Chapter 6

Warehouse-Scale Computers to Exploit Request-Level and Data-Level Parallelism:
Introduction

- Warehouse-scale computer (WSC)
  - Provide Internet services
    - Search, social networking, online maps, video sharing, online shopping, email, cloud computing, etc.
  - Provide Software as a Service (SaaS)
    - Millions of users, millions of independent requests
    - Threads rarely need to synchronize
    - Request-level parallelism (RLP)

- Differences with HPC “clusters”:
  - Clusters have higher performance processors and network
  - Clusters emphasize thread-level parallelism, WSCs emphasize request-level parallelism
Introduction

- Important design factors for WSC:
  - Cost-performance
    - Small savings add up
  - Energy efficiency
    - Affects power distribution and cooling
    - Work per joule
  - Dependability via redundancy
  - Network I/O
  - Interactive and batch processing workloads
  - Ample computational parallelism is not important
    - Most jobs are totally independent
    - “Request-level parallelism”
  - Operational costs count
    - Power consumption is a primary, not secondary, constraint when designing system
  - Scale and its opportunities and problems
    - Can afford to build customized systems since WSC require volume purchase
Prgrm’g Models and Workloads

- Batch processing framework: MapReduce
  - **Map**: applies a programmer-supplied function to each logical input record
    - Runs on thousands of computers
    - Provides new set of key-value pairs as intermediate values
  - **Reduce**: collapses values using another programmer-supplied function
Example:

- map (String key, String value):
  - // key: document name
  - // value: document contents
  - for each word w in value
    - EmitIntermediate(w,"1");  // Produce list of all words

- reduce (String key, Iterator values):
  - // key: a word
  - // value: a list of counts
  - int result = 0;
  - for each v in values:
    - result += ParseInt(v);  // get integer from key-value pair
    - Emit(As[String(result)]);
Prgrm’g Models and Workloads

- MapReduce runtime environment schedules map and reduce task to WSC nodes
- Apache Hadoop is an open-source alternative
- Supports availability
  - Use replicas of data across different servers
- Scales with workload demands
  - Often vary considerably
Prgrm’g Models and Workloads
Computer Architecture of WSC

- WSC often use a hierarchy of networks for interconnection
- Each rack holds servers connected to a rack switch
- Rack switches are uplinked to switch higher in hierarchy
  - Goal is to maximize locality of communication relative to the rack
- Array switch connects an array of racks
  - Array switch should have at least 10X the bandwidth of rack switch
  - Cost of $n$-port switch grows as $n^2$
Computer Architecture of WSC

Array switch

Rack

switch

1U Server

Copyright © 2012, Elsevier Inc. All rights reserved.
Computer Architecture of WSC

![Diagram showing the architecture of WSC](image)

Key:
- CR = L3 core router
- AR = L3 access router
- S = Array switch
- LB = Load balancer
- A = Rack of 80 servers with rack switch
Storage

Storage options
- Use disks inside the servers, or
- Network attached storage through Infiniband
- WSCs generally rely on local disks
- Google File System (GFS) uses local disks and maintains at least three replicas
- Apache Hadoop uses a similar approach
### WSC Memory Hierarchy

- Servers can access DRAM and disks on other servers using a NUMA-style interface

<table>
<thead>
<tr>
<th></th>
<th>Local</th>
<th>Rack</th>
<th>Array</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM latency (μs)</td>
<td>0</td>
<td>100</td>
<td>300</td>
</tr>
<tr>
<td>Disk latency (μs)</td>
<td>10,000</td>
<td>11,000</td>
<td>12,000</td>
</tr>
<tr>
<td>DRAM bandwidth (MB/sec)</td>
<td>20,000</td>
<td>100</td>
<td>10</td>
</tr>
<tr>
<td>Disk bandwidth (MB/sec)</td>
<td>200</td>
<td>100</td>
<td>10</td>
</tr>
<tr>
<td>DRAM capacity (GB)</td>
<td>16</td>
<td>1,040</td>
<td>31,200</td>
</tr>
<tr>
<td>Disk capacity (GB)</td>
<td>2000</td>
<td>160,000</td>
<td>4,800,000</td>
</tr>
</tbody>
</table>
Infrastructure and Costs of WSC

- **Location of WSC**
  - Proximity to Internet backbones, electricity cost, property tax rates, low risk from earthquakes, floods, and hurricanes

- **Power distribution**
Infrastructure and Costs of WSC

- **Cooling**
  - Air conditioning used to cool server room
  - 64 F – 71 F
    - Keep temperature higher (closer to 71 F)
  - Cooling towers can also be used
    - Minimum temperature is “wet bulb temperature”
Infrastructure and Costs of WSC

- Cooling system also uses water (evaporation and spills)
  - E.g. 70,000 to 200,000 gallons per day for an 8 MW facility

- Power cost breakdown:
  - Chillers: 30-50% of the power used by the IT equipment
  - Air conditioning: 10-20% of the IT power, mostly due to fans

- How many servers can a WSC support?
  - Each server:
    - “Nameplate power rating” gives maximum power consumption
    - To get actual, measure power under actual workloads
  - Oversubscribe cumulative server power by 40%, but monitor power closely
Measuring Efficiency of a WSC

- **Power Utilization Effectiveness (PEU)**
  - $\text{PEU} = \frac{\text{Total facility power}}{\text{IT equipment power}}$
  - Median PUE on 2006 study was 1.69

- **Performance**
  - Latency is important metric because it is seen by users
  - Bing study: users will use search less as response time increases
  - Service Level Objectives (SLOs)/Service Level Agreements (SLAs)
    - E.g. 99% of requests be below 100 ms
Measuring Efficiency of a WSC
Google Servers

- Containers holding two rows of 29 racks each (58 racks)
- Each rack holds 20 servers (1160 total servers)
- Commodity hardware such as
  - Dual-core AMD Opteron processors (2.2 GHz)
    - Downclocked FSB (666 MHz -> 533 Mhz)
  - 8 GB DDR2 DRAM
  - 1 or 2 SATA disk drives
  - 160 watts peak, 85 watts idle
Google Servers
Google Servers
Google Servers
Cloud Computing

- Cheaper to rent computing time than to maintain servers (e.g., $0.10 per machine/hour)
- Uses virtual machines
  - Protects users from each other
  - Simplifies software distribution
  - Easily to control resource usage
  - Limited access to physical resources
  - Hide actual hardware details
- Amazon Web Services (AWS)
  - Simple Storage Service (S3)
  - Elastic Compute Cloud (EC2)
Fallacies

- Cloud computing providers are losing money.
- Capital costs for WSC are higher than for the servers that it houses.
- Given improvements in DRAM dependability and the fault tolerance of WSC systems software, you don’t need to spend extra for ECC memory in a WSC.
- Turning off hardware during periods of low activity improves cost-performance of a WSC.
- Replacing all disks with flash memory will improve cost-performance of a WSC.
Pitfalls

- Trying to save power with inactive low-power modes versus active low-power modes
- Using too wimpy a processor when trying to improve WSC cost-performance