Warehouse-scale Computing

The machinery that runs the cloud

Luiz André Barroso, Google Inc.

As high-bandwidth Internet connectivity becomes more ubiquitous, more applications are being offered as Internet services that run on remote data center facilities instead of on a user's personal computer. The two classes of machines enabling this trend correspond to the very small and the very large ends of the device spectrum. On the small end are mobile devices which focus on user interaction and Internet connectivity but with limited processing capabilities; on the large end are the massive computing and storage systems that implement many of today's Internet (or Cloud) services, referred to here as *Warehouse-scale Computers* (WSCs) [Barroso09].

Cost-efficiency is critical for Internet services since their revenue often derives from online advertisement, with only a small fraction of service requests resulting directly in revenue. WSCs are particularly efficient for popular compute and data intensive on-line services, such as Internet search or language translation. A single search request may query the entire Web, including images, videos, news sources, maps and product information. Such services require a computing capacity well beyond the reach of a personal computing device, and therefore can only be economically feasible when amortized over a very large user population. In this article we provide a brief description of the hardware and software in WSCs and highlight some of the key technical challenges in this space.

The Hardware

WSC hardware consists of three primary sub-systems: the computing equipment itself, the power distribution systems, and the cooling infrastructure. A brief description of each sub-system follows below.

Computing systems:
WSCs are built out of low-end or mid-range server-class computers connected together in racks of 40-80 units by a first-level networking switch, with each of these switches in turn connecting to a cluster-level network fabric that ties together all the racks. These clusters tend to be composed of several thousands of servers, and constitute the primary unit of computing for Internet services (WSCs can be composed of one or multiple of such clusters). Storage is either provisioned as disk drives connected to each server or as dedicated file serving appliances(*). The use of near PC-class components is a departure from the supercomputing model of the 1970’s which relied on extremely high-end technology, and reflects the cost-sensitivity of this application space. Lower-end server components can leverage technology and development costs invested in the much higher volume consumer markets, therefore achieving extremely high cost-efficiency.

Power distribution:
The peak electricity needs of the computing systems in a WSC can reach above 10MW -- roughly equivalent to the average power usage of eight thousand US households. At those levels they need to tap into high-voltage long-distance power lines (typically 10-20kV), which need to be converted down to the low voltage levels appropriate for

distribution within the facility (400-600V). Power is then fed to an uninterruptible power supply (UPS) system that acts as an energy supply buffer against utility power failures, before being distributed to computing equipment. UPS systems are sized to support less than a minute of demand, since a set of diesel generators can jump into action within 15 seconds of an utility outage.

Cooling infrastructure:
Virtually all energy provided to computing equipment becomes heat, which needs to be removed from the facility so that the equipment can remain within designed operating ranges. This is accomplished by air conditioning units inside the building that supply cold air (18-22C) to the machinery, coupled by a liquid coolant loop to a cooling plant situated outside the building. The cooling plant uses chiller or cooling towers to expel the heat to the environment.

Relative costs:
The capital cost breakdown of a WSC among the three main subsystems above varies depending on the facility design. The cost of the non-computing components is proportional to the peak power delivery capacity, with the cooling infrastructure generally being more expensive than the power distribution. If high-end energy efficient computing components are deployed, computing system costs will tend to dominate. If lower-end and less energy efficient computing components are deployed, cooling and power distribution system costs can dominate total costs. Energy therefore affects WSC costs in two ways: directly through the price of the electricity consumed, and indirectly through the cost of cooling and power plants.

Design challenges:
Designing a WSC represents a formidable challenge. Some of the most difficult issues include deciding between scale-up and scale-out approaches (bigger servers or more servers) and determining the right aggregate capacity and performance balance among the various subsystems (example: do we have too much CPU firepower and too little networking bandwidth?). These decisions ultimately rely on workload characteristics. For example, search workloads tend to compute heavily within server nodes and exchange comparatively little networking traffic. Video serving workloads do relatively little processing and yet have heavy networking demands. An Internet services provider that offers both classes of workloads might be driven to design different WSCs for each class of workload, or to find a common sweet spot that accommodates the needs of both. Common designs, when possible, are preferable since they allow the provider to dynamically re-assign WSC resources to workloads as the business needs change -- business priorities tend to change frequently in the still young Internet services area.

Energy Efficiency:
Given the impact of energy on the overall cost of WSCs, it is critical to understand where energy is being used. The data center industry has developed a metric named power usage effectiveness (PUE), which objectively characterizes the efficiency of the non-computing elements in a facility. PUE is derived by measuring the total energy entering a facility and dividing it by the amount that is consumed by the computing equipment. Typical data centers are rather inefficient, with PUEs hovering around 2 (one Watt used, one Watt wasted). State-of-the-art facilities have reported PUEs as low as 1.13 [Google10]; at such levels, the energy efficiency focus shifts back to the computing equipment itself.

Mobile and embedded devices have been the main target of low-power technology development for decades, and much of the energy-saving features making their way to servers are rooted in that class of devices. However, mobile systems have focused on techniques which save power when components are idle, a feature that is less useful for WSCs which are rarely completely idle. Making WSCs energy efficient requires energy proportionality, a system behavior that yields energy efficient operation across the activity range [Barroso07].

The Software

The software running on WSCs can be broadly divided into two layers: infrastructure software and workloads, as described below.

Infrastructure software:

The software infrastructure in WSCs includes some basic components that enable the coordinated scheduling and use of their resources. For example, each Google WSC cluster has a management software stack that includes a scheduling master and a storage master, and corresponding slaves in each machine. The scheduling master takes submitted jobs and creates job task instances in various machines. Enforcing resource allocations and performance isolation among tasks is accomplished by the per-machine scheduling slaves in coordination with the underlying operating system (typically Linux-based). The role of storage servers is to export the local disks to the cluster-wide file system users.

Workloads:

While WSC workloads can span thousands of individual job tasks, with diverse behavior and communication patterns, they tend to fall into two broad categories: data processing and online services. Data processing workloads are the large batch computations needed to analyze, reorganize, or convert data between different formats. Stitching individual satellite images into seamless Google Earth tiles, or building a Web index from a large collection of crawled documents would be examples of data processing workloads. Their structure tends to be relatively uniform, and the keys for high performance are finding the right way to partition them among multiple tasks, and placing those tasks closer to their corresponding data. Programming systems such as MapReduce [Dean04] have been successful in simplifying the task of building complex data-processing workloads.

Web search is the best example of demanding online serving workloads. For these services, user happiness is deeply affected by response times, and the system may need to process through tens of terabytes of index data in order to respond to a query. High processing throughput is a requirement for both data processing and online services workloads, but the latter has much stricter latency constraints per individual request. Obtaining predictable performance from thousands of cooperating nodes in sub-second time scales is the main challenge for this class of systems.

Programming challenges:

Similarly to the hardware design problem for WSCs, the complexity of software

development for a WSC hardware platform can be a remarkable obstacle for both workload and infrastructure software developers. The sources of complexity derive from a combination of scale and limits of electronic technology and physics. For example, a processor accessing its local memory can do so at rates over 10 GB/sec, but accessing memory attached to another processor across the facility may only be feasible at orders of magnitude lower rates. WSC software designers also need to cope with failures. Two server crashes per year may not sound absurdly high but if the software in question runs on five thousand servers, it should expect a server to fail nearly hourly. Dealing with heterogeneous hardware performance features and the need to survive high failure rates bring a additional level of difficulty to programming WSCs compared to traditional servers.

Wrap up

The explosion in popularity of Internet services as a model for provisioning computing and storage solutions has brought supercomputing-class machines beyond the domain of numerical and scientific computing problems. Some of the world's largest computing systems are the Warehouse-scale computers behind many of today's Internet services. Building and programming this emerging class of machines are among the most compelling research areas in computer systems today.

References

[Dean04] MapReduce: Simplified Data Processing on Large Clusters, Jeffrey Dean and Sanjay Ghemawat; OSDI'04: Sixth Symposium on Operating System Design and Implementation,
San Francisco, CA, December, 2004.

[Barroso07] The Case for Energy-Proportional Computing, Luiz André Barroso and Urs Hölzle, IEEE Computer, December 2007.

[Barroso09] The Datacenter as a Computer - an introduction to the design of warehouse-scale machines, Luiz André Barroso and Urs Hölzle, Synthesis Series on Computer Architecture, Morgan & Claypool Publishers, May 2009.

[Google10] Google Data Center Efficiency Measurements,
http://www.google.com/corporate/green/datacenters/measuring.html

(*) Storage systems based on FLASH memory technology (sometimes called solid state drives, or SSDs) are only now beginning to be considered for WSC systems as an intermediary layer between DRAM and magnetic disk drives.