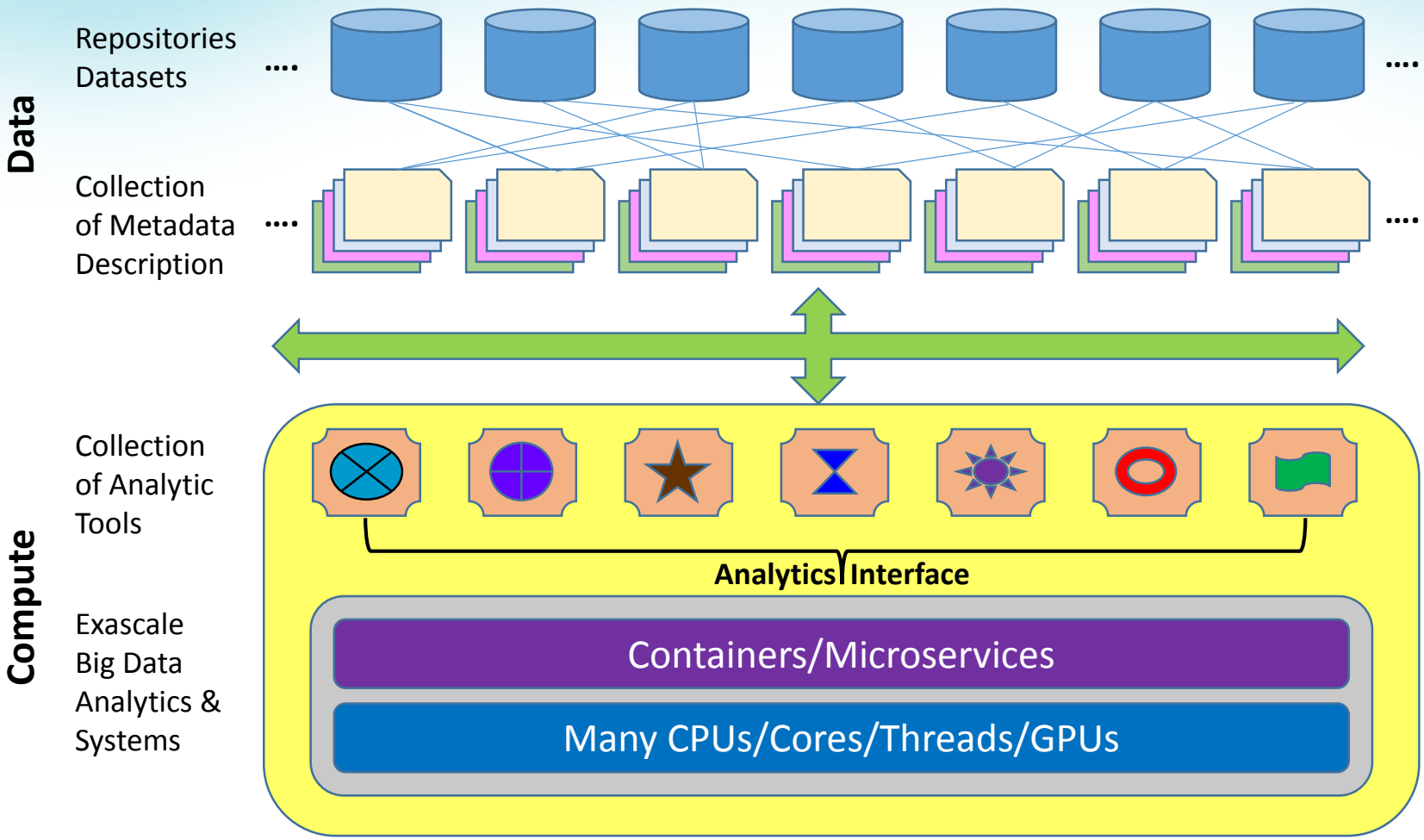
1. How best to combine the two stacks (HPC inside Big Data, Big Data inside HPC, or hybrid)?
2. What best standards interface to support them?
3. *Others…*

[1]: <need to find the original source>

Use Case Implementation: Long DNA Sequence Alignment

Sequence Pair sizes	1 KNC (SW1) [Using 237 threads] †		2 KNC (SW1) [Using 237* threads] †		4 KNC (SW1) [Using 237* threads] †		1 KNL (SW1+) [Using 41* Threads]		Nvidia K80 (SW2)		Nvidia TitanX (SW2)		Speedup (compare from 1KNC)				
	Time (s)	GCUPS	Time (s)	GCUPS	Time (s)	GCUPS	Time (s)	GCUPS	Time (s)	GCUPS	Time (s)	GCUPS	2 KNC	4 KNC	1 KNL	K80	TitanX
D44M vs. D46M $4.4x10^6$ X $4.6x10^6$	700 11.66m	29.2	396 6.60m	51.7	200 3.33m	100.7	1,258 29.9m	16.2	225 3.75m	91.0	120 2.00m	170.6	1.8	3.4	0.55	3.1	5.8
D23M vs. D33M $23x10^6$ X $33x10^6$	25,166 6.99h	30.0	14,105 3.91h	53.5	6,855 1.90h	110.1	26,397 7.33h	28.5	8,190 2.27h	92.1	4,930 1.36h	154.0	1.8	3.7	0.95	3.0	5.1
D23M vs. D42M $23x10^6$ X $42x10^6$	32,209 8.94h	30.0	17,958 4.98h	53.9	8,746 2.42h	110.6	33,978 9.43h	28.4	10,553 2.93h	91.6	6,342 1.76h	152.5	1.8	3.7	0.94	3.0	5.1
D23M vs. D50M $23x10^6$ X $50x10^6$	38,452 10.67h	30.0	21,291 5.91h	54.1	10,385 2.88h	111.0	40,519 11.25h	28.4	12,646 3.51h	91.1	7,754 2.15h	152.5	1.8	3.7	0.94	3.0	4.9
D33M vs. D42M $33x10^6$ X $42x10^6$	45,868 12.74h	30.1	25,553 7.09h	54.0	12,381 3.43h	111.4	45,617 12.67h	30.2	15,352 4.26h	89.9	9,043 2.51h	152.4	1.8	3.7	1.01	2.9	4.9
D33M vs. D50M $33x10^6$ X $50x10^6$	54,582 15.16h	30.1	30,402 8.44h	54.0	15,499 4.30h	106.0	54,340 15.09h	30.2	18,175 5.04h	90.3	10,751 2.98h	152.7	1.8	3.5	1.00	3.0	5.0
D42M vs. D50M $42x10^6$ X $50x10^6$	70,053 19.45h	30.0	38,875 10.79h	54.1	18,902 5.25h	111.4	67,564 18.76h	31.15	22,990 6.38h	91.5	13,615 3.78h	154.5	1.8	3.7	1.03	3.0	5.1

SW1=*SWAPHI-LS* for KNC, SW1+=modified SW1 for KNL, SW2=*SW#* using CUDA/GPU, *=maximum threads used before performance decrease, "h"=hour, "m"=minute
GCUPS=billion cell updates per second. †- http://ieeexplore.ieee.org/document/6968772/?arnumber=6968772&tag=1

Credit to: Michelle Luo, Yuechen Chen, Eddie Banuelos-Casillas, Cory Wang (all George Washington University intern students at NIST)

Needs: Convergence of Data + Compute