Standard Performance Evaluation Corporation (SPEC)

Power and Performance Benchmark Methodology V2.2

7001 Heritage Village Plaza, Suite 225
Gainesville, VA 20155,
USA

SPECpower Committee

Table of Contents

SVN Revision: 1158

SVN Date: 2014/12/03 18:56:18

1. Preface

The latest version of this document can be found on line at http://www.spec.org/power/docs/SPEC-Power_and_Performance_Methodology.pdf

1.1. Who Should Read This Document

This document is intended for performance benchmark designers and implementers who want to integrate a power component into their benchmark. The document may also serve as an introduction to those who need to understand the relationship between power and performance metrics in computer systems benchmarks. The assumption is that the business model and benchmark application are already selected and may already be implemented. Guidance is provided for including power metrics in existing benchmarks, as well as altering existing benchmarks and designing new ones to provide a more complete view of energy consumption.

1.2. Industry Demand for Power Improvements

Over the years, computing solutions have become less expensive to purchase and maintain – delivering more and more computing capacity at lower and lower equipment and operational costs. At the same time, the cost of energy has continued to rise. In some areas the amount of power that is available can no longer grow with the demand. We are at the point where the cost of electricity to run and cool computer systems exceeds the cost of the initial purchase.

This phenomenon has initiated a shift in thinking for Information Systems investment and management. In some cases, a statement is made that new computing solutions can only be accepted if they require no more power than their predecessors. In other cases, there is simply recognition that it is good business to not use excess power at times when maximum compute power is not required. The real shift in philosophy comes with the recognition that it can be satisfactory to not have the maximum compute power be instantly available, as long as the quality of service delivered meets a predetermined standard – One would not keep their automobile running in the garage all night, just for the benefit of being able to leave a little quicker in the morning, so why run a computer system at full capability just so that the very first transaction request can be satisfied as quickly as possible?

The result is that, in a growing number of cases, “satisfactory performance at lower energy cost” may be a better business answer than “always the best performance possible”.

In addition to business reasons for energy conservation, the energy costs of computing have caught the attention of a variety of government and standards groups – each recognizing that there are opportunities in the area of Information Technology (IT) to improve both economic and environmental demands of computing solutions.

1.3. Disclaimers

While these disclaimers are written in terms of this methodology, similar disclaimers should be included in any benchmark that is created using this methodology.

1.3.1. Use of Benchmarks Defined With This Methodology

The purpose of any general benchmark is to provide a comparison point between offerings in a specific environment. In performance benchmarks, it is important to select a benchmark that relates well to the desired environment for a computer installation. Different solutions perform differently in various benchmarks, and a single benchmark cannot be considered to be a direct indicator of how a system will perform in all environments.

The same statement is true for a metric that combines performance and power. Since performance is a very significant part of this equation, it is important to consider the workload being performed by the benchmark and to ensure that it is relevant.

Power metrics add an additional dimension to this caution. Power characteristics depend heavily on both workload and configuration. Thus, when examining power metrics between two systems, it is important to know whether the configurations of the systems being compared are relevant to an environment that is important to you.

The use of a performance benchmark or of a power-&-performance benchmark should be focused on comparison of solutions, and not for specific planning purposes. Unless the workload and configurations of the benchmarked solution match your planned solution, it could be very misleading to assume that a benchmark result will equate to reality in a production data center.

Benchmark designers can help to prevent customers and analysts from drawing inappropriate conclusions by creating power-related metrics that allow easy comparisons, but that do not readily lead to predictions of real power characteristics in a customer environment.

1.3.2. The Relationship Between Performance Benchmarks and "Reality"

Performance benchmarks, especially those whose measure is based on some amount of throughput in a given period of time, are notoriously "steady state" and they are invariably measured at the highest system utilization possible. They are designed this way to ensure consistent and repeatable measurements and to show the most positive view of performance possible. However, except for those rare, compute-intensive environments where there is always more work available than there is resource to address it, IT computing environments are almost exclusively NOT steady state and are almost exclusively NOT run at 100% of the available compute potential of the system. There can be significant differences between power characteristics for a computer system at high utilization compared to one operating at low utilization. There can also be significant differences in the way a power management feature manages power at a low, but steady utilization point, compared to the way it manages power in an environment where the workload fluctuates rapidly between high and low volumes.

The description or design document of a benchmark used for power measurement should include a statement of the amount of variability that is included in the workload – both in the range of workload demand levels measured and in the dynamics employed to simulate fluctuations in workload demand.

1.3.3. The Relationship Between Power Benchmarks and "Reality"

While performance is often variable, even in controlled benchmark environments, there can be even more variability in measures of power. Computer components tend to be designed to achieve at least some specified performance with at most some specified power requirement. However, the "at most" power limit (sometimes called the "Name Plate" power) is usually quite liberal, allowing significant latitude in the power requirements of subcomponents within the unit.

This means that, even if a performance per watt benchmark delivers extremely consistent results from measurement to measurement on the same configuration, it may not deliver the same result on an identically configured system that includes physically different components. The effect of these potential component or subcomponent differences will be most dramatic in benchmarks that only require simple configurations – where differences in processors or memory or storage may show an exaggerated affect.

Consumers must be informed that performance per watt benchmarks do not represent the exact power consumption that they will see in their own data centers. Such benchmarks provide an indication of relative efficiency, but are not good sizing tools for other environments.

Benchmark owners should include fair benchmarking rules that suggest that implementers do not sort the components that they test so as to misrepresent the average characteristics of the components that they sell.

1.3.4. Benchmarks Tend to Represent New Configurations.

Computer systems typically perform better when they are new than after they have aged for some time. Contributing reasons for this are the scattering of data, alteration of code paths from applying fixes to operating system and software packages and the increase in the size of information that is processed over time. Similarly, power requirements for aged systems may not be the same as those of newly configured systems. Error correction routines for storage and memory may, over time, require slightly more power to accomplish the same work. Other components may also become less efficient over time.

The intent of this methodology is to assist with the definition of benchmarks that can be used to compare systems that are configured in controlled environments. While such a benchmark might be used to track the relative power requirements of an individual configuration from month to month or year to year, no assumption should be made that the results generated on a new system will be comparable to those generated on a system that has been allowed to age for several years.

2. Introduction

2.1. Key Elements of a Power and Performance Benchmark:

A performance benchmark requires:

- An application or specification representing an application, usually satisfying a particular business model
- A method for driving the application in a consistent way, including ways to ensure that the system under test (SUT) is in a similar state at the start of each benchmark run
- A definition of the metrics of the benchmark and how they are derived
- A set of rules that provide for reasonable and fair comparison between results, possibly including restrictions on what the SUT configuration may be, how results will be reported and how information from the benchmark may be used in public comparisons.

To add measures of power to a performance benchmark requires:

- A decision on what changes, if any, are needed to measure the benchmark application with power – perhaps at points where the SUT is not exercised at peak capacity.
- A more complete definition of the SUT, to know what components must be included in the power measurement.
- A means for measuring power of the SUT, including definitions of what measurements are required, what instruments are acceptable to make the measurements, and a means for selecting what instruments are acceptable.
- Environmental limits that may be required to ensure consistent and comparable power measurements.
- Additional methods for driving the benchmark, so as to collect both performance and power information.
- Further definition of metrics to include power metrics, such as performance/watt or watts-at-idle.
- Additional rules for reporting and comparing data associated with the benchmark.

Each of the items in the above list is covered in a separate section of this document.

2.1.1. Wide Variety of Computing Environments

Both to understand where we are today and to quantify future improvements, it is important to have relevant benchmarks that can quantify the relationship between power requirements and computing capacity. This paper defines a methodology that can be followed to define a benchmark for a given business model. Since multiple business models are satisfied in the IT environment, it is assumed that multiple benchmarks will be required to represent “all” of IT.

For example, some environments have application servers that require very little disk and only minimal memory and fairly low networking capabilities; some environments may store a fairly small amount of information on disk, but may require large amounts of memory to process the information that are sent from other servers; some environments process and retain large amounts of data on storage devices, but need relatively little in the way of processing power; and some environments place a heavy load on all of the processor, memory, and disk.

2.1.2. One Benchmark Satisfies Only One Business Model

It is not possible for a single benchmark to represent the energy efficiency for all of the possible combinations of an IT environment. Even a benchmark that merges components from several environments is limited. Similarly, it would be very difficult to use benchmark results to predict the power characteristics of a specific IT environment, unless the environment is very closely replicated by the configuration of the benchmark. Furthermore, an active consumer environment will almost certainly use computing resources in ways that are different from a benchmark – in the configuration required to accomplish the workload, the variation of resource usage, the periods of the day when usage varies and the rate of change between usage patterns.

However, it is very possible for a single benchmark to give an indication of the relationship between power and performance for the components that the benchmark stresses. A benchmark that focuses primarily on the processor within a server should not be ignored, just because it does not focus on disk or networking. Rather, it should be viewed as a focused view of the processor component, with the knowledge that there may be other components that are not quantified in the benchmark. Because a complete view of the relationship between performance and power cannot be gained from a single benchmark, SPEC recommends that information be gleaned from multiple benchmarks that measure power and performance over a spectrum of computer configurations.

3. Defining Power Components within Performance Benchmarks

3.1. Types of Performance Benchmarks

Most performance benchmarks fit into three categories:

- Throughput-based benchmarks, where a steady flow of work requests is used to determine the number of requests that can be satisfied in a given period of time;
- Time-based benchmarks, where a specific amount of work is timed from start to finish; and
- Hybrid benchmarks, where the performance metric is a composite of multiple measurements of either Throughput-based or Time-based benchmarks.

It is possible to include a power component to the benchmark metrics for each of these because they all have distinct definitions for the beginning and end of a measurement period. Hybrid benchmarks may create a challenge in defining a power-related metric, but likely no more than that of defining the original performance metric.

The business model for Throughput-based benchmarks is often based on some kind of transactional load. Such business models tend to be dynamic in real cases, with some periods where the transaction flow may be very high and other cases where it may be near zero. It is strongly recommended to alter the benchmark driver to request work at intermediate points where the system is not running at maximum capacity, so that power characteristics at these intermediate points can be quantified. This may not be feasible for all Throughput-based benchmarks. However, valuable information will be gained if measurements can be made at intermediate points.

The business model for Time-based benchmarks is often based on generation of a specific amount of work to be accomplished, and measuring the duration of time to accomplish that work. It will often only be possible to measure power at peak capacity of the active benchmark.

3.2. Active-Idle

For all types of benchmarks, it is important to measure at least the power used while generating the maximum performance metric and the power used during an Active-Idle period, where the system is ready to do work, but has not done work for some period of time. These are the logical best and worst cases for work done per unit of power, and can serve as high and low bounds to define the power characteristics of a system when configured for the business model of the benchmark

The treatment of “Active Idle” in this methodology is meant to simulate the environment that results when there are no work requests and there is sufficient time available for the system to “cool down”. How “idle” a system can be should depend on the business model of the benchmark and care should be made to define this. The system must be “ready for work”, with all processes needed to accomplish the work of the benchmark enabled, but there are degrees of readiness. For example, a benchmark that focuses on typical “daytime” activity may represent a system that sits idle for extended nighttime periods, when the system reaches near hibernation state, perhaps even with storage devices spinning down and with memory and cache being cleared. However, most benchmarks will likely focus on an environment where requests could come at any time, day or night. In this case, the idle state should be very similar to the active state, with the ability to rapidly react to a transaction request.

To define the level of “idleness” of Active-Idle, benchmark implementers could define two quality-of-service characteristics:

1) What quality of service (i.e. response time) is expected of the first transaction request that is initiated in the “idle” state?
2) How quickly must the system be able to shift from Active-Idle to full performance capabilities?

Implementers need to determine the most appropriate time to measure Active-Idle. If the business model of the benchmark includes rapid response from idle, a measurement that begins shortly after (perhaps 1-2 minutes) the performance run might be appropriate. If the model includes long periods of idle time, where immediate response to a first request might not be expected, it might be appropriate to delay 30 or 60 minutes before measuring a “less than active Active-Idle.” To promote repeatability, a specific time delay should be specified, however. The typical choice may be some period after all work for the performance measurement has been completed, so that there will be confidence that the system is truly in a “ready” state. However, there are valid reasons for choosing a period prior to the measurement, a period in between other measurement intervals, or a period much later than the end of the performance measurement.

Once the decision for when and how Active-Idle should be measured is made, it is recommended that the measurement be built into measurement automation, so that it is measured consistently for all benchmark runs.

The role of Active-Idle in the performance per power metric is also dependent on the benchmark and its associated business model. As mentioned, above, it is important to measure power use during Active-Idle because it represents an endpoint in the spectrum of performance per watt measurements. However, the use of an Active-Idle measurement in the calculation of a primary metric for the benchmark should depend on its importance in the business model. It may be a critical component of the overall performance per power metric. It may also be excluded from this metric and only reported as a supplementary, stand-alone piece of information.

3.3. New Benchmark Development Efforts:

If we accept the premise that the measure of power characteristics will continue to be a key element of the evaluation of the performance capabilities of a computing solution, we should expect to integrate power measurement methodologies into the base design of future benchmarks. It is preferable to maintain a single benchmark development path to satisfy both performance and power measures, rather than to maintain dual paths for each benchmark environment.

New benchmark developments should include the steps listed in the clauses below this section. The key item that is different from “traditional” benchmark development is the ability to drive the workload in a controlled manner at a rate that is less than the maximum. Including this and the other concepts listed below will provide a benchmark that can deliver more relevant information to the customer and that will have greater flexibility for academic exercises, as well.

3.4. Existing Benchmark that Can Run only at = 100%

Generally, a time-based benchmark, where a given amount of work is timed to completion, can be run only at maximum performance. It may also be the case that the controls for a throughput-based benchmark are complex enough that intermediate measurement intervals would be difficult to define and regulate for consistently comparable measurements.

In order to accommodate this methodology for including power metrics with the performance benchmark, the application that initiates the benchmark run and that collects the performance information during or after the run must be enhanced to include Active-Idle “runs” and collection of power information. If a control program does not exist, one should be invented. The sequence of events is:

1. System is made ready for measurement, including some routine that ensures that the system is fully ready to execute the benchmark application, such as the execution of a small portion of the benchmark.
2. Benchmark harness application collects environmental data, as appropriate and automated.
3. If required by the benchmark, harness starts the benchmark warm-up cycle
4. Harness starts power and thermal measurements.
5. Harness starts benchmark run
6. Benchmark completes
7. Harness ends collection of performance data
8. Harness ends power and thermal collections
9. If required by the benchmark, harness executes the benchmark ramp-down cycle
10. Harness delays an appropriate period experimentally determined to be sufficient to achieve an idle state.
11. Harness starts power and thermal measurements
12. Delay a minimum period (e.g. SPECpower_ssj2008 uses 240 seconds) for Active-Idle measurement
13. Harness ends power and thermal collection
14. Benchmark harness application collects environmental data, as appropriate and automated
15. Harness post-processes performance, thermal and power data

Comment: Power at Idle is measured because even systems designed to run at maximum capacity sometimes stand idle. For a given benchmark, the owners of the benchmark can decide whether the benchmark metric is to be defined using the idle power, as shown in the first example of clause 7.2 or with only the benchmark run, itself. However, the power at idle should always be included as at least a reported value in the benchmark reports. For example, the Transaction Processing Performance Council's TPC-Energy specification (http://www.tpc.org/tpc_energy/default.asp) requires that the primary metric be computed based only on the power measured at maximum throughput, but also requires that a measurement be made and reported for a configuration in an idle state.

3.5. Existing Benchmark Capable of Graduated Throughput Levels

Benchmarks that are driven by some mechanism to deliver a throughput result are good candidates to adjust to drive at a variety of load levels. Since power consumption will likely vary at differing load levels, such benchmarks should be enhanced to allow the drivers to request work at intermediate points between zero and the maximum throughput value.

For a benchmark that has a measure based on throughput, the sequence of events is:

1. System is made ready for measurement.
2. Harness starts environmental measurements
3. If required, initiate calibration process to determine maximum throughput
4. Compute intermediate measurement targets relative to maximum throughput
5. Iterate:
    a. Harness starts benchmark segment run at throughput interval X, where X begins at the highest target throughput and reduces each iteration until a zero-throughput interval can be measured, to obtain an Active-Idle measurement
    b. Delay as needed for benchmark synchronization and to achieve steady state
    c. Harness starts power measurements
    d. Harness or benchmark collects power and performance metrics.
    e. Harness ends collection of performance and power measurements
    f. Delay as needed for benchmark synchronization
    g. Benchmark segment completes
    h. Harness delays as needed for synchronization
6. Harness ends environmental measurements
7. Harness post-processes performance and power data

3.5.1. Defining the intermediate measurement intervals

Methods for intelligently determining a load level as a percentage of peak workload are reasonably easy to develop, and whenever possible, should be used to allow multiple runs that deliver consistent results.

Note: The intermediate intervals should be based on the maximum throughput value for the system under test, and not on any measured utilization. For example, if a 20% measurement interval is desired from a benchmark that can achieve 55,555 transactions per hour at 100% of system capacity, the benchmark should be run at 11,111 transactions per hour, even if the processor utilization shows 15%, or 35%. Not all hardware and software architectures quantify processor utilization in a way that is comparable at low utilization levels. This is particularly true when a power management feature is employed that actually slows the processor in lightly loaded situations. The only measure that can be viewed with confidence is the percent of maximum throughput.

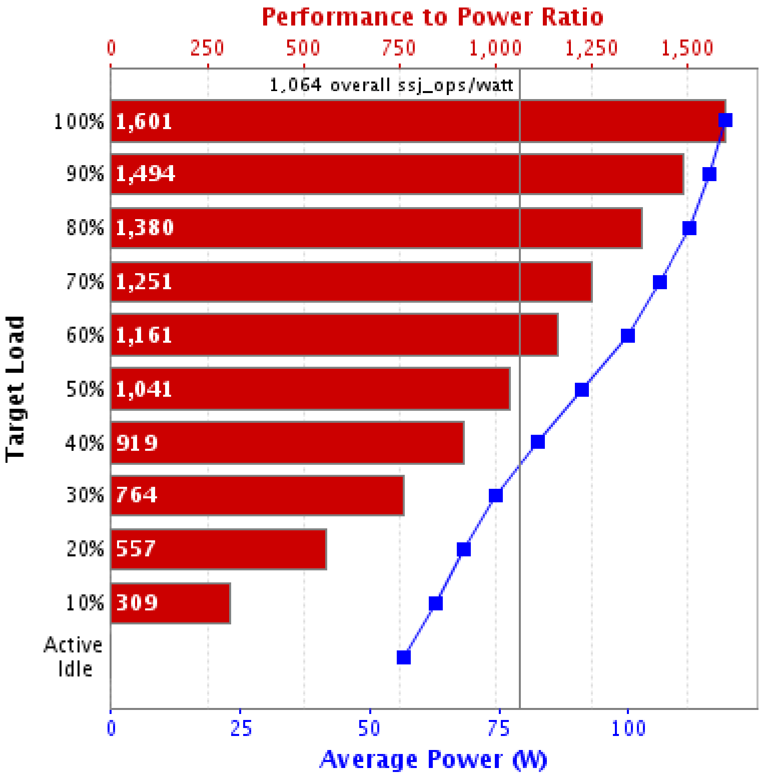
Since the benchmark was not designed initially to accommodate running at a range of throughputs, it is likely that the benchmark code, or at least the driver code, will require alteration to achieve this. This means that the performance results of the altered benchmark may not be comparable with those of the original. Clearly, the ideal method is to design the initial benchmark with intermediate measurement intervals in mind, so that power metrics can be achieved without impacting the benchmark results.

While the selection of intermediate measurement intervals can be an arbitrary decision made by the benchmark owners, the decision should include measurement levels that are representative of systems in the environment represented by the business model of the benchmark. Many customer environments have been observed to have systems that have average usage in the 5-20% ranges. Other environments may operate at closer to system capacity. In order to quantify the impacts and benefits of modern power management functions, the benchmark should require a range of several throughput levels. Idle and 100% capacity measurement intervals should also be included – 100% because it measures the limit of the system and usually the best performance/power ratio available; idle because many systems are left sitting idle on occasion (and often more frequently) or drop to an idle state in between periods of operation.

One possible method is to measure at 11 measurement intervals: 100%, 90%, 80% … 20%, 10% and Active-Idle, represented by a 0% throughput measurement. Measurement from high throughput to zero throughput points guarantees that the system under test is at “Active-Idle” state for the idle measurement. It also tends to provide a smoother idle measurement than initiating an idle measure directly following the initial measurements that may be used to calibrate the measurement points.

Note: The term “Active-Idle” is used to designate a state where the system is only temporarily idle, but has done real work prior to this time and is likely to do real work again in the near future. Other examples in the industry of graduated measurement intervals are SPECweb2009, which uses measurements at 100%, 80%, 60%, 40%, 20% and Active-Idle, and the SAP Server Power benchmark Version 1.1, which uses measurements of 50%, 100%, 65%, 20%, Active-Idle, 30%, 80%, 40% and 10%.

The following example is taken from a measurement of the SPECpower_ssj2008 benchmark on a single system. The red bars represent the efficiency (throughput per watt) for each measurement interval. The blue line shows the average power requirement at each of the 11 measurement intervals, including the Active-Idle measurement, where no throughput is shown:



Note: some benchmarks, such as SPECjbb2005, appear to have a natural ramp-up by initiating first one task that runs at maximum capacity, then a second task at maximum capacity, and so on until there are at least as many tasks as there are logical processors on the system (twice as many, for SPECjbb2005). This is usually not a satisfactory methodology for a general power metric, because most environments will have many more tasks that work intermittently, rather than a very few tasks running constantly. Running a benchmark with fewer tasks than logical processors on the system could provide an unrealistically positive view of a power management system. However, the benchmark should not preclude the use of emerging technologies for workload management that control the number of active threads based on workload.

3.5.2. Determining the Maximum Target Throughput

Since the maximum throughput is the anchor for determining the throughput steps in the measurement, a key requirement is to be able to determine the maximum value of the benchmark throughput. Of course, the original benchmark is likely designed to identify the maximum throughput as a part of the run procedures, but the power-measurable benchmark is an adaptation of the original, so the maximum may be different than it was with the original benchmark. There are multiple methods for calibrating to the maximum. Any are valid as long as the benchmark run rules dictate consistency from measurement to measurement.

Among the options are:

- Run the benchmark at high use once and take the measured value to be the maximum
- Run the benchmark at high use three times and average the 2nd and 3rd values for the "maximum", or perhaps run the benchmark at high use a larger number of times, but still average the last two runs to derive the target maximum.
- Run the benchmark at high use multiple times. When the test harness recognizes that the benchmark result is lower than the result of the prior run, run the benchmark one more time and average the last three runs.
- Set the benchmark maximum at an arbitrary fraction of what the original benchmark run was.

While any of these methods would work, it is recommended that the benchmark driver program employ one of the first three – where there is an automated method for determining a maximum throughput target – for official benchmark runs. The driver program could also support others, including the last one, so that controlled experiments can be done in an engineering or academic environment by setting the maximum to the same value time and time, again.

Note: The second option, listed above, is the one employed by official SPECpower_ssj2008 measurements.

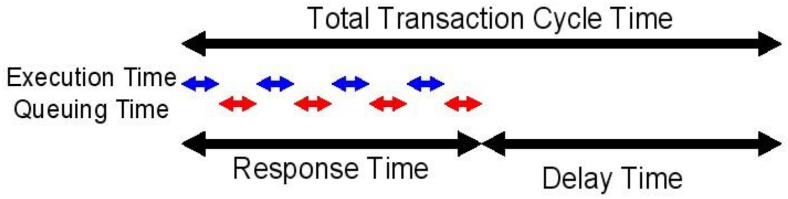
3.5.3. Controls to Achieve the Intermediate Measurement Intervals

In the real world, light loads are demanded of the server by a combination of moderate transaction intensity, relatively few users making requests and long durations between those requests. If a benchmark workload is driven by an external driver that simulates real users (that is to say, the benchmark harness not only controls the benchmark execution rules, but also simulates the individual workload transaction requests), chances are there are “key-think” delays that are already in the driver.

In this case, the intermediate measurement intervals of the measurement curve can be driven in a two-step process:

1. Compute the approximate key-think delay needed to run at the reduced transaction rate and measure the result.
2. If the result shows the throughput to be outside of a predefined tolerance level, adjust the key-think further and run again.

We should point out that this measurement is inherently approximate. The simple graphic below, represents the “life” of a transaction.



A given “user” will request work from the system and receive a result after some response time. The “user” will then delay some period of time before requesting another transaction. In many benchmarks, the delay time between transaction requests is zero, in order to drive more throughput through the system under test. The response time is not constant, however, because it is comprised of the combination of execution time and queuing time. Queuing time can be very long when the system is heavily used and it can be very short if the system is lightly used.

The result is that a simple calculation of a delay between transaction requests will likely generate more throughput at the lower utilization levels than expected, because the response time component of the “transaction life cycle” will be much shorter. Of course, if there is a power management routine employed, it will sense that the power management function can slow down the processor and the transaction run time may actually go up. To achieve some level of precision in targeting the throughput for each measurement interval, the use of a delay between transactions may require some iteration to speed or slow the transaction rate to meet the target throughput. Nonetheless, if the delays are introduced in an external driver of some kind, this method will likely yield measurement intervals that are near the target throughput percentages.

A more accurate control is to work with total “transaction life cycle” for the work submitted in the benchmark, by controlling the timing of the beginning of each transaction request, rather than the delay time between transactions. This will automatically account for any variation in the response times due to queuing changes.

Conceptually, one would look at the number of microseconds that should be spent on a transaction, if the throughput rate was to be xx percent of the maximum throughput and initiate each transaction at that microsecond interval. However, it is often difficult to control time delays at that granularity, because benchmark transactions are often much lighter than “real world” transactions. Thus, it is recommended that the cycle time for blocks of transactions be used.

For example, if a thread runs to maximum capacity at 800 operations per second, and you wish to run at 10% of maximum throughput, you want a total of 80 operations per second to be accomplished. This can be done by scheduling 80 operations to run, and not submitting more work until the start of the next second – or it can be done by scheduling 160 operations and not submitting more work for two seconds, or by scheduling 40 operations each half second.

The actual throughput control methods used can vary by benchmark. Some benchmarks will be most easily controlled by maintaining a constant number of simulated user tasks and varying the transaction life cycle. Others may be better controlled by maintaining a constant transaction life cycle per simulated user task, and varying the number of simulated users.

3.5.4. Need for Variability in Workload Demand

Including variability in performance workloads is important for two reasons: First, a set of regular iterations of delay for all tasks in the system can lead to a harmonic “drum-beat” situation where the work is all-on, then all-off. Second, extremely regular transaction flow is not representative of customer reality. When a system is operated at 10% of its compute capacity, it is almost certain that the workload demand will not be a steady 10%, but will be a highly variable sequence of load requests that happen to average 10%. This is important from an energy consumption perspective, because power management routines need to be designed to react to the variable workload requests on systems that are operating at low average fractions of capacity. To the extent possible, while maintaining measurement comparability, a variable load should be included in the driver mechanism, particularly for measurement segments that are targeted for less than 50% of capacity.

To avoid harmonic amplification and to provide a more realistic/random arrival rate for work requests, the cycle times for each task should vary in some way, such as using a negative exponential distribution. A negative exponential distribution is generally accepted by queuing theorists as being representative of the inter-arrival time of work requests. Values for the distribution can be computed “on the fly”, but will be more controlled if a deck of specific values that match the distribution is shuffled randomly prior to the measurement run and then each transaction cycle is determined by pulling a single card from the deck.

3.5.5. Example of Workload Driver with Built-in Variability

For Java-based applications, this scheduling can be accomplished using the ScheduledThreadPoolExecutor function of Java5. Other benchmark applications will need to be controlled with whatever methods are available in the implementation language of the driver.
The following example is an excerpt from the SPECpower_ssj2008 benchmark:

```
public class TransactionManager implements Runnable {
        private long intendedArrivalTime = Long.MIN_VALUE;

        // Miscellaneous benchmark-specific fields and methods
        // ...

        public void run() {
                if (intendedArrivalTime == Long.MIN_VALUE) {
                        intendedArrivalTime = System.nanoTime();
                }

                // Perform some benchmark-specific setup and bookkeeping
                // ...

                for (int i = 0; i < transactionsPerBatch; i++) {
                        // Execute a transaction
                        // Record the results (transaction response time, etc)
                }

                scheduleNextBatch();
        }

        private void scheduleNextBatch() {
                long now = System.nanoTime();
                long totalResponseTime = now - intendedArrivalTime;
                if(getRunMode()==RunMode.RECORDING){
                        timerData.batchComplete(totalResponseTime);
                } else {
                     warehouseRawBatchCount++;
              warehouseRawResponseTime+=totalResponseTime;
              }

                // Schedule the next batch of transactions
                long delay;
                if (meanDelayNs <= 0) {
                        // Calibration
                        delay = 0;
                        intendedArrivalTime = now;
                } else {
                        // getNextDelayNs returns the next delay from a negative
                        // exponential distribution, with the appropriate mean
                        // for this load level
                        long nextDelay = getNextDelayNs();
                        intendedArrivalTime += nextDelay;
                        delay = intendedArrivalTime - now;
                }

        try {
                        // Delay could be negative if intended arrival time has already
                // past.  Or it could be 0 if we are calibrating.
                        if (delay <= 0) {
                        sched.submit(this);
                        } else {
                                sched.schedule(this, delay, TimeUnit.NANOSECONDS);
                }
        } catch (RejectedExecutionException x) {
                // New tasks will be rejected during shutdown.  So ignore this
                // exception.
        }
}
```

3.6. Hybrid Benchmarks

Some performance benchmarks are actually a composite of smaller benchmarks. For example, SPECint2006 and SPECfp2006 are each comprised of a suite of individually timed tests. SPECweb2005 is another example, where there are three distinct benchmark suites that are designed to measure different business models within the web serving space.

In such cases, power requirements should be measured for each distinct measurement interval. Experimental data indicates that the power requirements can vary significantly, even when each test runs at maximum load. If there is justification to quantify performance at a unique point, similar arguments can be applied to report power at the same point. These metrics will not have a uniform “stair-step” appearance like those of intermediate steps in benchmarks that are described in Clause 4.6. However, they will provide valuable information regarding the energy characteristics of each distinct component of the performance benchmark. Of course, it could be possible for intermediate steps to be defined within a major component of a performance benchmark, in order to achieve a graduated measurement of power requirements.

Methods for merging the power and performance measurements into a single metric will be discussed in Clause 8.2 below.

4. System Under Test (SUT)

Boundaries of the computer systems for which performance and power will be measured should be defined in the benchmark. It is highly recommended that the SUT include all components of the configuration that would contribute to the application represented by the business model of the benchmark. The entire SUT should be measured for power requirements, although benchmark implementers may elect to focus metrics on specific subsystems of the SUT, as described in Clause 4.5.

Different server configurations are often selected for reasons other than pure performance or power use. Therefore, benchmark implementers should consider differentiation between server types in defining power metrics or comparison points.

4.1. Servers versus Personal Systems

Most of this methodology is focused on measurement of computer systems that are considered to be “servers”. However, the methods described here could also apply to single-user or “personal” systems.

For the purposes of this designation, a ‘server’ is defined as a computer system that is marketed to support multiple tasks from multiple users, simultaneously. A ‘personal system’ is a computer system that is primarily marketed to serve a single individual, even though multiple tasks may execute simultaneously. Because a personal system is intended for a single user, this methodology recommends that a display device and a user-input device be a part of the measured configuration. Since a server will likely be connected to many personal systems, these devices are not included in the recommended server configuration.

To avoid misleading comparisons, if personal systems are allowed for the benchmark, comparisons should be restricted so that only servers can be compared with servers and personal systems compared with personal systems. The test sponsor should be required to declare whether the SUT is a Server or a Personal System, depending on the primary use for the SUT and how the test sponsor intends it to be compared.

4.2. Discrete Server (Tower or Rack-mounted)

This class of server is defined as a general-purpose computer enclosure with room-level distribution (AC or DC). All elements of the SUT, such as boot device(s) and data storage, are contained within the single enclosure. The power measurements are taken simultaneously at all power cords to the SUT.

Note: On a discrete server which uses multiple power supplies, the combined power consumption from the multiple power supplies must be measured.
Note 2: On a discrete server configured with a redundant power supplies option, all power supplies included in the configuration, including redundant ones, must be measured.
Note 3: The intent is that configurations are measured as they would be installed at a customer location, without disabling non-essential equipment.

Even at this level of distinction, there can be significant differences between configurations for the SUT. For example, a discrete server may be designed to include up to two disk drives or up to eight. There may have room for one integrated Ethernet port or it may have space for multiple I/O adapters; it may be designed to fit in a rack on a machine room floor or it may be designed to sit desk-side in a quiet office area. It will not be possible to create categories that delineate all of these areas, but sufficient information should be required to be disclosed so that readers of benchmark data can draw their own conclusions.

4.3. Disaggregated Server

For many server configurations, the elements of the SUT, such as boot device(s) and data storage are contained in multiple enclosures. Some benchmarks may require a disaggregated server simply because they require sufficient storage that cannot be contained in a discrete server.

Servers of this type can come in many shapes and sizes. Similar to the discrete server, there may be servers that are expandable to 64, 128 or 256 processor cores, or they may only be expandable to 8 processor cores. There may be SUT configurations that are cooled entirely with air, configurations that are water cooled, or configurations that employ both. Depending on the types of configurations allowed for a benchmark, some restrictions in the types of comparisons that are allowed may be advisable.

The power measurements must be taken simultaneously at all of the power cords to the multiple enclosures of the SUT. While it may be possible to draw some conclusions about the power characteristics of distinct subsets of the SUT, configuration variability will make this a difficult task and it should be discouraged. For example, one system may include 8 internal disks within the enclosure that houses the processor chips, while another may only allow 2. If the benchmark requires a total of 30 disk drives, and an attempt was made to compare the external storage subsystems, the system that only required 22 external disks would show an advantage over the system that required 28 external disks.

4.4. Blade Server (Blade Enclosure-mounted)

A blade server environment has a unique relationship between performance and power because multiple servers with independent operating systems are able to share key resources with regard to energy consumption. In particular, they are likely to share power supplies, cooling fans, network components, and potentially storage subsystems.

Other environments may exist where multiple servers are sharing power-critical or thermal resources. It may be worthwhile to restrict this category to configurations that include servers that are dependent on power and other resources in a single enclosure. The minimum requirement should be that the servers involved share at least one significant component that provides for power efficiency. For example, an entire rack may use a consolidated power supply or may rely on a cooling system that is designed to handle all components in the rack. On the other hand, a configuration of multiple servers that all plug into the same power distribution strip in a rack, but do not share any other resources, would not fit into this category.

The importance of these types of configurations in saving space and energy for the performance capacity delivered makes it desirable to accommodate them in the design of the benchmark. This means that the benchmark should be designed to drive the workload on multiple servers.

Note 1: It could be argued that when a group of servers share storage resources in a storage area network (SAN) or network attached storage (NAS) configuration, they are sharing a power-efficient component and the entire collection should be treated as a single SUT. For the purposes of containing benchmark configurations to meaningful, measurable, and comparable sizes, this methodology advocates excluding this case, unless other energy-saving techniques are employed (common rack cooling, shared power supplies, etc.)

Note 2: For pure performance benchmarks, there is not a critical need to support multiple servers in the same benchmark. However, because of the unique relationship that these configurations have in energy saving, it makes sense to include them in a benchmark that relates performance to power.

4.4.1. Enclosure Measurements

For blade and “blade-like” configurations, the entire enclosure should be measured, populated with as many servers as the benchmark sponsor cares to measure. It would be unfair to require the measurement to be contained to a single server, since many of the resources used by the server are designed to be shared by many servers in the configuration. Performance measurements should be reported on the entire SUT and on a per-blade basis.

4.4.2. Measurement Allocation to Blades

For blade and “blade-like” configurations, the reduction of a performance/power metric to a single blade should be avoided, since this misrepresents the power requirements to drive a single blade. It would be unfair to compare the fractional result for a single blade against a discrete server, since the comparison would use only a fraction of the power for the shared components. At a minimum, comparisons made between blade configurations and other server configurations should require full disclosure of the configuration types being used, the performance per individual server (or blade), and such other information as may be needed to fully describe the configurations.

4.5. Measurement of Subsystems within a SUT

Many benchmarks require work to be completed within multiple logical subsystems of a SUT. Often, the implementation of these subsystems is physical. For example, a benchmark may require a significant amount of activity from a logical networking subsystem. The networking subsystem may be physical, meaning that there are separate physical components to satisfy the networking functions, or it may be logical-only, meaning that all network components reside within the server, or there may be a combination of the two. The same can be said for storage subsystems, application-serving subsystems, data-serving subsystems, etc.

In this case, it may be important to have a power measurement for just the specific subsystem. It remains important to include a requirement to report the entire SUT power, so as to discourage implementations that artificially load the unmeasured components in order to optimize the performance, and therefore the performance-per-watt of the measured component(s). It is also important to include the option of having a primary metric that is focused on the entire SUT, since some physical configurations may not be able to differentiate between physical subsystems.

For example, it may be completely appropriate for a benchmark that requires both a logical server and a logical storage subsystem to allow a benchmark sponsor to highlight the excellent performance-per-watt of their server, as long as they also show the power information of the storage subsystem and the entire SUT as a whole.

As noted, there will be cases where logical subsystems are co-resident on the same physical component of the SUT, or where a significant portion of a logical subsystem resides within the physical component that contains another subsystem. For example, a benchmark may require 20 disk units, 8 of which are configured to be in the server enclosure and 12 of which are configured to be in a separate drawer. In this case, subsystem metrics should not be allowed, nor should comparisons to the subsystem metrics of other measurements be allowed because it is difficult to separate the contribution from the subsystem from other logical subsystems.

It is recommended that

- Benchmark developers decide which subsystems are important enough to have distinct metrics (for example, network may be included within the server metric, but storage may be distinct, or all three may be distinct)
- Benchmark developers declare how much overlap is allowed before a physical subsystem ceases to represent the logical subsystem. A good guide is that at least 90% of the logical subsystem function must reside on the physical subsystem. In the 20-disk example, above, 40% of the disks are in the server enclosure, so unless it can be demonstrated that over 90% of the storage work is going to the 12 external disks, the configuration should not be allowed for subsystem metrics. However, if there were 100 disks, where 92 were physically housed in a separate storage subsystem, it would be possible to consider the remaining 8 disks to be a part of the server subsystem, with a distinct storage subsystem made up of the other 92 drives and their associated enclosures.
- Benchmark developers include run and reporting rules that allow subsystem comparisons only between benchmark implementations where subsystem metrics meet these criteria.
- Benchmark developers explicitly label subsystem metrics in a way that does not allow confusion with metrics that are associated with the full SUT. This is a particular concern with performance-per-watt metrics, since the subsystem watts will be less than the total SUT watts, resulting in a larger value for the subsystem performance-per-watt metric than for the SUT performance-per-watt metric.

If the benchmark designers generate rules that disallow the comparisons of results that are optimized for a single component with results that are intended to show the power requirements of the entire SUT, it is acceptable to provide a benchmark category where the primary performance per power metric does not include the power of the entire SUT, as long as the overall SUT power is required to be reported. For example, a benchmark could have two non-comparable categories, one where the entire SUT power is used for the primary metric and the sponsor would have the option of reporting subsystem metrics; and one where the entire SUT power is reported, but the primary metric is associated with the power of the server.

4.6. Measurement of systems that include one or more batteries

For SUT components that have an integrated battery, the battery should be fully charged at the end of each of the measurement intervals in the benchmark. It may be appropriate to limit the inclusion of systems that are designed to run on battery power for extended periods of time, as the technology in these systems relies heavily on battery power management and this could affect the validity of the result.

Since the power benchmark is designed to measure actual power required during the benchmark, a valid result should require that the system be measured while connected to an external power source and that proof is available that it is not relying on stored battery power while the measurement is in progress. Systems that have an option to drain the battery periodically and recharge it at a later time (a typical means of preserving battery life) should be required to have this feature disabled. If a system is configured with a battery that is intended to sustain it when AC power is not available, data should be required to demonstrate that the battery is fully charged at the end of each measurement period (e.g. screen shots or tabular battery charge data that includes time stamps).

Note that integrated batteries that are intended to maintain such things as durable cache in a storage controller can be assumed to remain fully charged. The above paragraphs are intended to address “system” batteries that can provide primary power for the SUT.

4.7. Optimizing the SUT Hardware and Software

Most benchmarks have language associated with both encouraging innovation and discouraging benchmark-only optimization. Similar language should be included for power efficiency optimization. Innovation should be encouraged for power-saving techniques and configurations that are intended to serve a larger audience than the specific benchmark application. Optimizations that improve power consumption for the benchmark only should be specifically disallowed.

4.8. Electrical Equivalence

Many benchmarks allow duplicate submissions for a single system sold under various names or physical configurations. For example, similar processors, memory, cache, etc. may be available for separately named desk-top, desk-side and rack-mounted systems. While it may be reasonable to assume the same performance for each of these, yielding three results from one measurement, it is more difficult to prove that their power requirements are equivalent. Each system will, at a minimum, have different cooling characteristics, which can affect the power draw of the system. For this reason, it is recommended that electrically equivalent submissions without a measurement be either disallowed or highly restricted in a power and performance benchmark.

5. Power Measurement

5.1. Alternating Current (AC) and Direct Current (DC) Powered Data Centers

Computing systems typically operate using Direct Current (DC) power internally. However, most external power comes from Alternating Current (AC) sources. Computing components that use AC external power rely on power supply units that convert the AC power to DC and regulate it to provide the consistent levels of DC power required by the internal sub-components within the unit. The efficiency of this conversion is one of the critical focus areas for overall power efficiency within the industry.

Theoretically, a computer server or other major computing component could be more efficient if, instead of converting external AC power, it could simply regulate power from an external DC power source. Consequently, there is a growing interest in development of DC-powered computing components.

As of Version 1.7.0 of SPEC’s Power and Temperature Daemon, the tool designed to support both AC and DC power measurements. However, SPEC does not recommend the comparison of AC-based results with DC-based results from power efficiency benchmarks or tools. The reasons are twofold:

1) There are substantial differences between these implementations. Data Center infrastructure changes that must be made to facilitate the use of DC power. Since the power received from the external power supplier is almost certainly AC, an additional converter is required to provide a data center with DC power. The power loss associated with this is not measured in a benchmark setting.

2) Measurement technology for measurement of AC power and DC power differs, particularly in the calculation of uncertainty. It is difficult to look at a combination of AC and DC measurements and know for certain that they have comparable levels of uncertainty.

5.2. Power Analyzer Requirements – AC Power Analyzers

Any benchmark that requires measurement of power usage during the benchmark must specify the requirements of the equipment used to collect the power-related information, as well. The measurement control that starts and stops the performance measurement should also start and stop recording for the power analyzer. The power analyzer should have an interface to automatically upload the results.

Performance benchmarks are inherently variable, with run-to-run variations often being on the order of several percent. In that regard, it is not necessary for the power instrumentation to be extraordinarily precise. However, power analyzers that are used to provide benchmark information should satisfy a minimal set of criteria that will instill confidence in the measurement and provide for a level playing field for comparison of information. Note that a power analyzer may meet these criteria when used in some power ranges but not in others, due to the dynamic nature of power analyzer uncertainty and Crest Factor.

Characteristics of the power analyzer should include:

Measurements	True RMS Power (watts) and at least two of volts, amps, and power factor must be reported by the analyzer.
Logging	The analyzer must store measurements to an external device, with a reading/reporting rate of at least 1/sec and an averaging rate that is 1 (preferred) or 2 times the reading interval. "Data averaging interval" is defined as the time period over which all samples captured by the high-speed sampling electronics of the analyzer are averaged to provide the measurement set.
Control	Either the start and stop recording/logging functions of the analyzer must be able to be controlled from an outside program (see Clause 9.1of this methodology document) or the logging function must include sufficient time-stamp information that data points which are outside of a measurement interval can be ignored.
Uncertainty	Measurements must be reported by the analyzer with an overall uncertainty of less than 1%for the ranges measured during the benchmark run. Overall uncertainty means the sum of all specified analyzer uncertainties. Note that analyzer uncertainty is dependent on range settings and on measured load.
Calibration	Must be able to calibrate the analyzer by a standard traceable to NIST (U.S.A.) (http://nist.gov) or counterpart national metrology institute in other countries. The analyzer must have been calibrated within the past year.
Crest Factor	The analyzer must be capable of measuring an amperage spike of at least 3 times the maximum amperage measured during any 1-second-average sample of the benchmark test. If an analyzer does not specify a Crest Factor, this may be satisfied by requiring that the maximum amperage measured during any 1-second period be no more than 1/3 of the maximum supported by the analyzer at measurement settings.

Uncertainty examples:

- An analyzer with a vendor-specified accuracy of +/- 0.5% of **reading** +/- 4 digits, used in a test with a maximum power value of 200W, would have "overall" accuracy of (((0.5%*200W)+0.4W)=1.4W/200W) or 0.7% at 200W.
- An analyzer with a wattage range 20-400W, with a vendor-specified accuracy of +/- 0.25% of **range** +/- 4 digits, used in a test with a maximum power value of 200W, would have "overall" accuracy of (((0.25%*400W)+0.4W)=1.4W/200W) or 0.7% at 200W.

The Power Analyzer should detect and report through a tool such as the SPEC PTDaemon when a sample is returned with a value of "unknown", when a sample would have an uncertainty value that exceeds the maximum uncertainty allowed for the benchmark and the average uncertainty value for a complete measurement interval (See Clause 8.2.2,)

5.3. Power Analyzer Requirements – DC Power Analyzers

Recommended requirements for DC analyzers are the same as those for AC analyzers except for one change and one addition. Methods for measuring and computing uncertainty are different for direct current than they are for alternating current. Instead of the recommended 1% uncertainty limit recommended for AC analyzers, an uncertainty limit of 1.5% is recommended for DC. As with AC measurements, care should be taken to ensure the analyzer meets the uncertainty requirement for the full spectrum of measurements being made.

The additional requirement comes from the fact that many external DC power sources retain some marginal AC component. Consequently, the analyzer must be capable of measuring both the DC and the AC components of the measured power. The controlling tool, such as the SPEC PTDaemon, will need to place the analyzer in a mode that is capable of measuring both components. This may appear to the end user that the analyzer is set to AC mode. Many analyzers screen out the AC component when set to measure in DC mode.

5.4. Power Analyzer Setup

The power analyzer must be located between the AC Line Voltage Source and the SUT. No other active components are allowed between the AC Line Voltage Source and the SUT.
Power analyzer configuration settings that are set by the SPEC PTDaemon must not be manually overridden.

5.4.1. Automatic Range Transitions

Many power analyzers support automatic range transitions depending on the load applied to them. Experience has shown that data can be inaccurate or lost during the period when the range transition is taking place. Uncertainty calculations are difficult when automatic range transitions take place because they depend on the current range setting. The use of automatic range support should be discouraged, and perhaps forbidden for a power per performance benchmark.

5.5. Process for Accepting Power Analyzers for Benchmark Measurements

Although many analyzers may appear to meet the above criteria, it is recommended that an acceptance process be employed to ensure that the analyzers used match criteria needed for confident measures of a benchmark. Some analyzers may show varied results in terms of accuracy over a range, or when switching ranges, or in the number of valid data instances that are recorded, etc. A list of power analyzers that have been accepted for the SPECpower_ssj2008 benchmark can be found at http://www.spec.org/power/docs/SPECpower-Device_List.html. These analyzers have passed tests defined by the SPEC Open Systems Group Power committee which are described at http://www.spec.org/power/docs/SPEC-Power_Analyzer_Acceptance_Process.pdf.

6. Environmental Considerations

Some environmental conditions can affect the power characteristics of computer equipment. These conditions may need to be controlled, and should be required to be reported as a part of the benchmark disclosure.

6.1. Temperature

Temperature has a very substantial effect on power characteristics. In general, the lower the temperature from the air cooling the equipment, the lower the power needed to operate the equipment.

6.1.1. Air-cooled vs liquid cooled equipment.

Note that this discussion assumes "traditional" air-cooled equipment. For larger configurations, alternative cooling methods, such as liquid-cooled, liquid-assist and sealed air-flow systems can be much more power-efficient than cooling with ambient air. If the business model of the benchmark supports configurations of systems that use these cooling methods, some consideration to encourage them may be advisable. For example, the minimum intake temperature requirement could be removed if advanced cooling methods are employed.

Care should be taken, however, when comparing power ratings for air-cooled and liquid-cooled equipment, because the power characteristics of the air-conditioning equipment and the liquid cooling equipment are not included in this methodology and are quite probably different. At a minimum, full disclosure of the cooling methods should be required.

6.1.2. Temperature Requirements

Benchmarks should be defined to restrict the minimum acceptable temperature to an appropriate level. A minimum of 20 degrees Celsius (68 degrees Fahrenheit) is a good starting point. ASHRAE recommendations for data center equipment are for 18-27 degrees Celsius, with a notation that significant savings in cooling costs can be achieved by maintaining higher temperatures. A minimum higher than 20 degrees could be specified, such as 23 or 25 degrees Celsius (73-77 degrees Fahrenheit), although this could be problematic, as many data centers are maintained in the 22-23 degrees Celsius range.

It is not necessary to specify a maximum temperature threshold, as an increase in temperature will likely increase the power consumption of a computer system.

6.1.3. Temperature Measurements

Temperature should be measured no more than 50mm (approximately 2 inches) upwind of the main inlet for airflow to the equipment being benchmarked. If there are multiple inlet locations, a survey of temperatures should be taken, and the inlet with the lowest ambient temperature should be used to monitor temperature. Even for a configuration that includes several servers and/or storage units, a single temperature probe should be sufficient. There should be language in the rules of the benchmark that the test sponsor take steps to ensure that they are measuring at the coolest inlet.

Run rules should require some minimum threshold for temperature. For example they could require that the minimum measured temperature be no less than 20 degrees Celsius. Alternately, the rules could require that the average measured temperature be no less than 22 and that the minimum measured be no less than 18.

To ensure comparability and repeatability of temperature measurements, the following attributes for the temperature measurement device should be required in the benchmark:

- Logging - The sensor should have an interface that allows its measurements to be read and recorded by the benchmark harness. The reading rate supported by the sensor should be at least 4 samples per minute.
- Uncertainty - Measurements should be reported by the sensor with an overall uncertainty of +/- 0.5 degrees Celsius or better for the ranges measured during the benchmark run.

Temperature should be measured on a regular basis throughout the measurement. The low temperature measured should be reported with the benchmark disclosure. The test sponsor should be required to include a statement that they did not do anything to intentionally alter the temperature for any equipment inlet during the run of the measurement and that they measured the temperature at the air inlet expected to have the lowest ambient temperature.

6.2. Humidity

Current data indicates that humidity has an almost negligible impact on power characteristics. Therefore, it is not necessary to set minimum or maximum boundaries for relative humidity for a benchmark.

However, to ensure that currently unknown patterns do not exist, the test sponsor should be required to either report the relative humidity at the beginning of the test run or state that the relative humidity during the measurement was within documented operating specification of SUT

6.3. Air Pressure/Altitude

Air pressure can affect the density of the air that is used to cool computer equipment, and therefore can affect the rate of heat exchange as air flows through the equipment. The effect of air pressure is not nearly that of temperature, but higher pressure will provide a beneficial effect in air-cooled systems. There is little concern with moderate variations in pressure due to weather or altitude, but rules should be enforced to prevent measurement in a hyper pressurized environment. A maximum of 1050 mill bars is reasonable to handle virtually all natural atmospheric conditions.

To ensure that currently unknown patterns do not exist, the test sponsor should be required to report either the air pressure in the room where the equipment is measured or the elevation where the measurement took place and include a statement asserting that nothing was done to overtly alter the air pressure during the course of the benchmark measurement.

6.4. Air Flow

Empirical evidence has shown that, except in extreme conditions, the flow of air around the equipment being measured makes little difference in the power characteristics of the equipment. Increased air flow can improve the delivery of cool air to the inlets of the equipment, but this will be monitored by measuring the temperature as directed in Clause 6.1.

The test sponsor should be required to report that nothing was done to overtly direct air toward or away from the measured equipment during the course of the benchmark measurement in a way that would be considered inconsistent with normal data center practices.

6.5. Power Source

The line voltage and other characteristics can affect the efficiency of equipment that is connected to it.

The preferred Line Voltage source used for measurements is the main AC power as provided by local utility companies. Power generated from other sources often has unwanted harmonics that may not be measured correctly by many power analyzers, and thus would generate inaccurate results.
The benchmark should specify that the Line Voltage Source needs to meet the following characteristics:

- Frequency: (50Hz, 60Hz) ± 1%
- Voltage: (100V, 110V, 120V, 208V, 220V, 230V or 400V) ± 5%

If a system is marketed in specific geographic locations, the use of the external power that is compatible with those locations should be recommended, and perhaps enforced. For example, if standard voltage is 100V, but a system is measured at 220V, an improvement in efficiency is likely to be measured that would not be delivered in a real environment.

The usage of an uninterruptible power source (UPS) as the line voltage source should be allowed, but the voltage output must be a pure sine-wave. This usage should be required to be specified in the benchmark disclosure.

Systems that are designed to be able to run normal operations without an external source of power (such as an internal battery) cannot be used to produce valid comparisons with systems that operate on external power, since the charging of the battery is not measured.

In some environments, a stable external power source is not available and must be supplemented with an artificially controlled power source. This should be allowed, if documented in the benchmark disclosure and if the following requirements are met:

- Total Harmonic Distortion of source voltage (loaded), based on IEC standards: < 5%
- The AC Power Source needs to meet the frequency and voltage characteristics defined by the benchmark
- The AC Power Source must not manipulate its output in a way that would alter the power measurements compared to a measurement made using a compliant line voltage source without the power source.

The intent should be that the AC power source does not interfere with measurements such as power factor by trying to adjust its output power to improve the power factor of the load.

7. Performance/Power Metrics

7.1. Power Measurements at Distinct Benchmark Measurement Intervals

All power-measured benchmarks will have at least two distinct measurement segments: Benchmark at full performance and Active-Idle after benchmark. Some benchmarks will also include intermediate throughput intervals in the benchmark definition. The average power for each distinct measurement segment must be reported. This reporting is important because different customer environments will have different interests with regard to system utilization. A pool of single-purpose servers may operate at very low utilization most of the time, where Idle and 10% throughput intervals are most interesting. A large system designed to consolidate through virtualization could run near the 80% interval for much of the time and at idle for some of the time. For compute intensive environments, the target may be to use the system at 100% capacity almost all of the time.

The throughput and power measurements should be reported for every measurement segment, so that readers can interpret a collective metric and can draw specific conclusions at utilization points that are of interest to them.

7.2. Computing a Benchmark Performance-per-Power Value

Each benchmark that is defined to include power measurements will have a DIFFERENT Performance per Power rating, both because the Performance per Power calculation depends on the throughput values for the benchmark involved and because each benchmark targets a different business model. As such, the metric should be labeled in such a way as to discourage comparison across benchmarks. Terms like “ssjPerformance-per-Power” and “mailPerformance-per-Power” or “webPerformance-per-Power” appropriately separate the power metrics from different benchmarks.

Note: The term “Performance-per-Power” is used here to differentiate a performance/power metric from a typical performance metric. It does not imply that the metric must be so named. However, it is suggested that the performance rating be listed in the numerator and the power rating in the denominator. This is consistent with SPECpower_ssj2008, the first industry standard benchmark to include a performance-per-power metric. The choice of numerator and denominator is made in this way so that the efficiency rating will have a larger value when there is greater power efficiency, much like miles per gallon. While a power-per-performance metric would also yield comparable results (liters per 100 km), it was felt that a performance-per-power rating would be more intuitive.

In determining the method for computation of a performance-per-power metric, the business model for the benchmark and the available measurement points should be evaluated.

Three key points:

1) Reporting information for benchmarks that include performance-per-power metric(s) must include at least the energy measurements from the SUT, as defined by the benchmark. It is recommended that the primary power metric include the energy requirements of the entire SUT (see note, below).
2) The performance-per-power metrics from different benchmarks are not comparable.
3) Within a benchmark, a higher performance-per-power value means that more work can be achieved for the same expenditure of energy.

Note: For benchmarks that rely on the performance and power characteristics of specific subsystems, such as a storage subsystem or a network switch, it may be advisable to require both the total SUT power measurement and measurements associated with specific subsystems. In some cases, a benchmark may be defined to use only the power from one subsystem as the primary metric. However, the power of the entire SUT must still be reported to ensure that the information is fairly portrayed. This is further described in Clause 4.

Secondary metrics may also be defined, such as maximum throughput and the performance/power ratio at maximum throughput. If secondary metrics are defined to focus on a subset of the SUT, great care must be taken to ensure that they can be accurately represented and fairly compared. Secondary metrics should never be publicly reported without including the associated primary metrics.

7.2.1. Case for Measurement Intervals of Equal Importance

Case 1: If each point measured is of equal or near equal importance for the business model.

The performance-per-power result could be computed, as follows:

1. The total performance metric for each of the distinct measurement segments is computed and these totals are summed.
2. The average power measured for each benchmark segment, including the Active-Idle measurement is added together.
3. The quotient of the summed performance metric and the summed power measurements is reported as the performance-per-power value.

For example, consider a benchmark that has the following characteristics:

Segment	***Throughput***	***Measured Power***
100% Interval	100 furlongs/fortnight	400 watts
Active-Idle	0	225 watts

In this case, the performance-per-power value would be (100+0)/(400+225) = 0.160 Furlongs/Fortnight per watt (or 160 Furlongs/Fortnight per kilowatt).

Note: In this case, the Active-Idle measurement holds a full 50% of the weight of the power portion of the metric. A better representation may be to require two primary metrics – the 100% performance/power metric and the Active-Idle metric.
An example with multiple benchmark segments follows:

Segment	***Throughput***	***Measured Power***
100% Interval	200 whatsits/hour	400 watts
80% Interval	158 whatsits/hour	375 watts
60% Interval	119 whatsits/hour	350 watts
40% Interval	82 whatsits/hour	300 watts
20% Interval	41 whatsits/hour	250 watts
Active-Idle	0	225 watts

In this case, the performance-per-power value would be
(200+158+119+82+41+0)/(400+375+350+300+250+225) = 0.316 whatsits/hour per watt (or 316 whatsits/hour per kilowatt

- **Note:** Even for the same system configuration, the two values are not comparable because the performance metric used for the numerator of the calculation is significantly different for each benchmark. They also differ because the relative importance of the idle measure in the first benchmark is replaced by the importance of multiple measures on a sliding scale in the second.

- **Note:** The units of this computed metric are throughput/watts. However, it is not an actual description of pure throughput/watts, since the values are computed over a range of measurements. A choice could be made to either make the result unit-less or to build a ratio between the result and a reference machine, which would default to be unit-less.

Example from SPECpower_ssj2008: Using the same measurement result shown in Clause 3.5.1, above, the following shows the calculations for the performance-per-watt metric used in that benchmark. As one would expect, the system delivers the best performance per watt ratios at high utilization. Since the model used for the benchmark is to weigh each measurement interval equally, the actual benchmark rating is computed from the 11 measurement intervals and reported at the bottom of this table. This demonstrates the need to ensure that only benchmark ratings that are computed with similar rules be compared with each other.

Performance			Power	Performance to Power Ratio
Target Load	Actual Load	ssj_ops	Average Power (W)	
100%	99.8%	190,234	119	1,601
90%	90.7%	172,967	116	1,494
80%	80.8%	154,130	112	1,380
70%	69.7%	132,811	106	1,251
60%	60.8%	115,866	99.8	1,161
50%	49.6%	94,582	90.9	1,041
40%	39.7%	75,792	82.5	919
30%	29.8%	56,857	74.4	764
20%	19.9%	37,980	68.2	557
10%	10.2%	19,410	62.8	309
Active Idle		0	56.7	0
∑ssj_ops / ∑power =				1,064

7.2.2. Case for Measurement Intervals of Varied Importance

Case 2: Benchmarks with distinct measurements that may not be of equal importance for the business model used.

When a benchmark includes distinct measurement intervals that contribute unequally to the performance metric, or that have different levels of importance relative to the business model of the benchmark, a strictly even weighting of the measurements as described in Clause 7.2.1 may not be appropriate. For example, while showing the power measurement at Active-Idle may be important, that value may not have as much importance as the power measurement at full performance.

The Active-Idle measurement does not give any indication of the system's capacity to do work, so it could be that this metric should be downplayed in the final metric. The automotive equivalent would be to decide whether or not to give equal weights to the miles per gallon achieved when the automobile is moving and to the amount of fuel expended when the automobile is sitting at a stop sign. It is important to be efficient at the stop sign, but it may not be as important as efficiency while moving on the road. On the other hand, if the business model of the automotive benchmark were to require long periods of leaving the automobile parked but running, the idle efficiency would become much more important.

In this case, a decision must be made by the benchmark owners as to the importance of each measurement interval and how to deal with it in the primary performance-per-power metric.

Several possible decisions can be made, some of which are:

- Use only the power at maximum throughput for the primary metric, but report other measurements for information
- Measure energy use for each measurement interval that contributes to the primary performance metric and report the benchmark throughput metric (operations per unit of time) divided by the total energy use (Note that operations per second divided by kilowatt-hours yields a value that is effectively equivalent to operations per kilowatt.)
- Apply a weighting factor to the power measured at each measurement interval, and include that in the formula.

Each of these, and quite likely other valid methods, will yield different mathematical results. The key points are to stay as true to the business model of the benchmark as possible and to deliver a metric that is both comparable across systems and easy to understand.

8. Reporting

As with typical performance benchmarks, a significant amount of reporting is required to ensure that benchmarks were executed correctly and completely. This includes information regarding the measurement environment, the SUT, the performance characteristics and the power characteristics. Reporting of this information helps to ensure reproducibility, comparability and reliability of measurements and allows them to be verified for compliance. Most of the areas listed in the earlier sections of this document will require some level of reporting.

8.1. Environment and Pre-measurement Reporting

Recommended information to be verified before preparing the result report, preferably before starting the measurement includes:

- A complete description of the SUT, including orderable/configurable components that can affect either performance or power measures
  - Sufficient information to allow the system configuration to be replicated by another party, if desired, and
  - Tuning or other optimization changes made to achieve the benchmark result
- It is also advisable to require reporting of benchmark control system configurations, benchmark and control code levels and similar information that may be needed to both reproduce the results and to verify that the benchmark was correctly measured.
- Line voltage characteristics must comply with one of the supported specifications
  - Power Line Frequency (50Hz or 60Hz or DC)
  - Power Line Voltage (100V, 110V, 120V, 208V, 220V, other)
  - Power Line Phase characteristics (single phase, two-phase, three-phase)
  - Power Line source (wall, UPS, regulator, etc.)
- Altitude of measurement laboratory and a statement if artificial air pressure was employed
- The basic information to verify the correct calibration of the power analyzer should be required to be reported and verified
  - Information to identify the specific power analyzer device (vendor, model, serial number)
  - Information to identify the calibration event (calibration institute, accrediting organization, calibration label)
  - Calibration date
  - Voltage and Amperage range-settings of the power analyzer(s)
- Temperature sensor used
- Any other data required to be reported for the performance measurements

Note: Some geographical locations do not have stable line voltages and may require some form of regulator or uninterruptible power supply to deliver consistent power to the SUT. In other cases, a measurement for a non-local configuration (such as European line characteristics in a United States measurement or vice versa) may require a device to perform power conversion. This should be allowed as long as it is declared in the benchmark disclosure and no steps are taken to artificially optimize the measurement. The power analyzer(s) must be placed between the power source device and the SUT.

8.2. Measurement Reporting

8.2.1. Performance Reporting

Performance reporting is typically dictated by the performance benchmark that is being used to create a performance-per-power benchmark. Performance information (throughput, response times) and stability information (reported and recovered errors, for example) is likely required for at least every consistent measurement interval in the benchmark.

There are sufficient examples of required performance reporting in the various benchmarks maintained by SPEC that further detail is unnecessary here.

8.2.2. Power Reporting

For each analyzer used and each distinct measurement interval, the following information should be reported:

- Average voltage
- Average current
- Average power
- Average power factor
- Minimum ambient temperature
- Performance metric or sub-metric
- Power readings as reported by SPEC PTDaemon
    - The benchmark should report the number of samples that may have a return value "Unknown."
    - The benchmark should report the number of samples that may have an uncertainty value that is greater than the maximum allowed for each measurement interval. (SPEC PTDaemon is capable of reporting this.)
    - The benchmark should report the Average uncertainty for all sampling periods of a measurement interval.

If subsystem metrics are used, these values should be reported for each subsystem.

For the complete benchmark, the above values should also be reported, with suitable calculations to merge the data points, except that average power factor is not meaningful when multiple measurements are involved.

8.2.3. Power Factor Considerations

At the present time, this methodology only recommends reporting of the average power factors for each analyzer at each measurement interval. However, it may be appropriate to include additional requirements on the power factor. For example, since the power factor contributes to the actual energy that must be expended to produce the power consumed by the System Under Test, a benchmark could require a minimum power factor to achieve a valid result, or could penalize a result with a low power factor by applying some reduction in the metric if the power factor is below a specified threshold.

8.2.4. Measurement and Reporting from Multiple Components

Power characteristics should be measured for the entire SUT. If there are several components in the SUT that are separately powered, they should be separately measured and reported or connected through a power strip or power distribution unit that is directly connected to a power analyzer. Some single components have dual line cords. In this case, the power characteristics of both lines must be measured, either with separate analyzers or by using a power consolidation strip between the SUT and the analyzer.

Some benchmarks may have such complex configurations that there are multiple identically configured components needed to satisfy the performance requirements of the benchmark (for example, multiple storage drawers in a rack). While the preferred scenario is clearly to measure all components of the SUT, the benchmark owners may elect to devise some method for measuring power on a subset of components and computing power requirements for the entire SUT. The accuracy of such a methodology is questionable, so benchmark owners may decide to apply some contingency factor to ensure that such a calculation always shows at least as much power as is really used by the SUT.

8.2.5. Aggregation of Multiple Power Analyzer Measurements

In situations where multiple power analyzers are used to measure power requirements of a SUT, or where approximations are made to apply measurements of one component in the SUT to multiple "identical" components, obtaining accurate measures of the recommended metrics will be more challenging.

Average Total Power will be the sum of the average power measures for each subset measured.

Average

Average Power Factor is not recommended. Prorating the power factor across multiple analyzers may be of limited value. It is more appropriate to only require reporting of the measured power factor from each individual power analyzer, without attempting to aggregate them.

Average Voltage and **Average Current** MIGHT be meaningful if the target voltage is the same for all components of the SUT. However, it will not be meaningful if some components are operating on a different voltage base from others (115 volts versus 208 volts, which are both available in the United States, for example.) Consequently, this methodology does not recommend aggregation of voltage and current from multiple power analyzers.

Peak Voltage, Peak Current and Peak Power, if desired for benchmark reporting, can only be accurately represented if the second-by-second measurements are prorated by the average power percentage of the total. Even this is not a completely accurate representation and it may be best to report these items only as a part of detailed reporting of individual power analyzers used in the measurement and not as aggregated results.

Note: This methodology does not include these values as essential for benchmark reporting, but some benchmarks may choose to include them.

Examples: Consider the following 5-second measurement of a SUT connected through three analyzers. The measures are exaggerated from what one would expect in reality to make the calculations more interesting. Text in blue, standard font is what would be reported by the individual power analyzers. Text in *red*, *italic* font is additional computation using the methods described above.

Observe that, because Analyzers A, B and C operate in the 115 volt, 208 volt and 230 volt ranges, respectively, that the row labeled “ProRated Volts” shows values that are essentially meaningless, which also means the row labeled “ProRated Amperes” is equally meaningless. The ProRated Power Factor is also of little value. The only aggregated value of confidence is the Total Watts row, which is a valid measure even if portions of the SUT are operating at different voltage levels.

The final averages, shown in ***bold italic*** values in the lower right are the same whether they are calculated from the end-of-measurement averages in the right hand column for each analyzer or from the prorated values for the total SUT in the lower rows. Thus, except for when peak values or detailed plotting of results is desired, it will be easiest to compute the values from the averages delivered by each power analyzer for each interval. As previously mentioned, the average total value with the most confidence in the table, below, is the average Watts. Other values require formulas that prorate them based on relative power contribution for each analyzer. The results of these formulas may be difficult to understand and could be misleading. Only total watts for each distinct measurement period are recommended as a requirement for a benchmark with a power metric.

	Second 1	Second 2	Second 3	Second 4	Second 5	Averages
Analyzer A Volts	100	110	115	110	120	111
Analyzer A Amperes	2.5	3.0	2.5	3.0	2.5	2.7
Analyzer A Watts	220	260	250	280	230	248
Analyzer A PF	.88	.79	.87	.85	.82	.84
Analyzer B Volts	208	208	208	208	208	208
Analyzer B Amperes	5.0	6.0	7.0	8.0	9.0	7.0
Analyzer B Watts	1000	1100	1200	1300	1400	1200
Analyzer B PF	.96	.88	.82	.78	.75	.84
Analyzer C Volts	230	225	220	225	235	227
Analyzer C Amperes	2.0	2.1	2.2	2.3	2.4	2.2
Analyzer C Watts	420	420	420	420	420	420
Analyzer C PF	.91	.89	.91	.74	.74	.84
ProRated Volts – no value or misleading	*186*	*182*	*183*	*183*	*190*	***185***
ProRated Amperes – no value or misleading	*33*	*42*	*38*	*45*	*32*	***38***
ProRated PF – no value or misleading	*0.94*	*0.87*	*0.85*	*0.78*	*0.76*	***0.84***
Total Watts	*1640*	*1780*	*1870*	*2000*	*2050*	***1868***

9. Automation and Validation

Much of the information that is reported will require validation to ensure that the benchmark was run correctly. This can either be done manually, or as a part of the automation tools that will be created to control the benchmark. While neither automated benchmark control nor automated validation is required to collect and report benchmark results, including automation tools will greatly enhance the probability of generating results that are correct and comparable.

9.1. Integration of Commands to Collect Power and Thermal Data

Measurements of performance and power will be most easily accomplished if the controlling application or tool used to initiate, control and record results for the performance benchmark can also exercise control and record results for the power monitoring and thermal monitoring devices. Many monitoring devices have command interfaces where actions can be controlled via the call of APIs. If devices that do not support API control interfaces are allowed to be used in the benchmark, then sufficient run requirements must be in place to ensure that the power metrics can be properly aligned with the performance metrics.

SPEC benchmarks can make use of the SPEC Power and Temperature Daemon (SPEC PTDaemon) that can control several popular power analyzers and temperature sensors The SPEC PTDaemon translates vendor-specific command and hardware interfaces into a consistent API that hides the power analyzer and thermal sensor details from the benchmark harness. Much of the SPEC PTDaemon code was developed for SPEC by member companies, and the current license agreement is for non-profit organizations only. For organizations outside of SPEC, inquiries should be made to info@spec.org. There are many analyzers that are supported by the code – not all of which will produce compliant benchmark results. Current power analyzers and temperature monitors supported by the software can be found at http://www.spec.org/power/docs/SPECpower-Device_List.html. Analyzers that are compliant with the SPEC_ssj2008 benchmark are so designated in this list. As additional measurement devices are identified that satisfy the criteria listed above, the interfaces for these devices may be integrated into the SPEC code. Internal and external inquiries should be directed as listed above.

Note: This does not imply that all power analyzers and temperature sensors listed above satisfy the requirements identified in this document. A separate list of measurement devices that are qualified to be used for published results should be included in the run rules for the benchmark.

9.2. Controlling Power and Performance Benchmarks

Performance benchmarks can be driven by a driver that is either internal to the SUT or external to it. Typically, unless the driver comprises such a small fraction of the workload that it doesn't affect the result, the best performance is achieved with an external driver. This allows the driver to regulate and monitor workload requests to the SUT. On the other hand, many benchmarks are self-driven, where the driver function is a part of the system load. In this case, the only external interface that is needed is the command to begin the benchmark.

For power measurements, it is almost certain that an external measurement device will be required. Even if a SUT has the capability to monitor power characteristics internally, a benchmark requires measurement methods that are certifiably consistent, which would likely exclude internal measures.

Since the power metric will almost certainly be obtained from an external analyzer and since the performance metric could be obtained from an external source, this methodology is written from the perspective of having the workload driver and the monitors for throughput and power external to the SUT. Except for the actual power analyzer, it is logically possible to integrate these functions within the SUT, but they will be represented as external to the SUT, here.

The more automated controls that can be included in a benchmark, the easier it will be to check results for validity and achieve comparable results. This is true for performance benchmarks. It is also true for benchmarks that measure power and performance. Areas that can be automated include

- Documenting the configuration of the SUT

- Documenting the power analyzer used and the settings for the analyzer during the measurement
- Automating the execution of the workload
- Synchronizing the measurement of power and temperature with the workload
- Validation that the workload met performance criteria
- Validation that the benchmark environment was always in compliance with benchmark criteria
- Validation that the power analyzer(s) and temperature probe(s) are among the lists of accepted devices for the benchmark
- Validation that the power measurements met power criteria
- Validation that the uncertainty in power measurements is within the criteria set for the benchmark

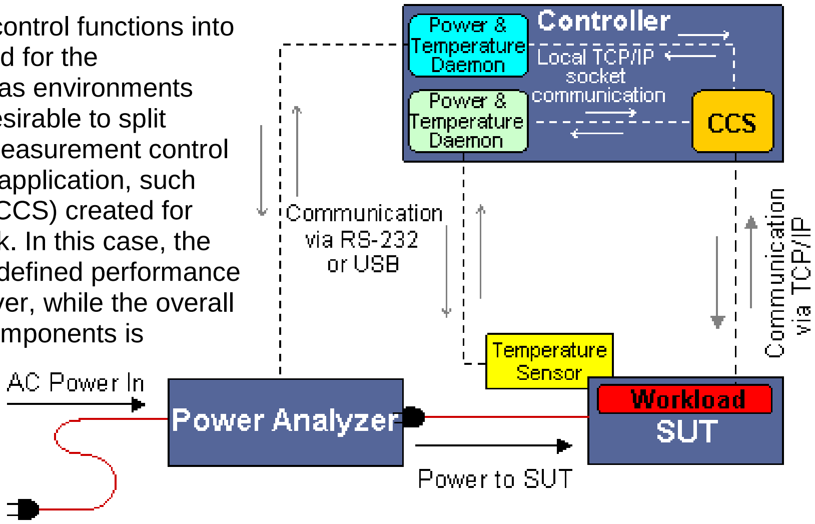
The figure below represents a generic benchmark environment that is capable of driving and monitoring both performance and power components of the workload:

The functions of the measurement server are:

1. Start and stop each segment (phase) of the performance benchmark
2. Control the workload demands for the performance benchmark
3. Start and stop collection of power data from the power analyzer so that power data and performance data from each phase can be correlated.
4. Store log files containing benchmark performance information and benchmark power information.
5. Include logic to convert the raw data to the format needed for reporting, submission and validation of the benchmark.
6. Collect and store environmental data, if automated for the benchmark. Optionally, there may also be some logging functions required within the SUT, such as resource utilization and potentially some throughput performance information.

9.2.1. SPECpower Control and Collect System

It will often be possible to integrate control functions into the driver that is specifically designed for the performance benchmark. However, as environments become more complex, it may be desirable to split these functions, where the overall measurement control is maintained in a separate, flexible application, such as the Control and Collect System (CCS) created for the SPECpower_ssj2008 benchmark. In this case, the performance measurement of each defined performance step is controlled via the defined driver, while the overall coordination between the various components is managed by the CCS.


Power & Temperature Daemon
Controller
Local TCP/IP socket communication
Power & Temperature Daemon
CCS
Communication via RS-232 or USB
Communication via TCP/IP
Temperature Sensor
Workload
SUT
AC Power In
Power Analyzer
Power to SUT

SPEC benchmarks can make use of a Control and Collect System application that was implemented for the SPECpower_ssj2008 benchmark to manage the steps identified in clause 3.5 of this document. The design is flexible enough to be adapted to multiple power analyzers and multiple SUT servers, although there is clearly some benchmark-specific logic that would need to be altered for new benchmark implementations. Much of this code was developed for SPEC by member companies, and the current license agreement is for SPEC benchmarks only. For organizations outside of SPEC, inquiries should be made to info@spec.org.

10. Fair Use Considerations

In order to promote fair use of benchmark results, rules should be established to control how benchmark information can be used in public information. The following is a partial list of recommended rules:

- Do not allow estimated results to be publicly disclosed or compared to measured results
- Do not allow comparisons of power or performance per power results that come from multiple benchmarks
- Require that if any measured data from the disclosure is used, the primary metric for the systems being compared must be disclosed in close proximity.
- Require that when comparing measured performance and/or power data from any target load level, both the performance and the power results for that target load also must be disclosed in close proximity.
- Require that when comparing performance and/or power measurements at different target load levels, the comparisons must also include the performance and power at the 100% target load level in close proximity.
- Consider what restrictions should be placed on what types of systems can be compared. For example, it is not recommended to allow comparisons between servers and personal systems
- In a comparison of benchmark results that include multiple nodes, require that the number of nodes for each SUT must be stated.
- In a comparison of benchmark results that include multiple nodes, when deriving performance and/or power information from a multi-node result, the derivations must also include the number of nodes and the calculation method in close proximity.
- Since the Active Idle measurement interval does not have a performance load level, comparisons of Active Idle intervals must include the primary metric and the performance at the 100% target load level in close proximity.
- Specifically identify what benchmark information can be compared and what, by omission in the list, cannot. For example,
  - The performance and the performance per watt at each load point might be allowed for comparisons, but detailed transaction response data that is used to validate the benchmark result should not be compared.
  - Calibration measurement information should not be used for comparisons

"Close proximity" as used above is defined to mean in the same paragraph, in the same font style and size, and either within 100 words or on the same presentation slide.

The following paragraphs are examples of acceptable language when publicly using SPECpower_ssj2008 results for comparisons.

1. When fully loaded, Server X provides more performance and consumes less power than Server Y. Server X scores: (95,853 ssj_ops and 276W) @ 100% target load vs. Server Y: (40,852 ssj_ops and 336W) @ 100%. The SPECpower_ssj2008 overall ssj_ops/watt are Server X: 203 and Server Y: 87.4 [1].
2. Server X provides greater efficiency than Server Y. The SPECpower_ssj2008 overall ssj_ops/watt for 4-node Server X is 203 and for 2-node Server Y is 87.4 [1].
3. Server X does not pass 250W until near full load, whereas Server Y reaches it much earlier. Server X scores (79,346 ssj_ops and 252W) @ 90% target load while Server Y scores (8,237 ssj_ops and 254W) @ 30%. When fully loaded, Server X scores (95,853 ssj_ops and 276W) @ 100% and Server Y scores (40,852 ssj_ops and 336W) @ 100%. The SPECpower_ssj2008 overall ssj_ops/watt are Server X: 203 and Server Y: 87.4 [1]
4. Server X uses only 50W at the Active Idle point, compared to 255W at Active Idle for Server Y. Server X scores (185,000 ssj_ops and 200W) @ 100% target load and Server Y scores

(240,000 ssj_ops and 200W) @ 100%. The SPECpower_ssj2008 overall ssj_ops/watt are Server X: 512 and Server Y: 450 [1]

5. Server X provides better performance and uses less power than the individual nodes of Server Y. The single node server X scores (766 ssj_ops and 1,050W) @ 100% target load level. Server Y is a 10-node server which scores (2,550 ssj_ops and 10KW) @ 100% -- which means that on average each of the nodes uses (255 ssj_ops and 1000W) @ 100%. The SPECpower_ssj2008 overall ssj_ops/watt results are Server X: 415 and Server Y: 325 [1]

   Note: The above examples assume the inclusion of a footnote similar to:
   [1] Comparison based on results for the named systems as published at www.spec.org as of 26 January 2011. SPEC® and the benchmark name SPECpower_ssj® are registered trademarks of the Standard Performance Evaluation Corporation. For more information about SPECpower, see www.spec.org/power_ssj2008/.

11. Appendix – Terminology

11.1. RMS

The root mean square (RMS) is a statistical measure of the magnitude of a varying quantity. The RMS value of a signal is the amplitude of a constant signal that yields the same average power dissipation. The following formula defines the RMS value of current (see also 9.4 Current):

$$I_{RMS} = \sqrt{\frac{1}{T}\int_{t=0}^{T} i(t)^2\,dt}$$

The RMS value is proportional to the power consumption in an ohmic resistor. Usually it is defined over one period of the signal.
The analogue formula is defined for voltage.

11.2. True RMS

Power analyzers are called True RMS or TRMS, if the instrument measures the RMS values independent from the waveshape of the signal.
Some older or low cost analyzers measure in fact just the rectified value and multiply it with the form factor 1.11 (see also 9.8 Form Factor) to get the RMS value of the signal. But this is only valid for sinusoidal signals. For non-sinusoidal signals the form factor may be quite different. In reality signals are often deformed, So these analyzers are not appropiate for SPECpower measurements as the relative error may exceed drastically.

11.3. Crest Factor

The crest factor is the ratio between the peak value and the TRMS value of a signal.

$$U_{cff} = \frac{U_{Peak}}{U_{TRMS}}$$

The crest factor is an indicator for the capacity of the power analyzer. For sinusoidal AC the crest factor is

$$U_{cff} = \sqrt{2}$$

This means that the analyzer has to handle much higher peak values, e.g. the analyzer measuring 2 Ampere with 230V has to handle peak values of 2.82 Ampere. In reality the signals are not optimally sinusoidal ,so SPECpower requires a crest factor of at least 3 for conform measurements.

11.4. Current

Current is defined as the flow of an electric charge.

- The flowing charge can be either a positive charge, an ion, or a negative charge, an electron.

- The flow of an ion from a positive side of a power source to the negative side of a power source is the conventional idea behind current flow, and is therefore called conventional current.
- However, the flow of an electron from the negative side of a power source to the positive side of a power source produces the same electric current as in the conventional current model.
- As a result of this effect, all flowing charges, and therefore current, are assumed to have a positive polarity when measuring current, or solving electrical circuits.
- Current is measured in Amperes (A), or Amps for short, which is a base unit. The definition of electric charge, a Coulomb, is derived from the formula, $1A = \frac{1Coulomb}{1\sec ond}$. This provides the following equations for current, $I = \frac{Q}{t}$ or $Q = I * t$. In these equations, I is current, Q is electrical charge, and t is time.
- The electric charge represented by the 1 Coulomb value in the formula in the previous line can either be a constant charge, or it can vary with time.
- This provides the basis for the two different types of current, Alternating current (AC) and Direct current, (DC). These are discussed in the following clause, 11.5.
- Current can also be calculated by using Ohm's Law, $I = \frac{V}{R}$. Although Ohm's Law assumes an ideal resistor through the applied voltage, this equation can be used as long as a total resistance or impedance can be found.

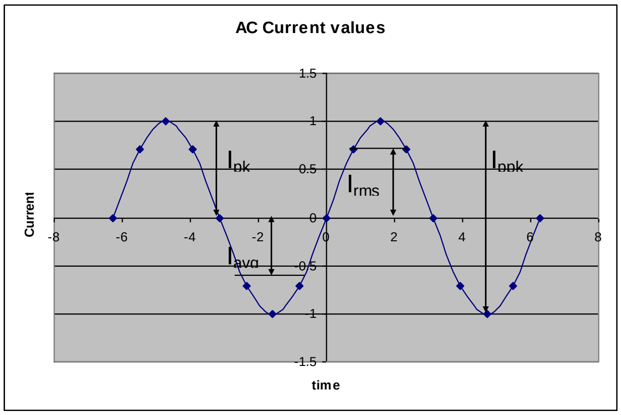
11.5. Types of current - AC / DC

11.5.1. AC - Alternating current

Alternating current is defined by the flow of electrons, or the flow of an electrical current in general, which varies direction on a cyclical basis.

- Alternating current is usually represented by a sine wave when referring to AC power, but it can also be represented by a triangle or square wave.
- Alternating current used in most countries is generated and delivered at a frequency of 50 Hz or 60 Hz. However, some countries do use a mixture of 50 Hz and 60 Hz power supplies. For a description of frequency, refer to clause 11.6.
- In the United States for example, the AC signal is generated and delivered at 60 Hz whether it is the standard 120V, single phase AC that is supplied from a wall outlet; the 208V, single phase that many servers run off of; or the 240V, 3 phase AC which is used for applications such as air conditioners.
- Due to the change in frequency of an AC signal, many calculations become more complex and more variables are introduced. This can be seen in clause 11.7.1when talking about AC voltage.
- An AC signal can have root mean square (rms) values, average values, peak values, and peak-to-peak values. For sinusoidal signals these values are calculated using the following equations.
  - $I_{avg} = \frac{2}{\pi} * I_{pk} \approx 0.637 * I_{pk}$
  - $I_{rms} = \frac{\sqrt{2}}{2} * I_{pk} = \sin(\frac{\pi}{4}) * I_{pk} \approx 0.707 * I_{pk}$
  - $I_{pk} = 0.5 * I_{ppk}$
  - $I_{avg} = \frac{1}{\pi} * I_{ppk} \approx 0.318 * I_{ppk}$
  - $I_{rms} = \frac{\sqrt{2}}{4} * I_{ppk} \approx 0.353 * I_{ppk}$
  - $I_{avg} = \frac{2\sqrt{2}}{\pi} * I_{rms} \approx 0.900 * I_{ppk}$

- Below is a graph that represents the different current values relative to one another on a sine wave.



11.5.2. DC - Direct current

Direct current is defined by the constant flow of electrons, or an electrical current in general, in a single direction.

- Direct current does not vary direction, and therefore stays at one constant frequency of zero, and constant voltage value.
- Most DC power sources do not maintain an absolutely constant voltage. Some variation is normal and is quantified as a “non-DC” component of the DC current source.
- A common source of a DC signal would be a battery of any sort. A battery has a constant voltage and delivers a constant current.
- Also, since the current is constant in a DC signal, many DC calculations are straight forward and follow simple formulas such as Ohm’s Law shown in clause 11.4 or the first equation for voltage and power given in clause 11.9.1.

11.6. Frequency

Frequency is the measurement of the recurrence of an event, or number of cycles, per unit time. The period is the reciprocal value of frequency, and is defined as the duration of one cycle of a recurring event.

- Frequency is measured in Hertz (Hz).
- The period is measured in seconds (s).
- Frequency and period are represented by the equation, $f = \frac{1}{T}$.
- 1 Hz indicates that an event repeats once per second.
- 2 Hz indicates that an event repeats twice per second.
- One frequency that alternating current is generated and delivered at is 50 Hz. Most of Europe has their electrical systems operating on this frequency.
- Another frequency that alternating current is generated and delivered at is 60 Hz. The United States and some other countries have their electrical systems operating on this frequency.

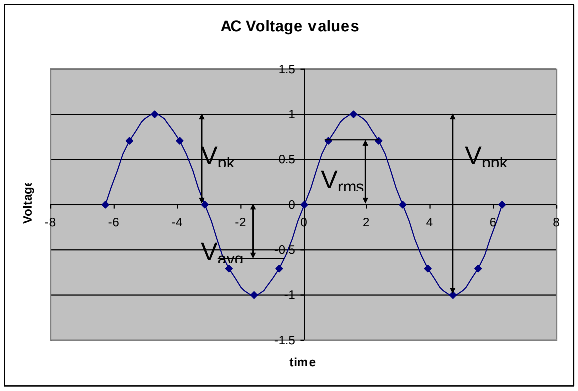
11.7. Voltage

Voltage is defined as the electrical potential difference between two points.

- Voltage is measured in Volts (V).
- A basic equation for voltage shows that voltage is the product of current and resistance, as seen here: $V = I * R$.
- Voltage is also used to compute real power. The general formula for real power (P) is P=V*I, where P is real power (Watts), V is voltage (Volts), and I is current (Amps). The multiple types of power, which are discussed further in clause 11.9, all use some form of voltage to compute their values.

11.7.1. AC Voltages

- When measuring voltage in with an Alternating Current (AC), there are multiple types of voltage that can be used in calculations. These types of voltage include V_{avg}, the average voltage over a given time period; V_{rms}, the root mean square voltage; V_{pk}, the peak voltage of an AC signal; and V_{ppk}, the peak-to-peak voltage of an AC signal.
- A brief list of the different AC voltage types and the equations relating sinusoidal signals to each other can be seen in here:
  - $V_{avg} = \frac{2}{\pi} * V_{pk} \approx 0.637 * V_{pk}$
  - $V_{rms} = \frac{\sqrt{2}}{2} * V_{pk} = \sin(\frac{\pi}{4}) * V_{pk} \approx 0.707 * V_{pk}$
  - $V_{pk} = 0.5 * V_{ppk}$
  - $V_{avg} = \frac{1}{\pi} * V_{ppk} \approx 0.318 * V_{ppk}$
  - $V_{rms} = \frac{\sqrt{2}}{4} * V_{ppk} \approx 0.353 * V_{ppk}$
  - $V_{avg} = \frac{2\sqrt{2}}{\pi} * V_{rms} \approx 0.900 * V_{ppk}$
- Below is a graph that represents the different voltage values relative to one another on a sine wave.



Note: The voltage measurements taken during the benchmark are measured in V_{rms}.

11.7.2. DC Voltages

- However, when measuring voltage with a Direct Current (DC), only a single type of voltage is used. This comes from the fact that a DC signal is at a constant current, constant voltage, and zero frequency. This means that the signal does not vary over time like an AC signal, and therefore, there is only one type of voltage to measure.
- For an explanation on AC and DC signals and a comparison of the two, refer to clause 11.5.

11.8. Form factor

The form factor is the ratio of the RMS value of a signal and the rectified value. As the simple average value of sinusoidal signals is zero, the rectified value was introduced:

$$U_{\mathrm{Re}\,ct} = \sqrt{\frac{1}{T} \int_{t=0}^{T} | u(t) | * dt}$$

The rectified value has another proportional factor for every shapeform to get the RMS value.

$$U_{ff} = \frac{U_{RMS}}{U_{\mathrm{Re}\,ct}}$$

For sinusoidal signals the form factor is 1.11

11.9. Power

Power is defined as the transmission rate of electrical energy.

- Just as with voltage, there are multiple types of power. Power can be broken down into DC Power, and AC Power.

11.9.1. DC Power

- Power in a DC circuits is straight forward and easy to measure and calculate.
- The instantaneous power in a DC circuit is found though Joule's Law. Using this law, the equation for instantaneous power is $P = V * I$.
- Here P represents power, measured in Watts (W), V represents voltage, measured in Volts (V) and I represents current, measured in Amps(A).
- However, if one is measuring or calculating the lost in a resistor, the either of the following equations can be used. $P = I^2 * R$ or $P = \frac{V^2}{R}$.

11.9.2. AC Power

- In an AC circuit, the measurement and calculations of power get a bit more complex since there are multiple types of power to consider. Among the different types of AC power are real power, reactive power, complex power, and apparent power.
- Even though watts (W) is the unit for all forms of power, it is generally reserved as the unit for real power since real power is the actual power consumed in a system.

11.9.3. Real Power, P

- Real power is also known as true power, effective power, or active power.
- This power is calculated in a very similar manner to power in a DC circuit.
- Real power describes the power which is transferred to the load and does not return while a defined time. So real power is the average value of the power oscillation. It's formula is:

$$P = \frac{1}{T} \int_{t=0}^{T} (v(t) * i(t))dt$$

- The equation for real power is $P = V_{RMS} * I_{RMS} * \cos\varphi$. Here P is the average power, measured in Watts (W). And φ represents the phase angle between the voltage and the current signals (see also 9.10 Power factor).
- The smaller the phase angle gets, the closer the voltage and current are to being in phase with each other. This causes the real power to get larger and act more like power in a DC circuit. In this situation, the real power is then sometimes called effective power.
- Conversely, the larger the phase angle gets, the more out of phase the voltage and the current are when compared to one another. This causes the real power to get smaller and allows reactive power (Q) to take over in the circuit.

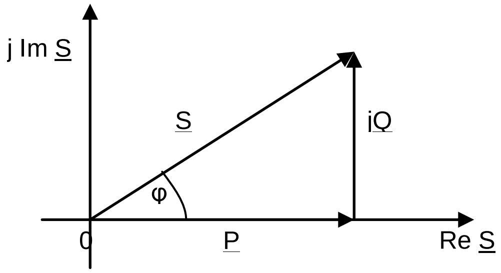
11.9.4. Reactive Power, Q

- Reactive power, Q, is the imaginary part of the total AC power equation when it is put together with real power. It is the power which is transferred to the load and returns while a defined time.
- Reactive power is measured in volt-amperes reactive (Var) and 1 Var =1V*A.
- The equation for reactive power is $Q = V_{rms} * I_{rms} * \sin\varphi$.

- The unit of a Var represents the power consumed by a reactive load.
- A reactive load is a load that consists of capacitors and inductors.
- In this situation there is no net power flow (P=0), and energy just flows back and forth between the components of the circuit.
- Reactive power is not delivered to or consumed by the load.
- VArs can be minimized by balancing the reactive loads in a circuit, or by add off-setting reactive loads. By minimizing VArs in an AC circuit, one maximizes the transmission efficiency of the real power.

11.9.5. Complex and apparent power

- Complex and apparent power are very closely related to each other. In fact, apparent power is nothing more than the absolute value of complex power, and they both carry the same unit of measurement, the volt-ampere (VA).
- Apparent power is the consumption a load seems to have if only measuring the True RMS voltage and True RMS amperage and multiplying them. The phase shift or distortion of the signals are not taken into calculation
  $S = V_{RMS} * I_{RMS}$
- Complex power is the combination of real and reactive power. It is represented by the symbol S.
- The formula for complex power is as follows $S = P + jQ$. Where P is real power, Q is reactive power, and j is the imaginary unit.
- Since apparent power is the absolute value of the complex power, apparent power will always be higher than real power. This leads to apparent power being used as the power rating for different devices.
- A graphical view of how the AC powers are related can be seen below.



11.10. Power factor

The power factor of an AC circuit is defined as being the ratio of the real power to the apparent power.

- This number is always between 0 and 1 and is often expressed as a percentage.
- When power factors are not expressed as a percentage, leading or lagging will is typical written behind the value. Leading and lagging refer to whether the current is ahead of the voltage, or the voltage is ahead of the current. The leading and lagging varies depending on the type of load.
- Leading and lagging notation also shows the sign of the phase angle. A leading mark indicates a negative sign.
- The equation for the power factor is $\frac{P}{S} = |\cos\varphi|$ where φ is the phase angle, P is real power, and S is apparent power.
- If the power factor is 1 then the load of the circuit is consuming all the power, meaning there is only real power.
- If the power factor is 0 then the load of the circuit is purely reactive and there is only reactive power.
- Ideally the power factor should be as close to 1 as possible

- It is possible to correct the power factor of load by adding or subtractive reactive loads from the circuit.
- In many electrical data sheets, the power factor is represented by $\cos\varphi$.

11.11. Energy

The international standard measure for energy is the Joule, or one Newton-Meter/Second

- The electrical equivalent for energy is measured in watt-hour (Wh) or Kilowatt-hour (KWh). 1Wh=3600Joules
- The accumulated transmission rate of electrical energy
- Used in long-term contexts, e.g., annual energy usage

11.11.1. Conversion: Power [Watt] -> Energy [kilo Watt hour]

Question:

How much kWh is used by a 300 Watt server in a year under maximum load?

Calculation:

300 (W/1000)*(24hours*365.25days)) =
300 W*8.766kh = 2629.8 kWh

Answer:

A 300 W server used 2629.8 kWh per year under maximum load.

11.12. Efficiency

Efficiency is a unit less ratio of output-power to input-power.

- The equation for efficiency is $\eta = \frac{output - power}{input - power}$, where η is the symbol for efficiency.
- The Maximum Power Theorem clearly shows that devices transfer the maximum power to a load when running at 50% of the electrical efficiency.

11.13. Temperature Conversion

Measured in Degree Fahrenheit (F) or Degree Celsius

- Fahrenheit = (Celsius value * 9/5) + 32
- Celsius = (Fahrenheit value -32) * 5/9

Fahrenheit	-4.00	32.00	68.00	104.00	140.00	176.00	212.00
Celsius	-20.00	0.00	20.00	40.00	60.00	80.00	100.00