Data Intensive Scalable Computing

Its Role in Scientific Research

Randal E. Bryant
Carnegie Mellon University

http://www.cs.cmu.edu/~bryant

Examples of Big Data Sources

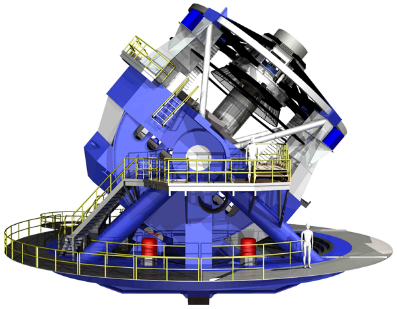
Google / Yahoo! / Microsoft

- Search over > 20 B web pages
- Also maps, images, videos, …

Wal-Mart

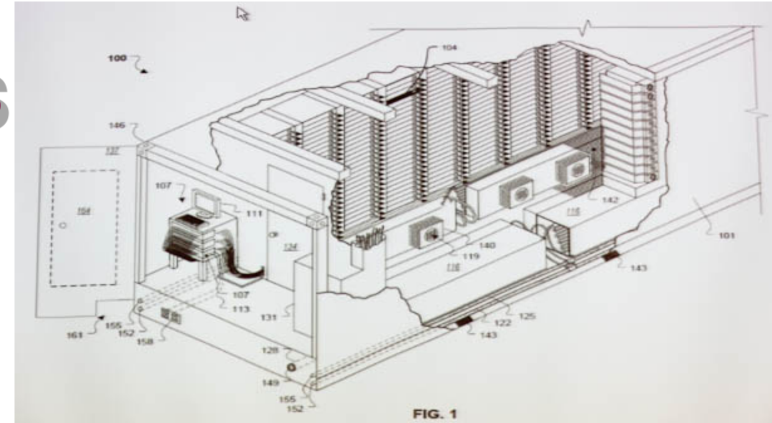
- 267 million items/day, sold at 6,000 stores
- HP built them 4 PB data warehouse
- Mine data to manage supply chain, understand market trends, formulate pricing strategies

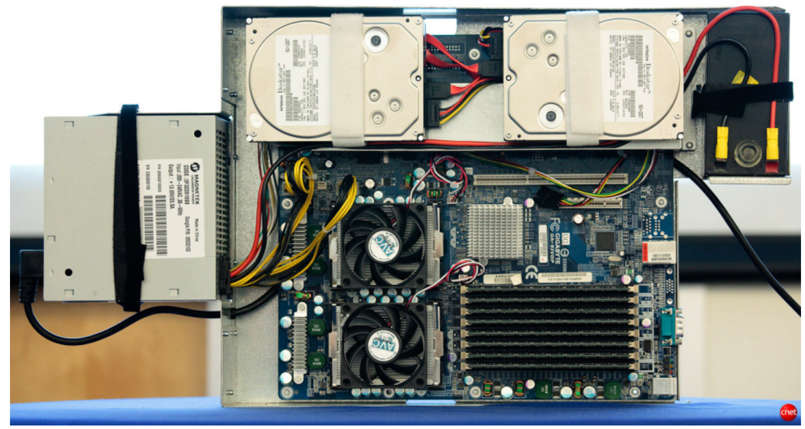
LSST



- Chilean telescope will scan entire sky every 3 days
- Generate 30 TB/day of image data

Google Data Centers





Dalles, Oregon

- Hydroelectric power @ 2¢ / KWH
- 50 Megawatts
  - Enough to power 6,000 homes

- Engineered for maximum modularity & power efficiency
- Container: 1160 servers, 250KW
- Server: 2 disks, 2 processors

Why So Much Data?

We Can Get It

- Automation + Internet

We Can Keep It

- Seagate Barracuda
- 1.5 TB @ $110 (7.3¢ / GB)

We Can Use It

- Scientific breakthroughs
- Business process efficiencies
- Realistic special effects
- Better health care

Could We Do More?

- Apply more computing power to this data

Seagate Barracuda 7200.11 1.5 TB 7200RPM SATA 3Gb/s 32MB Cache 3.5 Inch Internal Hard Drive ST31500341AS-Bare Drive (Amazon Frustration-Free Packaging)

Other products by Seagate

(277 customer reviews) | More about this product

Size Name:

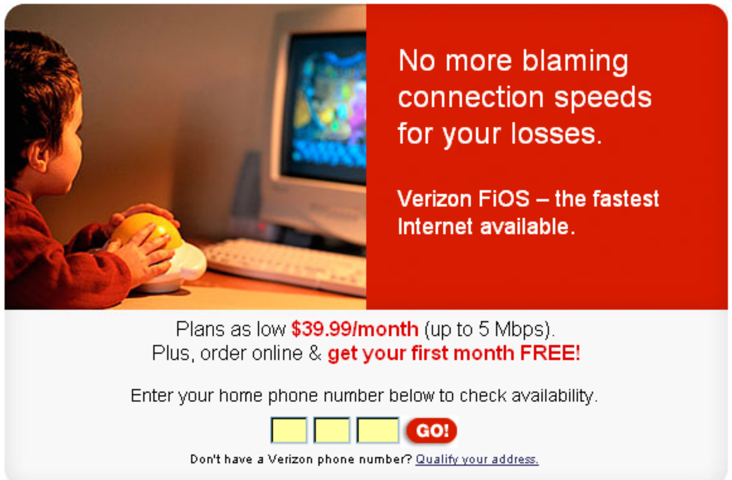
1.5 TB: $109.99

List Price: ~~$199.99~~

Price: $109.99 Free Standard Shipping (3-5 days)
Details

You Save: $90.00 (45%)
Special Offers Available

Oceans of Data, Skinny Pipes



Seagate

1 Terabyte

- Easy to store
- Hard to move

Disks	MB / s	Time
Seagate Barracuda	115	2.3 hours
Seagate Cheetah	125	2.2 hours
Networks	**MB / s**	**Time**
Home Internet	< 0.625	> 18.5 days
Gigabit Ethernet	< 125	> 2.2 hours
PSC Teragrid Connection	< 3,750	> 4.4 minutes

Data-Intensive System Challenge

For Computation That Accesses 1 TB in 5 minutes

- **Data distributed over 100+ disks**
  - Assuming uniform data partitioning
- **Compute using 100+ processors**
- **Connected by gigabit Ethernet (or equivalent)**

System Requirements

- **Lots of disks**
- **Lots of processors**
- **Located in close proximity**
  - Within reach of fast, local-area network

Is This Cloud Computing?

"I don't want to be a system administrator. You handle my data & applications."

- Hosted services
- Documents, web-based email, etc.
- Can access from anywhere
- Easy sharing and collaboration

"I've got terabytes of data. Tell me what they mean."

- Very large, shared data repository
- Complex analysis
- *Data-intensive scalable computing* (DISC)

Desiderata for DISC Systems

Focus on Data

- Terabytes, not tera-FLOPS

Problem-Centric Programming

- Platform-independent expression of data parallelism

Interactive Access

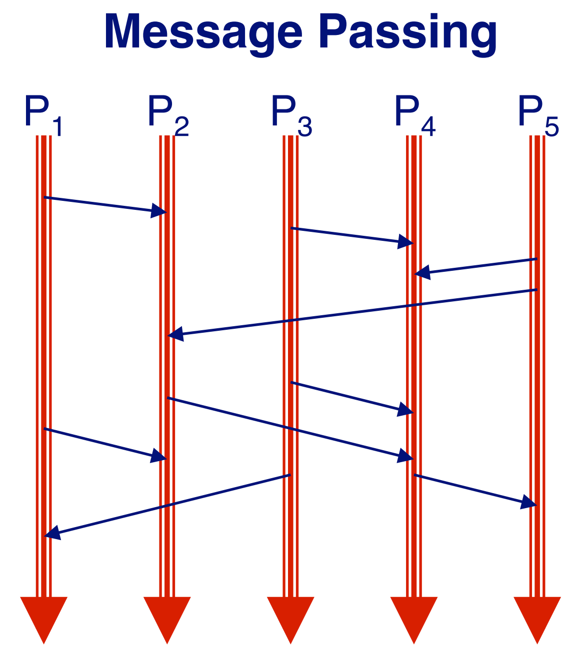
- From simple queries to massive computations

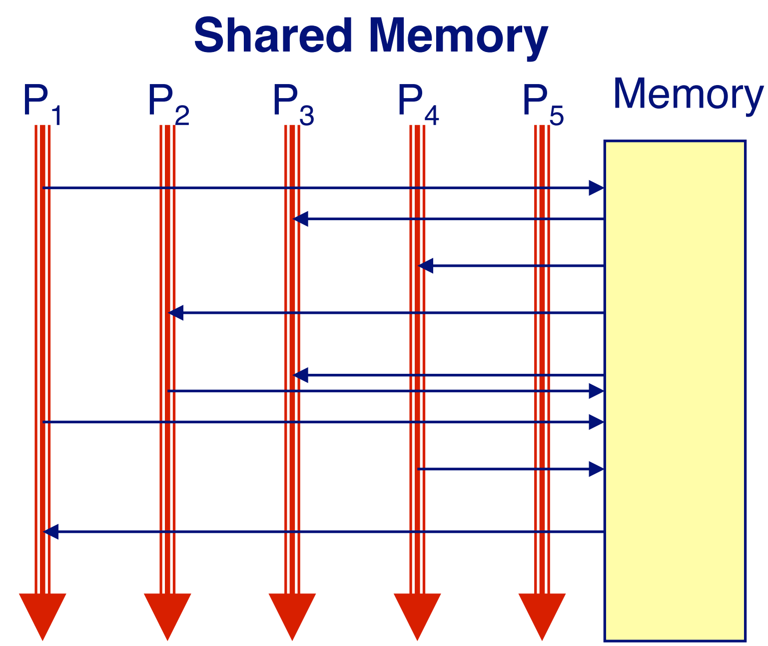
Robust Fault Tolerance

- Component failures are handled as routine events

Contrast to existing High Performance Computing (HPC) systems

Typical HPC Operation





Characteristics

- Long-lived processes
- Make use of spatial locality
- Hold all program data in memory (no disk access)
- High bandwidth communication

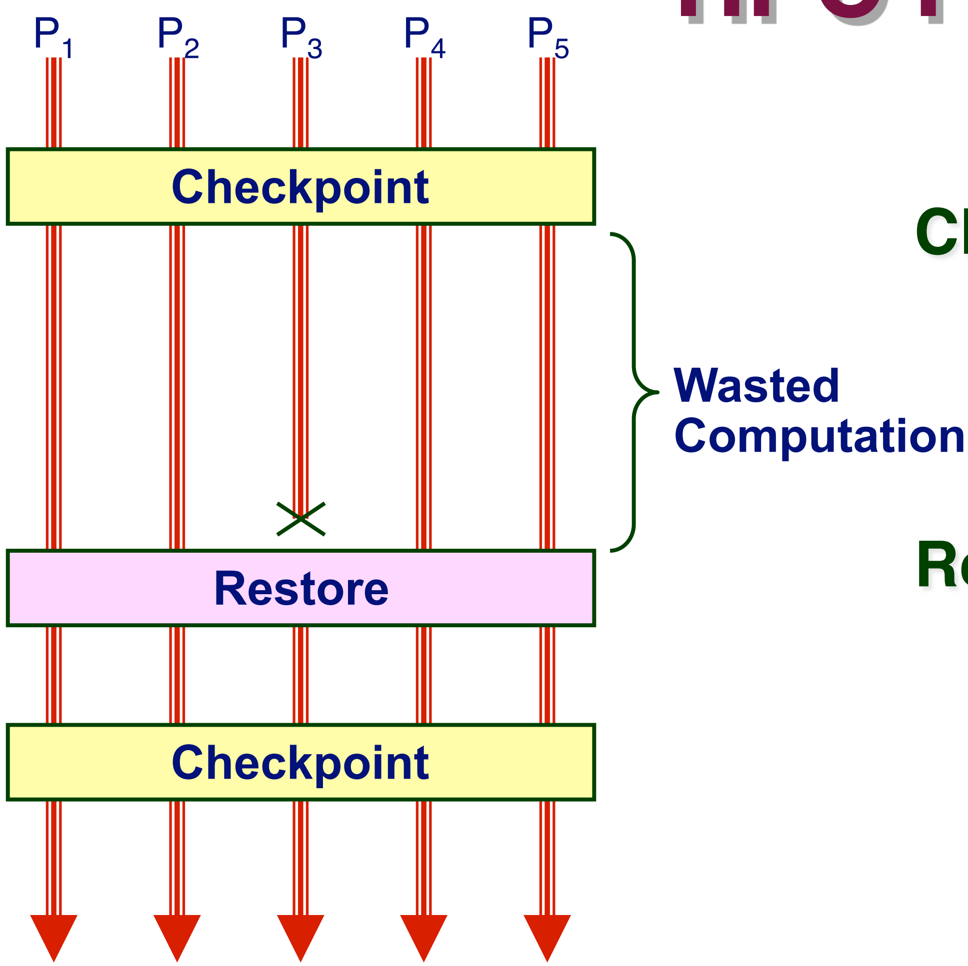
Strengths

- High utilization of resources
- Effective for computationally-intensive applications

Weaknesses

- Requires careful tuning of application to resources
- Intolerant of any variability

HPC Fault Tolerance



Checkpoint

- Periodically store state of all processes
- Significant I/O traffic

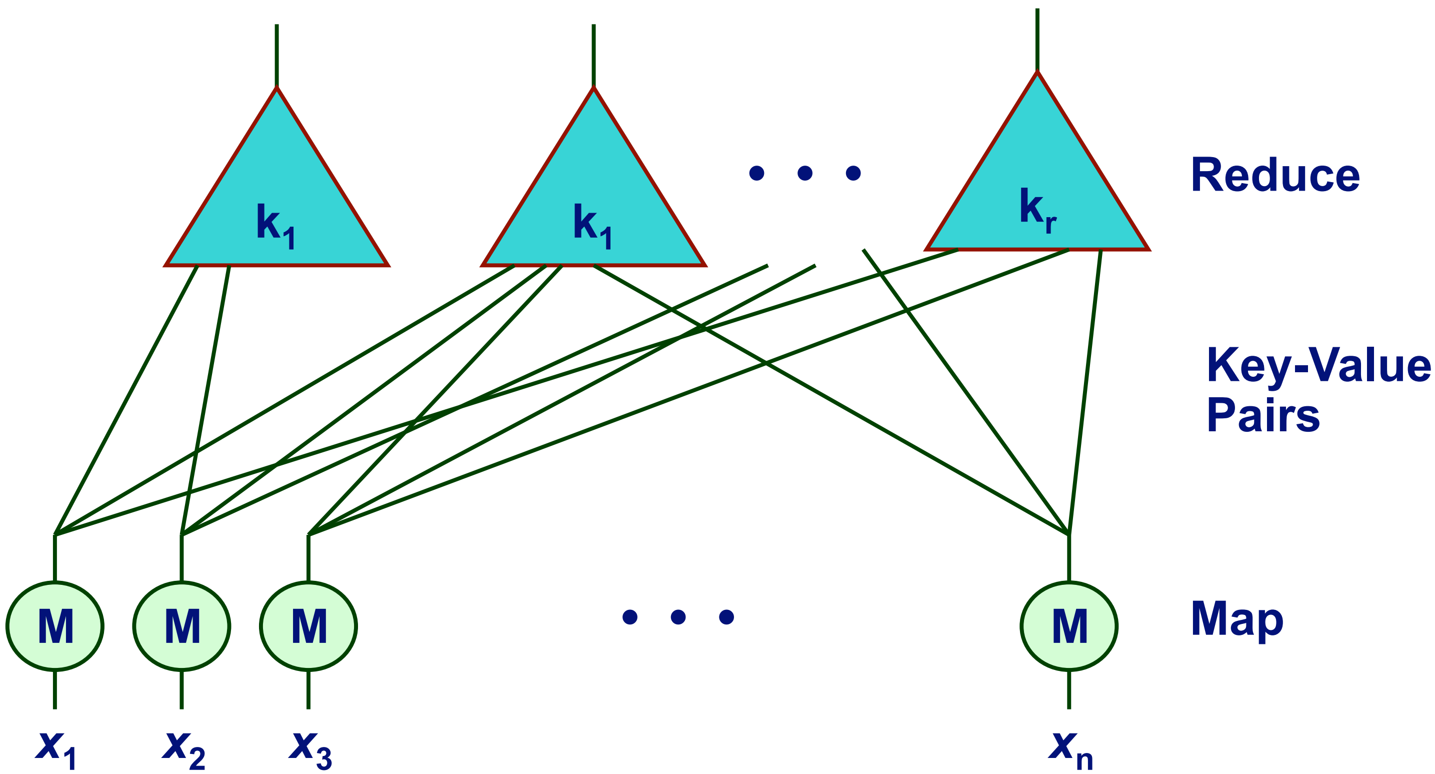
Restore

- When failure occurs
- Reset state to that of last checkpoint
- All intervening computation wasted

Performance Scaling

- Very sensitive to number of failing components

Map/Reduce Programming Model

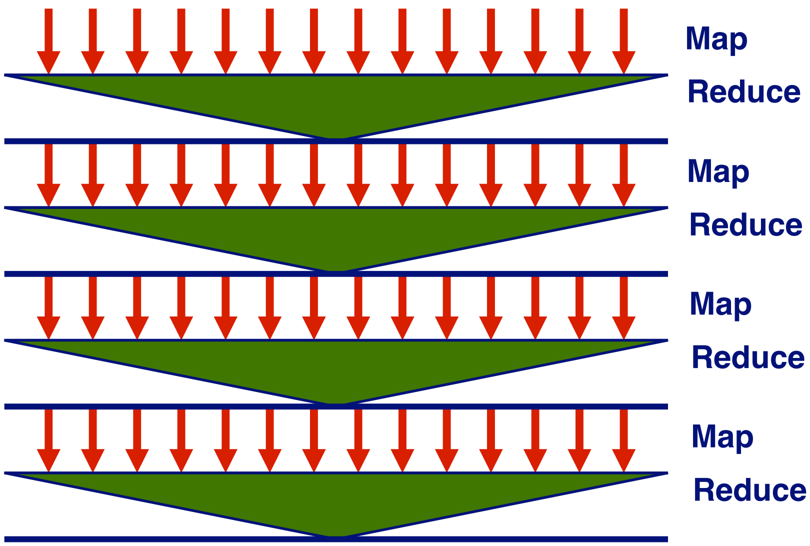


- **Map computation across many objects**
  - E.g., 10^{10} Internet web pages
- **Aggregate results in many different ways**
- **System deals with issues of resource allocation & reliability**

Dean & Ghemawat: "MapReduce: Simplified Data Processing on Large Clusters", OSDI 2004

Map/Reduce Operation

Map/Reduce



Characteristics

- Computation broken into many, short-lived tasks
  - Mapping, reducing
- Use disk storage to hold intermediate results

Strengths

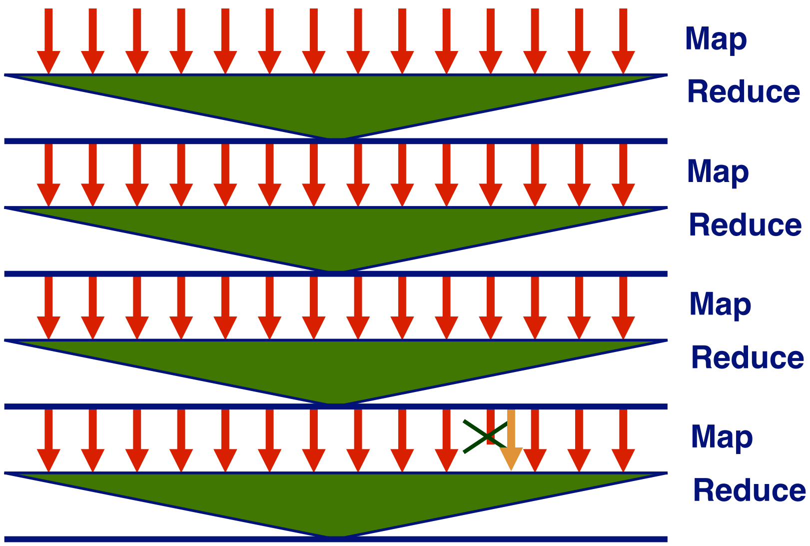
- Great flexibility in placement, scheduling, and load balancing
- Can access large data sets

Weaknesses

- Higher overhead
- Lower raw performance

Map/Reduce Fault Tolerance

Map/Reduce



Data Integrity

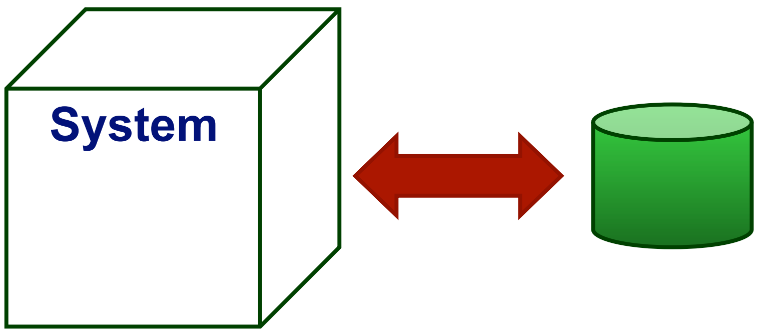
- Store multiple copies of each file
- Including intermediate results of each Map / Reduce
  - Continuous checkpointing

Recovering from Failure

- Simply recompute lost result
  - Localized effect
- Dynamic scheduler keeps all processors busy

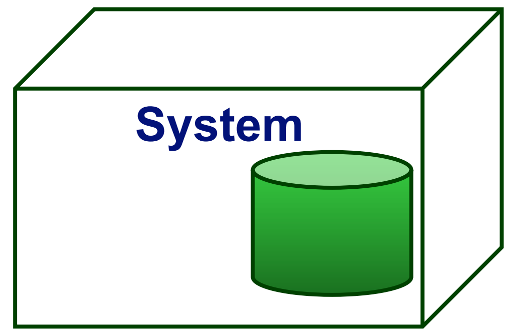
System Comparison: Data

Conventional HPC



- **Data stored in separate repository**
  - No support for collection or management
- **Brought into system for computation**
  - Time consuming
  - Limits interactivity

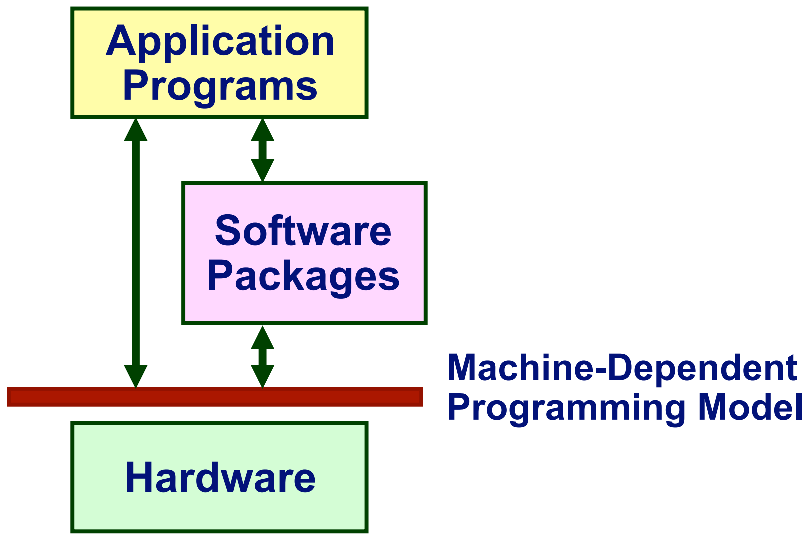
DISC



- **System collects and maintains data**
  - Shared, active data set
- **Computation collocated with storage**
  - Faster access

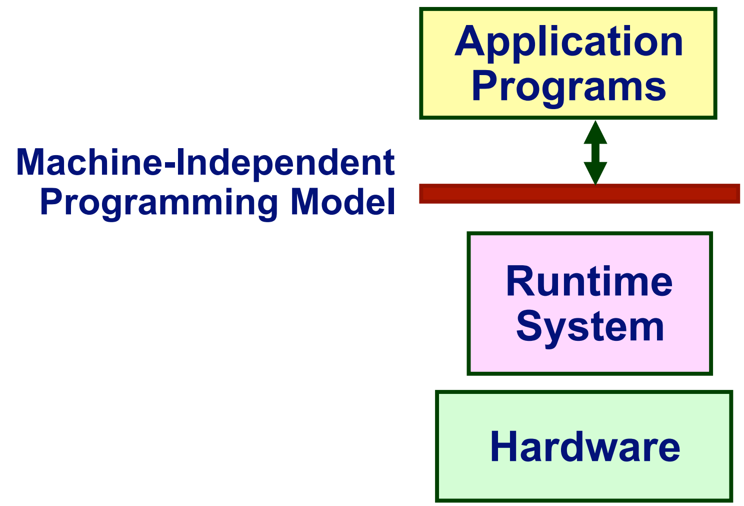
System Comparison: Programming Models

Conventional HPC



- Programs described at very low level
  - Specify detailed control of processing & communications
- Rely on small number of software packages
  - Written by specialists
  - Limits classes of problems & solution methods

DISC



- Application programs written in terms of high-level operations on data
- Runtime system controls scheduling, load balancing, …

System Comparison: Reliability

Runtime errors commonplace in large-scale systems

- Hardware failures
- Transient errors
- Software bugs

Conventional HPC

"Brittle" Systems

- Must back up entire computation to most recent checkpoint
- Must bring down system for diagnosis or repair

DISC

Flexible Error Detection and Recovery

- Runtime system detects and diagnoses errors
- Selective use of redundancy and dynamic recomputation
- Nonstop operation
- Requires flexible programming model & runtime environment

Compare to Database Technology

Structured Data

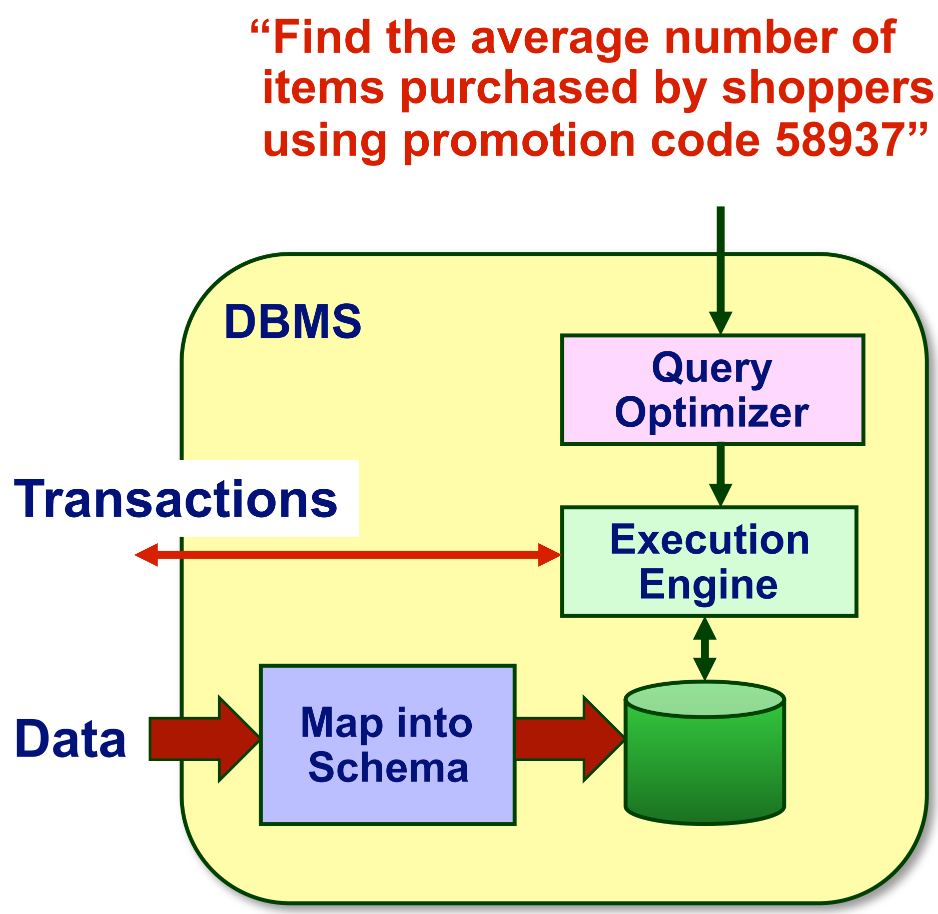
- Based on previously conceived schema design
- Bulk loaded or via transactions

Access with Queries

- Declarative language
- Extract, cross-reference, and aggregate data

Monolithic System

- Implementation details hidden (in principle)
- Often proprietary



Simplistic Comparison

DBMS

- Data stored according to schema
- Declarative query language
- Many sophisticated optimizations
- Support small & large queries
- Limited scaling

Map/Reduce

- Data stored as unstructured files
- User-defined map & reduce functions
- Runtime system fairly straightforward
- Batch processing of data only
- Designed to operate on massive scale

DBMS : Map/Reduce Convergence

Computational Support in DBs

- **User defined functions**
  - Lets user write code to operate on data
  - Recently adding map / reduce expressive power

Database Support in M/R Environments

- **Layer database features onto file system**
  - Limit expressive power in favor of scalability
- **Layer query language on top of Map / Reduce**
  - e.g., Pig, Hive

Getting Started

Goal

- **Get university faculty & students involved in DISC**

Software

- **Hadoop Project**
  - Open source project providing file system and Map/Reduce
  - Supported and used by Yahoo
  - Rapidly expanding user/developer base
  - Prototype on single machine, map onto cluster

Access to Hardware

Rent from Amazon

- **Elastic Compute Cloud (EC2)**
  - Generic Linux cycles for $0.10 / hour ($877 / yr)
- **Simple Storage Service (S3)**
  - Network-accessible storage for $0.15 / GB / month ($1800/TB/yr)

Borrow from Others

- **IBM / Google providing access to cluster through NSF**
- **Yahoo providing access to selected universities through M45 program**
- **OpenCirrus Consortium: Intel, HP, Yahoo, and others**

Build Your Own

- **Universities acquiring clusters of 10—100 nodes.**

(Potential) Impact on Science

Analyzing Measured / Acquired Data

- **Spatio-temporal data (astronomy, accelerators, …)**
- **Unstructured / semistructured data (DNA databases, …)**
- **Different needs and approaches**
  - Variations of HPC, DB, and Map/Reduce

Analyzing Synthetic Data

- **Simulations generate massive data sets**
- **Need new tools to evaluate and understand**

Impact on Traditional Scientific Computing

- **Simple hardware, extreme scalability, abstract programming model**
- **Current performance well short of HPC for computationally -intensive applications**