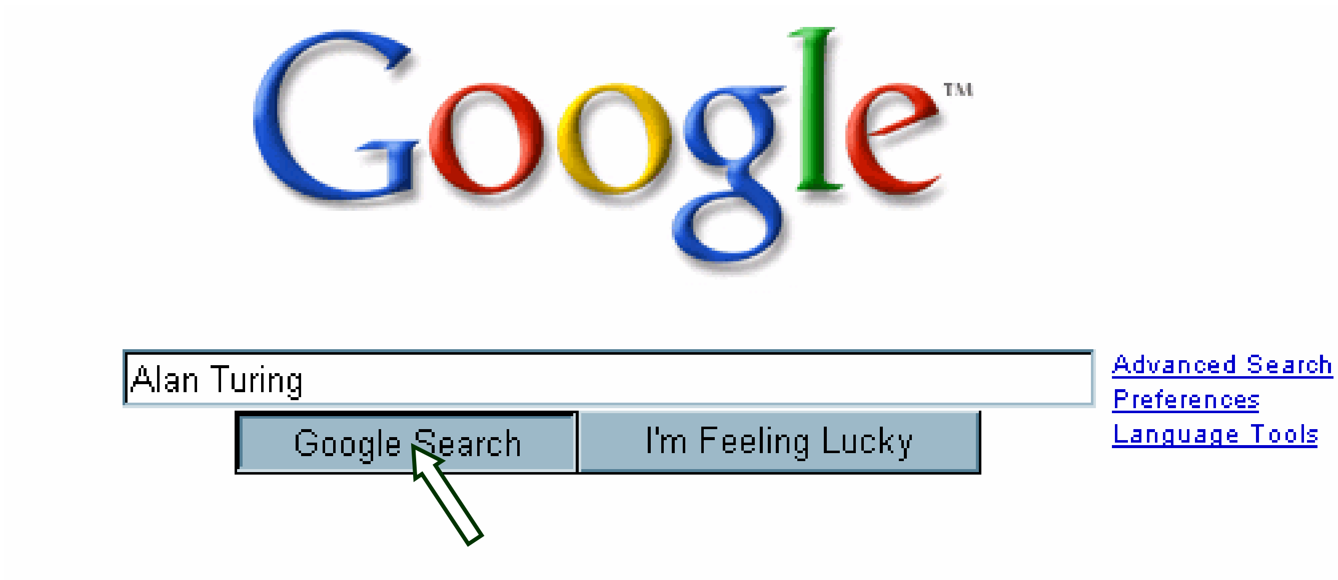
Data Intensive Super Computing

Randal E. Bryant
Carnegie Mellon University

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

Motivation



- **200+ processors**
- **200+ terabyte database**
- **10^{10} total clock cycles**
- **0.1 second response time**
- **5¢ average advertising revenue**

Google's Computing Infrastructure

System

- ~ 3 million processors in clusters of ~2000 processors each
- Commodity parts
  - x86 processors, IDE disks, Ethernet communications
  - Gain reliability through redundancy & software management
- Partitioned workload
  - Data: Web pages, indices distributed across processors
  - Function: crawling, index generation, index search, document retrieval, Ad placement

Barroso, Dean, Hölzle, "Web Search for a Planet: The Google Cluster Architecture" IEEE Micro 2003

A Data-Intensive Super Computer (DISC)

- Large-scale computer centered around data
  - Collecting, maintaining, indexing, computing
- Similar systems at Microsoft & Yahoo

Google's Economics

Making Money from Search

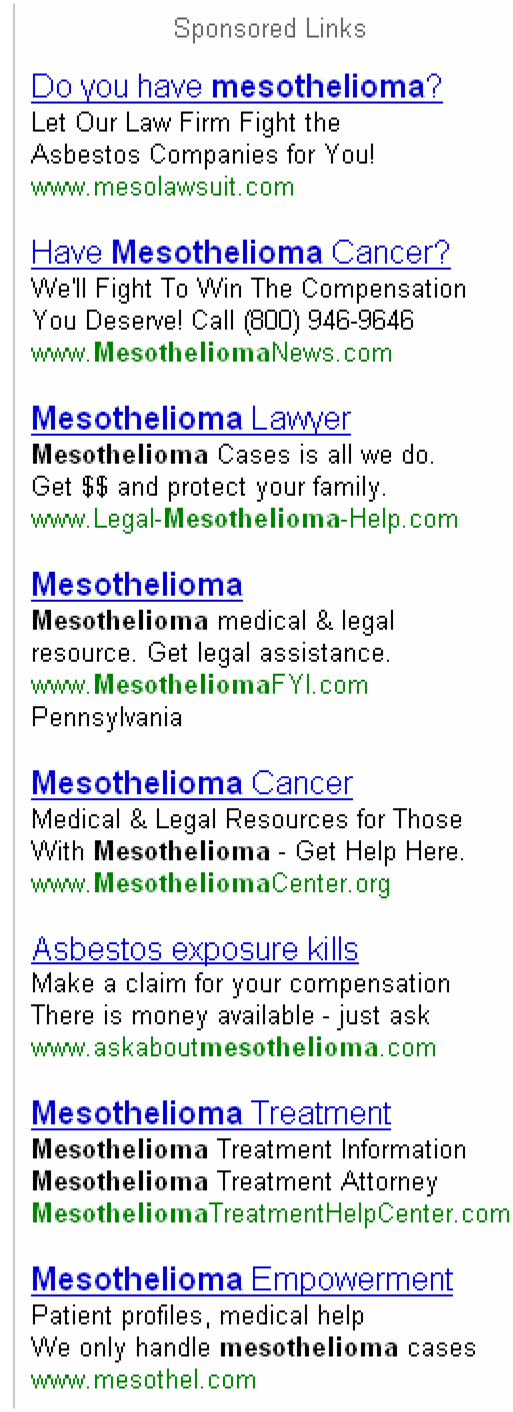
- $5B search advertising revenue in 2006
- Est. 100 B search queries
- ➔ 5¢ / query average revenue

That's a Lot of Money!

- Only get revenue when someone clicks sponsored link
- Some clicks go for $10's

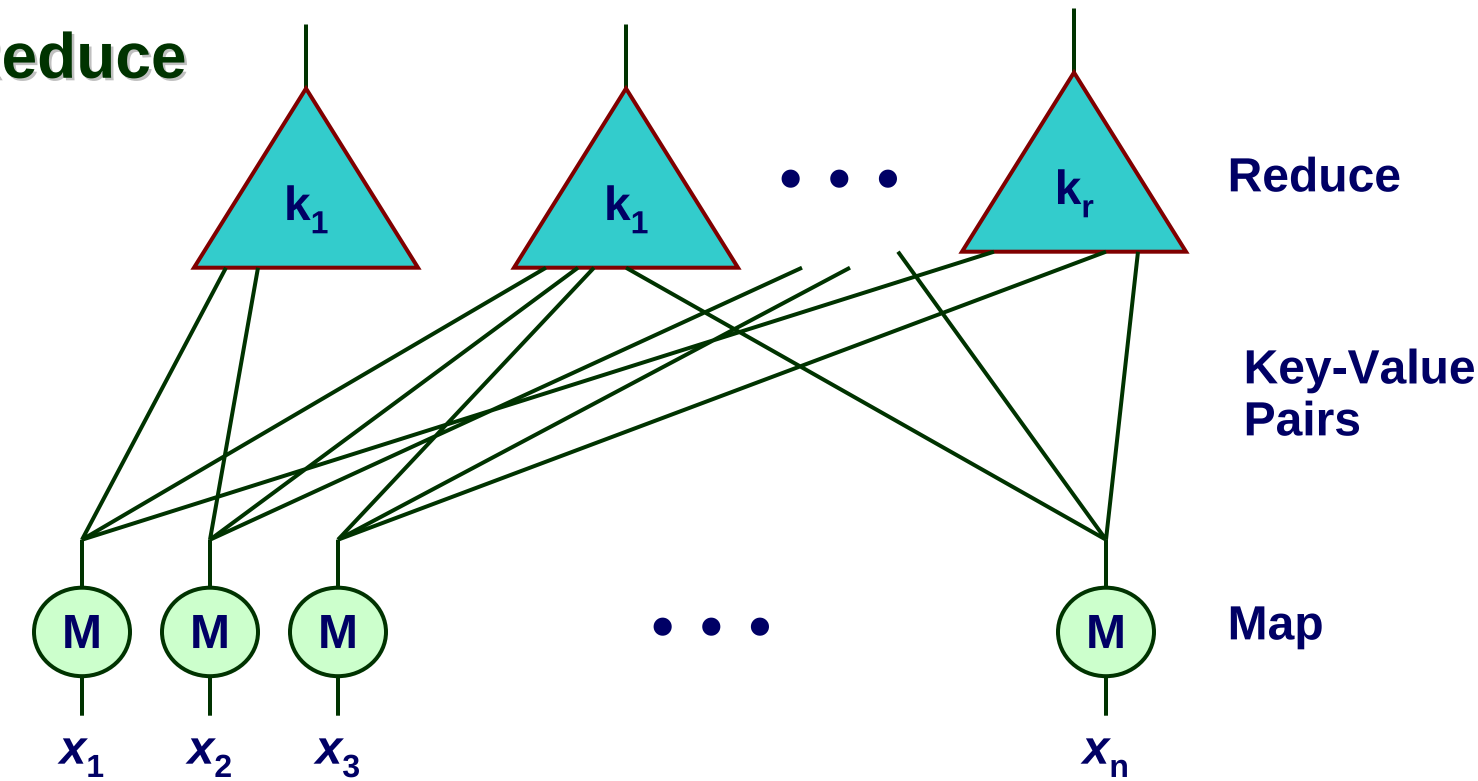
That's Really Cheap!

- Google + Yahoo + Microsoft: $5B infrastructure investments in 2007



Google's Programming Model

MapReduce



- 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

DISC: Beyond Web Search

Data-Intensive Application Domains

- **Rely on large, ever-changing data sets**
  - Collecting & maintaining data is major effort
- **Many possibilities**

Computational Requirements

- **From simple queries to large-scale analyses**
- **Require parallel processing**
- **Want to program at abstract level**

Hypothesis

- **Can apply DISC to many other application domains**

The Power of Data + Computation

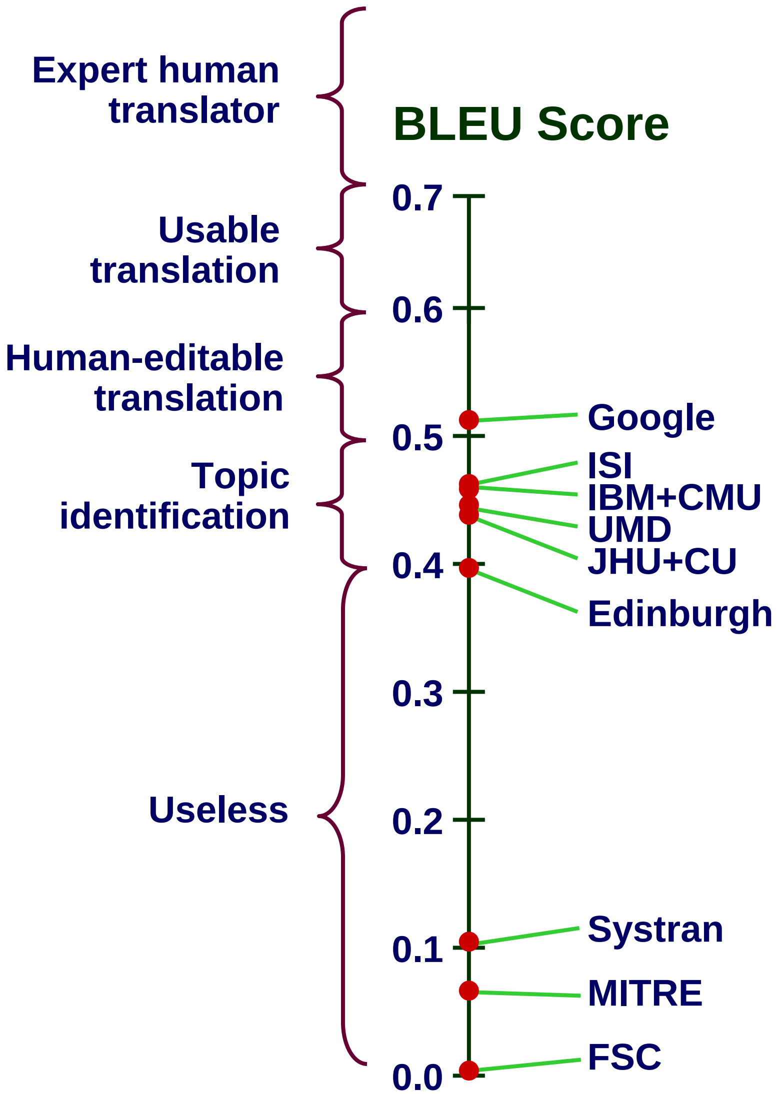
2005 NIST Machine Translation Competition

- **Translate 100 news articles from Arabic to English**

Google's Entry

- **First-time entry**
  - Highly qualified researchers
  - No one on research team knew Arabic
- **Purely statistical approach**
  - Create most likely translations of words and phrases
  - Combine into most likely sentences
- **Trained using United Nations documents**
  - 200 million words of high quality translated text
  - 1 trillion words of monolingual text in target language
- **During competition, ran on 1000-processor cluster**
  - One hour per sentence (gotten faster now)

2005 NIST Arabic-English Competition Results



BLEU Score

- Statistical comparison to expert human translators
- Scale from 0.0 to 1.0

Outcome

- Google's entry qualitatively better
- Not the most sophisticated approach
- But lots more training data and computer power

Our Data-Driven World

Science

- Data bases from astronomy, genomics, natural languages, seismic modeling, …

Humanities

- Scanned books, historic documents, …

Commerce

- Corporate sales, stock market transactions, census, airline traffic, …

Entertainment

- Internet images, Hollywood movies, MP3 files, …

Medicine

- MRI & CT scans, patient records, …

Why So Much Data?

We Can Get It

- **Automation + Internet**

We Can Keep It

- **Seagate 750 GB Barracuda @ $266**
  - 35¢ / 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**

Some Data-Oriented Applications

Samples

- Several university / industry projects
- Involving data sets ≈ 1 TB

Implementation

- Generally using scavenged computing resources
- Some just need raw computing cycles
  - "Embarrassingly parallel"
- Some use Hadoop
  - Open Source version of Google's MapReduce

Message

- Provide glimpse of style of applications that would be enabled by DISC

Example: Wikipedia Anthropology

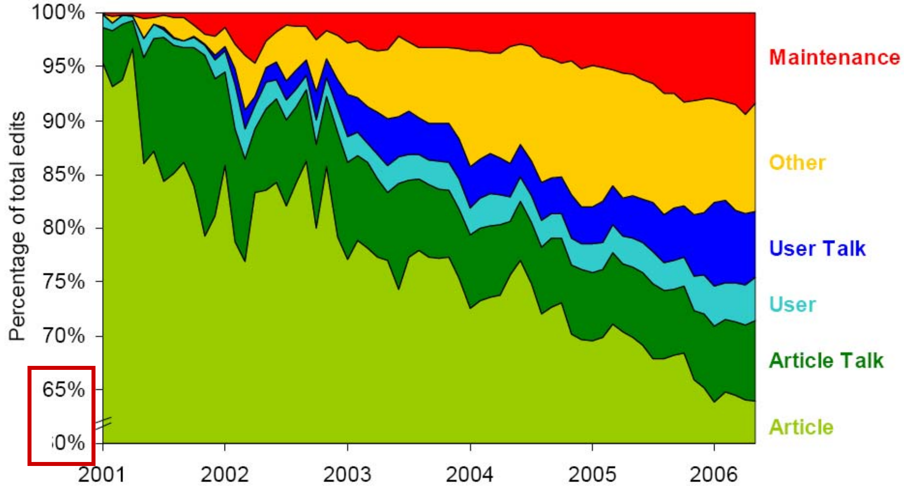


Figure 4. Changing percentage of edits over time showing that decreasing direct work (article) and increasing indirect work (article talk, user, user talk, other, and maintenance).

Kittur, Suh, Pendleton (UCLA, PARC), "He Says, She Says: Conflict and Coordination in Wikipedia" CHI, 2007

Increasing fraction of edits are for work indirectly related to articles

Experiment

- **Download entire revision history of Wikipedia**
- **4.7 M pages, 58 M revisions, 800 GB**
- **Analyze editing patterns & trends**

Computation

- **Hadoop on 20-machine cluster**

Example: Scene Completion

Hays, Efros (CMU), “Scene Completion Using Millions of Photographs” SIGGRAPH, 2007

Image Database Grouped by Semantic Content

- 30 different Flickr.com groups
- 2.3 M images total (396 GB).

Select Candidate Images Most Suitable for Filling Hole

- Classify images with gist scene detector [Torralba]
- Color similarity
- Local context matching

Computation

- Index images offline
- 50 min. scene matching, 20 min. local matching, 4 min. compositing
- Reduces to 5 minutes total by using 5 machines

Extension

- Flickr.com has over 500 million images …

Example: Web Page Analysis

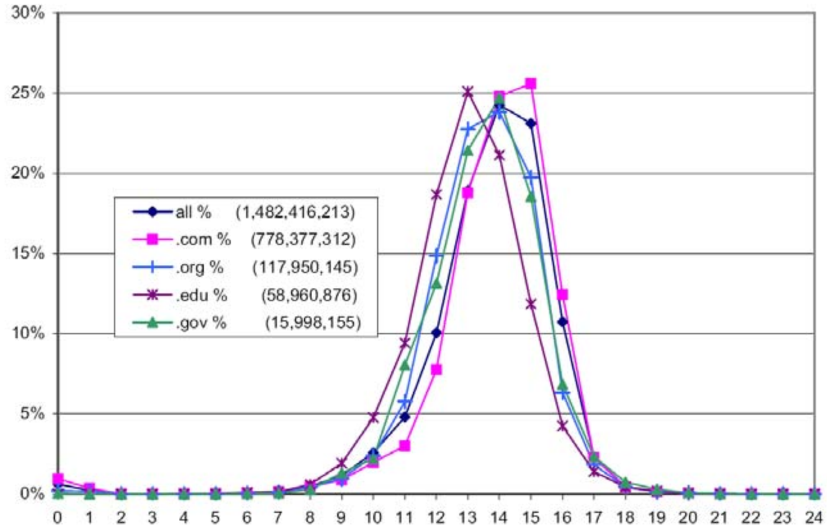


Figure 2. Distribution of document lengths overall and for selected top-level domains.

Fetterly, Manasse, Najork, Wiener (Microsoft, HP), "A Large-Scale Study of the Evolution of Web Pages," Software-Practice & Experience, 2004

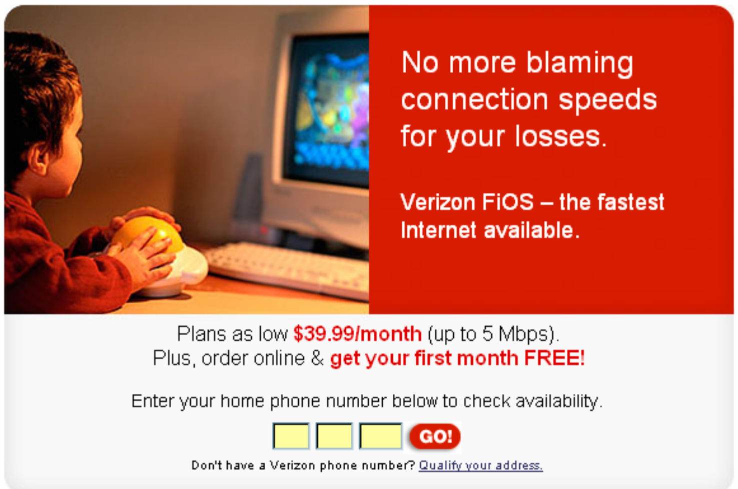
Experiment

- **Use web crawler to gather 151M HTML pages weekly 11 times**
  - Generated 1.2 TB log information
- **Analyze page statistics and change frequencies**

Systems Challenge

"Moreover, we experienced a catastrophic disk failure during the third crawl, causing us to lose a quarter of the logs of that crawl."

Oceans of Data, Skinny Pipes



Seagate

1 Terabyte

- Easy to store
- Hard to move

Disks	MB / s	Time
Seagate Barracuda	78	3.6 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

Designing a DISC System

Inspired by Google's Infrastructure

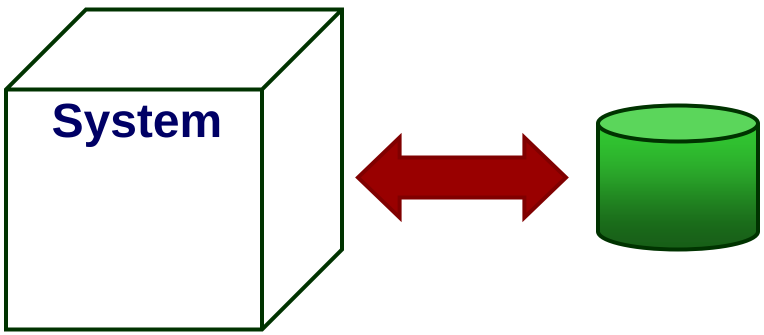
- **System with high performance & reliability**
- **Carefully optimized capital & operating costs**
- **Take advantage of their learning curve**

But, Must Adapt

- **More than web search**
  - Wider range of data types & computing requirements
  - Less advantage to precomputing and caching information
  - Higher correctness requirements
- **10^2–10^4 users, not 10^6–10^8**
  - Don't require massive infrastructure

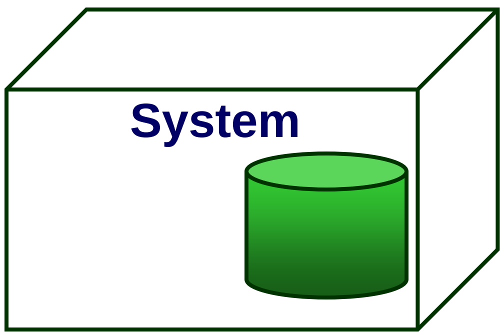
System Comparison: Data

Conventional Supercomputers



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

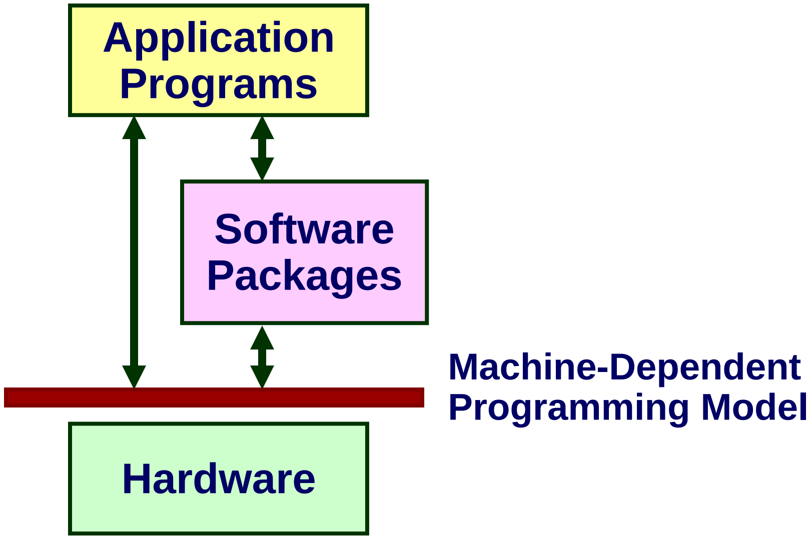
DISC



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

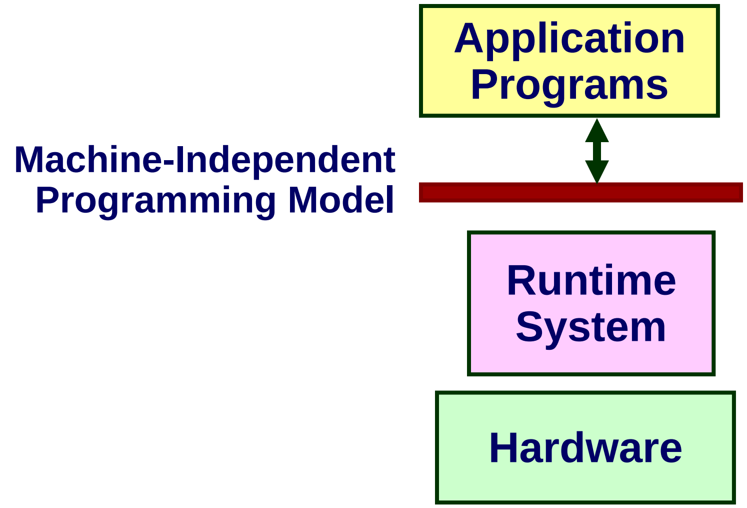
System Comparison: Programming Models

Conventional Supercomputers



- 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: Interaction

Conventional Supercomputers

Main Machine: Batch Access

- Priority is to conserve machine resources
- User submits job with specific resource requirements
- Run in batch mode when resources available

Offline Visualization

- Move results to separate facility for interactive use

DISC

Interactive Access

- Priority is to conserve human resources
- User action can range from simple query to complex computation
- System supports many simultaneous users
  - Requires flexible programming and runtime environment

System Comparison: Reliability

Runtime errors commonplace in large-scale systems

- Hardware failures
- Transient errors
- Software bugs

Conventional Supercomputers

"Brittle" Systems

- Main recovery mechanism is to recompute from most recent checkpoint
- Must bring down system for diagnosis, repair, or upgrades

DISC

Flexible Error Detection and Recovery

- Runtime system detects and diagnoses errors
- Selective use of redundancy and dynamic recomputation
- Replace or upgrade components while system running
- Requires flexible programming model & runtime environment

What About Grid Computing?

Grid: Distribute Computing and Data

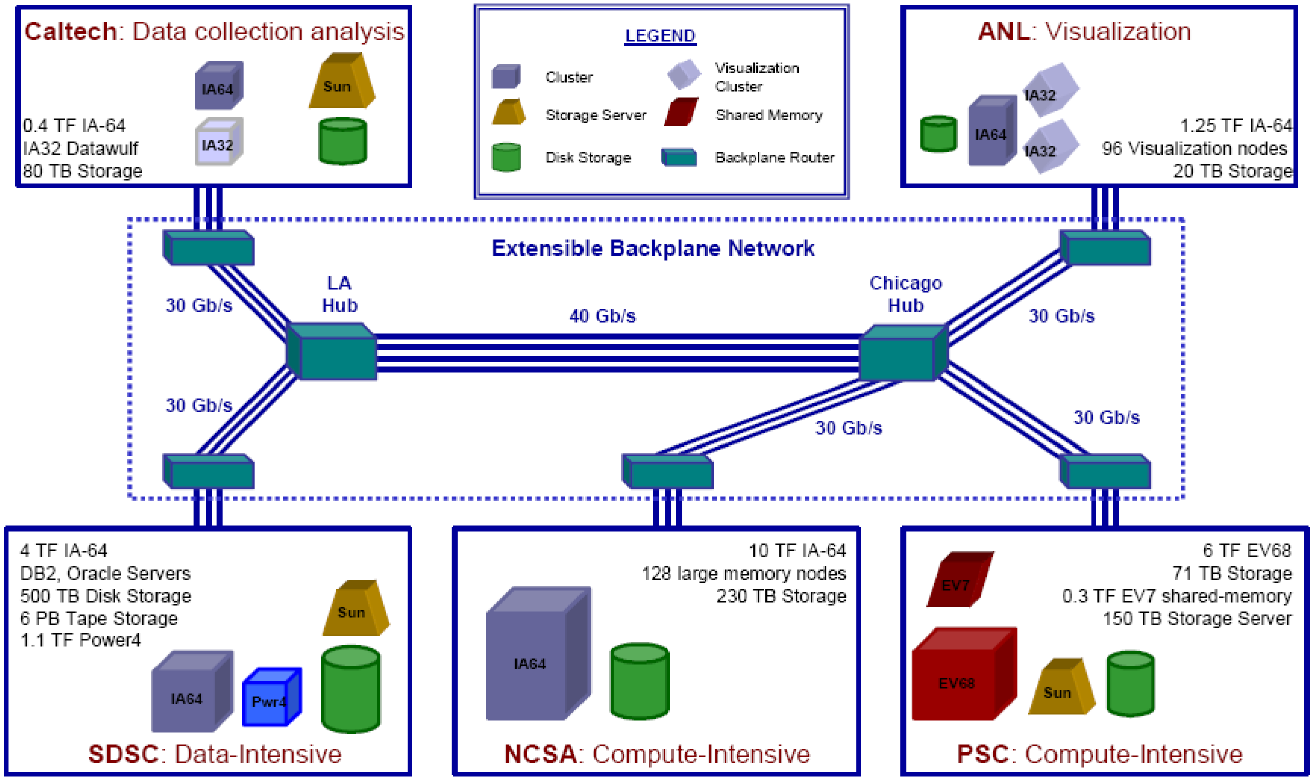
- **Computation: Distribute problem across many machines**
  - Generally only those with easy partitioning into independent subproblems
- **Data: Support shared access to large-scale data set**

DISC: Centralize Computing and Data

- **Enables more demanding computational tasks**
- **Reduces time required to get data to machines**
- **Enables more flexible resource management**

Part of growing trend to server-based computation

Grid Example: TeraGrid (2003)



Computation

- 22 T FLOPS total capacity

Storage

- 980 TB total disk space

Communication

- 5 GB/s Bisection bandwidth
- 3.3 min to transfer 1 TB

Compare to Transaction Processing

Main Commercial Use of Large-Scale Computing

- **Banking, finance, retail transactions, airline reservations, …**

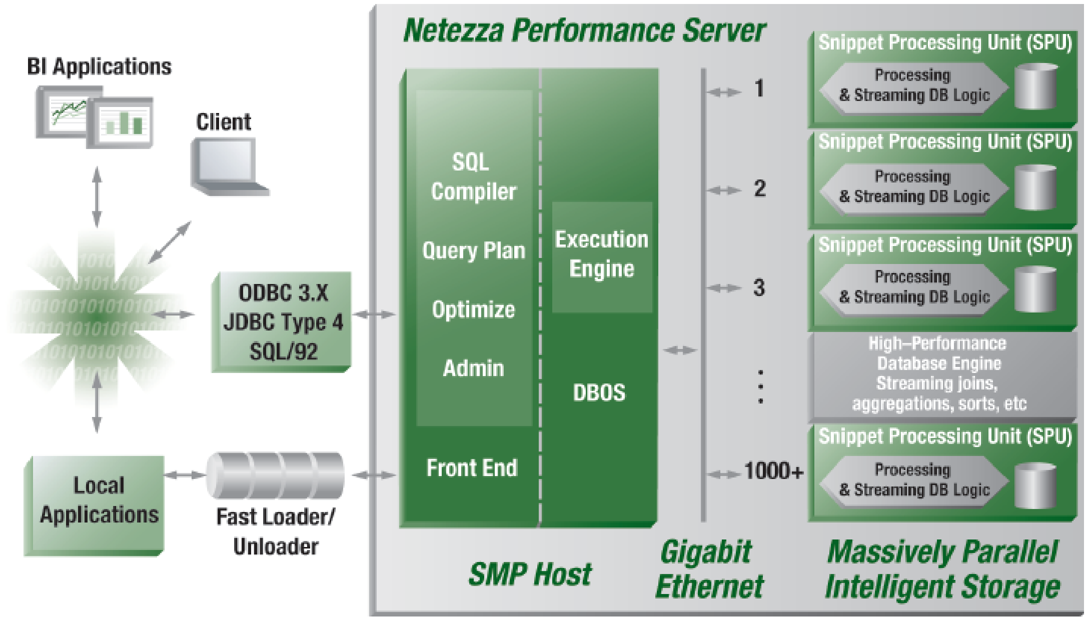
Stringent Functional Requirements

- **Only one person gets last $1 from shared bank account**
  - Beware of replicated data
- **Must not lose money when transferring between accounts**
  - Beware of distributed data
- **Favors systems with small number of high-performance, high-reliability servers**

Our Needs are Different

- **More relaxed consistency requirements**
  - Web search is extreme example
- **Fewer sources of updates**
- **Individual computations access more data**

A Commercial DISC



Netezza Performance Server (NPS)

- **Designed for "data warehouse" applications**
  - Heavy duty analysis of database
- **Data distributed over up to 500 Snippet Processing Units**
  - Disk storage, dedicated processor, FPGA controller
- **User "programs" expressed in SQL**

Solving Graph Problems with Netezza

Davidson, Boyack, Zacharski, Helmreich, & Cowie,
"Data-Centric Computing with the Netezza Architecture,"
Sandia Report SAND2006-3640

Evaluation

- Tested 108-node NPS
- 4.5 TB storage
- Express problems as database construction + queries

Problems tried

- Citation graph for 16M papers, 388M citations
- 3.5M transistor circuit

Outcomes

- Demonstrated ease of programming & interactivity of DISC
- Seems like SQL limits types of computations

Why University-Based Projects?

Open

- Forum for free exchange of ideas
- Apply to societally important, possibly noncommercial problems

Systematic

- Careful study of design ideas and tradeoffs

Creative

- Get smart people working together

Fulfill Our Educational Mission

- Expose faculty & students to newest technology
- Ensure faculty & PhD researchers addressing real problems

Who Would Use DISC?

Identify One or More User Communities

- **Group with common interest in maintaining shared data repository**
- **Examples:**
  - Web-based text
  - Genomic / proteomic databases
  - Ground motion modeling & seismic data

Adapt System Design and Policies to Community

- **What / how data are collected and maintained**
- **What types of computations will be applied to data**
- **Who will have what forms of access**
  - Read-only queries
  - Large-scale, read-only computations
  - Write permission for derived results

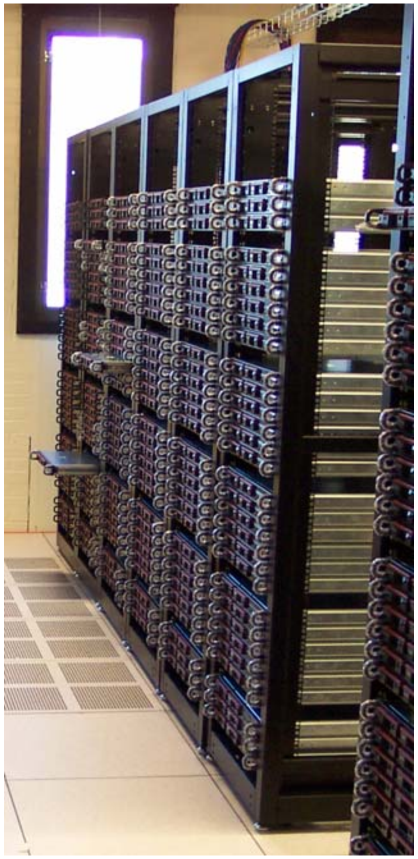
Constructing General-Purpose DISC

Hardware

- Similar to that used in data centers and high-performance systems
- Available off-the-shelf

Hypothetical "Node"

- 1–2 dual or quad core processors
- 1 TB disk (2-3 drives)
- ~$10K (including portion of routing network)



Possible System Sizes

100 Nodes $1M

- 100 TB storage
- Deal with failures by stop & repair
- Useful for prototyping

1,000 Nodes $10M

- 1 PB storage
- Reliability becomes important issue
- Enough for WWW caching & indexing

10,000 Nodes $100M

- 10 PB storage
- National resource
- Continuously dealing with failures
- Utility?

Implementing System Software

Programming Support

- Abstractions for computation & data representation
  - E.g., Google: MapReduce & BigTable
- Usage models

Runtime Support

- Allocating processing and storage
- Scheduling multiple users
- Implementing programming model

Error Handling

- Detecting errors
- Dynamic recovery
- Identifying failed components

CS Research Issues

Applications

- Language translation, image processing, …

Application Support

- Machine learning over very large data sets
- Web crawling

Programming

- Abstract programming models to support large-scale computation
- Distributed databases

System Design

- Error detection & recovery mechanisms
- Resource scheduling and load balancing
- Distribution and sharing of data across system

Sample Research Problems

Processor Design for Cluster Computing

- Better I/O, less power

Resource Management

- How to support mix of big & little jobs
- How to allocate resources & charge different users

Building System with Heterogeneous Components

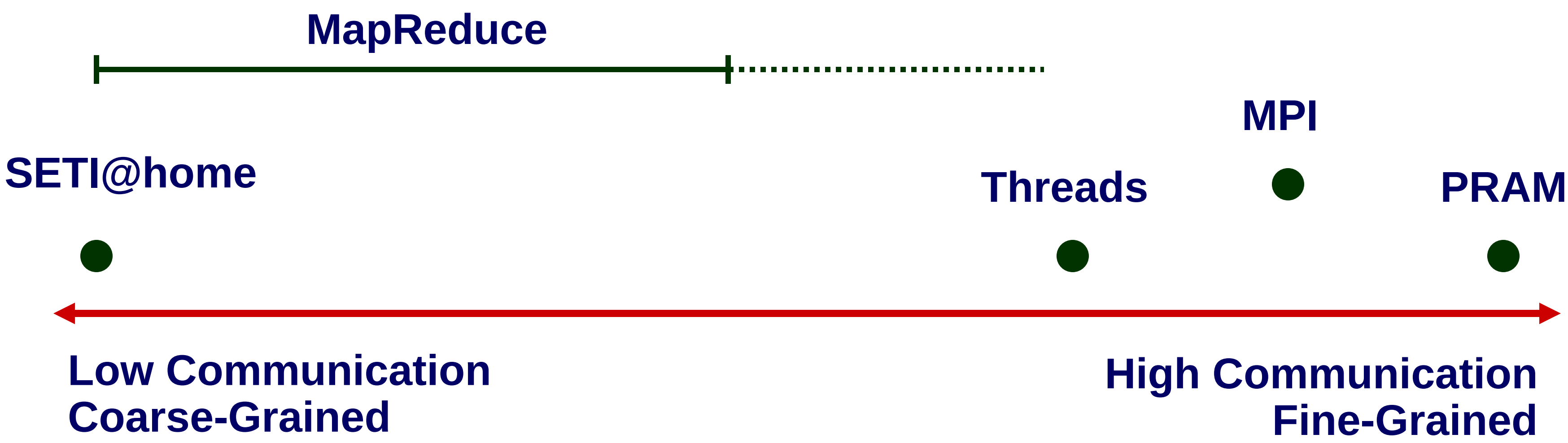
How to Manage Sharing & Security

- Shared information repository updated by multiple sources
- Need semantic model of sharing and access

Programming with Uncertain / Missing Data

- Some fraction of data inaccessible when want to compute

Exploring Parallel Computation Models



DISC + MapReduce Provides Coarse-Grained Parallelism

- Computation done by independent processes
- File-based communication

Observations

- Relatively "natural" programming model
  - If someone else worries about data distribution & load balancing
- Research issue to explore full potential and limits
  - Work at MS Research on Dryad is step in right direction.

Computing at Scale is Different!

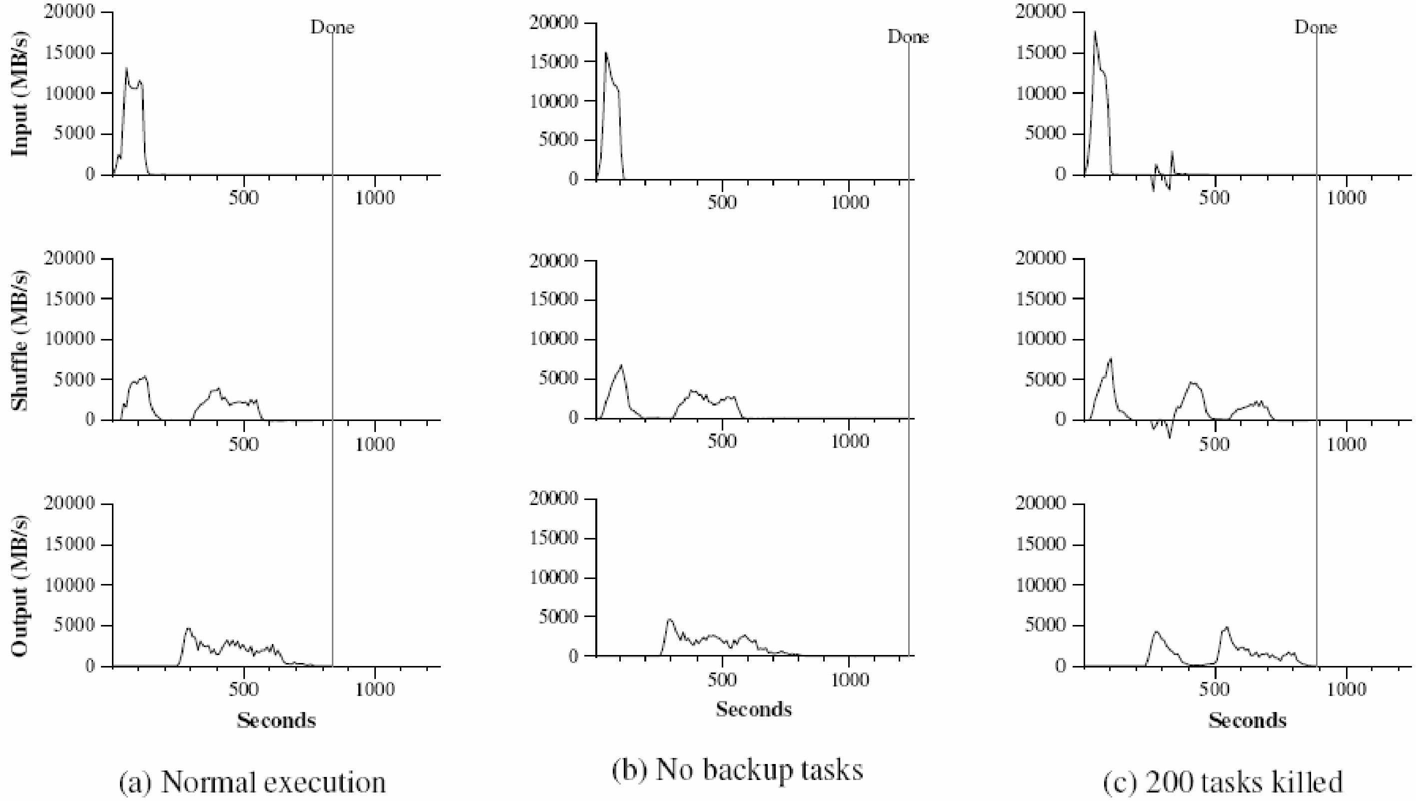


Figure 3: Data transfer rates over time for different executions of the sort program

- Dean & Ghemawat, OSDI 2004
- Sorting 10 million 100-byte records with 1800 processors
- Proactively restart delayed computations to achieve better performance and fault tolerance

Jump Starting

Goal

- **Get faculty & students active in DISC**

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)
- **Example: maintain crawled copy of web (50 TB, 100 processors, 0.5 TB/day refresh) ~250K / year**

Software

- **Hadoop Project**
  - Open source project providing file system and MapReduce
  - Supported and used by Yahoo

Impediments for University Researchers

Financial / Physical

- Costly infrastructure & operations
- We have moved away from shared machine model

Psychological

- Unusual situation: universities need to start pursuing a research direction for which industry is leader
- For system designers: what's there to do that Google hasn't already done?
- For application researchers: How am I supposed to build and operate a system of this type?

Overcoming the Impediments

There's Plenty Of Important Research To Be Done

- **System building**
- **Programming**
- **Applications**

We Can Do It!

- **Amazon lowers barriers to entry**
- **Teaming & collaborating**
  - The CCC can help here
- **Use Open Source software**

What If We Don't?

- **Miss out on important research & education topics**
- **Marginalize our role in community**

Concluding Thoughts

The World is Ready for a New Approach to Large-Scale Computing

- **Optimized for data-driven applications**
- **Technology favoring centralized facilities**
  - Storage capacity & computer power growing faster than network bandwidth

University Researchers Eager to Get Involved

- **System designers**
- **Applications in multiple disciplines**
- **Across multiple institutions**

More Information

"Data-Intensive Supercomputing: The case for DISC"

- **Tech Report: CMU-CS-07-128**
- **Available from** `http://www.cs.cmu.edu/~bryant`