DRAFT

Metrics of IT Equipment — Computing and Energy Performance

Bruce Nordman[1]
LBNL High Tech Buildings Project
Lawrence Berkeley National Laboratory
March 10, 2005

*Comments are solicited on this draft by **April 18** to be incorporated into a final version.*
Relevant data are also sought; these need not show absolute power or capacity.
Contact the author to be notified of updates.
This paper and the summary are available as the last links of:
http://hightech.LBL.gov/datacenters.html

Summary

This report summarizes work to date on the development of a simple, standard method of characterizing the degree to which a single server reduces its energy consumption when operating at low levels of computation compared to what it consumes at peak computing capacity (the "part-load" condition). The goal is to bring more attention and rigor to the issue, and lead to future servers which save energy by having lower power use at part load.

Section 1 provides background from the perspective of energy consumption and efficiency research and policy. Section 2 reviews key terms and background data. Section 3 discusses relevant existing benchmarks. Section 4 addresses specific issues for data centers and methods for reducing power use at part load. Section 5 reviews relevant measured power data. Section 6 describes approaches to the proposed energy vs. load metric. Section 7 addresses some related areas of work, and Section 8 presents conclusions and next steps.

1. Energy Context

A balanced portfolio of efficiency research and action on data centers should cover all major energy uses, including the Information Technology (IT) equipment itself. This is primarily servers, but also includes network equipment, data storage products, and other devices.

For the IT industry, adequately cooling data center equipment (particularly servers) is a problem, both locally within a device chassis, and more globally among racks in data centers. In addition, some equipment racks are limited in equipment capacity by the amount of power that can be provided to them rather than physical space in the rack.

The area of "energy-aware computing" [Lefurgy] encompasses more than just efficiency; for example, some systems detect temperatures or aggregate power consumption levels outside of operating limits and reduce system capacity (and power use) to bring them back into compliance[2].

On assessing server performance, "their energy efficiency is difficult to quantify given the lack of standard and agreed-upon metrics" [Felter].

---

[1] Bruce Nordman, BNordman@LBL.gov, 510-486-7089

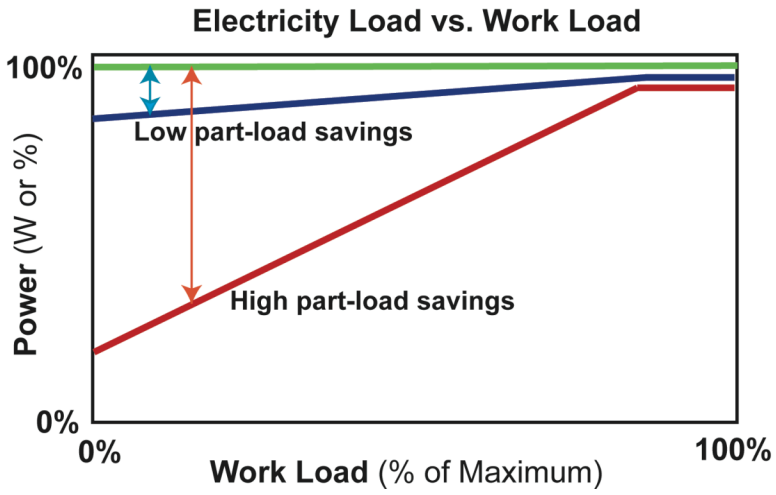
[2] In [Kant] modulating capacity for cooling purposes is called "power-out management" in contrast to "power-in management" which encompasses the mechanisms for reducing energy use for energy efficiency reasons.

A barrier to clear understanding of present consumption and efficiency opportunities is the lack of standard methods to correlate IT equipment energy consumption with the useful information processing tasks being performed. Such measurements are needed across devices (particularly servers), for the same device at different levels of activity, as well as groups of devices. This discussion is an initial effort to remove that barrier.

A standard "energy vs. load" metric would help explain how IT electrical loads vary within the envelope of maximum consumption, and show the potential for, or document the success of, mechanisms to maximize the reduction of energy consumption when IT processing loads are well below a system's maximum capability — the dominant mode for most servers.

Figure 1 presents such a metric graphically. The work load is a percentage scale, since the metric is not intended for measuring absolute system performance. Example computations are web pages served, database transactions completed, or calculations performed. The key is that the amount of computation be driven by external sources so that only a certain amount "needs" to be done at any particular time. This is unlike most scientific tasks in which large amounts of computing take long periods of time to complete, and are to be done as quickly as possible. The energy vs. load metric does not directly apply to most scientific computing.

Figure 1. Conceptual Diagram of Energy vs. Computation Metric



Computer benchmarks most commonly compare the speed of one system to another. Measurements that compare a system only to itself are inherently simpler to construct and gain acceptance for and so be used by industry. This approach — comparing a system to itself for various levels of computing activity — is the subject of this paper. Shifting future IT equipment to have higher part-load savings can save large amounts of electricity, even if the maximum consumption values do not change.

Ultimately, a simple energy vs. work load metric could be utilized by industry[3] to document the energy-saving features of their products to customers for specific applications, and could possibly be a basis for an Energy Star labeling program for data center IT equipment.

[3] Companies that we have had some contact with to date on this topic include: AMD, HP, IBM, Intel, and Sun, in addition to a number of academics.

2. Background

2.1 Terms

Capacity

A system's maximum capacity for information or data processing. Synonyms for capacity include maximum work load, and maximum throughput. Common metrics of capacity are MFlops/second, transactions/second, web content throughput (bytes/second), and web connections maintained/second.

Work Load

An actual amount of information or data processing done by a system, expressed either in absolute terms, or as a percent of a system's maximum capacity.

Power

Unless otherwise specified, power refers to AC power input to a system, or for DC-powered systems, the single supply point of such power into the power supply.

Part-load Condition

Operation of a server (or other device) at less than its maximum capacity. This term derives from HVAC equipment and is in contrast to the maximum load condition.

Part-load Efficiency

The degree to which a system's power use drops as the utilized capacity drops from the 100% maximum capacity state.

Performance States / Operating Points

Different states of a server system in which capacity (or functionality) and power consumption may vary. Common mechanisms include reducing frequency and voltage, or powering down or off parts of the system that are not being utilized. For clusters this may involve powering down some nodes while leaving others capable of performing useful work.

2.2 Server (Application) Classification

Power vs. work load metrics are most easily understood with single-processor systems with their own power cord. A single processor could have multiple cores. Multiple processor systems and clusters of coordinated servers provide for efficiency measures with how work is distributed among machines — including powering down or off some systems. Optimization techniques that operate within a single system have been called "local" in contrast with those for groups or clusters [Bianchini].

Widespread application types that lead to variable system utilization are web page serving, database query serving, and transaction processing generally. Servers in non-data center contexts also commonly see this usage pattern, such as those for printing, file storage, and email. It is common to divide commercial servers into three tiers: "front-end Web servers, application servers, and storage and database servers" [Bianchini]. The first of these are sometimes labeled "edge servers" since they are the ones that interact directly with machines elsewhere on the Internet [Felter].

The distinction is often made between transaction and batch processing. While the part-load savings potential of transaction processing is the focus of this discussion, batch activities

can also provide such opportunities if the batch program does not require the entire window of time available for its running; the system can be run at a lower performance rate and still meet the users needs[4] [Lefurgy].

2.3 Server Utilization Modes

When measuring server power, it is critical to correctly interpret and label system power and performance states to get the desired results and properly evaluate them. There are usually several distinct maximum power values, many active states, one or more idle modes, sleep states (possibly several), and several off modes. For all of these modes, short-term or transitory spikes are not of interest; rather, it is long-term stable states that contribute significantly to annual energy consumption and so merit attention for saving energy.

Maximum Power

Server manufacturers have methods ("stress tools") for driving systems to their maximum possible power consumption — levels that may not actually ever be reached in ordinary use[5]. These can be used for system rating, reliability, and safety purposes. This figure is not of interest for this discussion; rather, we are interested in the maximum power encountered in actual use. There are two different maximums that may arise: the maximum produced while doing some particular type of operation (e.g. a particular benchmark) or the maximum that can be produced with *any* ordinary application.

For a particular set of hardware and period of time there will be an "observed" maximum as used which could be considerably less than what the system as configured could be driven to. This figure depends on the time granularity of average power measurements. It is probably most appropriate to use a time period at or comparable to those used to assess stable states (see Section A.4).

One example from hardware system documentation provides figures for "typical power consumption" (that measured when "running power intensive applications") and "maximum power" ("the sum of the worst case power consumption of every subsystem in the box") with the latter for safety/power infrastructure purposes [HP]. The typical value is about 70% of the maximum. Another example provides "Typical power" figures which "may be used to assess average utility cost of cooling and electrical power" and "breaker" power levels [HP2]. In these examples, typical power is about 80% of breaker power.

Note that these are all measureable power values, in sharp contrast to "nameplate" power ratings which are not measured attributes (see Section 4.1).

Idle

One mode that is easily measurable and provides a gross indication of the shape of the power vs. load curve is the "idle" power of a machine which has no applications active (also called the "base" power of a system [Bianchini]). For systems with Microsoft Windows, this is typically called Windows Idle[6]. The appearance of this mode in servers has been called

---

[4] For example, a daily report generation routine might need to be done between midnight and 6 am but only require one hour at full speed to complete; it could be run significantly more slowly than this and still meet the 6 am deadline.

[5] Intel calls this "A power virus … an unusually intensive workload that maximizes power consumption. Most useful applications draw only a fraction of the power a power virus consumes. http://developer.intel.com/technology/itj/2005/volume09issue01/art06_interface_materials/p01_abstract.htm

[6] One source referred to this for unix systems as the "unix prompt" mode — presumably meaning that no user applications are running.

"idleness — or *slack*" [Lefurgy].

For systems which utilize sleep modes, idle time may inherently occur only in modest-sized periods of time. Idle power measurements should not include sleep modes unless the transparency of such modes in functionality and latency makes them nearly indistinguishable from idle.

Active

Active states cover the continuum from just above idle up through and including maximum values. One source refers to system states as "Active-high" and "Active-low" to distinguish among performance states [Miyoshi]. How heavily a system is loaded — how active it is — has been called the "utilization rate" [Bianchini].

Below Idle

For applications such as office file servers, print servers, or mail servers, there may be long periods of time in which there is no activity. These offer the potential for systems going into power states lower than the normal "idle" state (usually "sleep"), so long as full network connectivity is maintained. It may be reasonable to tolerate relatively long wakeup times on the relatively infrequent wakeups on these systems, so long as no data is lost.

For individual server systems, off modes are rarely relevant, though clusters can utilize these. "Hibernate" is best understood as an off and not a sleep mode (see Appendix VIII of [Nordman]).

In disk power management, the "standby" state is a higher power state than "sleep" (note that this usage contrasts with other contexts in which "standby" is usually mapped to the off state).

2.4 Server Loading

While desktop and notebook PCs can make good use of sleep modes, most servers have difficulty ever doing this for a variety of reasons.

- They often have no sufficiently long periods of no activity that would make sleep modes viable;
- Latency requirements often preclude the delay that sleep modes can create; and
- The compromise in network connectivity inherent in most current network interfaces is not acceptable.

Working towards making use of sleep modes more viable is desirable but beyond the scope of this discussion.

Most servers in commercial use perform a considerable portion of their activity based on many individual requests received over a network connection. While only a portion of commercial server use, loads driven by web-based applications are a clear and convenient example to use to illustrate the general issue.

There seems to be a clear consensus on the variable and often low nature of web loads in most commercial settings as illustrated by the following quotes:

> "Data center servers typically run at a very low average utilization." [Kant]
>
> "In reality, servers seldom operate at full load for long periods of time. Several studies have found that Web workloads are bursty in nature, and support the intuitive notion that Web servers tend to be busiest during some peak hours during the day and almost idle in others." [Felter]

> "… previous studies have observed that Web servers are relatively idle for large fractions of time." [Elnozahy]
>
> "A detailed analysis and characterization of Web server access patterns … [has] received considerably less attention in the research literature due in part to the lack of data." [Iyengar]

While there is little reason to doubt this fact, specific data to document it in the public domain are scarce and often old. For example, the [Felter] paper is from late 2003 but the reference cited on this topic is from 1996. Similarly, the [Elnozahy] paper used a source from three years before its publication. When asked about this issue, people interviewed for this paper agreed that the low loadings of commercial servers was a fact of life but generally had no relevant data to document or quantify it.

There are many reasons for the low average loading of servers, many quite justifiable for good business reasons. Among these are:

- Accommodating known peak demands (including seasonal spikes and special events),
- Accommodating hoped-for as-yet-unseen peak demands,
- Allowing for a percentage of units going off-line at any one time,
- Needing extra capacity as hardware is phased in and out,
- Having extra capacity to accommodate software changes, and
- The usual engineering "safety factor" between potential expected demand and that provisioned for.

A review of a data-intensive web service (the Microsoft Terraserver) incidentally notes the daily and weekly patterns in the volume of activity [Barclay]. The weekly minimum is about 20% of the weekly maximum value, with the average over a week being roughly half of the peak. The weekend peak is about half of the weekday peak.

[Elnozahy] examined web traces of several sites, including some for various Olympics games. As these were serving a global audience, they would be even more evenly distributed than web servers that serve primarily a national or regional audience. One of these traces was also analyzed in [Bohrer] which reports that the average load was only 25% of the *observed* peak, though the actual peak capacity of this system is not known so that the average load percentage could be well below this. Other web traces averaged from 11% to 50% of the observed peak. A graph of 7 days of data in [Iyengar] for a globally relevant web site (1998 Olympics) show average requests at about 50% of the peak.

One week of web trace for a major IT company (a U.S.-based but global) showed that average web activity was about 60% of the weekly maximum, with the minimum about 25% [Chase]. Other traces discussed had lower average loads as a percent of peak.

Network link loading (not server loading) is assessed by [Odlyzko]. He finds very low typical utilization rates of network capacity: 3-5% for private networks, 10-15% for internet backbone links, and sees this general pattern as likely to continue to hold for the foreseeable future. To the degree that server activity is driven by network traffic, variations in the traffic can be a simple and aggregate indicator of activity, and the absolute value an indirect indicator that the total amount of activity is well below capacity.

It would be helpful to have more (and more current) examples of server loading data, and some sense of how servers generally are loaded. However, lack of such data need not impede progress on part-load efficiency since there seems no reason to doubt that a huge number of servers are operating as relatively low load levels much of the time.

More virtualization of servers (including pay-as-you-go and grid computing methods) could in principle lead to higher utilization rates (though whether this will actually occur is speculative). Even if it is a significant effect, part-load conditions are likely to remain a dominant characteristic of most commercial servers.

In sum, it seems likely that many — if not most — web servers operate at around 25% of their observed peak level on average, with most of the rest probably not more than 50%. As these are relative to the observed peak, not the maximum capacity of the system, the average as a percent of maximum capacity will be considerably lower.

3. Computing Benchmarks

3.1 Benchmark Overview

A variety of sources have described what makes a good benchmark. A useful benchmark must be Relevant, Portable, Scaleable, and Simple [Gray]. Key requirements are: Linearity (proportional increases in performance result in the same proportional increase in the metric), Reliability (performance ranking by the metric should directly correspond to ranking of performance in general), Repeatability, Ease of Measurement (so that it will be used, and used correctly), Consistency (of application to different systems), and Independence (of parties biased towards a particular manufacturer) [Lilja]. One industry representative stated that a good benchmark must be: Portable, Scalable, and Platform-independent.

Benchmarks may be designed to exercise only a specific part or characteristic of a system (e.g. memory access time), or to include the effects of many parts of the hardware (including processor, memory, network interface, and disks) and software (operating system and application).

A core issue is that “While there may be some debate on whether industrial benchmarks represent actual workloads, they nevertheless serve as a fair vehicle for comparing different designs and implementations under the same conditions” [Felter].

Most scientific computing applications have a chunk of computation to be done as *fast* as possible, or a limited window of computing time in which to do as *much* computation as possible. In both cases, systems are generally run at close to their maximum capacity so that the type of part-load condition described here applies not at all, or much less than with commercial applications. For this reason, this paper does not address benchmarks designed for scientific applications.

3.2 Benchmark Types

We group existing benchmarks into three categories: Comprehensive, Simple, and Synthetic. Details are found in Appendix A.

Comprehensive Benchmarks

Comprehensive benchmarks use real application programs in realistic ways and so exercise all parts of a system. The most prominent examples currently are the SPEC and TCP families. While the SPEC family includes some benchmarks for scientific applications, both are oriented to transaction processing such as database and web applications.

Simple Benchmarks

Several tools for measuring web server performance make simplifications that make the absolute result quantity not directly useful but they provide a reliable relative index of system performance for these purposes. Hence, these are "simple, but real". Other benchmarks of this type are numeric calculation oriented, which can be combined with processor dispatch algorithms to simulate transaction system environments.

Synthetic Benchmarks

"Synthetic benchmark programs are artificial programs that do no real, useful work" but instead execute a mixture of instructions intended to reflect that done by a type of application [Lilja]. Some synthetic benchmarks are oriented to stressing particular parts of a system, e.g. memory or input/output capacity.

"Application benchmark programs" as those that "are complete, real programs that actually produce a useful result" [Lilja]. This could presumably apply to either comprehensive or simple benchmarks.

3.3 Measurement

The facilities required to conduct industry-standard performance measurements (e.g. TCP) are very expensive in terms of the needed IT equipment as well as the highly skilled professionals who run the tests[7]. This makes it a difficult proposition to conduct such tests solely for the principal reason of making energy measurements. More plausible is to add the energy measurements to already-planned performance benchmarking, either strictly recording the power use of the machine during the standard test, or also adding a few computational variations to the standard test designed to reveal the relevant energy figures.

One industry source mentioned variations from platform to platform in the capacity of a web server reported by the Webstress tool that were not readily explicable by known platform differences. This raises the question of the reliability of the peak value of such a test and hence the power values generated by percentages of the peak.

The difficulty of finding the peak capacity of a system in an event-driven context is noted in several of the papers reviewed and in discussions with individuals. Several industry sources recommended that a relatively simple, synthetic, benchmark be used for assessing how power consumption varies with load.

4.0 Goals and Technologies

4.1 Data Center Focus

The Data Center design charrette organized by the Rocky Mountain Institute in February 2003 [RMI] covered all aspects of energy use in data centers. The recommendations from that event included:

- Designing servers whose maximum consumption is less than what is typical of the market (including by substituting more lower-capacity processors for fewer higher-capacity ones),

[7] For example, one industry source reported needing 50 dual-processor clients to generate the work load for a server system in one test. On the other hand, another test simply used a client with more processing power than the web server being tested.

- Specifying higher efficiency power supplies, and
- Using higher efficiency components.

A call was made for better and standard publishing of typical power consumption values for standard configurations to be used in power and heat removal requirements rather than nameplate values which can be much higher. The recommendations also included items relevant to this discussion such as "scaling CPU power to data/load", powering down some systems when not needed, using energy consumption as a consideration in allocating workloads to machines, and creating an Energy Star specification for servers. This latter item envisioned using sleep modes for servers that experience long periods of non-use (e.g. in office environments) and added hardware to facilitate maintaining network presence in sleep.

ASHRAE Technical Committee 9.9 crafted a guide to provide information and standard conditions needed by those designing space conditioning systems for data centers [ASHRAE]. It notes the difference between the "nameplate rating" of power consumption and actual "measured power" that actually occurs, specifically:

> "Nameplate ratings should at no time be used as a measure of equipment heat release. The purpose of a nameplate rating is solely to indicate the maximum power draw for safety and regulatory approval."

Nameplate power values always exceed actual consumption, often by substantial amounts. A nameplate value may also be specified for an equipment chassis which has the ability to hold varying numbers of varying power cards. With most installations having fewer than maximum cards of less than maximum average power consumption, the difference in this case between nameplate and measured is even larger.

The discussion of measured power in [ASHRAE] explicitly draws on that for telecommunications environments (it cites Telcordia GR-3028-CORE). The power consumption values are to be derived from (among other criteria) conditions with "user controls or programs set to a utilization rate that maximizes the number of simultaneous components, devices, and subsystems that are active". The intent of this appears to be to result in the maximum power that a customer could encounter in actual use.

The ASHRAE document also calls for standard reporting of power consumption values for "representative conditions", and in an example, show an example manufacturer report which lists the power consumption for minimum, maximum, and typical hardware configurations. This is consistent with the earlier recommendation from the RMI Charette [RMI]. Several manufacturers we spoke with about this topic also agreed with the value of a standard reporting format for power consumption values for data center equipment, with servers the logical place to start. Possible values to include are: nameplate power, worst case achievable, typical maximum, typical (perhaps at 50% computational load or median of some test), and idle.

There is presently no Energy Star specification for servers (though workstations can be covered). The current proposal for the new Energy Star Computer Specification [EPA] includes "desktop-derived" servers and covers idle power and power supply efficiency. Tier II proposals include benchmarking systems to "performance per unit energy" (specific procedure not yet specified) and fixing the "network problem". The latter would allow some servers to go to sleep when the latency on waking was within performance bounds.

4.2 Mechanisms

The most commonly proposed and implemented method of modulating server power consumption in response to demand is to scale frequency and voltage (or just frequency) to match the total needed capacity. The reason for this is that a substantial portion of processor power is determined by the product of the frequency and the square of the voltage, and reducing the frequency allows for reducing the voltage [Pouwelse]. Thus, the drop in power consumption from a reduction in frequency is far more than the linear proportion of the frequency reduction, as much as 90% reduction at low frequencies (CPU only) [Pouwelse] [Rohrer].

Another method is "request batching" to halt (and even put to sleep) a server when it has no outstanding requests, accumulate a queue of requests to process, then submit the queue to the machine when it has reached a sufficient size [Bianchini]. This method was found to be most beneficial at relatively low levels of capacity factor, with frequency and voltage scaling most effective at higher levels; the combination of the two saved the most. In [Elnozahy], request batching is capped at 100 ms to avoid excessive response times.

In all cases, energy saving strategies need to crafted to not compromise performance (quality of service). Total response time for individual requests (or averages of many requests) is a common criterion. Another is the rate at which requests are not responded to by the system at all.

The concept of "Critical Power Slope" is explored in [Miyoshi]. This is based on the power performance of systems in different operating modes. The goal is to measure the effect of different "operating points" to guide system operation to those that are most energy-efficient, and away from those that actually increase consumption relative to a base case. Mechanisms they explore are frequency scaling, clock throttling, and dynamic voltage scaling (DVS).

With the wide variety of types of applications and their use, knowing in real time how much systems slowdown is reasonable or optimal is "a non-trivial task" [Pouwelse]. The operating system can make better decisions about this with relevant information from applications (assuming that they are written to provide it), and can be more ambitious in pursuing slowdown as latency allowances are relaxed.

In detailed DC power measurements of systems, [Bohrer] found memory energy consumption approximately doubling from the idle to completely busy state (though real applications might not reach the amount of memory use intensity as this test generated). They also found over a 40% drop in disk power from a busy to idle (but still spinning) state. The absolute power savings from the processor were several times the combined memory and disk potential savings.

Clusters

Power saving opportunities in clusters (groups) of servers are greater than that available for single-processor systems. One method for reducing cluster server power is to power down (to sleep or off modes) a portion of a cluster when the capacity of all systems (nodes) is not needed. This is variously called "processor packing" [Lefurgy], "load concentration" [Bianchini2], and "node vary-on/vary-off (VOVO)" [Elnozahy2]. Since the power levels of products in sleep or off are usually much lower than idle levels, the savings can be significant. In many cases there is no hardware change needed to gain these savings; rather, it is operating system or other software which monitors processing needs and manages the transitions. Also, because the time needed to bring resources online is much greater than with single-processor solutions, application-specific requirements regarding latency and speed of changes in capacity needs enter the picture, providing significant complication to generic analysis of the relative energy-

efficiency of a solution. The various methods can be used in combination, and among servers, voltage scaling can be coordinated [Elnozahy2].

The energy efficiency potential of "dense servers" in clusters is assessed in [Felter]. These are machines that individually have relatively lower performance than others, but have very low power consumption, small space requirements, and capability of powering down or off quickly (and also recovering quickly) in response to variations in demand. Their test system used Power Aware Request Distribution (PARD) to manage the powering up and down of servers and appropriately allocating requests to the machines available.

Related to energy savings from management of server clusters is the potential raised by using virtualization to map multiple logical systems into one physical one [Kant].

4.3 Technologies in Products

Frequency and voltage scaling were first introduced in processors designed for the mobile market, and have been moving into the desktop and server lines since. AMD uses the names PowerNow! and Cool'n'Quiet for their version of this method. Intel uses the terms SpeedStep and Demand Based Switching (DBS).

PowerNow! is said to "reduce CPU power at Idle by 75%" [AMD] in processors designed for workstation and server applications, with similar savings for processors designed for the desktop. Changes in operating point can be as frequent as 30 per second.

Demand-Based Switching [Intel] is said to save up to 30% of total server power (note the distinction from measurements of processor power only) — this on top of the 10% reduction in power use at idle from fully active that occurs on systems even without the technology enabled. DBS also covers frequency and voltage scaling, and is largely the same as the SpeedStep technology used in processors designed for mobile systems. While DBS covers only the processor, Intel notes that "In the future, this technology might be extended to other system components to further reduce overall power consumption" [Intel].

For the PowerPC processor, there seems to not be a marketing name for frequency and voltage scaling, but states of half and fourth of the base speed are supported, along with a "deep nap" state at 1/64 of the base processor speed [Purgatorio]. The voltage is also dropped with the frequency in these states.

Just how much of a system can be reduced in power consumption is a moving target; industry is seeking to increase the savings by covering more and more system elements. Which components are affected is not relevant to the energy/load metric; rather, it is the measured result of this that is important. For processors with multiple cores, it is possible for each to run at a different performance state [Intel2]. Intel is also investigating ways to reduce idle state power by powering down more parts of a system, and making lower power states more useable by reducing latency times to resume to full activity.

5. Data

5.1 Data in Literature

When simulating web workloads that span one or more days, it is common to speed up the process and scale the number of actual requests per second to capacity of the system being measured. This method seems reliable [Felter] among others.

Some data from the literature are based on simulations of system performance and energy use. In all cases, the authors validated their simulators against real systems to provide confidence in the results. As such, this discussion does not dwell on whether particular data are from direct measurements or simulations.

Some studies disaggregate server power consumption into major components such as memory, I/O, network interfaces, processors, etc. Understanding these can help explain why measured values vary as they do, though the metrics explored here only report the result.

Active Savings

One reported test of request batching found savings of 3.1 to 27%, from dynamic voltage scaling from 8.7 to 38%, and from the combination of the two, 17 to 42% [Elnozahy]. These tests were made with a performance criterion of "90th percentile first packet response times at or better than 50 ms".

Measurement of the effect of clock throttling on one system found that power dropped by about half when the system was run at 1/8 of compute capacity [Miyoshi]. This actually resulted in an increase in energy consumption (though only by a few percent) for a sample piece of computation due to the increased time taken to do it at the slower clock rate and the relative consumption of the idle state that the full clock rate scenario used. In a frequency reduction test, CPU power dropped by about 10% over the range tested. Use of httperf to stress a system at different rates found a drop in power of about 50% at a 7% load using frequency scaling only.

Reported Saving of 23 to 36% of energy on a sample server were obtained by using frequency and voltage scaling [Bohrer]. In these tests, the web workload was scaled so that the peak request rate experienced would raise the processor to its highest performance state. How close this was to its maximum capacity is not known, but it should be reasonably close. A test of a system at different request rates showed power savings from 100% capacity to zero to be about 50% savings. The relation at intermediate points was quite close to linear.

Idle Savings

In measurements of six different servers, idle power ranged from 64% to 83% of the maximum. Two pairs of measurements were of the same hardware model but with different operating systems, and each showed more than 5% difference in idle power depending on the operating system used (and one case showed different maximum power) [Chase].

The topic has been addressed by those in the energy efficiency community. One example shows measurements of a circa 2001 desktop PC (AMD 1 Ghz processor) being used as a print server [Calwell]. From a baseline of "Intensive Printing/Computation" of 97W, "Background Printing" uses 19% less power and at an idle state it uses 23% less power. Measurement periods are on the order of one to two minutes. Another report includes two measurements for a Dell Power Edge 2400 functioning as a Web/SQL server [White]. Power at 100% processor time usage is about 112 W; idle periods of 1-2% processor time utilization show about an 18% drop in power consumption.

In a discussion of server clusters of very low power (13W) machines, [Felter] reported that "roughly 3 W per blade could be conserved during system idle time simply by halting the processor in the idle loop".

Clusters

In one test, a 38% energy savings resulted from a test that utilized a seven node cluster, using a historic web trace as a basis that as applied, required all seven nodes during part of the test [Bianchini]. The baseline test in which no nodes were powered down dropped about 5% in power from the peak and trough load levels. The efficiency test dropped the load about 75%, with the lowest levels being served by only two of the seven nodes. Savings of 43% were found in a test with 8 nodes [Bianchini2].

[Elnozahy2] report energy savings of 20% for voltage/frequency scaling only to up to 50% when combined with powering off servers entirely when demand allows it (this for a simulated cluster of 10 nodes).

The quantitative results from [Felter] show that without a power-sensitive dispatch policy, a sample workload dropped only about 5% with a 90% reduction in demand. With the PARD mechanism in operation, power dropped about 85% in the test which utilized a system with eight servers in the cluster. Overall energy savings over the course of the simulated day were about 40%.

One test with five servers found a 29% energy savings in a web trace when systems could be powered down (the workload was scaled to require all five at peak) [Chase]. This paper also estimated savings as they increase with additional servers providing more granularity in the dispatch of resources. Even just two servers provided over 20% of savings, with a full 16 providing just under 40%.

The general result that emerges from these data are about 1/3 savings in energy in typical workloads from the use of frequency/voltage scaling. This is on top of idle power values about 20% less than that at maximum capacity in systems without frequency/voltage scaling.

5.2. Data from Manufacturers

Many manufacturers have reported to LBNL making measurements that illustrate the power vs. computation relationship. However, most of these data are considered proprietary at present and would need to be scrubbed of some information to be able to be made public. Both axes of the graph in Figure 1 can be made into percentages of the maximum value rather than reported as absolute levels of power or computation. In addition, the information about the precise application being run and system configuration can similarly be masked. This could facilitate discussion of the method by allowing more data sharing to without revealing confidential information.

Active Savings

One source reported that without extra effort to reduce power at low levels of demand, power dropped from the 100% demand level approximately linearly to be reduced about 22% when completely idle (see Figure 2). With power savings mechanisms enabled (in this case voltage and frequency scaling), power use at 100% capacity dropped about 2% (perhaps reflecting that 100% utilization is never actually fully busy) to about a 43% reduction when idle. Interestingly, most of the absolute savings over the base case occurred in the 80% utilization case, so that this reduction was decidedly non-linear.

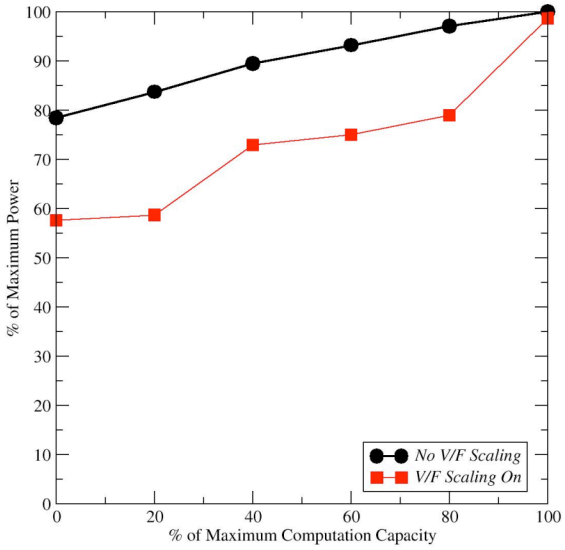
Idle Savings

Another manufacturer reported that a multiprocessor system dropped about 25% in power consumption from the fully loaded case to fully idle. A third source reported a 20% drop in power consumption on a system from an intensive computational load to an idle one.

6.0 Energy vs. Load Metrics

There are several possible approaches to a power vs. load metric: a complex benchmark, a simple-but-real benchmark, or a synthetic benchmark. In all cases the system is run full-out to determine what that peak capacity is, then caused to be driven at different "speeds" below that peak.

Figure 2. Power vs. load data for a current 4-processor server



Complex Benchmark

The SPEC and TCP families of benchmarks are obvious examples of complex benchmarks. These are the benchmarks most widely used by vendors in rating their systems. Vendors could add energy benchmarks to their existing testing of systems to minimize the extra effort such a test requires. The database transaction benchmarks are more complicated to run than the web versions, and as we don't expect that the energy vs. load relation would differ between these two cases (something that could be tested on a few systems to confirm), it would be better to use the simpler web benchmarks.

However, using these benchmarks as the basis for the standard energy vs. load test would reduce the degree to which it could be used since many individuals and organizations lack the hardware, software, and human resources to conduct them. In addition, if there is a simpler option that produces a substantially similar result, then that should be taken.

Simple Real Benchmark

Another approach (that taken by [Bohrer][8]) is to drive a web server in a simplistic way that does not require the significant infrastructure that the complicated one does. Simplifications such as repeatedly asking for the same page will not exercise disk I/O since the page can always be served out of cache. Not asking for dynamic pages changes the absolute number of peak pages servable, but might not affect the effect on power consumption and so not be an issue for a power vs. load metric.

Synthetic Benchmark

A third approach is to run a simple program (process) that does some mindless calculation (or I/O) then sleeps variable amounts of time to produce different levels of system activity. The initial case would involve the shortest possible sleep periods to simulate system saturation and obtain the 100% busy case. For this to work on real systems it may be necessary or desirable for there to be several identical processes running that do the same thing. Some understanding of how systems know when they can change performance state may be necessary to assure that the benchmark doesn't unrealistically engage or sabotage the method.

There are methods that occupy spaces between these, such as using some existing benchmark that does continuous computation and dividing it up into chunks that can be dispatched to simulate a transaction-driven environment (e.g. stream2). One source suggested that a synthetic FFT program was a good basis for a sample load.

Implementation

In both of the simpler cases, it would be highly desirable to test them on a system in which the complex test was also done to confirm that the power vs. load relation is similar enough across the tests to verify that the simpler tests are viable alternatives. It seems likely that this will be true, but confirmation is needed. This would need to be tested by several manufacturers who have complex benchmarks that have already been run at different speeds, and could readily make simple benchmark measurements on the same systems.

For timing, one minute of stabilization time seems quite sufficient before test periods, and detailed data from some initial tests could provide a good basis for abbreviating this further. This period should also be used between tests at different "speeds". One minute also seems sufficient for the test period itself, with the proviso that power should not vary within this period by more than a certain amount to reflect system stability. There seems no need for a post-test period for this purpose. Sampling of power and load about once a second is desirable.

Reporting for these tests should include the hardware configurations (including memory, disks, network interfaces, etc.), and operating system used. While measurements of relative power are informative, absolute power levels are preferred.

Interpretation

This measurement will show how a system's consumption varies with load, but does not show the absolute energy efficiency in terms of performance/watt. A system could show great variation in consumption across loading levels but do so from an inefficient performance/watt level. Thus, this metric shows a desirable characteristic but is by no means a comprehensive assessment of energy efficiency.

---

[8] In this test a web server was repeatedly sent the same URL; it showed a clear signature of the relation between power and work load, with the power varying close to linearly with load.

7. Future Work

There are several additional topic areas closely related to it that would be useful to pursue for reducing IT equipment energy use.

- Crafting energy vs. load metrics specifically for clusters of servers;
- Comparing the performance (in terms of computational work and energy consumption) of different machines;
- Developing a standard method of reporting basic characteristics of all the IT equipment in a data center or portion thereof to facilitate aggregation of information about diverse devices in a standard way (comparable to gathering basic demographic data about a population of people); and
- Applying these energy vs. load principles to network equipment and storage products.

Server Clusters

Clusters present issues for how to fairly evaluate power vs. load performance beyond that which applies to individual systems. Because individual nodes can be powered down to sleep or off, the ability to match capability to load exceeds that of individual systems. However, issues of latency need to be more carefully specified or reported.

Inter-machine comparisons

A single, simple, universal standard for assessing server absolute energy efficiency would be particularly useful. However, no such metric presently exists, and whether consensus on one could ever be arrived at is speculative. Many barriers exist, including the diverse applications and performance needs of customers. However, exploring this topic further may provide useful results, and metrics which usefully assess energy performance for specific domains.

One paper proposes a metric of SpecWeb/Watt [Bohrer]. Another defines the metric of "power efficiency", as "the benchmark rating, divided by average power consumption in watts", with a variety of benchmarks suitable for the task [Felter]. As with miles per gallon ratings for autos, greater efficiency leads to higher values for the metric.

Network Equipment

For network equipment (switches, routers, firewalls, etc.), there may be similar opportunities to those with servers to scale capacity to demand through voltage and frequency scaling. While powering off network links is unlikely except in peculiar circumstances, reducing the data link rate may be relatively easy to implement and have significant national savings — especially in residential and commercial environments. Dynamic link data rate reduction has been proposed and outlined [Gunaratne] for times of low traffic, and saves energy at both ends of the network link (switch and server). As network speeds increase, the savings per link from this method rises dramatically. If switch hardware that could go to sleep during times of low or no traffic, there is clear possibility to save significant energy without compromising performance [Gupta].

Storage

Storage system energy consumption is not addressed in this paper, but has some energy savings opportunities. For example, [Carrera] report up to 23% savings in disk energy by use of multi-speed disks. The potential for savings in storage likely depends greatly on the nature of the stored data and access patterns to it.

8. Conclusions

It is clear that many commercial servers operate at low levels of activity for much of the time. No current standard metric shows how this affects power consumption for current products, or could do so for future ones designed to exploit this fact. There is a need for such a metric and for clear and consistent definitions of relevant terms. There are a variety of benchmarks that could be applied to the problem. The simplest one that correctly reflects system performance should be selected and then used.

9. References

[AMD] AMD PowerNow!™ Technology with Optimized Power Management (OPM) – Coming First Half of 2005, www.amd.com/us-en/0,,3715_12353,00.html (accessed February 22, 2005).

[ASHRAE] ASHRAE, Thermal Guidelines for Data Processing Environments, American Society of Heating, Refrigerating, and Air-Conditioning Engineers, 2004.

[Barclay] Barclay, Tom, Jim Gray, and Wyman Chong, TerraServer Bricks — A High Availability Cluster Alternative, Microsoft Research, October 2004, MSR-TR-2004-107.

[Bianchini] Bianchini, Ricardo, and Ram Rajamony, Power and Energy Management for Server Systems, IEEE Computer, November 2004.

[Bianchini2] Bianchini, Ricardo, Research Directions in Power and Energy Conservation for Clusters, Rutgers University, DCS-TR-466, 2001.

[Bodas] Bodas, Deva, New Server Power-Management Technologies Address Power and Cooling Challenges, Technology @ Intel magazine, 2003. www.intel.com/update/contents/sv09031.htm

[Bohrer] Bohrer, Pat, Elmootazbellah N. Elnozahy, Tom Keller, Michael Kistler, Charles Lefurgy, Chandler McDowell, and Ram Rajamony The Case For Power Management In Web Servers, Chapter 1 of Power-Aware Computing (Robert Graybill and Rami Melhem, editors). Kluwer/Plenum, 2002.

[Calwell] Calwell, Chris, Impact of Duty Cycle and Load Conditions on PC Energy Use, Ecos Consulting, March 18, 2004.

[Carrera] Carrera, Enrique V., Eduardo Pinhiero, and Ricardo Bianchini, Conserving Disk Energy in Network Servers, Rutgers University, ICS 2003.

[Chase] Chase, Jeffrey S., Darrell C. Anderson, Prachi N. Thakar, Amin M. Vahdat, and Ronald P. Doyle, Managing Energy and Server Resources in Hosting Centers, 2002.

[Chen] Chen, J. Bradley, Andrew Wharton, and Mark Day Benchmarking the Next Generation of Internet Servers, 1997. www-128.ibm.com/developerworks/lotus/library/ls-Benchmarking_Internet_Servers/

[EPA] Energy Star Program Resquirements for Computers — Eligibility Criteria: Preliminary Draft, 2005. www.energystar.gov/index.cfm?c=revisions.computer_spec

[Gunaratne] Gunaratne, Chamara, Ken Christensen, and Bruce Nordman, Managing Energy Consumption Costs in Desktop PCs and LAN Switches with Proxying, Split TCP connections, and Scaling of Link Speed, to appear in the International Journal of Network Management, 2005.

[Elnozahy] Elnozahy, Mootaz, Michael Kistler, and Ramakrishnan Rajamony, Energy Conservation Policies for Web Servers, IBM Research (Austin), 2002.

[Elnohazy2] Elnozahy, E. N., M. Kistler, and R. Rajamony, Energy-Efficient Server Clusters, from Proceedings of the Second Workshop on Power Aware Computing Systems, 2002.

[Felter] Felter, W.M. T.W. Keller, M.D. Kistler, C. Lefurgy, K. Rajamani, R. Rajamony, F.L. Rawson, B.A. Smith, and E. Van Hensbergen, On the Performance and Use of Dense Servers, IBM Journal of Research and Development, Vol. 47, No. 5/6, September/November 2003.

[Gray] Gray, Jim (editor), The Benchmark Handbook, 1991, Morgan Kaufman.

[Gupta] Gupta, M., S. Grover and S. Singh, " A Feasibility Study for Power Management in LAN Switches", Proceedings of the 12th IEEE International Conference on Network Protocols, October 2004. http://www.cs.pdx.edu/~singh/papers.html

[HP] Site Preparation Guide: HP Integrity rx8620-32 Server, Third Edition, A7026-96016, May 2004. http://www.docs.hp.com/en/A7026-96016/A7026-96016.pdf

[HP2] Site Preparation Guide: HP Integrity Superdome and HP 9000 Superdome, Fourth Edition, A5201-96032. http://www.docs.hp.com/en/A5201-96032/A5201-96032.pdf

[Intel] Meeting the Power Challenge through Chip Design, Intel Technology and Research, 2005. www.intel.com/technology/silicon/power/chipdesign.htm

[Intel2] Intel, Designing for Power: Intel Leadership in Power Efficient Silicon and System Design, 2005.

[Iyengar] Iyengar, A., and Mark Squillante and Li Zhang , Analysis and Characterization of Large-Scale Web Server Access Patterns and Performance World Wide Web vol. 2 #1, 2, June 1999.

[Lilja] Lilja, David J., Measuring Computer Performance : a practitioner's guide, Cambridge University Press, 2000.

[Kant] Kant, Krishna and Prasant Mohapatra, Internet Data Centers, IEEE Computer, November 2004.

[Lefurgy] Lefurgy, Charles, Karthick Rajamani, Freeman Rawson, Wes Felter, Michael Kistler, and Tom W. Keller, Energy Management for Commercial Servers, IEEE Computer, December 2003.

[Miyoshi] Miyoshi, Akihiko, Charles Lefurgy, Eric Van Hensbergen, Ram Rajamony, and Raj Rajkumar, Critical Power Slope, Understanding the Runtime Effects of Frequency Scaling, ACM ICS '02, 2002.

[Mosberger] Mosberger, David, and Tai Jin, httperf — A Tool for Measuring Web Server Performance, 1997. http://www.hpl.hp.com/personal/David_Mosberger/httperf.html

[Nordman] Nordman, Bruce, The Power Control User Interface Standard, prepared for the California Energy Commission, Public Interest Energy Research Program, 500-03-012F 2002. http://www.energy.ca.gov/pier/final_project_reports/500-03-012f.html

[Odlyzko] A. Odlyzko, Andrew, Data networks are lightly utilized, and will stay that way, Review of Network Economics, Vol. 2, No. 3, pp. 210-237, September 2003.

[RMI] Design Recommendations for High-Performance Data Centers, report of the Integrated Design Charrette, February 2003, prepared by the Rocky Mountain Institute, http://www.rmi.org/sitepages/pid626.php

[McCalpin] McCalpin, John D., Stream, www.cs.virginia.edu/stream/Code/stream_d.c

[McCalpin2] McCalpin, John D., The STREAM2 Home Page, www.cs.virginia.edu/stream/stream2

[Microsoft] MS Web Application Stress Tutorial, www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webtutor.mspx

[Microsoft2] MS Web Application Stress Tool, http://www.microsoft.com/technet/archive/itsolutions/intranet/downloads/webtutor.mspx

[Pouwelse] Pouwelse, Johan, Koen Landgendoen, and Henk Sips, Dynamic Voltage Scaling on a Low-Power Processor.

[Purgatorio] Purgatorio, Helena, Improvements in power management techniques in IBM PowerPC microprocessors, June 2004 – New Product Focus, 2004. www-03.ibm.com/chips/products/powerpc/newsletter/jun2004/newproductfocus.html

[SPEC] SPEC, SPECweb99 Release 1.02 Run and Reporting Rules, version 2.9, 2003. www.spec.org/web99/docs/runrules.html
[Strahl] Strahl, Rich, Load Testing Web Applications using Microsoft's Web Application Stress Tool, www.west-wind.com/presentations/webstress/webstress.htm, 2000.
[TPC] Transaction Processing Performance Council (TPC), TPC Benchmark W (Web Commerce), Version 1.8, 2002. www.tpc.org
[White] White, Steve, Power Consumption Versus CPU Activity, EPRI-PEAC, personal communication, December 2004.
[Webstone] Mindcraft — Webstone Benchmark Information, www.mindcraft.com/webstone

Appendix A — Benchmark Details

A.1 Comprehensive Benchmarks

Two benchmark families that use real application programs (in realistic ways) and so exercise all parts of a system are SPEC and TCP.

The SPEC series of benchmarks began in 1988 with numeric computation tests (e.g. SPECint and SPECfp) but more recently have had web serving and java tests added. Tests using the SPECweb99 benchmark are to be "meaningful, comparable to other generated results, and repeatable" [SPEC]. Systems are not to be optimized in ways that improve benchmark results without providing comparable benefit to similar applications generally. Compliant tests are those in which no requests are lost or exceed the response time criterion. Each test is an "iteration" and three must be performed. The result is the median of the three tests, and is expressed as the number of simultaneous connections that can be maintained. Reporting of hardware includes the "System model number, type and clock rate of processor, number of processors, and main memory size" along with information about memory, disks, etc. In addition, both operating system and application software information is required, as are characteristics of the network connection. Finally, details of the load generating clients are also required.

The TPC (Transaction Processing Council) began in 1988 with benchmarks of commercial systems for accessing and updating databases; it also has had web tests added in more recent years (beginning in 1998). For TPC-W, among the performance data to be collected and reported beyond the web serving metric is CPU Utilization, which an example graph shows to be between 60 and 100% for various sub-tests. [TPC]

One source stated that SPECweb99_SSL was a better benchmark than the original SpecWeb as the SSL version is more realistically cpu-centric. An updated SPECweb2005 test is in development. Another source recommended TPC-C and TPC–W (a database and web benchmark respectively) for power benchmarking.

A.2 Other Web Benchmarks

The program httperf, dates back to at least 1997 [Mosberger]. The paper notes the value of explicitly supporting multiple clients running httperf to be able to provide a sufficient quantity of requests to test a high powered server. One issue the authors caution to be aware of is configuration limitations on the client or server that may limit system capacity artificially, such as TCP port space, open file descriptors, or socket buffer memory.

In 1999, Microsoft developed the Web Application Stress (WAS) tool to "realistically simulate" client loads that a server might experience [Microsoft, Microsoft2]. The tool can be run on as many client machines as is necessary to reach the limit on the server, but it was designed to minimize the number of discrete clients needed. Data reported in its use include the number of requests processed per second as well as the processor time utilization in percent of the clients and server. Latency is reported is Time To First Byte (TTFB) — from the request to receipt of response and Time To Last Byte (TTLB). The tutorial also suggests that above 80% processor utilization, the clients may become unstable, and that if the server reaches 80 to 85% utilization, then it may have reached its peak sustainable capacity. Another source [Strahl] categorizes a system as overloaded once the processor runs "close to 100%" and response times exceed 10 seconds. may be run close to 100%. The WAS tool is intended

for “performance testing, stability or stress testing, [and] capacity planning”. Only performance testing is relevant to this energy analysis.

For both WAS and httperf, as requests made rises, requests fulfilled rises in tandem until system capacity is reached, after which it will modestly drop as the unfulfillable requests take some time out of fulfilling those that can be. In an example in the httperf documentation, failed requests actually begin to appear well in advance of the maximum capacity being reached, showing the importance of specifying such performance criteria.

A web test tool that seems to have fallen into disuse of late is DBench [Chen]. It was developed to exercise more of web server systems than other tools did at that time (1996), and used real web trace log data as the basis for tests rather than more predictable generated load from other tests.

Other software for generating web server test requests include Webstone (which dates back to 1995 and seems to lack many of the features of httperf and WAS [Webstone]) and WebBench (which is no longer supported by Veritest).

Peak performance has been defined as the maximum number of transactions served per second while maintaining the 90^{th} percentile of response time within the specified performance requirement [Felter]. The 95^{th} percentile criterion has also been used [Bohrer]. In DBench, peak capacity can be reached when requests fail to be served, when average response time exceeds a limit (5 seconds), or maximum response time exceeds another limit (10 seconds) [Chen].

A.3 Other Small-scale Benchmarks

The Stream2 program shows the capability of a particular machine in regard to memory access by accessing memory rapidly over varying sizes of arrays [McCalpin2]. It covers four different sub-benchmarks and results show memory speed for varying amounts of memory accessed during the test. It is based on an earlier program called (not surprisingly) Stream [McCalpin]. While Stream2 is not directly useful for measuring power dependence on load as it attempts to run full-out, it could be utilized with process sleep functions to be the synthetic load for a simple power benchmark. Stream dates back as least as far as 1995 and Stream2 to 1999.

Among others, IO Meter stresses input/output capacity (www.iometer.org), and MLBench is a collection of programs designed to stress the CPU.

One source stated that power consumption values for LinPack (a set of benchmarks focusing on scientific calculations) and stream2 were similar.

A.4 Timing Issues

Any standard measurement system needs to have criteria for how the data are collected over time to assure that they are stable and reflective of long-term performance. For these types of tests there are several key times: the duration a system is to be run before measurement to assure that it has reached a stable condition; the length of the actual measurement test period; and any possible time of running after the test. In addition, there is the rate of data sampling within the test period beyond simply the total energy consumption.

Pre-Test Period

For SPECweb99, when running a test, systems are to be run for at least 20 minutes of

"warmup time" before any test, and have a 5 minute gap of "rampup time" between successive tests [SPEC]. Webbench specifies a 30 second "ramp up" [Chen].

Test Period

For how long the test period is to be, SPECweb99 refers to each test as an "iteration" and three must be performed. Run times for each test are to be 20 minutes.

The TPC-W specification [TPC] provides for calculating the figure of merit of performance for relatively short intervals (a maximum of 30 seconds) but provides that at least 60 of these must be reported and graphed to show consistent system operation. Sampling of performance data should be once per second or faster.

For httperf, an example test duration of three minutes is mentioned [Mosberger]. A Webstone test run is to be of at least 10 minutes in duration [Webstone]. A Webbench test is 5 minutes long [Chen].

One manufacturer stated that power measurements in their laboratory are typically done over a 2-5 minute period, though when presented with constant load drivers, the systems generally quiesce within seconds.

Data in [Bohrer] were collected for 30 second intervals at each request rate and from a graph of power over time it is clear that the system equilibrates within seconds to the new power level.

Post-Test Period

SPECweb99 specifies 5 minutes after the last test of "rampdown time" and Webbench 30 seconds of "ramp down". Why such a period is even necessary is not clear.

Sampling

The httperf discussion uses a 5 second time limit for when a connection request is deemed to have failed. httperf records throughput and error statistics every 5 seconds so that the variation in these in the course of a test can be observed.

[Felter] recorded power and utilization figures once per second. The WAS tutorial [Microsoft1] suggests data sampling interval of 5 seconds.

Cluster-specific data

In [Elnozahy2], nodes in a cluster are powered on and off, which is assumed to take 30 seconds. Thus, a delay period of 60 seconds is utilized before reassessing system state to allow time for equilibration. In [Bianchini2], bringing an node in a cluster from off to available takes about 100 seconds (shutting it down takes 45).