

CS 444: Big Data Systems

Chapter 2. Computing Trends for Big Data

Chase Wu

Professor of Data Science
Director of Center for Big Data
New Jersey Institute of Technology

Collaborative Research Staff
Computer Science and Mathematics Division
Oak Ridge National Laboratory

Outline

- **Introduction**
- **Challenges & Objectives**
- **Computing Technologies**
 - **High-performance computing**
 - **Supercomputing and Cluster Computing**
 - **Grid computing**
 - **Computing continuum: from edge to fog to cloud**
 - **Mobile computing**
- **Ongoing Research in Our Group**
 - **Data-intensive computing**
 - **High-performance networking**

Albert Einstein
Old Grove Rd.
Nassau Point
Peconic, Long Island

August 2nd, 1939

F.D. Roosevelt,
President of the United States,
White House
Washington, D.C.

Sir:

Some recent work by E.Fermi and L. Szilard, which has been communicated to me in manuscript, leads me to expect that the element uranium may be turned into a new and important source of energy in the immediate future. Certain aspects of the situation which has arisen seem to call for watchfulness and, if necessary, quick action on the part of the Administration. I believe therefore that it is my duty to bring to your attention the following facts and recommendations:

In the course of the last four months it has been made probable - through the work of Joliot in France as well as Fermi and Szilard in America - that it may become possible to set up a nuclear chain reaction in a large mass of uranium, by which vast amounts of power and large quantities of new radium-like elements would be generated. Now it appears almost certain that this could be achieved in the immediate future.

This new phenomenon would also lead to the construction of bombs, and it is conceivable - though much less certain - that extremely powerful bombs of a new type may thus be constructed. A single bomb of this type, carried by boat and exploded in a port, might very well destroy the whole port together with some of the surrounding territory. However, such bombs might very well prove to be too heavy for transportation by air.

-2-

The United States has only very poor ores of uranium in moderate quantities. There is some good ore in Canada and the former Czechoslovakia, while the most important source of uranium is Belgian Congo.

In view of this situation you may think it desirable to have some permanent contact maintained between the Administration and the group of physicists working on chain reactions in America. One possible way of achieving this might be for you to entrust with this task a person who has your confidence and who could perhaps serve in an unofficial capacity. His task might comprise the following:

a) to approach Government Departments, keep them informed of the further development, and put forward recommendations for Government action, giving particular attention to the problem of securing a supply of uranium ore for the United States;

b) to speed up the experimental work, which is at present being carried on within the limits of the budgets of University laboratories, by providing funds, if such funds be required, through his contacts with private persons who are willing to make contributions for this cause, and perhaps also by obtaining the co-operation of industrial laboratories which have the necessary equipment.

I understand that Germany has actually stopped the sale of uranium from the Czechoslovakian mines which she has taken over. That she should have taken such early action might perhaps be understood on the ground that the son of the German Under-Secretary of State, von Weizsäcker, is attached to the Kaiser-Wilhelm-Institut in Berlin where some of the American work on uranium is now being repeated.

Yours very truly,

A. Einstein
(Albert Einstein)

Oak Ridge National Laboratory



1 Managed by UT-Battelle
for the Department of Energy



Oak Ridge National Laboratory



2 Managed by UT-Battelle
for the Department of Energy



Six Scientific Themes

*Born of necessity. Inspired by our quest to know.
We have always been called upon to address
America's greatest scientific challenges.*

NEUTRON SCIENCE

Leading the World

The Spallation Neutron Source and the High Flux Isotope Reactor together make Oak Ridge the world's foremost center for neutron science.

andersonian@ornl.gov

BIOLOGICAL SYSTEMS

Developing New Options

Whether converting biomass to fuel or understanding the impacts of climate change, biological research at ORNL is helping develop new options for energy, environmental protection, and human health. mannre@ornl.gov

ADVANCED MATERIALS

Strengthening American Industry

With DOE's first Nanoscience Center, the world's most powerful electron microscope, and the High Temperature Materials Laboratory, Oak Ridge plays a critical role in American industrial competitiveness.

buchananmv@ornl.gov

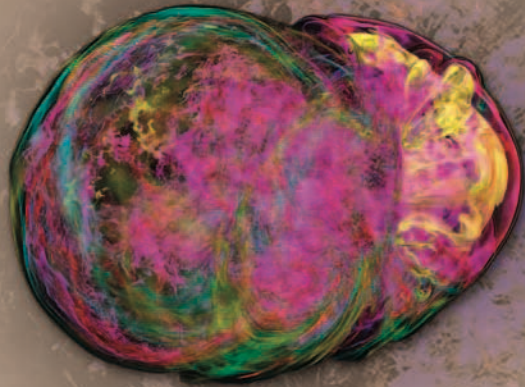
"Men love to wonder, and that is the seed of science."

... Ralph Waldo Emerson

NATIONAL SECURITY Guarding the Gates

From biochemical sensors to stopping the proliferation of nuclear weapons, technologies that make America safer are among the laboratory's top research priorities.

akersfhjr@ornl.gov



HIGH PERFORMANCE COMPUTING Tackling the Big Problems

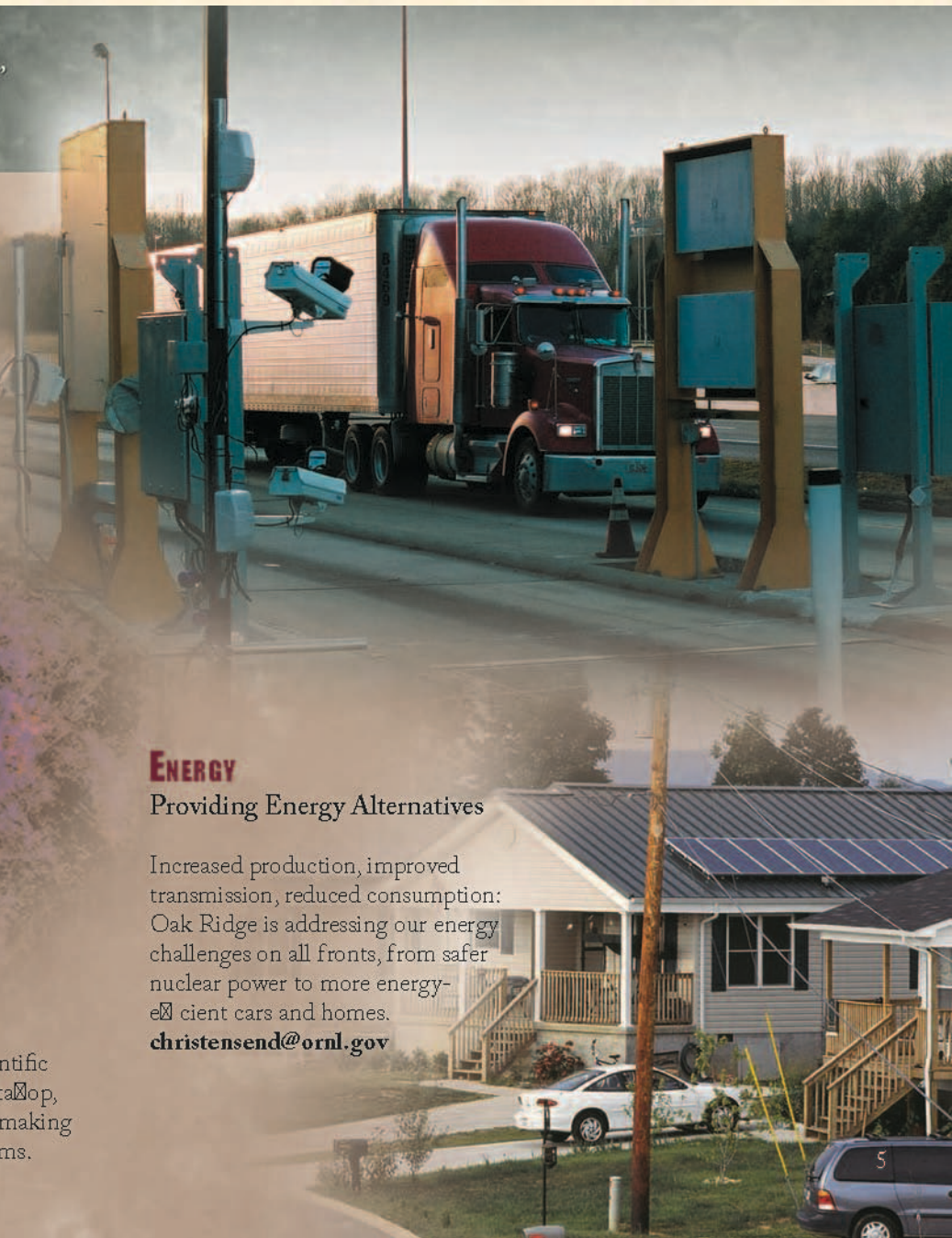
With unmatched computational capacity for open scientific research, Oak Ridge is on a path by 2009 to reach a petaFlop, or 1 quadrillion mathematical calculations per second, making it possible to model the most complex scientific problems.

zachariat@ornl.gov

ENERGY Providing Energy Alternatives

Increased production, improved transmission, reduced consumption: Oak Ridge is addressing our energy challenges on all fronts, from safer nuclear power to more energy-efficient cars and homes.

christensend@ornl.gov



World-class Facilities



Oak Ridge National Laboratory

Solving the
BIG PROBLEMS

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



Office

Supercomputer

3 Managed by UT-Battelle
for the Department of Energy



Big-data Applications

- **BIG DATA: rapidly increase from T to P, to E, to Z, to Y, and beyond...**
- **Science**
 - **Simulation**
 - Astrophysics, climate modeling, combustion research, etc.
 - **Experimental**
 - Spallation Neutron Source, Large Hadron Collider, microarray, genome sequencing, protein folding, etc.
 - **Observational**
 - Large-scale sensor networks, astronomical imaging/radio devices (Dark Energy Camera, Five-hundred-meter Aperture Spherical Telescope – FAST), etc.
- **Business**
 - **Financial transactions**
 - Wal-Mart, NY stock trading center, Amazon, Alibaba
- **Social media**
 - **YouTube, Facebook, Twitter, Weblogs, TikTok, WeChat**

No matter which type of data is considered, we need a high-performance end-to-end computing solution to support data generation, storage, transfer, processing, and analysis!

Big-data Workflows

- **Require massively distributed resources**
 - **Hardware**
 - Computing facilities, storage systems, special rendering engines, display devices (tiled display, powerwall, etc.), network infrastructures, etc.
 - **Software**
 - Domain-specific data analytics/processing tools, programs, etc.
 - **Data**
 - Real-time, archival
- **Feature different complexities**
 - Simple case: linear pipeline (a special case of DAG)
 - Complex case: DAG-structured graph
- **Different application types have different performance requirements**
 - Interactive: minimize total end-to-end delay for fast response
 - Streaming: maximize frame rate to achieve smooth data flow

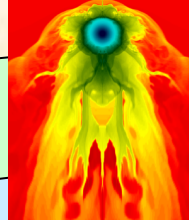
Computing Paradigms: an Overview

- **Client–Server Model**
 - *Client–server computing* refers broadly to any distributed application that distinguishes between service providers (servers) and service requesters (clients)
- **High-performance Computing (Supercomputing, Cluster Computing)**
 - Powerful computers: supercomputer, PC cluster
 - Used mainly by large organizations for critical applications, typically bulk data processing such as scientific computing, enterprise resource planning, and financial transaction processing
 - Programming models: MPI, OpenMP, CUDA, MapReduce/Hadoop, Spark
- **Grid Computing**
 - A form of distributed computing and parallel computing, whereby a “super and virtual computer” is composed of a cluster of networked, loosely coupled computers acting in concert to perform very large tasks
- **Cloud (Utility) Computing**
 - The packaging of computing resources, such as computation and storage, as a metered service similar to a traditional public utility, such as electricity
- **Service-Oriented Computing**
 - Cloud computing provides services related to computing while, in a reciprocal manner, service-oriented computing consists of the computing techniques that operate on software-as-a-service
- **Edge Computing**
 - A distributed computing paradigm that brings computation and data storage closer to the sources of data to improve response times and save bandwidth
 - Internet of things (IoT) is an example of edge computing
- **Peer-to-peer (P2P) Computing**
 - Distributed architecture without the need for central coordination, with participants being at the same time both suppliers and consumers of resources (in contrast to the traditional client–server model)
- **Mobile Computing**
 - Computing on the go!

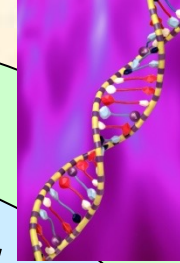
High-performance Computing (Supercomputing, Cluster Computing)

Supercomputing for Scientific Applications

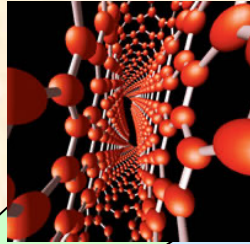
Astrophysics



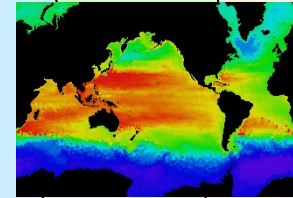
Computational biology



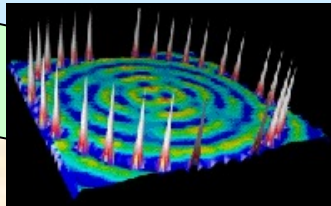
Nanoscience



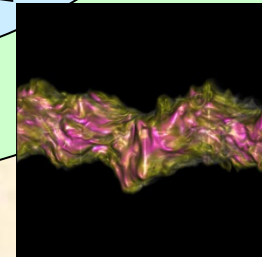
Climate research



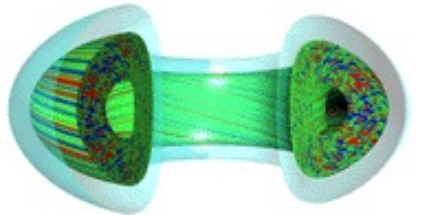
Computational materials



Flow dynamics



Fusion simulation



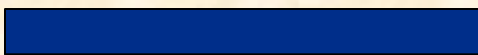
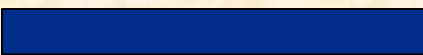



Neutron sciences

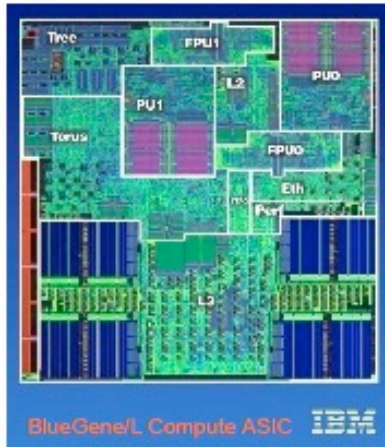


Why do we care about computing power?

Computer Security: Exhaustive Key Search

- **Two types of security**
 - Computational security
 - Unconditional security
- **Two types of encryption methods**
 - Conventional (a.k.a. single-key, secret-key, symmetric): DES/DEA
 - Public key-based (a.k.a. asymmetric): RSA, D-H
 - Either key could be used for encryption, but only the other key can be used for decryption
- **Attack on computational security**
 - Always possible to simply try every key
 - Most basic attack, proportional to key size
 - Assume to either know / recognize plaintext

Key Size (bits)	Number of Alternative Keys	Time required at 1 decryption/ μ s	Time required at 10^6 decryptions/ μ s
32	$2^{32} = 4.3 \times 10^9$		
56	$2^{56} = 7.2 \times 10^{16}$		
128	$2^{128} = 3.4 \times 10^{38}$		
168	$2^{168} = 3.7 \times 10^{50}$		
26 letters (permutation)	$26! = 4 \times 10^{26}$		



IBM BlueGene/L #1 212,992 Cores

Total of 26 systems all in the Top176

2.6 MWatts (2600 homes)

70,000 ops/s/person

(104 racks, 104x32x32)

212992 procs

Rack

(32 Node boards, 8x8x16)

2048 processors

Node Board

(32 chips, 4x4x2)

16 Compute Cards

64 processors

Compute Card

(2 chips, 2x1x1)

4 processors

Chip

(2 processors)

2.8/5.6 GF/s

4 MB (cache)

5.6/11.2 GF/s

1 GB DDR

90/180 GF/s

16 GB DDR

2.9/5.7 TF/s

0.5 TB DDR

180/360 TF/s

32 TB DDR

Full system total of
131,072 processors

The compute node ASICs include all networking and processor functionality. Each compute ASIC includes two 32-bit superscalar PowerPC 440 embedded cores (note that L1 cache coherence is not maintained between these cores). (20.7K sec about 5.7hours; n=2.5M)

"Fastest Computer"

BG/L 700 MHz 213K proc

104 racks

Peak: 596 Tflop/s

Linpack: 498 Tflop/s

84% of peak

Exascale Race/Technologies

Projected Exascale Dates and Suppliers

U.S.



- Sustained ES*: 2022-2023
- Peak ES: 2021
- Vendors: U.S.
- Processors: U.S. (some ARM?)
- Initiatives: NSCI/ECP
- Cost: \$600M per system, plus heavy R&D investments

EU



- PEAK ES: 2023-2024
- Pre-ES: 2021-2022
- Vendors: U.S., Europe
- Processors: Likely ARM
- Initiatives: EuroHPC
- Cost: \$300-\$350M per system, plus heavy R&D investments

China



- Sustained ES*: 2021-2022
- Peak ES: 2020
- Vendors: Chinese (multiple sites)
- Processors: Chinese (plus U.S.?)
- 13th 5-Year Plan
- Cost: \$350-\$500M per system, plus heavy R&D

Japan



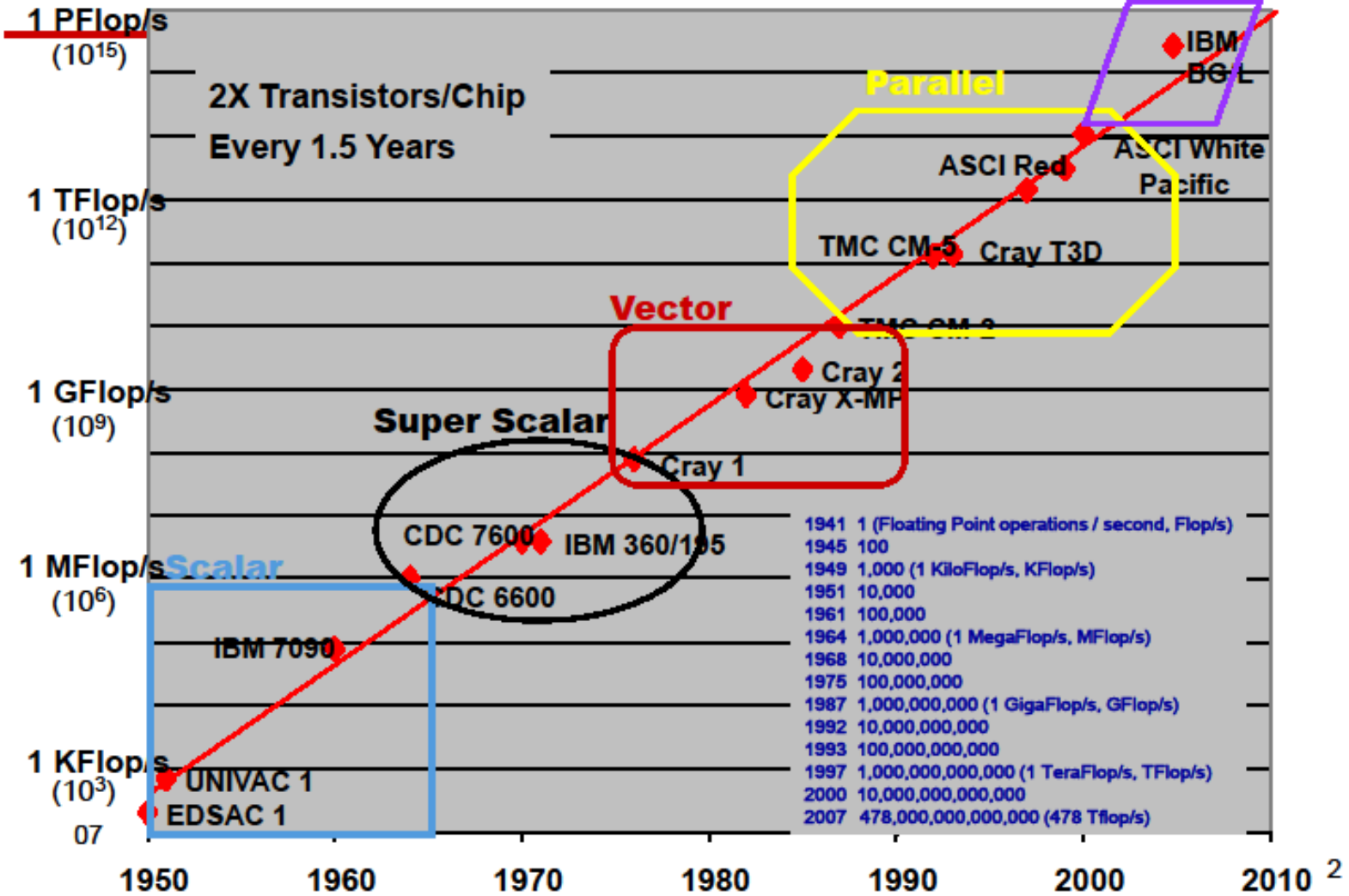
- Sustained ES*: ~2022
- Peak ES: Likely as a AI/ML/DL system
- Vendors: Japanese
- Processors: Japanese
- Cost: \$800M-\$1B, this includes both 1 system and the R&D costs, will also do many smaller size systems

** 1 exaflops on a 64-bit real application*

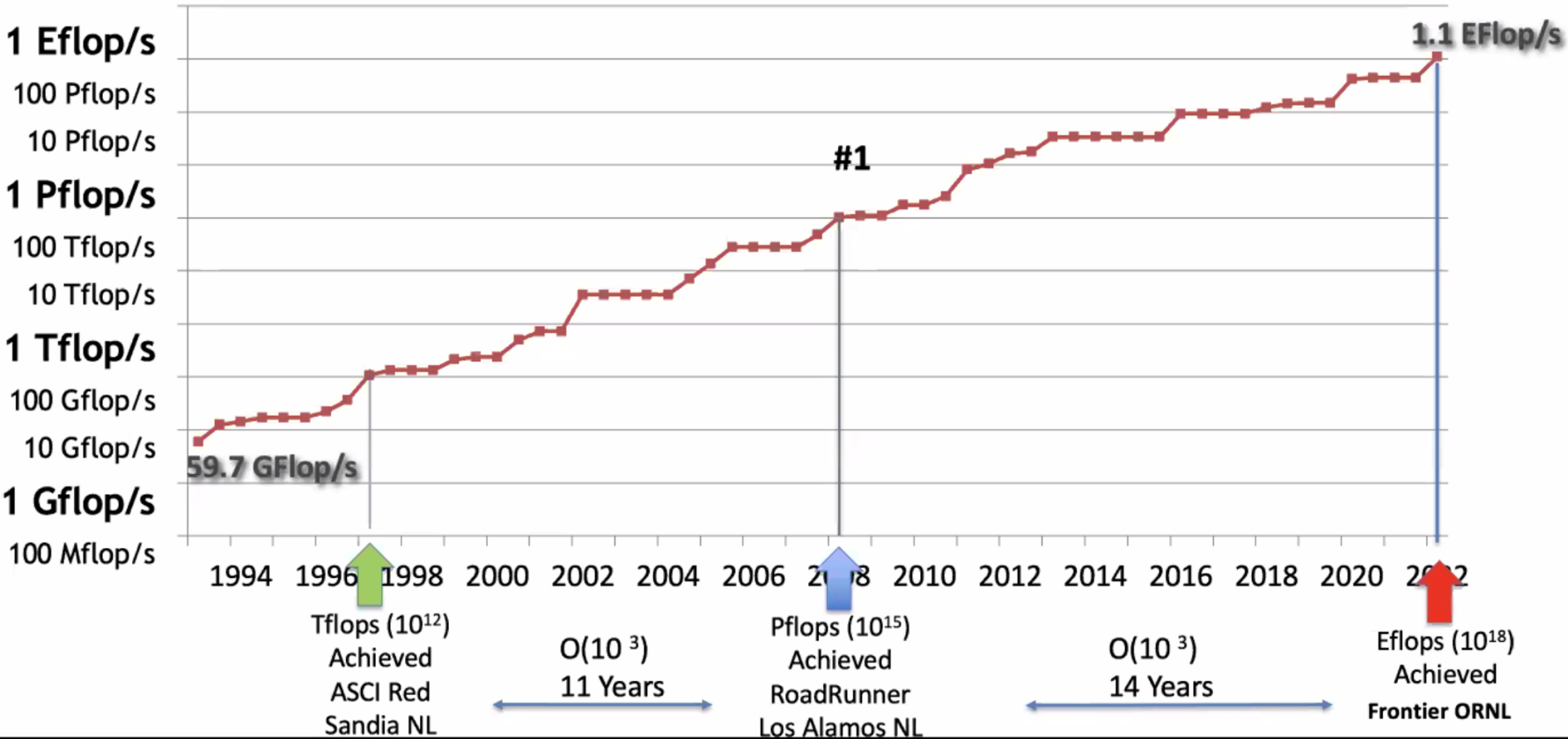


A Growth-Factor of a Billion in Performance in a Career

Super Scalar/Vector/Parallel



PERFORMANCE DEVELOPMENT



Top 500 June 2024 Release

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,699,904	1,206.00	1,714.81	22,786
2	Aurora - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11, Intel DOE/SC/Argonne National Laboratory United States	9,264,128	1,012.00	1,980.01	38,698
3	Eagle - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Azure Microsoft Azure United States	2,073,600	561.20	846.84	
4	Supercomputer Fugaku - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,630,848	442.01	537.21	29,899
5	LUMI - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE EuroHPC/CSC Finland	2,752,704	379.70	531.51	7,107

Current #1 System Overview

System Performance

- Peak performance of 2 Eflop/s for modeling & simulation

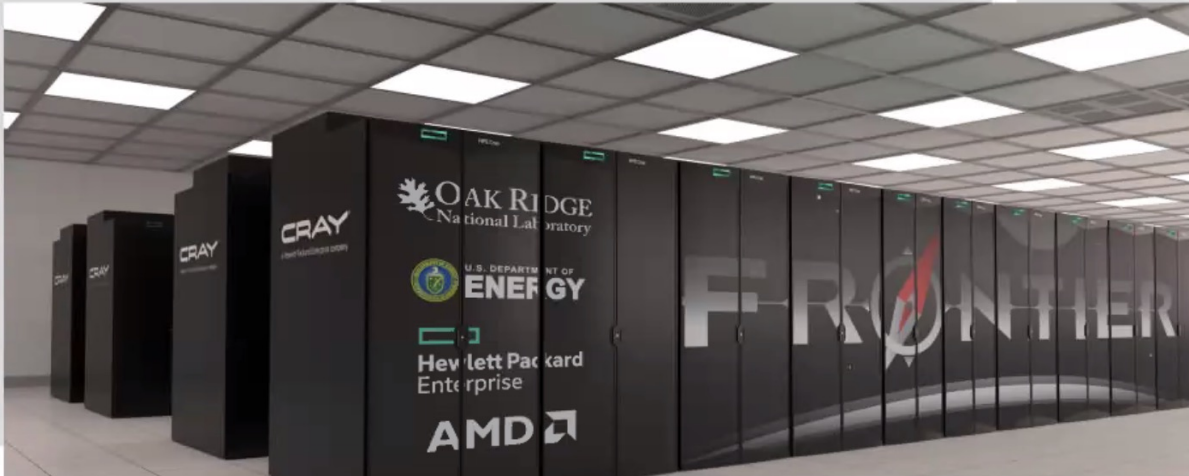
Peak performance of 2 Eflop/s for modeling & simulation

Each node has

- 1-AMD EPYC 7A53 CPU w/64 cores (2 Tflop/s)
- 4-AMD Instinct MI250X GPUs Each w/220 cores (4*53 Tflop/s)
- 730 GB of fast memory
- 2 TB of NVMe memory

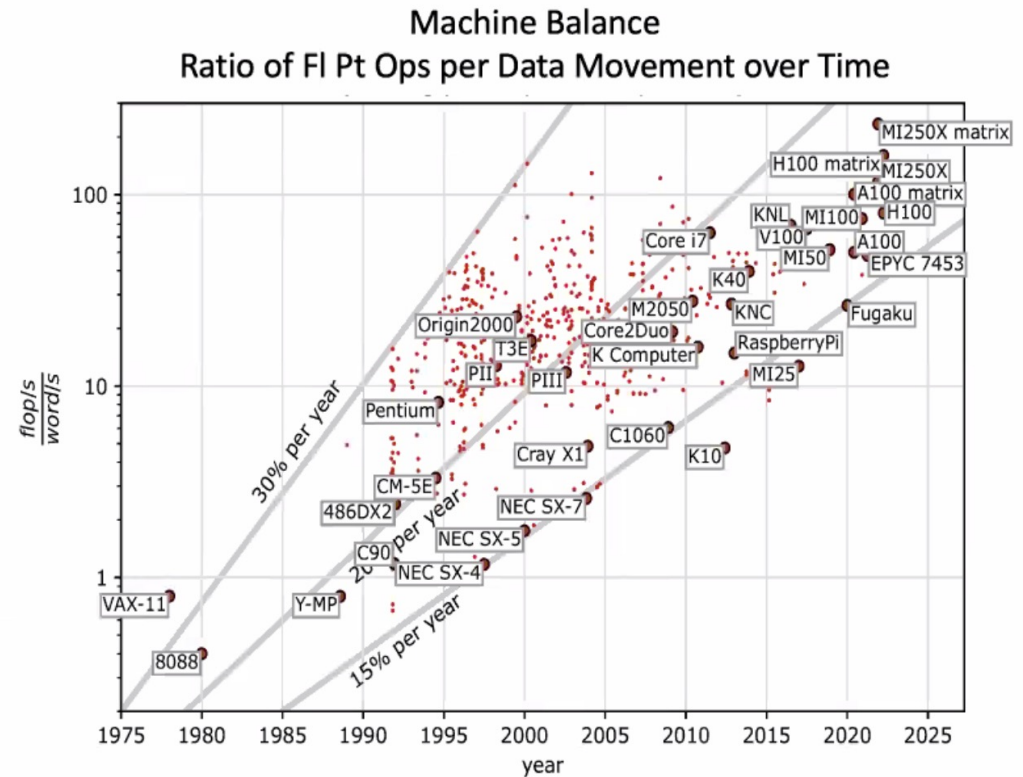
The system includes

- 9408 nodes
- Cray Slingshot interconnect
- 706 PB (695 PB Disk + 11 PB SSD)



When We Look at Performance in Numerical Computations ...

- Data movement has a big impact
- Performance comes from balancing floating point execution (**Flops/sec**) with memory->CPU transfer rate (**Words/sec**)
 - “Best” balance would be 1 flop per word-transferred
- Today’s systems are close to 100 flops/sec per word-transferred
 - Imbalanced: Over provisioned for Flops



Graph from Mark Gates

Plot for 64-bit floating point data movement & operations
(Bandwidth from CPU or GPU memory to registers)

Top 10 Challenges to Exascale

3 Hardware, 4 Software, 3 Algorithms/Math Related

- **Energy efficiency:**

- Creating more energy efficient circuit, power, and cooling technologies.

- **Interconnect technology:**

- Increasing the performance and energy efficiency of data movement.

- **Memory Technology:**

- Integrating advanced memory technologies to improve both capacity and bandwidth.

- **Scalable System Software:**

- Developing scalable system software that is power and resilience aware.

- **Programming systems:**

- Inventing new programming environments that express massive parallelism, data locality, and resilience

- **Data management:**

- Creating data management software that can handle the volume, velocity and diversity of data that is anticipated.

- **Scientific productivity:**

- Increasing the productivity of computational scientists with new software engineering tools and environments.

- **Exascale Algorithms:**

- Reformulating science problems and refactoring their solution algorithms for exascale systems.

- **Algorithms for discovery, design, and decision:**

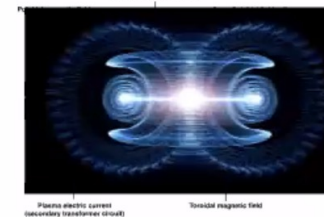
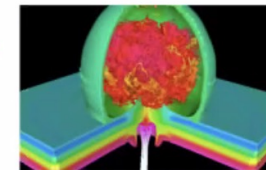
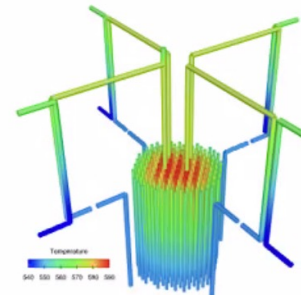
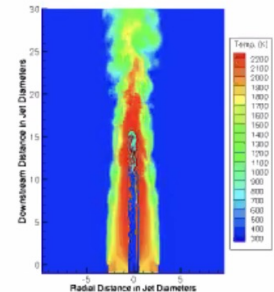
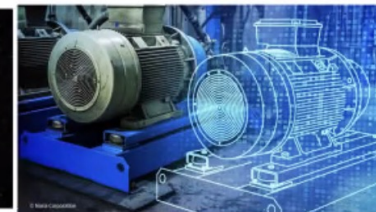
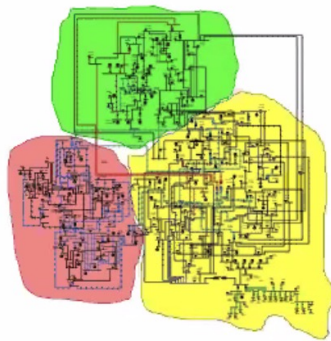
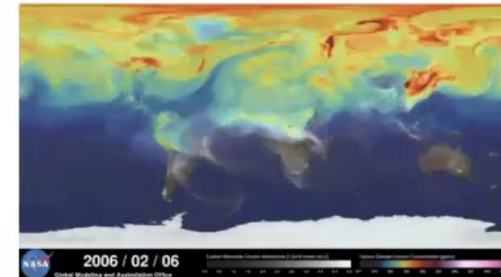
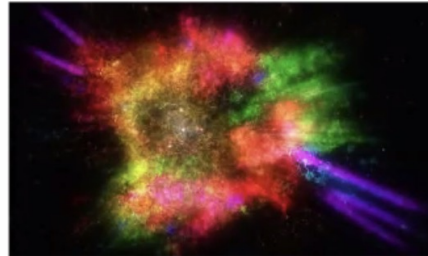
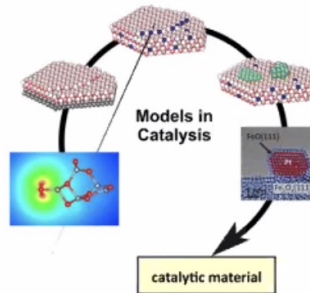
- Facilitating mathematical optimization and uncertainty quantification for exascale discovery, design, and decision making.

- **Resilience and correctness:**

- Ensuring correct scientific computation in face of faults, reproducibility, and algorithm verification challenges.

Today's Top HPC Systems Used to do Simulations

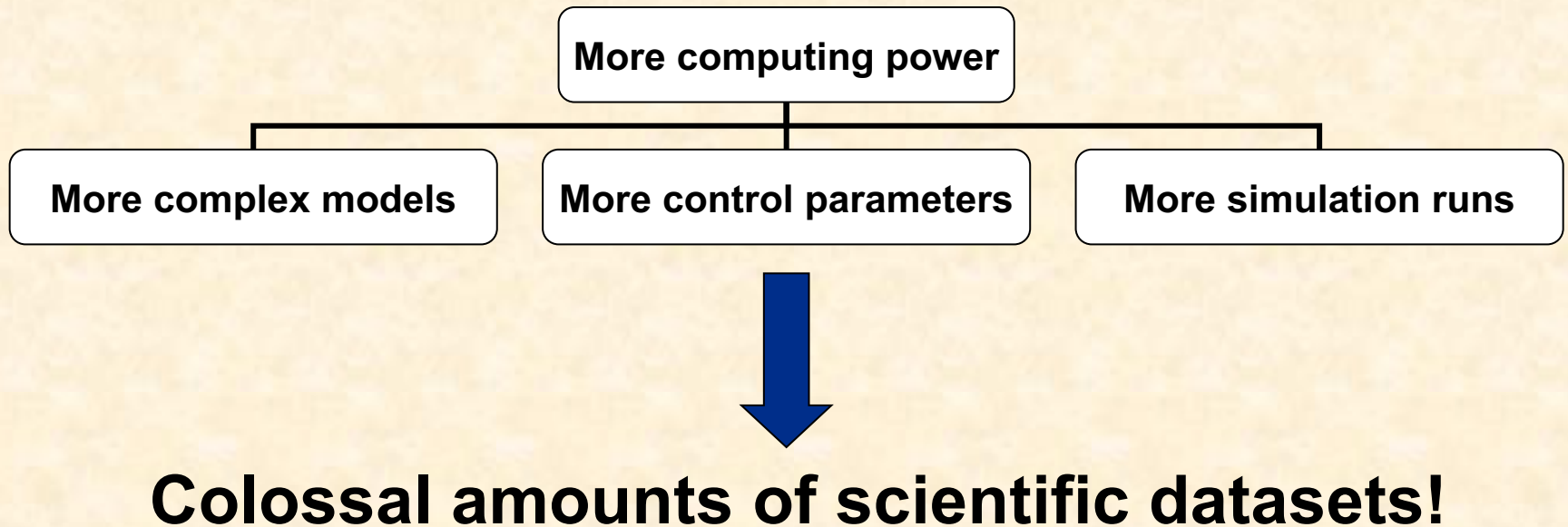
- *Climate*
- *Combustion*
- *Nuclear Reactors*
- *Catalysis*
- *Electric Grid*
- *Fusion*
- *Stockpile*
- *Supernovae*
- *Materials*
- *Digital Twins*
- *Accelerators*
- ...



- Usually 3-D PDE's
 - Sparse matrix computations, not dense

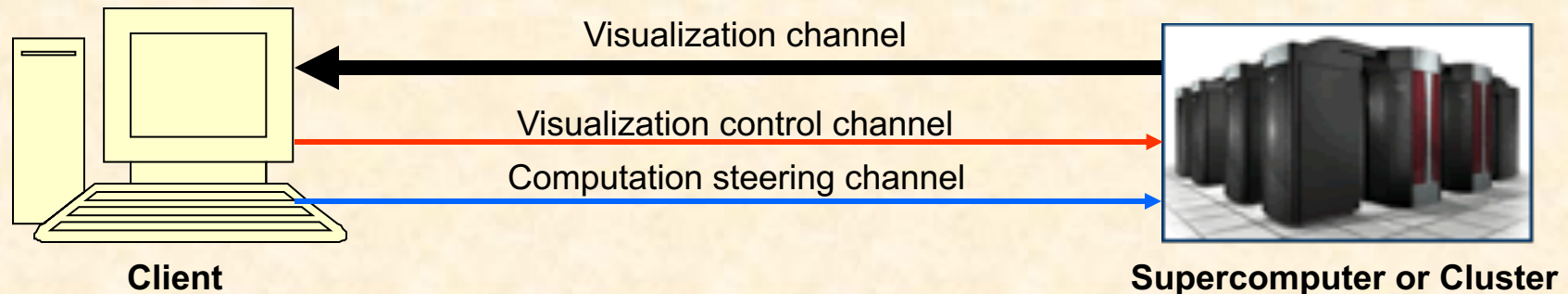
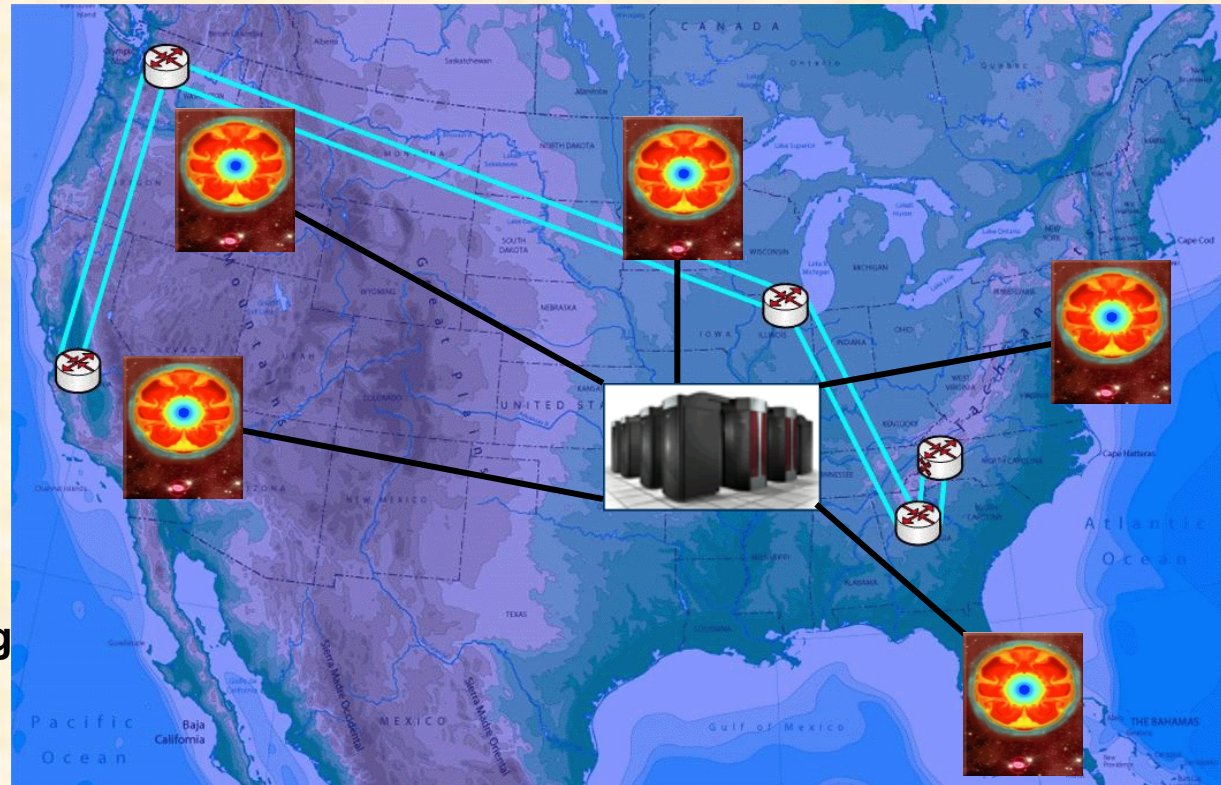
Computing power continues to increase over time

- Frontier: first ever reaching exascale supercomputing!

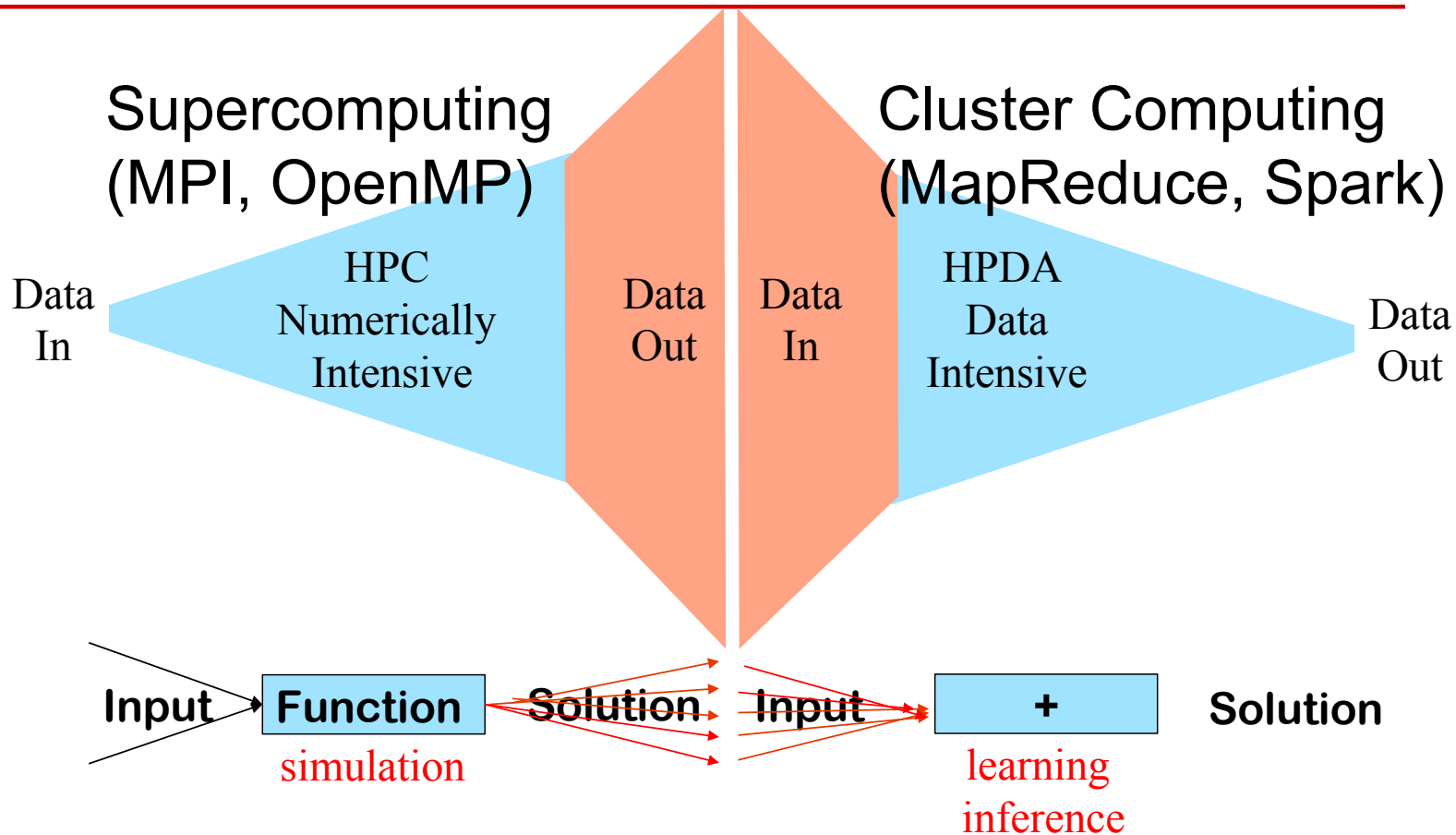


Terascale Supernova Initiative (TSI)

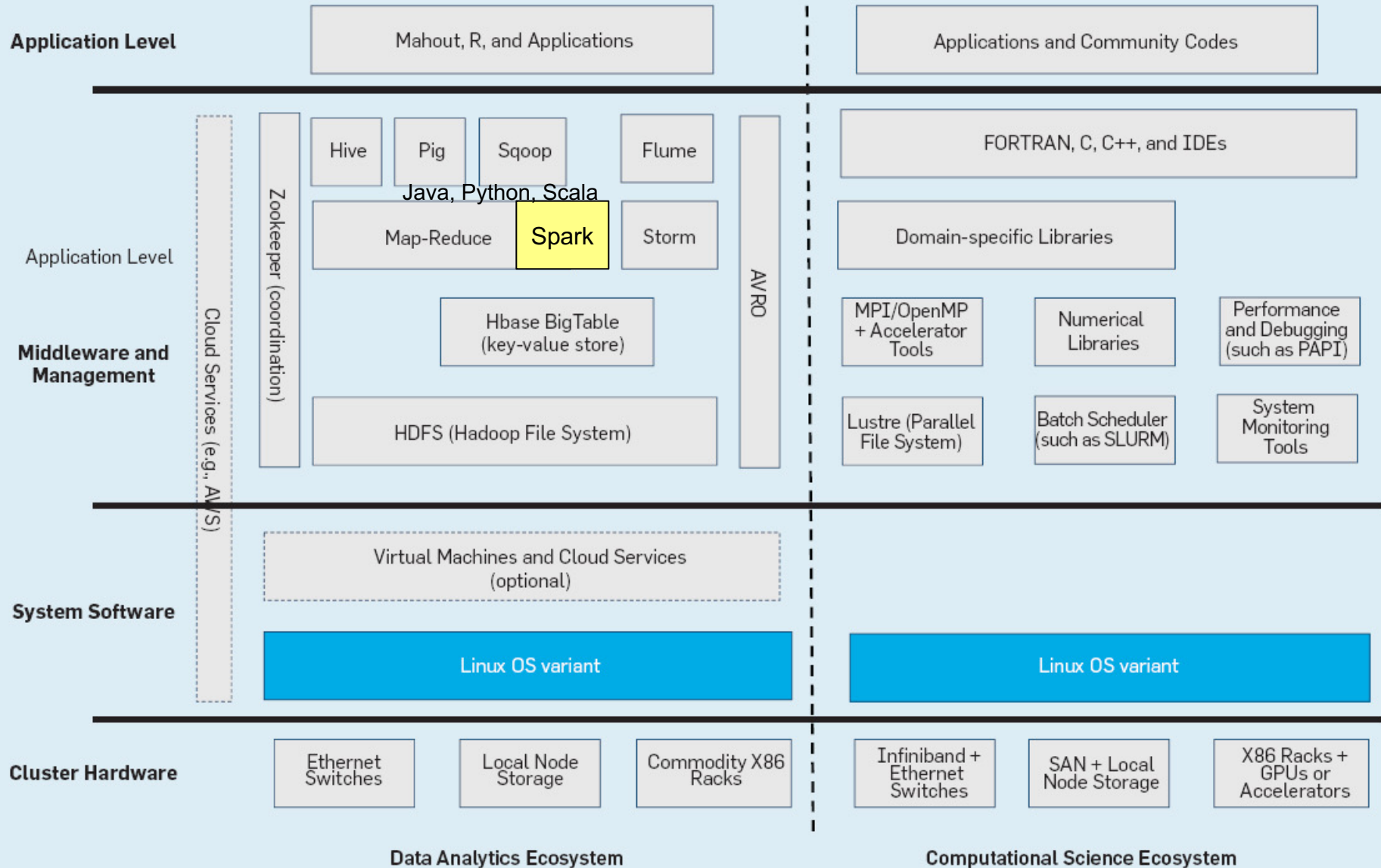
- Collaborative project
 - Supernova explosion
- TSI simulation
 - 1 terabyte a day with a small portion of parameters
 - From TSI to PSI to ESI
- Transfer to remote sites
 - Interactive distributed visualization
 - Collaborative data analysis
 - Computation monitoring
 - Computation steering



High Performance Computing

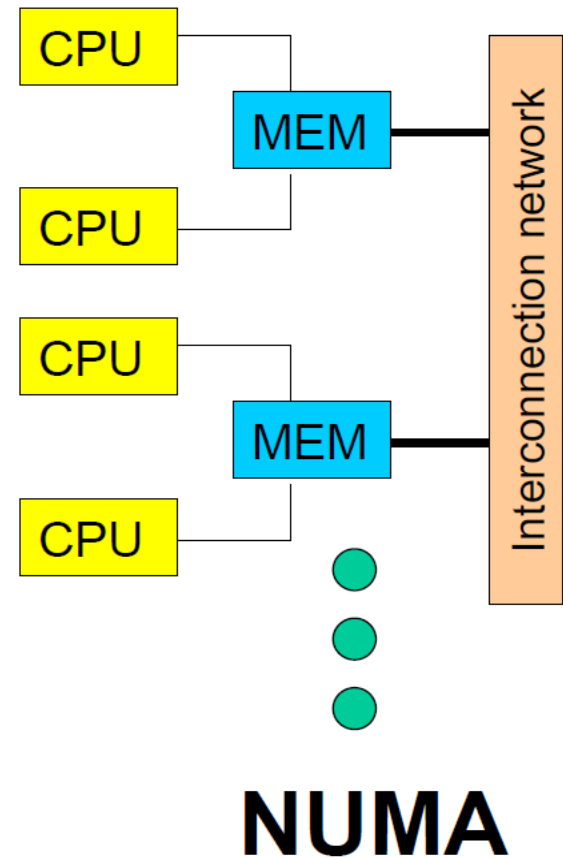


Comparison of Data Analytics and Computing Ecosystems



HPC Architectures

- **SMP – symmetric multiprocessing (up to 64CPUs)**
 - All processors access common memory on the same rights
 - Used in desktops
- **NUMA – nonuniform memory access**
 - Global address space (as in SMP)
 - Faster access to local memory
 - Slower to remote
- **Distributed memory multicomputers**
 - Communication via messages
- **Vector computers**
 - Multiple functional units performing the same operation on vector registers (very long ones)
 - E.g. vector addition, dot product
 - Almost disappeared
- **Distributed memory multiprocessors**
 - MPP: Massively Parallel Processing
 - Tightly integrated
- **Constellations**
 - Clusters of supercomputers



How are processors connected?

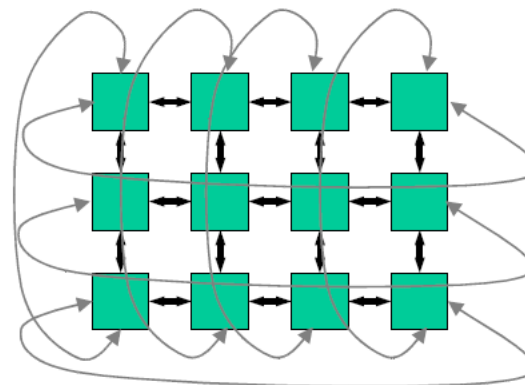
Network Topologies

- **Goal**

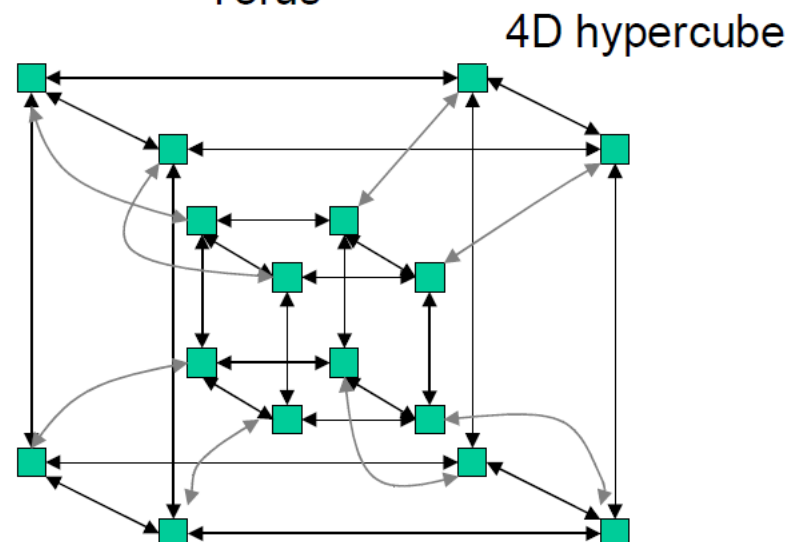
- Limited number of connections per node
- Small width
- Scalability

- **Topologies**

- Mesh
- Ring
- Torus
- Hypercube
- ...



Torus



4D hypercube

Type of Parallelization

Trivial:

- Each CPU does a part of the work independently.

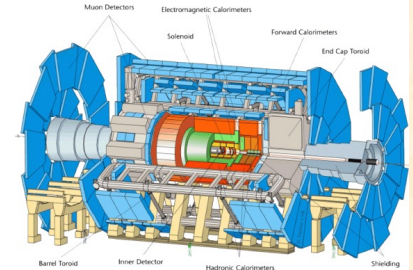
Nontrivial:

- Each CPU relies on its neighbor CPUs to complete the assigned work.

Parallel application example – HEP

- **Detector Simulation**
 - Simulate 10000 events with Geant4
 - One event – (e.g.) 1 minute
 - One week of computations!

Trivial parallelization – give parts of all the events to different CPUs



- **Event analysis**
 - Plot a quick histogram of 10000 events
 - One event – (e.g.) 0.1s
 - 20 minutes – This is not a quick histogram!

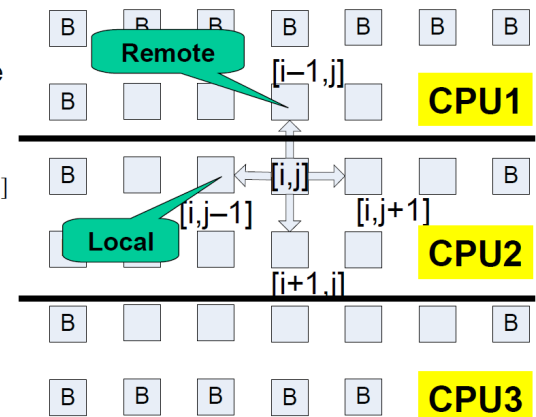
Trivial parallelization – each CPU analyzes a part of all the events. At the end histograms are added. But making it interactive and transparent is a challenge.

Nontrivial example – SOR

- **SOR – Successive overrelaxation**
 - Solution to the Laplace equation

$$A_{\text{new}}[i, j] = \frac{\alpha}{4} (A[i-1, j] + A[i+1, j] + A[i, j-1] + A[i, j+1]) + (1-\alpha)A[i, j]$$

Boundary rows have to be transferred between CPU's



Nontrivial but easy

Designing a parallel application

- **Partitioning**

- decompose data and computations into small tasks

- **Communication**

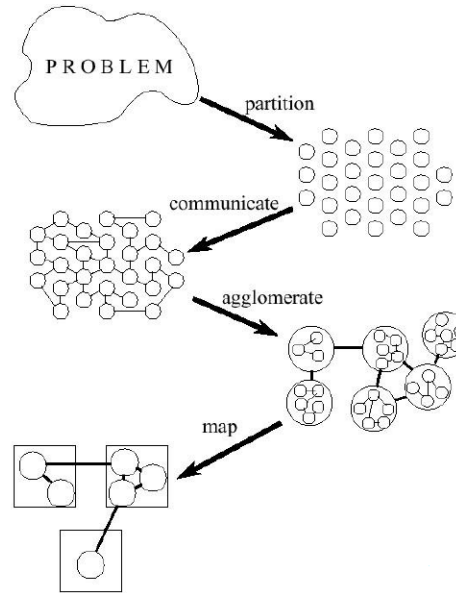
- Analyze communication required to coordinate tasks

- **Agglomeration – reduce communication**

- Increase granularity, improve locality

- **Mapping – map processors to tasks**

- Concurrent tasks on different CPUs
- Frequently communicating tasks on the same CPU



Performance Metrics

- **Execution time**

- Time when the last processor finishes its work

- **Speedup**

- $(\text{time on 1CPU}) / (\text{time on } P \text{ CPUs})$

- **Efficiency**

- $\text{Speedup} / P$

- **Minimize communication overhead**

- Data locality is important

- **Load balancing**

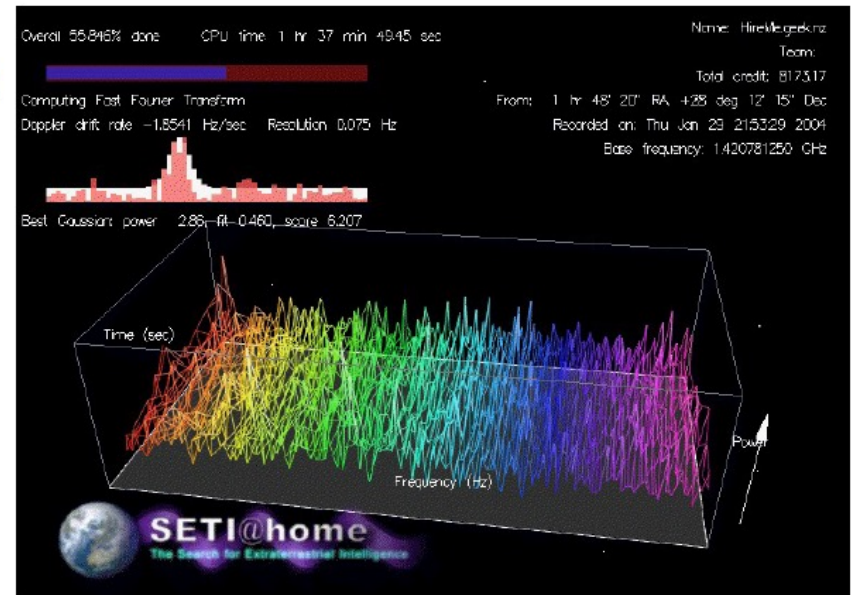
- Make sure processors are never idle
 - Dynamic load balancing: divide work at runtime

- **Take system architecture into account!**

- Fast local and slow remote memory for NUMA machines
- Hardware for broadcasting, reduction
- Faster access to neighboring nodes

Seti@home-like computing

- **Idle computers can be used**
 - Application running as a screen saver
- **Only computational-intensive application**
 - no communication while computing
- **Very successful – people are willing to share their computing power**
- **LHC@home – started recently**
 - Testing stability of the beam (60 particles 100k loops)



Grid Computing

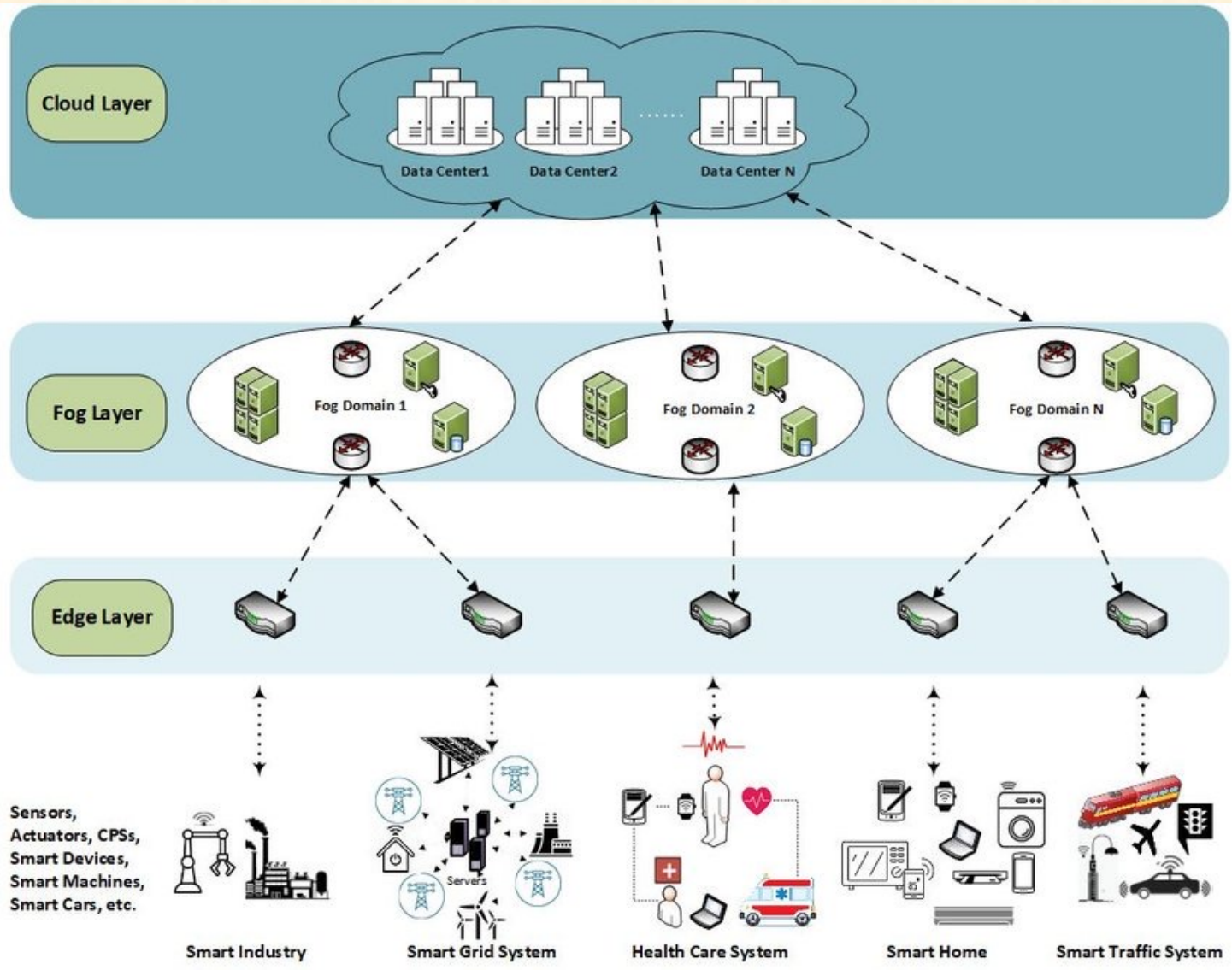
Grid Computing

- **Who needs grid computing?**
 - **Particular software capabilities**
 - Modelling, simulation, etc.
 - **High hardware/computing demands**
 - Processing, storage, etc.
 - **Large network bandwidth**
 - Circuit provisioning to support large data transfer
- **Problems, which are hard (or impossible) to solve at a single site, can be solved with the right kind of parallelization and distribution of the tasks involved.**
- **There are two primary types of grids**
 - **Computational grids**
 - Open Science Grid (OSG)
 - Worldwide LHC Computing Grid (WLCG)
 - **Data grids**
 - Earth System Grid (ESG)

Requirements

- **Computational grids**
 - Manage a variety of computing resources
 - Select computing resources capable of running a user's job
 - Predict loads on grid resources
 - Decide about resource availability
 - Dynamic resource configuration and provisioning
- **Data grids**
 - Provide data virtualization service
 - Support flexible data access, filtering, and transfer mechanisms
 - Provide security and privacy mechanisms
- **Grid computing environments are constructed upon three foundations**
 - Coordinated resources
 - Open standard protocols and frameworks
 - Non-trivial QoS

Computing Continuum: from Edge to Fog to Cloud



Cloud Computing

- **What is cloud computing?**
 - The phrase originated from the cloud symbol used to symbolize the Internet
 - A model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction
 - Provide computation, software, data access, and storage services that do not require end-user knowledge of the physical location and configuration of the system that delivers the services
- **Cloud architecture**
 - Involve multiple *cloud components* communicating with each other over application programming interfaces, usually web services and 3-tier architecture
 - Front end: seen by the user, such as a web browser
 - Back end: the cloud itself comprising computers, servers, data storage devices, etc.

Five Layers in Cloud Computing

Front and back ends:

- **Client**

- A *cloud client* consists of computer hardware and/or computer software that relies on cloud computing for application delivery, or that is specifically designed for delivery of cloud services

- **Server**

- The servers layer consists of computer hardware and/or computer software products that are specifically designed for the delivery of cloud services, including multi-core processors, cloud-specific operating systems and combined offerings

Three types of cloud computing services

- **Application**

- Cloud application services or "Software as a Service (SaaS)" deliver software as a service over the Internet, eliminating the need to install and run the application on the customer's own computers and simplifying maintenance and support

- **Platform**

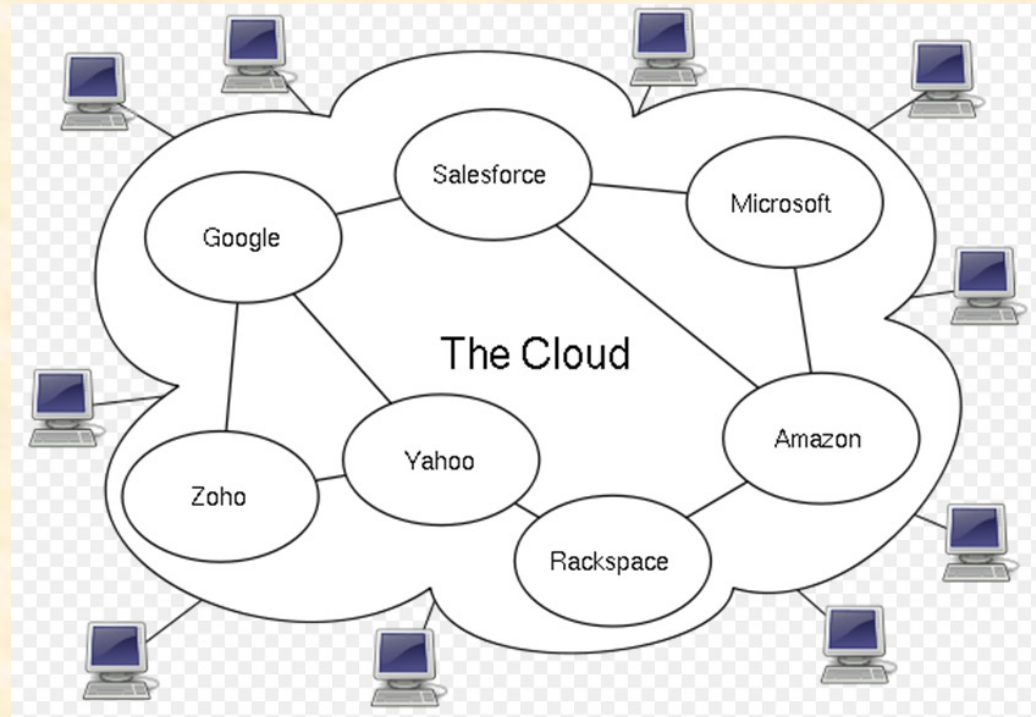
- Cloud platform services or "Platform as a Service (PaaS)" deliver a computing platform and/or solution stack as a service, often consuming cloud infrastructure and sustaining cloud applications

- **Infrastructure**

- Cloud infrastructure services, also known as "Infrastructure as a Service (IaaS)", delivers computer infrastructure – typically a platform virtualization environment – as a service

Real-life Cloud Computing Environments

- Microsoft Windows Azure
- Google Gov Cloud
- Amazon EC2
- Alibaba Cloud
- Baidu Cloud
- Eucalyptus (first open-source platform for private clouds, 2008)
 - A software platform for the implementation of private cloud computing on computer clusters
- Many others



Managing Amazon EC2 instances

- **AWS management console**
 - Web-based, most powerful
 - <http://aws.amazon.com/console/>
- **Command line tools**
 - http://aws.amazon.com/developertools/351?_encoding=UTF8&jiveRedirect=1
- **Third-party UI tools**
 - Example: ElasticFox browser add on

Login AWS management console



Services ▾ Edit ▾

jp238@njit.edu @ 934713874... ▾ Global ▾ Help ▾

Welcome

The AWS Management Console provides a graphical interface to Amazon Web Services. Learn more about how to use our services to meet your needs, or get started by selecting a service.

[Getting started guides](#)

[Reference architectures](#)

[Free Usage Tier](#)

Set Start Page

Console Home ▾



AWS Marketplace

Find & buy software, launch with 1-Click and pay by the hour.

Amazon Web Services

Compute & Networking

- Direct Connect**
Dedicated Network Connection to AWS
- EC2**
Virtual Servers in the Cloud
- Elastic MapReduce**
Managed Hadoop Framework
- Route 53**
Scalable Domain Name System
- VPC**
Isolated Cloud Resources

Storage & Content Delivery

- CloudFront**
Global Content Delivery Network
- Glacier**
Archive Storage in the Cloud
- S3**
Scalable Storage in the Cloud
- Storage Gateway**
Integrates on-premises IT environments with Cloud storage

Database

- DynamoDB**
Predictable and Scalable NoSQL Data Store
- ElastiCache**
In-Memory Cache
- RDS**
Managed Relational Database Service

Deployment & Management

- CloudFormation**
Templated AWS Resource Creation
- CloudWatch**
Resource & Application Monitoring
- Data Pipeline** NEW
Orchestration for data-driven workflows
- Elastic Beanstalk**
AWS Application Container
- IAM**
Secure AWS Access Control

App Services

- CloudSearch**
Managed Search Service
- Elastic Transcoder** NEW
Easy-to-use scalable media transcoding
- SES**
Email Sending Service
- SNS**
Push Notification Service
- SQS**
Message Queue Service
- SWF**
Workflow Service for Coordinating Application Components

Announcements

[Amazon Simple Workflow in Seven Additional Regions and Extends IAM Support](#)

[Announcing Amazon Elastic Transcoder](#)

[Announcing High Memory Cluster Instances for Amazon EC2](#)

[More...](#)

Service Health

[Edit](#)

Click [Edit](#) to add at least one service and at least one region to monitor.

[Service Health Dashboard](#)

Select AMI to create instance(s)

Create a New Instance Cancel X

Select an option below:

Classic Wizard

Launch an On-Demand or Spot instance using the classic wizard with fine-grained control over how it is launched.

Quick Launch Wizard

Launch an On-Demand instance using an editable, default configuration so that you can get started in the cloud as quickly as possible.

AWS Marketplace

AWS Marketplace is an online store where you can find and buy software that runs on AWS. Launch with 1-Click and pay by the hour.

Name Your Instance: Pick a meaningful name, e.g. Web Server

Choose a Key Pair:

Public/private key pairs allow you to securely connect to your instance after it launches.

Select Existing Create New None

Choose a Launch Configuration:



More Amazon Machine Images **NEW!**

Search through public and AWS Marketplace AMIs or choose from your own custom AMIs.



Amazon Linux AMI 2012.09

The Amazon Linux AMI 2012.09 is an EBS-backed, PV-GRUB image. It includes Linux 3.2, AWS tools, and repository access to multiple versions of MySQL, PostgreSQL, Python, Ruby, and Tomcat. **64 bit** **32 bit** Free tier eligible



Red Hat Enterprise Linux 6.3

Red Hat Enterprise Linux version 6.3, EBS-boot. **64 bit** **32 bit**



SUSE Linux Enterprise Server 11

SUSE Linux Enterprise Server 11 Service Pack 2 basic install, EBS boot with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, and Ruby 1.8.7 available. **64 bit** **32 bit**



Ubuntu Server 12.04.1 LTS

Ubuntu Server 12.04.1 LTS with support available from Canonical (<http://www.ubuntu.com/cloud/services>). **64 bit** **32 bit** Free tier eligible

Note: You can customize your settings in the next step.

Continue

[Submit Feedback](#) [Getting Started Guide](#)

Select instance type: <http://aws.amazon.com/ec2/instance-types/>

Assign instance to security group(s)

- A **security group** defines firewall rules for instances
- At launch time, instance can be assigned to multiple groups
 - Default group doesn't allow any network traffic
- Once an instance is running, it can't change to which security group(s) it belongs
- Can modify rules for a group at any time
- New rules automatically enforced for all running instances and instances launched in the future

Illustration of security group

The screenshot shows the AWS Management Console interface for EC2 Security Groups. The left sidebar contains navigation options like INSTANCES, IMAGES, ELASTIC BLOCK STORAGE, and NETWORK & SECURITY. The main content area shows a list of security groups, with 'test-cluster' selected. Below the list, the details for the 'test-cluster' security group are shown, including the 'Inbound' rules table.

Security Groups List:

	Name	VPC ID	Description
<input checked="" type="checkbox"/>	test-cluster		Group for Hadoop Slaves.
<input type="checkbox"/>	test-cluster-master		Group for Hadoop Master.
<input type="checkbox"/>	default		default group
<input type="checkbox"/>	mycluster1-master		Group for Hadoop Master.
<input type="checkbox"/>	mycluster1		Group for Hadoop Slaves.
<input type="checkbox"/>	quicklaunch-1		quicklaunch-1

Security Group: test-cluster - Inbound Rules:

ICMP	Port (Service)	Source	Action
	ALL	sg-7ff73516 (test-cluster-master)	Delete
	ALL	sg-73f7351a (test-cluster)	Delete
TCP	Port (Service)	Source	Action
	0 - 65535	sg-7ff73516 (test-cluster-master)	Delete
	0 - 65535	sg-73f7351a (test-cluster)	Delete

Launch instance



EC2 Dashboard

Events

INSTANCES

- Instances
- Spot Requests
- Reserved Instances

IMAGES

- AMIs
- Bundle Tasks

ELASTIC BLOCK STORAGE

- Volumes
- Snapshots

NETWORK & SECURITY

- Security Groups
- Elastic IPs
- Placement Groups
- Load Balancers
- Key Pairs
- Network Interfaces

Launch Instance Actions

Viewing: All Instances All Instance Types Search

1 to 2 of 2 Instances

<input type="checkbox"/>	Name	Instance	AMI ID	Root Device	Type	State	Status Checks	Alarm Status	Monitoring	Security Groups	Key Pairs
<input type="checkbox"/>	empty	i-9ccf74ec	ami-08ba6861	instance store	m1.small	terminated		none	basic	default	te
	empty	i-984df6e8	ami-1624987f	ebs	t1.micro	pending	initializing...	none	basic	quicklaunch-1	ec

No EC2 Instances selected.

Select an instance above

Instance IP addresses

- **Public IP and DNS**
 - Public addresses are reachable from the Internet
- **Private IP and DNS**
 - Private addresses are only reachable from within the Amazon EC2 network
- **Elastic IPs**
 - Static IP addresses associated with account, not specific instances
 - If instance using elastic IP fails, this address can be quickly remapped to another instance
 - DNS propagation may take a long time if remapping the name to another IP address

Checking instance IP/DNS

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with 'Services', 'Edit', and user information 'jp238@njit.edu @ 934713874...' and 'N. Virginia'. The left sidebar contains a navigation menu with categories like 'INSTANCES', 'IMAGES', 'ELASTIC BLOCK STORAGE', and 'NETWORK & SECURITY'. The main content area shows the 'Instances' page with a table of instances. One instance is selected, and its details are shown below. A blue box highlights the 'Network Interfaces' section, specifically the 'Public DNS', 'Private DNS', and 'Private IPs' fields.

Services Edit jp238@njit.edu @ 934713874... N. Virginia Help

EC2 Dashboard Events

Launch Instance Actions

Viewing: All Instances All Instance Types Search 1 to 2 of 2 Instances

Name	Instance	AMI ID	Root Device	Type	State	Status Checks	Alarm Status	Monitoring	Security Groups	Key Pairs
empty	i-9ccf74ec	ami-08ba6861	instance store	m1.small	terminated		none	basic	default	te...
empty	i-984df6e8	ami-1624987f	ebs	t1.micro	running	2/2 checks pa	none	basic	quicklaunch-1	ec...

Root Device Type: ebs **Tenancy:** default

IAM Role: - **Lifecycle:** normal

EBS Optimized: false

Block Devices: sda1

Network Interfaces:

Public DNS: ec2-23-20-83-139.compute-1.amazonaws.com

Private DNS: ip-10-196-102-134.ec2.internal

Private IPs: 10.196.102.134

Secondary Private IPs: --

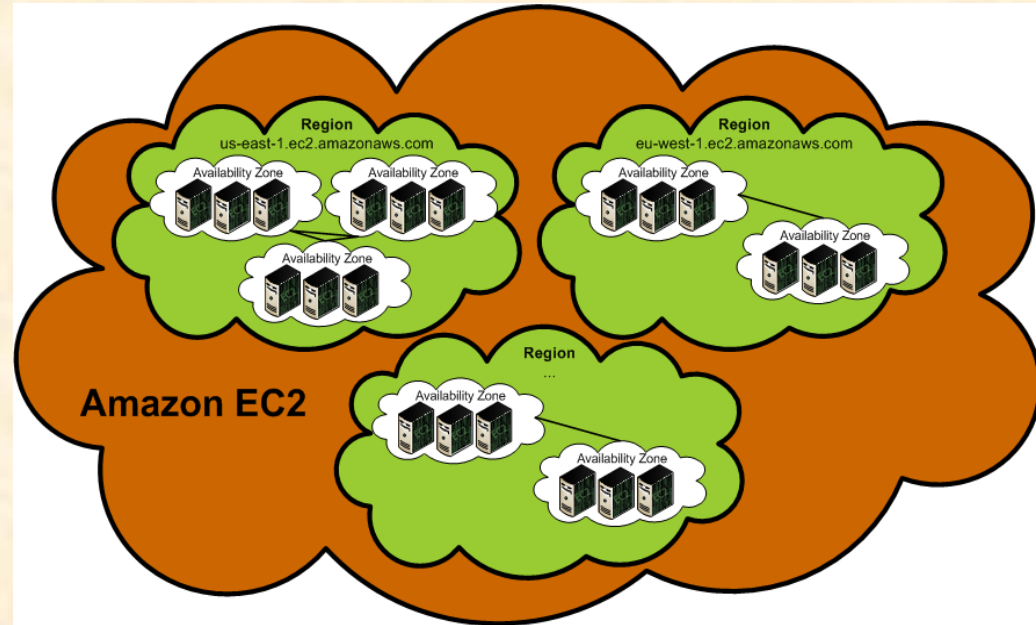
Launch Time: 2013-01-31 13:58 EST (less than an hour)

State Transition Reason:

Termination Protection: Disabled

Product Codes:

Regions and availability zones



- **Regions are located in separate geographic areas (US: Virginia & California, Ireland, Singapore, etc.)**
 - Each Region is completely isolated
 - Failure independence and stability
- **Availability Zones are distinct locations within a Region**
 - Isolated, but connected through low-latency links
 - Failure resilience
- **Current pricing: <http://aws.amazon.com/ec2/pricing/>**

Connecting to instances

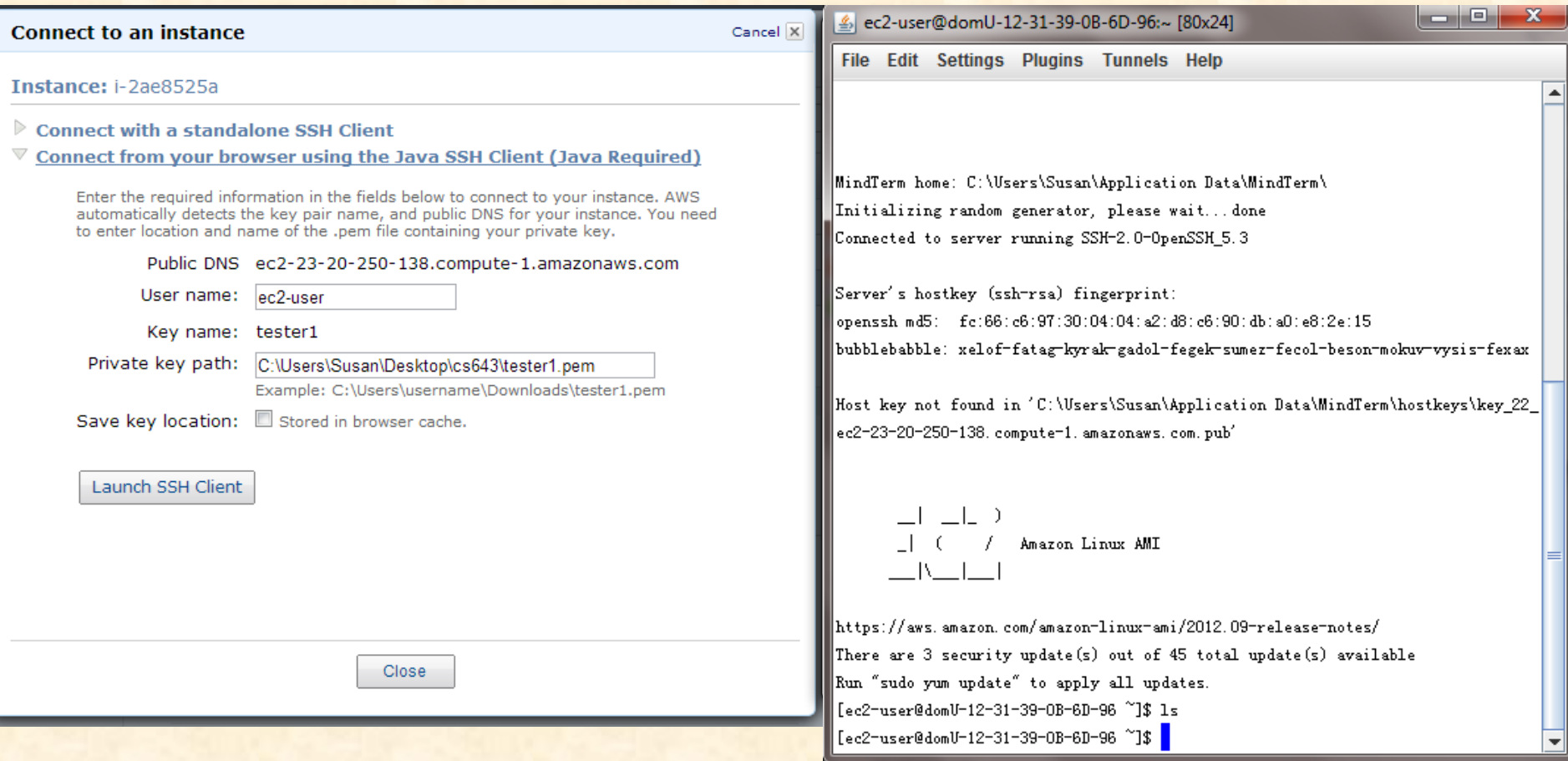
- **Windows instance**

- Connect from your browser using the Java SSH Client
- Connect using Remote Desktop

- **Linux Instance**

- Connect from your browser using the Java SSH Client
- Putty.exe/putty key generator
- SSH command on linux machine
 - `ssh -i key.pem ec2-user@ec2-23-20-83-139.compute-1.amazonaws.com`

Connect to Linux instance



The image shows two windows side-by-side. The left window is titled "Connect to an instance" and contains a form for connecting to an AWS instance. The right window is a terminal window titled "ec2-user@domU-12-31-39-08-6D-96:~ [80x24]" showing the output of the connection process.

Connect to an instance (Cancel)

Instance: i-2ae8525a

- Connect with a standalone SSH Client
- Connect from your browser using the Java SSH Client (Java Required)

Enter the required information in the fields below to connect to your instance. AWS automatically detects the key pair name, and public DNS for your instance. You need to enter location and name of the .pem file containing your private key.

Public DNS: ec2-23-20-250-138.compute-1.amazonaws.com

User name:

Key name: tester1

Private key path:
Example: C:\Users\username\Downloads\tester1.pem

Save key location: Stored in browser cache.

Terminal Window: ec2-user@domU-12-31-39-08-6D-96:~ [80x24]

```
File Edit Settings Plugins Tunnels Help

MindTerm home: C:\Users\Susan\Application Data\MindTerm\
Initializing random generator, please wait... done
Connected to server running SSH-2.0-OpenSSH_5.3

Server's hostkey (ssh-rsa) fingerprint:
openssh md5: fc:66:c6:97:30:04:04:a2:d8:c6:90:db:a0:e8:2e:15
bubblebabble: xelof-fatag-kyrak-gadol-fegek-sumer-fecol-beson-mokuv-vysis-fexax

Host key not found in 'C:\Users\Susan\Application Data\MindTerm\hostkeys\key_22_
ec2-23-20-250-138.compute-1.amazonaws.com.pub'

  _| _| )
  _| ( /  Amazon Linux AMI
  _|\_|_|

https://aws.amazon.com/amazon-linux-ami/2012.09-release-notes/
There are 3 security update(s) out of 45 total update(s) available
Run "sudo yum update" to apply all updates.
[ec2-user@domU-12-31-39-08-6D-96 ~]$ ls
[ec2-user@domU-12-31-39-08-6D-96 ~]$
```

Connect to Windows instance: get administrator password

Console Connect - Remote Desktop Connection Cancel X

Instance: i-34e55f44 **Public DNS:** ec2-54-242-92-134.compute-1.amazonaws.com

▼ [Log in with your credentials](#)

Log in to your instance with your credentials:

Public DNS: ec2-54-242-92-134.compute-1.amazonaws.com
Username: Administrator
Password: [Retrieve Password](#) *Click if you do not know your password.

You can download an RDP file for this instance which will launch Remote Desktop Connection and connect to your instance. You will need to note down your password because the Remote Desktop Connection software will open in a new window.

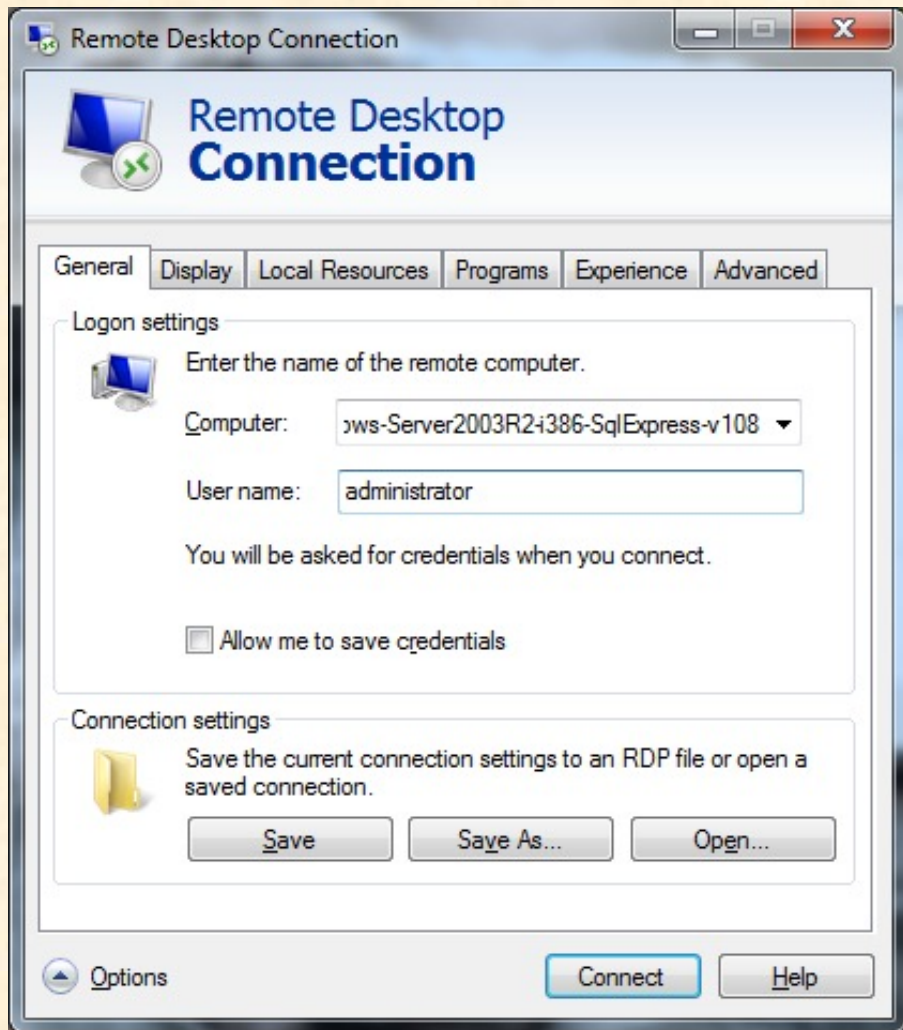
[Download shortcut file](#)

If you need help configuring your remote desktop software, click [here](#).

▶ [Retrieve Windows Administrator password](#)
▶ [Need help configuring your remote access software?](#)

Close

Connect to Windows instance



- Type in public DNS name of instance
- Use the retrieved administrator password

IAM – Identity Access Management

- **“AWS Identity and Access Management (IAM) helps to securely control access to AWS resources**
- **IAM is used to control who can use your AWS resources (*authentication*) and what resources they can use and in what ways (*authorization*).”**

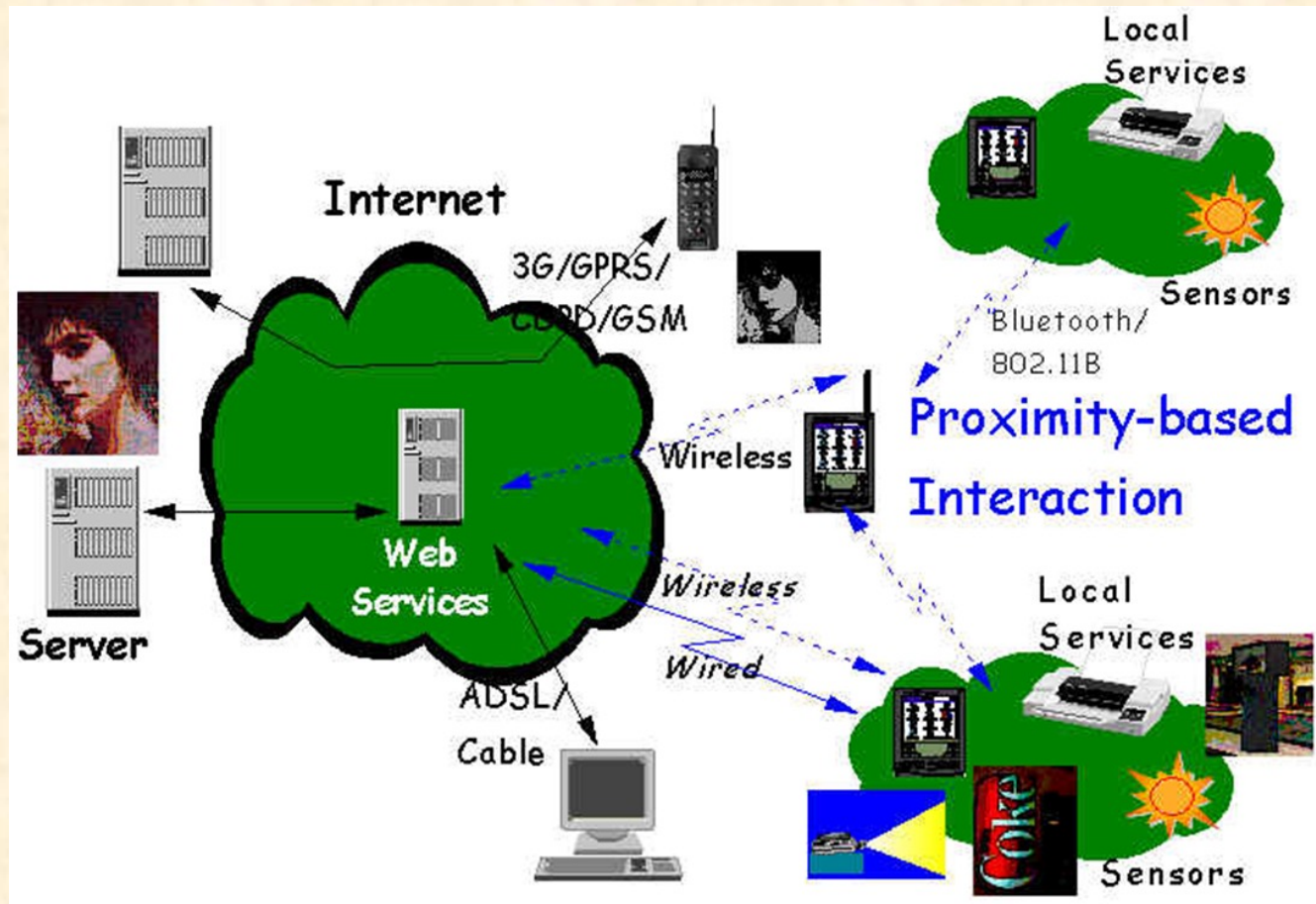
Mobile and Ubiquitous Computing



Mobile and Ubiquitous Computing



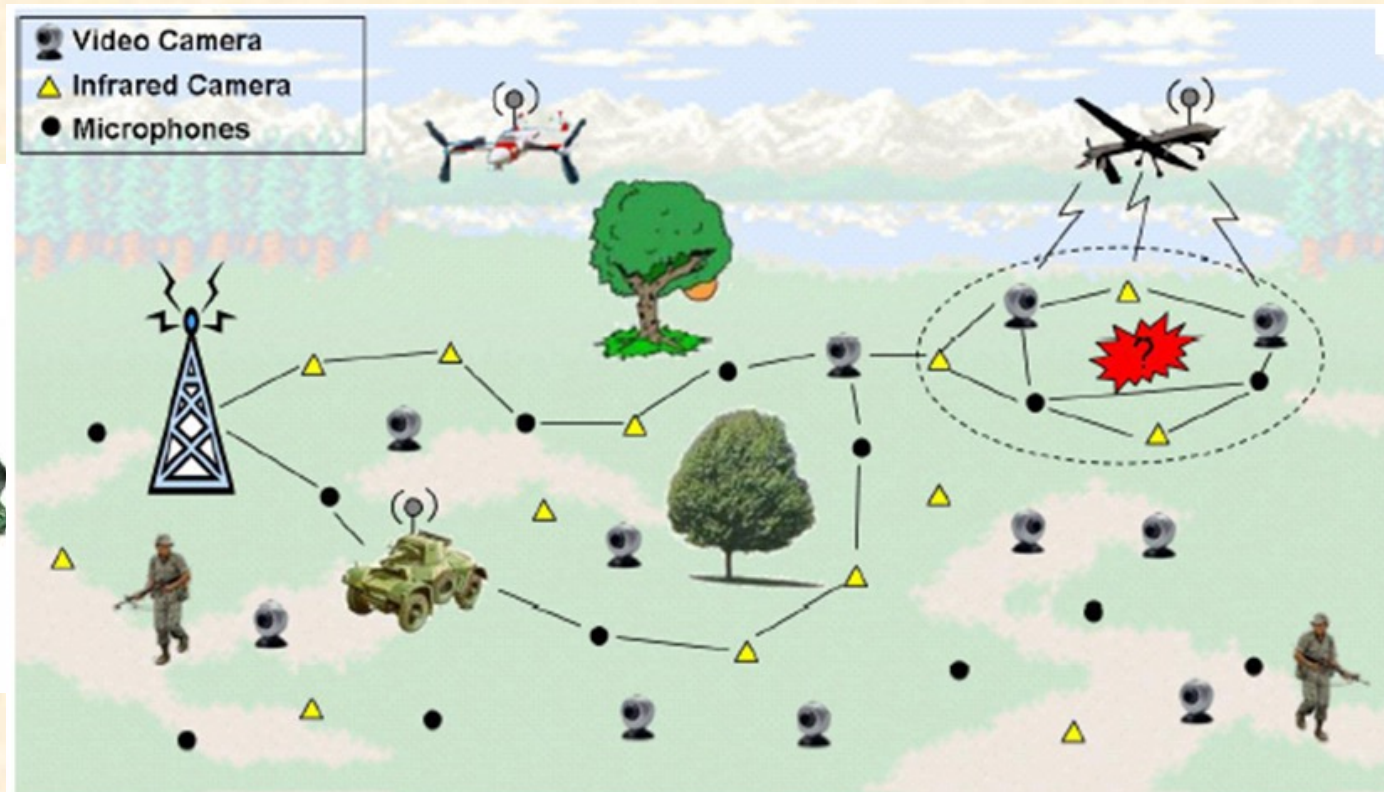
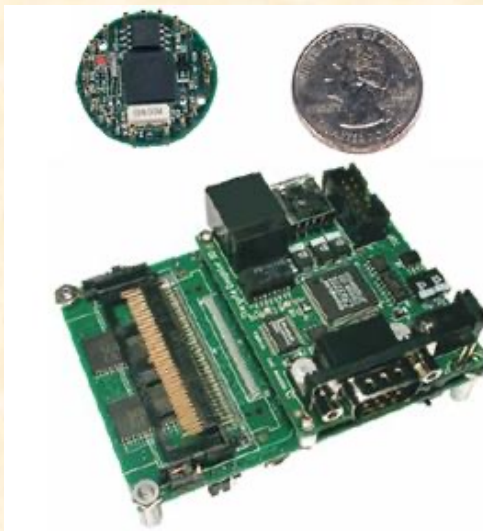
Mobile and Ubiquitous Computing Internet of Things (IoT)



- GPRS: General Packet Radio Service
- WTP: Wireless Transport Protocol
- WAP: Wireless Application Protocol with 2 components: WML and WMLScript
- WML: Wireless Markup Language
- cHTML: Compact HTML
- HDML: Handheld Device Markup Language

Wireless Sensor Networks

- **Pervasive applications**
 - **Agricultural**
 - **Military**
 - **Industrial**



Ongoing Research in Our Big Data Group

- **Data-intensive computing**
 - Big data ecosystem
 - AI and ML
 - Scientific Workflow optimization
 - Mapping, scheduling, modeling
- **High-performance networking**
 - Bandwidth scheduling
 - Transport control
 - Control plane design
- **Distributed sensor networks**
 - Deployment, routing, fusion
- **Cyber security**
 - Monitoring, game theory
- **Visualization and image processing**



Thanks! ☺

Questions ?