Big Data Analytics: The Apache Spark Approach

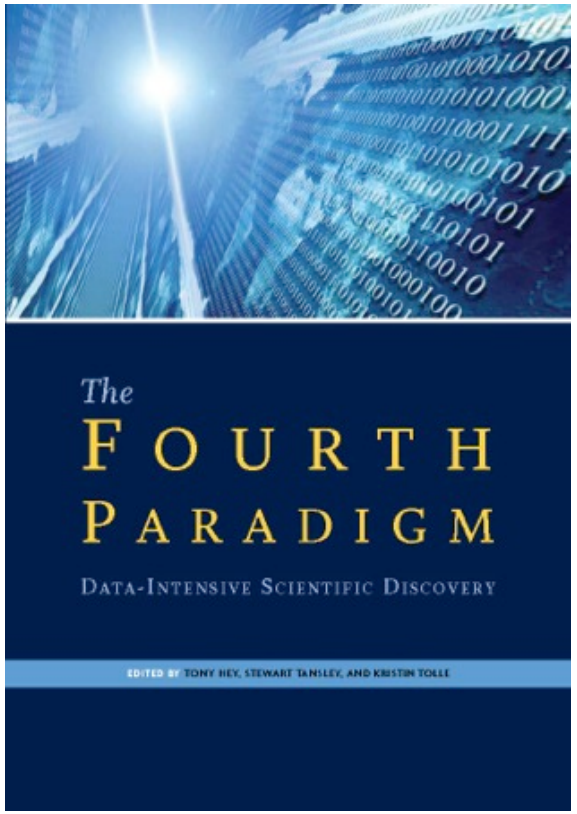
Michael Franklin
ATPESC
August 2017

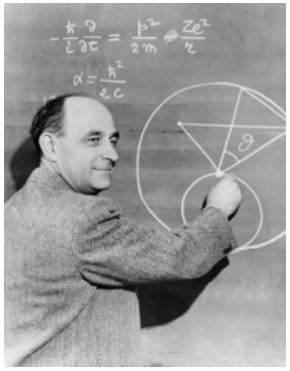
THE UNIVERSITY OF CHICAGO

Spark™

The Fourth Paradigm of Science

1. Empirical + experimental
2. Theoretical
3. Computational
4. Data-Intensive





Open Source Ecosystem & Context

2006-2010
Autonomic Computing & Cloud

2011-2016
Big Data Analytics

Usenix HotCloud Workshop 2010

Spark: Cluster Computing with Working Sets

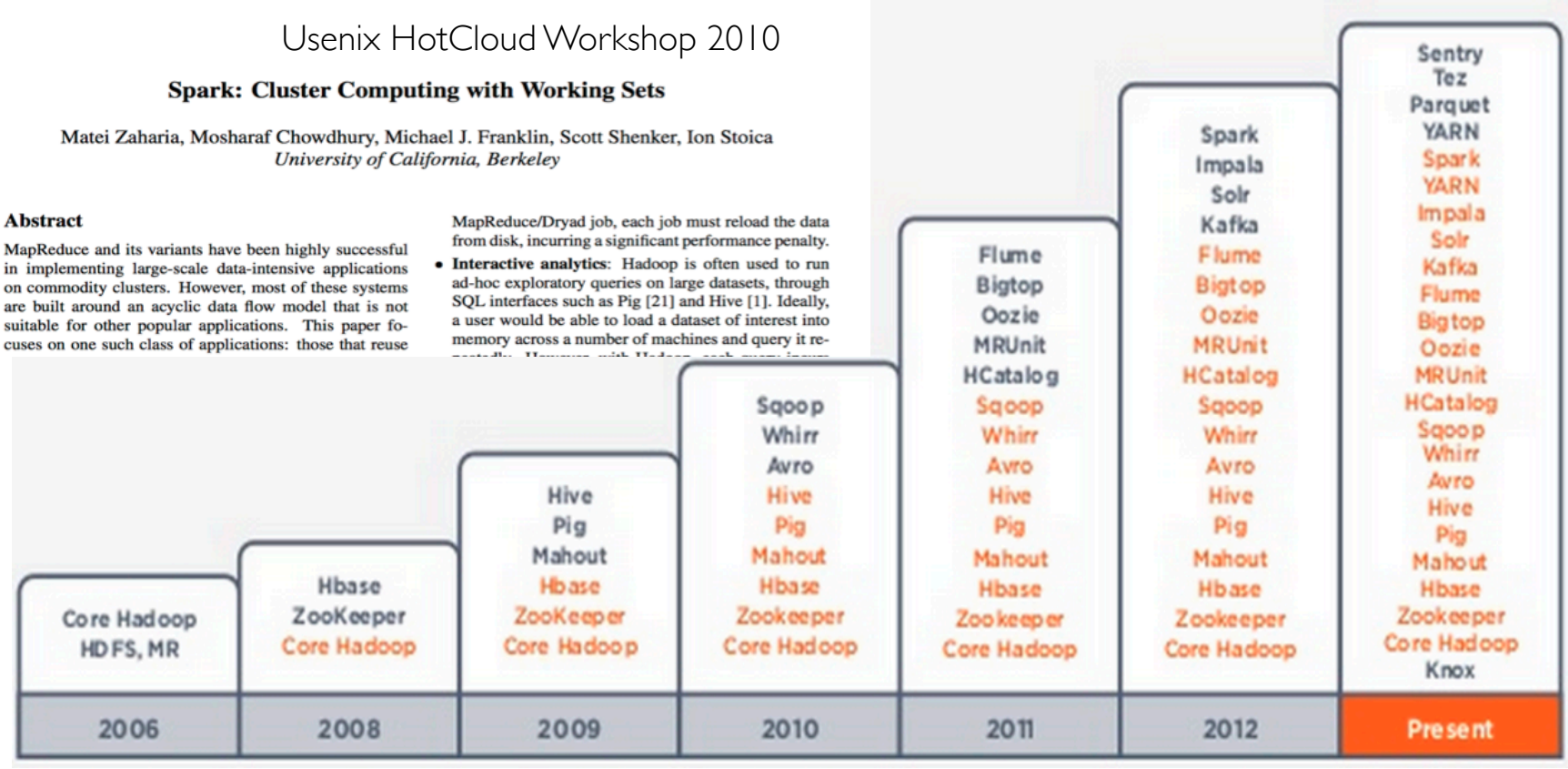
Matei Zaharia, Mosharaf Chowdhury, Michael J. Franklin, Scott Shenker, Ion Stoica
University of California, Berkeley

Abstract

MapReduce and its variants have been highly successful in implementing large-scale data-intensive applications on commodity clusters. However, most of these systems are built around an acyclic data flow model that is not suitable for other popular applications. This paper focuses on one such class of applications: those that reuse

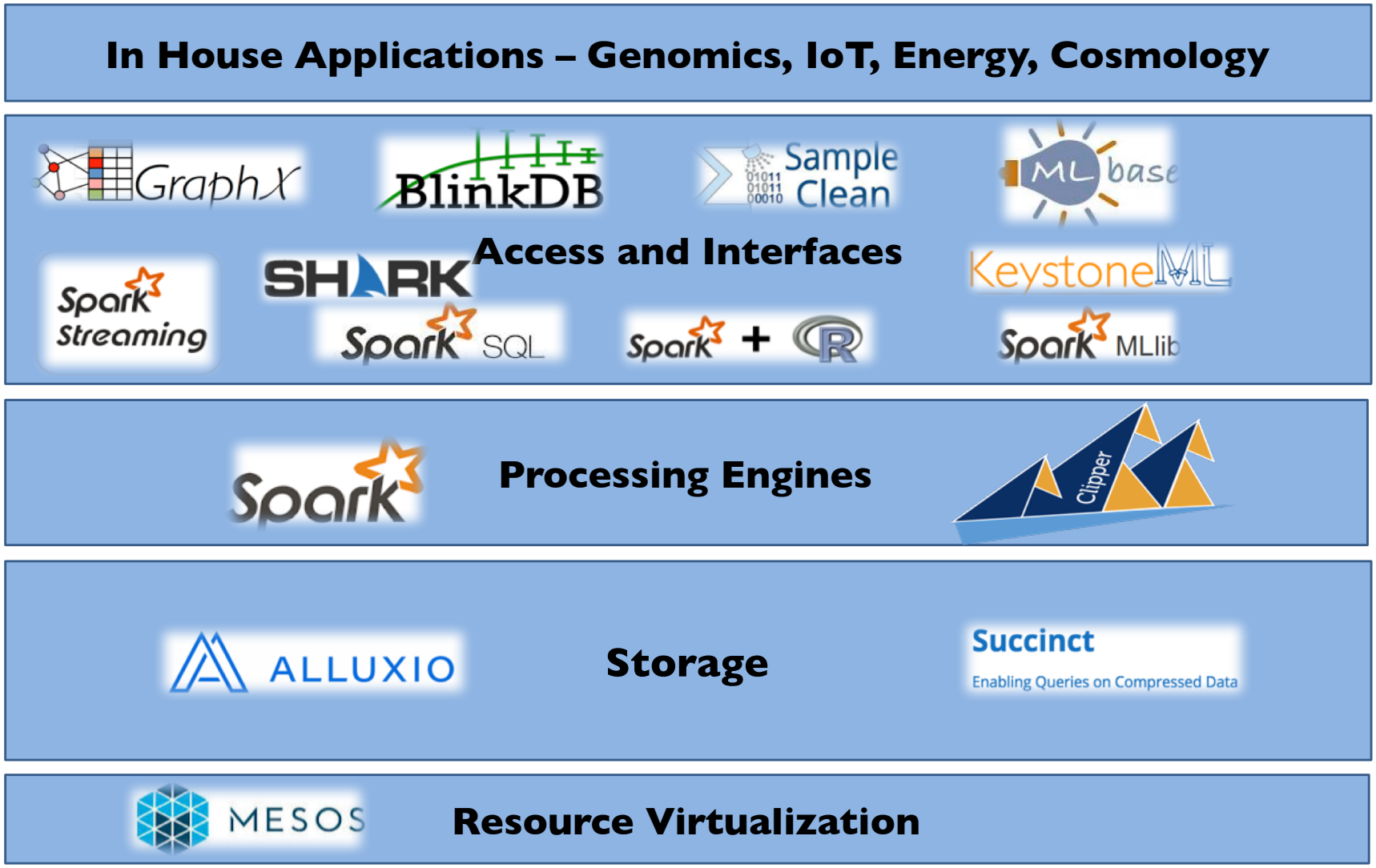
MapReduce/Dryad job, each job must reload the data from disk, incurring a significant performance penalty.

- **Interactive analytics**: Hadoop is often used to run ad-hoc exploratory queries on large datasets, through SQL interfaces such as Pig [21] and Hive [1]. Ideally, a user would be able to load a dataset of interest into memory across a number of machines and query it re-



...

Berkeley Data Analytics Stack



Pathways to Convergence

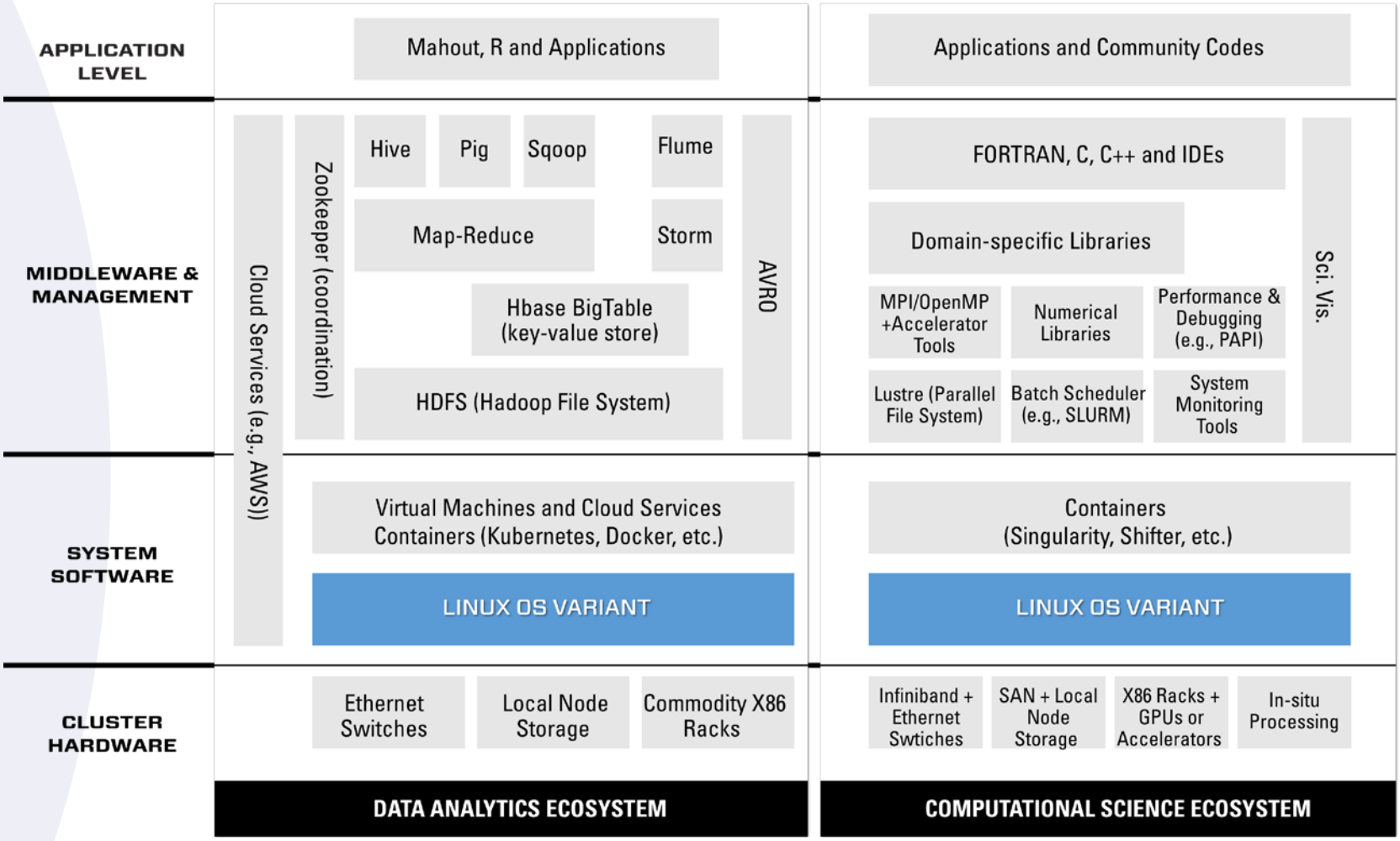
Mark Asch - mark.asch@u-picardie.fr

EXDCI – BDVA meeting

Bologna, July 4th 2017

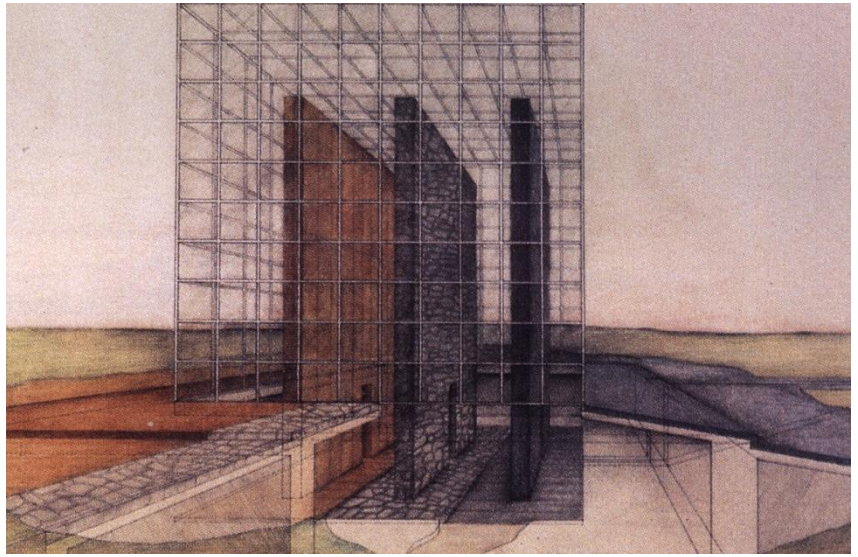
The EXDCI project has received funding from the European Unions Horizon 2020 research and innovation programmed under the grant agreement No 671558.

The 2 worlds: Big Data and HPC software stacks



Argonne Training Program on Extreme-Scale Computing (ATPESC)

Presented to
ATPESC 2017 Participants



http://deathofdrawing.com/wp-content/gallery/raimund-abraham/RA-House-With-Three-Walls.jpg

Peter Kogge

Data Intensive Computing, the 3rd Wall, and the Need for Innovation in Architecture

Q Center, St. Charles, IL (USA)
Date 08/04/2017

EXASCALE COMPUTING PROJECT

When Do We Need New Architectures

- Long-lasting architectural advances occur when a “wall” must be overcome
- 1st Wall – Mid 90s: the **Memory Wall**
- 2nd Wall – 2004: the **Power Wall**
- 3rd Wall – Now: the **Locality Wall**

And this is largely due to emergence of apps with *Data Intensive* Characteristics

What Do I Mean by *Data Intensive*?

- Computation dominated by data access & movement – **not flops**
- Large sets of data often persistent
  - but little reuse during computation
- No predictable regularity
- Significantly different scaling
- Streaming becoming important

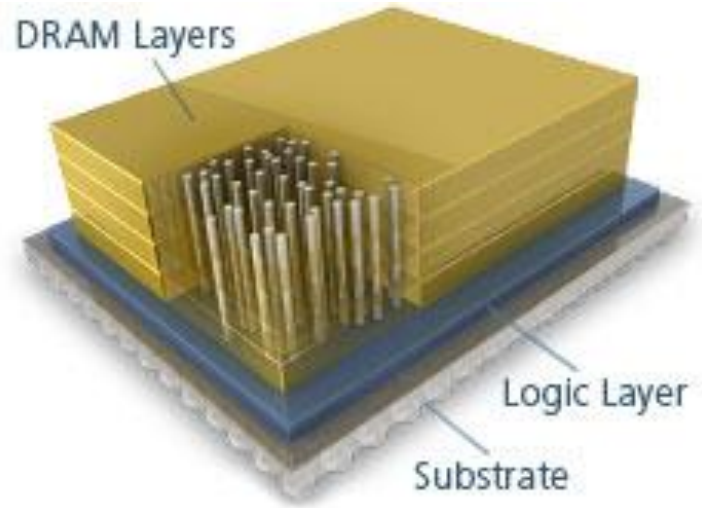
The "Locality" we have come to expect from our apps is disappearing

This Talk

- Moore's Law and the Prior Walls
- Today's Architectures
- Evidence of a New "Locality" Wall
  - Benchmarks
  - A Big Data Application
- Migrational Computing: a Possible Architectural Fix

Technology, Moore's Law, and Beyond

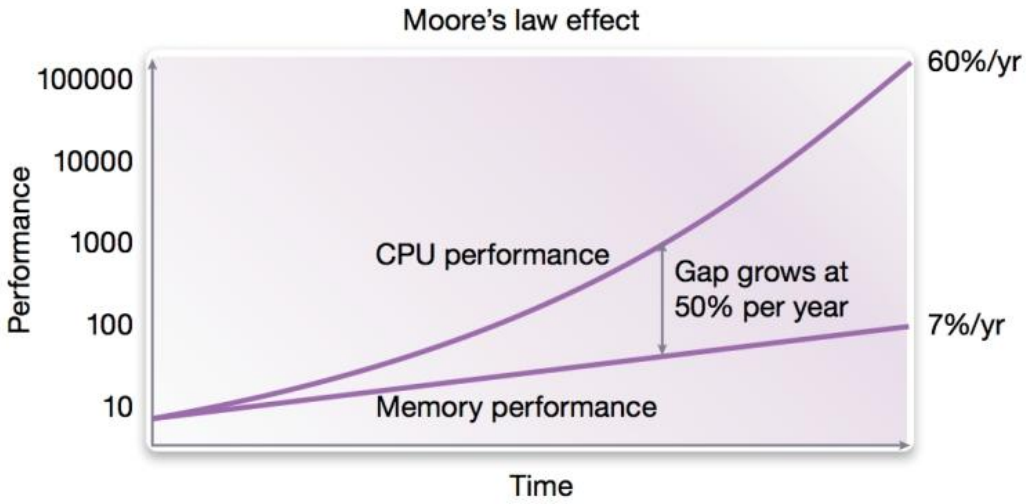
- Moore's Law: 2D transistors get smaller & faster
  - From 10um to 5nm feature size: **2,000X smaller & faster**
- Cores get smaller, faster, lower power
  - Power density approx. constant *as long as* V_{dd} declines
- Memory arrays get denser
  - To maximize density, access time *drops at best slowly*
  - Can increase bandwidth, but power skyrockets
- After Moore's Law: ***we're going 3D***!
  - With a mix of die types



http://www.micron.com/products/hybrid-memory-cube

The Memory Wall (mid 1990s)

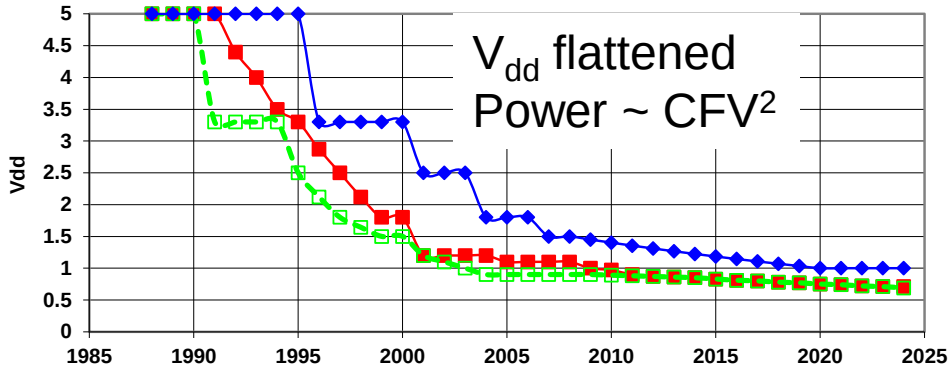
- Core clock speeds outran memory latency
- **Breaking the Wall**: Use extra transistors for
  - **Bigger on-chip SRAM caches**
  - More ILP to find more memory accesses
  - Add additional floating point capability
- Enablers: Applications had *plenty* of locality
- Example: Ax=b, A is large, dense, matrix
  - Tremendous temporal locality
  - Assume caches can save nxn patch of A
  - $O(n^2)$ to read nxn patch of A to cache
  - $O(n^3)$ operations on this patch
- With ***big enough cache***, don't care *how slow* memory is

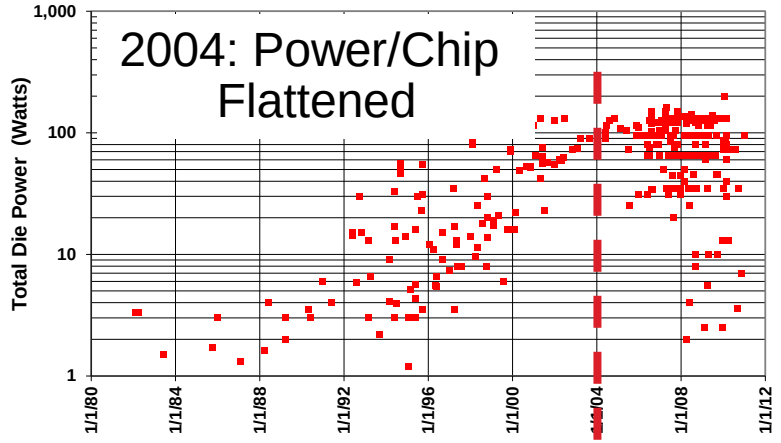


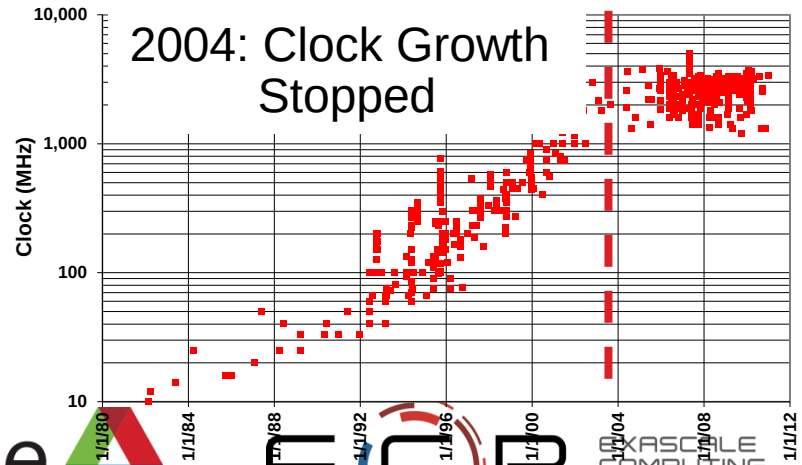
http://www.extremetech.com/wp-content/uploads/2014/07/140364245678419.jpg

The Power Wall (2004)

- Flattening V_{dd} increased power density
  - Bigger chips meant more logic to dissipate
- Result: at 120Watts, cooling uneconomical
- Breaking the wall:
  - ***Lower* the clock rate**
  - **Use *multiple* simpler cores**
  - **Increase *SIMD-style* parallelism**
- Side-effect: need more bandwidth
- Solution for dense apps: again **bigger caches**







Argonne NATIONAL LABORATORY

ECP EXASCALE COMPUTING PROJECT

“A Vision for Exascale: Simulation, Data and Learning”

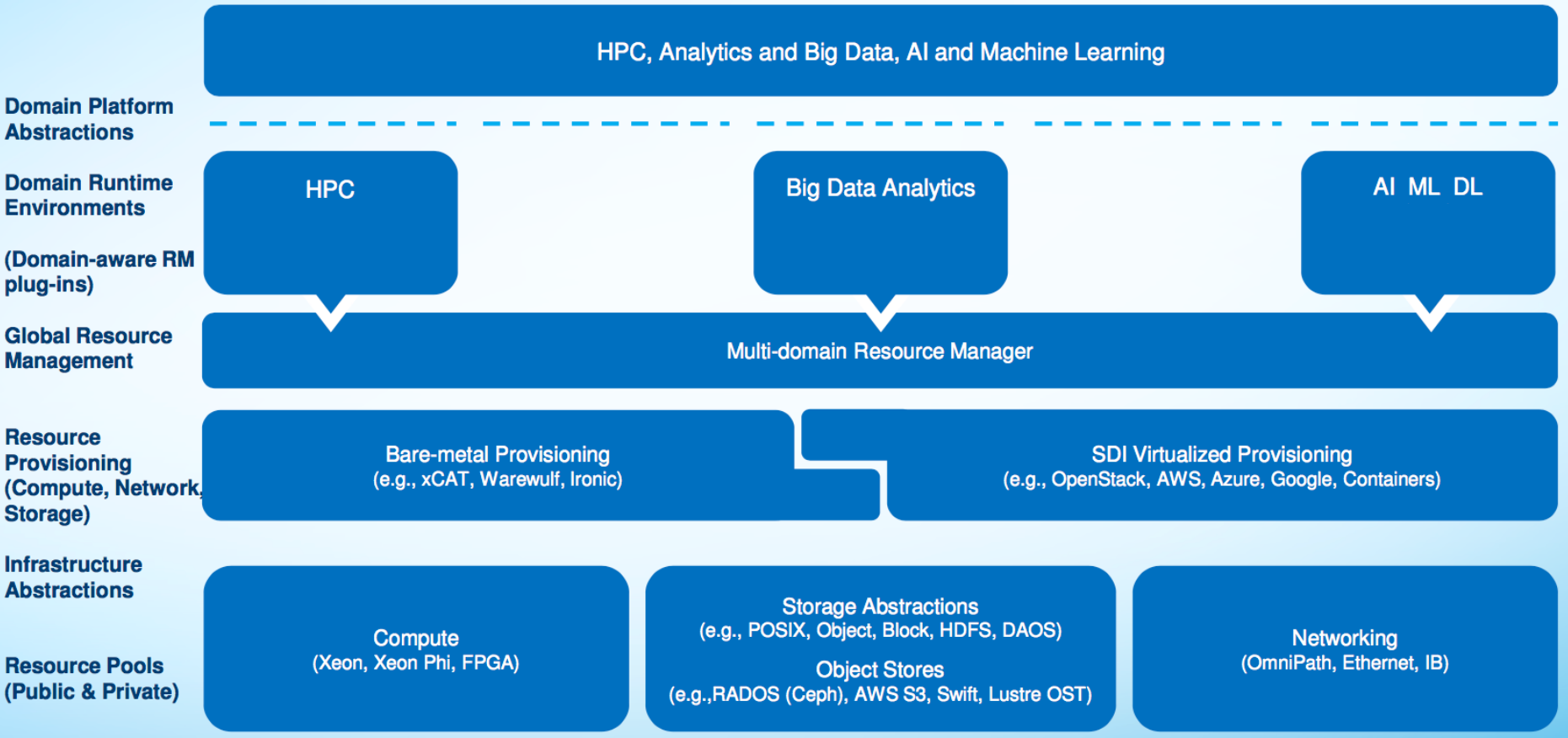
Rick Stevens
Argonne National Laboratory
The University of Chicago

Crescat scientia; vita excolatur

Aurora 21

- Argonne's Exascale System
- Balanced architecture to support three pillars
  - Large-scale Simulation (PDEs, traditional HPC)
  - Data Intensive Applications (science pipelines)
  - Deep Learning and Emerging Science AI
- Enable integration and embedding of pillars
- Integrated computing, acceleration, storage
- Towards a common software stack

Towards an Integrated Stack



The New HPC “Paradigm”