Power In, Dollars Out: How to Stem the Flow in the Data Center

Server Power Considerations for IT Administrators

July 2, 2010

Abstract

This document provides a comprehensive analysis of the server power landscape for information technology (IT) administrators. It explains the effect of server power usage on total cost of ownership (TCO) for IT organizations, shows the intricacies of the power-versus-performance tradeoff in the server realm, and provides detailed information to help IT administrators make power-conscious purchasing and usage planning decisions.

This information applies to the following operating systems:  
 Windows Server 2008   
 Windows Server 2008 R2

References and resources discussed here are listed at the end of this paper.

For questions or to provide feedback on this paper, contact:  
 [srvpwrfb@microsoft.com](mailto:srvpwrfb@microsoft.com)

The current version of this paper is maintained on the Web at:   
 <http://www.microsoft.com/whdc/system/pnppwr/powermgmt/Svr_Pwr_ITAdmin.mspx>

***Disclaimer****: This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.*

*Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.*

*This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.*

© 2010 Microsoft Corporation. All rights reserved.

Document History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Change |  |  |  |
| July 2, 2010 | Added descriptions of the new features in Windows Server 2008 R2. Updated the best practices list to reflect the new features. | | | |
| November 17, 2008 | First publication | | | |

Contents

[Introduction 4](#_Toc265841425)

[Background 4](#_Toc265841426)

[Simple Power-Saving Methods 5](#_Toc265841427)

[Power Efficiency Best Practices Checklist 5](#_Toc265841428)

[Shut Down Idle Machines 5](#_Toc265841429)

[Deploy Power-Efficient Hardware 6](#_Toc265841430)

[Windows Server Logo Additional Qualifier for Enhanced Power Management 6](#_Toc265841431)

[2.5-Inch Disk Drives 6](#_Toc265841432)

[Low-RPM Disk Drives 6](#_Toc265841433)

[Power-Efficient Processors 6](#_Toc265841434)

[Memory 6](#_Toc265841435)

[Power Supplies and Cooling Fans 7](#_Toc265841436)

[Remote Power Strips 7](#_Toc265841437)

[Track Power Consumption and Set Power Budgets 7](#_Toc265841438)

[Use Appropriate Operating System Power Plan Configurations 7](#_Toc265841439)

[Identify Inefficient Hardware, Drivers, and Software with Powercfg /energy 7](#_Toc265841440)

[Tune Processor Power Management Parameters in Windows Server 2008 8](#_Toc265841441)

[Increase Data Center Efficiency through Virtualization 8](#_Toc265841442)

[Use the Latest Windows Server and Service Pack Updates 8](#_Toc265841443)

[Minimize Unnecessary System Activity 8](#_Toc265841444)

[Power Costs, Tradeoffs, and Analysis 8](#_Toc265841445)

[The Increasing Cost of Data Center Power 9](#_Toc265841446)

[Power Capacity 9](#_Toc265841447)

[Environmental Effect 9](#_Toc265841448)

[Government Oversight 9](#_Toc265841449)

[The Power/Performance Tradeoff 9](#_Toc265841450)

[Efficiency Analysis, Benchmarks, and Metrics 10](#_Toc265841451)

[ACPI Overview 11](#_Toc265841452)

[ACPI Processor States 12](#_Toc265841453)

[C-States 12](#_Toc265841454)

[P-States 12](#_Toc265841455)

[T-States 13](#_Toc265841456)

[System-Level Power 13](#_Toc265841457)

[Active System Power Consumption 14](#_Toc265841458)

[Power Consumption of Idle Systems and Components 15](#_Toc265841459)

[Operating System Effects on Server Power 17](#_Toc265841460)

[Windows Server 2003 vs. Windows Server 2008 R2 18](#_Toc265841461)

[Power Plan Selections 19](#_Toc265841462)

[Advanced Processor Power Management Concepts 19](#_Toc265841463)

[Performance and Idle State Transitions: Single Processor 19](#_Toc265841464)

[Performance and Idle State Transitions: Multiple Processors 20](#_Toc265841465)

[Driver and Application Effects on Power Efficiency 20](#_Toc265841466)

[Maintaining Idle States 20](#_Toc265841467)

[Interrupts 21](#_Toc265841468)

[Timers 21](#_Toc265841469)

[Processor Affinity 21](#_Toc265841470)

[Measuring PPM Effectiveness 22](#_Toc265841471)

[Preventing Problems 22](#_Toc265841472)

[Removing Unnecessary Software 22](#_Toc265841473)

[Turning Off or Unplugging Unnecessary Hardware 22](#_Toc265841474)

[Using In-Box Drivers Where Possible 22](#_Toc265841475)

[Increasing Data Center Utilization 22](#_Toc265841476)

[Overprovisioning 23](#_Toc265841477)

[Monitoring 23](#_Toc265841478)

[System Consolidation 23](#_Toc265841479)

[Virtualization 23](#_Toc265841480)

[Multiple Roles in Windows Server 2008 R2 23](#_Toc265841481)

[Dynamic Provisioning 24](#_Toc265841482)

[Virtualization and Power Management in Windows Server 2008 R2 24](#_Toc265841483)

[Guest Operating System Enlightenments 24](#_Toc265841484)

[Performance 24](#_Toc265841485)

[Hardware Component Effects on Power 24](#_Toc265841486)

[Processors 24](#_Toc265841487)

[Performance States 25](#_Toc265841488)

[Idle States 25](#_Toc265841489)

[Low-Voltage Processors 25](#_Toc265841490)

[Memory 25](#_Toc265841491)

[Memory Family 26](#_Toc265841492)

[Bus Speed 26](#_Toc265841493)

[Memory Capacity 26](#_Toc265841494)

[Chip Density 26](#_Toc265841495)

[Additional Features 26](#_Toc265841496)

[Storage 27](#_Toc265841497)

[Size Reduction 27](#_Toc265841498)

[RPM Reduction 27](#_Toc265841499)

[Solid-State Disks 27](#_Toc265841500)

[RAID Selection 28](#_Toc265841501)

[Network Adapters 28](#_Toc265841502)

[Remote Power Strips 28](#_Toc265841503)

[Cooling 29](#_Toc265841504)

[Power Supply Units 29](#_Toc265841505)

[Power Supply Unit Efficiency 29](#_Toc265841506)

[Efficiency Programs 30](#_Toc265841507)

[Data Center Infrastructure 30](#_Toc265841508)

[Resources 30](#_Toc265841509)

[Appendix. Viewing PPM Counters by Using Perfmon.exe 33](#_Toc265841510)

[End Notes 39](#_Toc265841511)

# Introduction

This paper presents a comprehensive analysis of the server power landscape for information technology (IT) administrators. It explains how power affects the IT budget, shows power and performance tradeoffs, and describes in detail how hardware and software can affect overall power efficiency in the data center. You can use this information to make power-aware deployment and purchasing decisions, identify issues and inefficiencies on deployed systems, and maximize your organization’s power efficiency.

This paper provides background on server power concepts and issues, summarizes the most important and effective approaches for saving power, and provides information on some advanced server power topics.

# Background

This document focuses on present-day hardware components and software applications that work with the Windows Server® 2008 and the Windows Server 2008 R2 operating systems.

The goals of this document are to:

* Provide simple, usable suggestions for increasing server power efficiency.
* Help you make power-conscious purchasing and usage planning decisions that can reduce power costs and reduce wasted capacity.
* Enable you to identify inefficient hardware and software components and correct inefficiencies.

Calculations and estimates in the paper are based on the following assumptions:

* According to the United States Energy Information Administration, the cost of electricity in the United States is 10.51 cents per kilowatt hour (kWh), based on the average United States commercial rate as of June 2009. You should adjust calculations to match local electricity costs.
* A 24-hour-a-day, 7-day-a-week (24x7) server runs for 8,760 hours per year.

In addition, this paper uses the term “server” to refer only to in-box parts. External disk arrays, monitors, input devices, uninterruptible power supplies (UPSs), and other items are not included in power calculations.

To achieve any power-efficiency savings, you must understand the workload of a server. Selecting hardware or configuration parameters without understanding the workload of the machine can lead to poor performance or power efficiency.

First, gauge the utilization levels of server subsystems such as disk, network, CPU, and memory on an existing system. You can then reduce excess capacity or provision future capacity only in the subsystems where it is needed and opt for low-power parts if that is feasible.

For example, computationally intensive workloads usually do not need fast disks, large redundant array of independent disks (RAID) arrays, or quad-port 10‑gigabit per-second (Gbps) network adapters, but generally require as many processors as possible.

# Simple Power-Saving Methods

Some simple power-saving methods can help reduce the power footprint in most scenarios. This section describes some basic power concepts and methods to give you some good first steps for reducing power consumption before you investigate more detailed methods.

## Power Efficiency Best Practices Checklist

The following are best practices that will help you to achieve power efficiency:

* Shut down idle machines during off-peak times.
* Use 2.5-inch disk drives instead of 3.5-inch disk drives.
* Use low-revolutions-per-minute (RPM) disk drives where it is possible.
* Select power-efficient processors and memory.
* Install variable-speed fans and efficient power supplies in servers to reduce waste.
* Use remote-controlled power strips to completely eliminate electricity flow to servers that are turned off.
* Ensure that Windows Server 2008 and Windows Server 2008 R2 servers are configured to use the Balanced power plan.
* Use the PowerCfg utility with the /**energy** option in Windows Server 2008 R2 to identify energy efficiency issues.
* Use servers that meet the Windows Server Logo Advanced Qualifier (AQ) criteria for Enhanced Power Management.
* Use metering and budgeting features in Windows Server 2008 R2 to monitor power consumption and set power budgets.
* Use Microsoft® Hyper-V™ to consolidate workloads or combine server roles on idle and underutilized servers where it is possible.
* Use Windows Server 2008 R2 as the root and guest operating systems in virtualized deployments.
* Install the latest service packs and Windows Server releases.
* Remove or shut down unnecessary roles, applications, and devices.

The following sections describe these recommendations in more detail.

## Shut Down Idle Machines

Workloads vary over time. Some workloads run only at specific times of the day, whereas other workloads are dynamic and user driven. By identifying consistent, long periods of nonuse, you can shut down servers when they are not being used. For example, backup, test, and build servers are typically idle for long periods during the day. According to *TechNet Magazine*’s “Sustainable Computing: Is It Time to Turn Off Your Servers?,” servers typically consume more than 50 percent of their peak instant power at idle. Turning off servers when they are not needed can save a lot of electricity.

## Deploy Power-Efficient Hardware

Choosing power-efficient hardware when you deploy new servers or upgrade existing servers is a simple, cost-effective way to increase power efficiency. These components can cost more up front, but you can view the operational cost savings as a return on investment (ROI). Other than the initial deployment or installation, low-power hardware incurs no additional management overhead.

### Windows Server Logo Additional Qualifier for Enhanced Power Management

Use servers that meet the Windows Server Logo Additional Qualification (AQ) criteria for Enhanced Power Management. Servers that meet this criteria support power efficiency features such as processor power states, metering and budgeting, and support for operating system power management control or Processor Clocking Control (PCC). For more information on which servers qualify for the Additional Qualification, see the link for the Windows Server Catalog in “[Resources](#_Resources).”

### 2.5-Inch Disk Drives

Microsoft test data shows that power consumption for 2.5-inch disk drives is about half that of 3.5-inch disk drives that have comparable capacity and speed. For more information, see “[Size Reduction](#_Size_Reduction)” later in this paper.

### Low-RPM Disk Drives

For storage installations that have no strict latency requirements, 15,000-RPM enterprise-class disk drives may be unnecessary. If 10,000-RPM or even 7,200-RPM disk drives can adequately satisfy performance goals, use them instead of high-RPM disk drives to significantly reduce power consumption.

### Power-Efficient Processors

Using power-efficient processors can save significant wattage. Processor manufacturers have built power management features into processors for several years, and the manufacturers are working on additional power management features for future product lines. Some processor families incorporate low-power states, whereas other processor families are designed as low-voltage parts.

### Memory

Memory module power consumption varies widely from one module to another. Bus speed plays a large factor in memory power consumption, but so do density, rank, and operating voltage. On a system that has many sticks of RAM, the memory power footprint becomes a large percentage of system power and a prime target for savings. You should review module specifications and choose sticks that consume the least power but still meet your business goals and scenario requirements.

### Power Supplies and Cooling Fans

Power supplies and cooling fans are important components to consider when you want to reduce power waste in the data center. Investing in high-efficiency power supplies and variable-speed fans can reduce unnecessary power consumption by a significant percentage, which saves money in the long term and frees important capacity for other uses.

### Remote Power Strips

Currently, systems that are shut down can still consume tens of watts (W) of power. Unplugging systems is the only way to eliminate this power waste, but physically doing so might not be possible for all organizations. Remote power-control strips let you automate this process and can save large quantities of otherwise wasted electricity.

## Track Power Consumption and Set Power Budgets

Use the metering features in Windows Server 2008 R2 to monitor system power consumption. Server hardware that runs Windows Server 2008 R2 and contains an onboard metering device can communicate power consumption data to the operating system. You can access the data by using Windows Management Instrumentation (WMI) or through the Windows Power Management Pack for System Center Operations Manager 2007 R2.

If your server has power usage constraints, you can use the power budgeting features in Windows Server 2008 R2 that are paired with the power metering features. Note that implementing power budgeting can impact overall system power efficiency.

For more information, see “[Power Metering and Budgeting](http://msdn.microsoft.com/en-us/library/ff543910(v=VS.85).aspx)” on MSDN®.

## Use Appropriate Operating System Power Plan Configurations

A simple change that can generate power savings is to ensure that the power plan on deployed systems enables operating system power management technologies. The Balanced power plan, which is enabled by default in Windows Server 2008 and Windows Server 2008 R2, is most appropriate to deliver power efficiency across the widest range of server applications. For more detail, see “[Power Plan Selections](#_Power_Plan_Selections)” later in this paper.

## Identify Inefficient Hardware, Drivers, and Software with Powercfg /energy

Use the PowerCfg utility with the /**energy** option in Windows Server 2008 R2 to identify energy efficiency issues.

In Windows Server 2008 R2, the PowerCfg utility supports the new **/energy** command-line option that you can use to analyze the energy efficiency of a server. When you run the **powercfg** command with the **/energy** option, the utility performs a 60-second test to detect potential energy efficiency issues. The utility generates a simple HTML report in the current directory. The report lists available processor power management (PPM) states and identifies power issues, including USB devices and drivers that lack selective-suspend support and applications that request high-frequency system timer resolution. For more information on the **powercfg /energy** command, see “[Resources](#_Resources).”

## Tune Processor Power Management Parameters in Windows Server 2008

The default processor power parameter (PPM) settings in the Windows Server 2008 Balanced plan are “safe” values that reduce the potential for power savings algorithms to negatively affect performance. We have identified a set of parameters that can increase power efficiency by up to 10 percent on some workloads. For more information, see ”[Resources](#_Resources).”

This guidance applies only to Windows Server 2008. The power parameter defaults are updated in Windows Server 2008 SP2 and Windows Server 2008 R2.

## Increase Data Center Efficiency through Virtualization

Currently, the most power-efficient servers run at full utilization. However, according to the United States Environmental Protection Agency (EPA) “Report to Congress on Server and Data Center Energy Efficiency,” production servers run at anywhere from 5‑ to 15‑percent utilization on average. You might be justifiably worried about affecting Quality-of-Service (QOS) levels by increasing utilization, but in many situations, converting physical machines into virtual machines and consolidating them onto fewer servers by using Hyper-V can eliminate overhead, significantly reduce power footprint, and increase efficiency. For more information on saving power by using virtualization, see “Improve Energy Efficiency and Manage Power Consumption with Windows Server 2008 R2” on the WHDC website.

## Use the Latest Windows Server and Service Pack Updates

New power management features and settings are frequently delivered in new operating system editions and service packs. Updating can provide a more power-efficient experience. Results shown in “[Windows Server 2003 vs. Windows Server 2008 R2](#_Windows_Server_2003)” later in this paper demonstrate that Windows Server 2008 R2 is 14‑percent more power efficient at equivalent utilization levels than Windows Server 2003. Keeping current on operating system releases and service packs is the best way to ensure that your Windows servers are achieving maximum efficiency.

## Minimize Unnecessary System Activity

Servers at low utilizations can reduce power consumption by entering low-power states. Unnecessary or poorly written applications and device drivers can interrupt these states and prevent a system from achieving the lowest possible power consumption. To ensure that a system maximizes usage of idle states, you should remove all nonessential roles, applications, and devices.

# Power Costs, Tradeoffs, and Analysis

Before you try to optimize server power consumption, we recommend that you understand the cost motivations, tradeoffs, and analysis metrics that are involved in the process.

## The Increasing Cost of Data Center Power

The increase in power consumption for IT equipment over the last decade[[1]](#endnote-1) and the commoditization of server-class computer hardware have led to accelerated change in the dynamics of IT administration. *Electronics Cooling* magazine reports that annual utility costs just to turn on and keep a server cool are approaching the up-front cost of a server. Consider a server farm of one thousand servers whose idle power consumption is 100 W. Running 24x7, this server farm costs $92,067.60 per year *just to keep powered on*.

Server power consumption is also a primary target for cost reduction in the data center because of the multiplicative effect of cooling and infrastructure costs. A recent United States EPA “Report to Congress on Server and Data Center Energy Efficiency” suggests that equipment power represents only half the total electricity bill in the data center, with the other half going to support and infrastructure equipment such as air conditioning, fans, network switches, and UPSs. Each watt that is used to power a server can require an additional watt for support and cooling equipment.

### Power Capacity

According to the same EPA report, data center energy capacity is more often a motivation for reducing power footprint than electricity cost. Capacity expansion requires new power and cooling equipment or, in the worst case, new site construction. These costs are significantly more prohibitive than the cost of energy.

### Environmental Effect

Concern for the environment is another reason to reduce the power footprint of IT installations. A smaller power footprint increases the environmental sustainability of an organization.

### Government Oversight

Governments around the world are increasingly sensitive to energy issues. The United States EPA published its ENERGY STAR mandates for enterprise server power efficiency. Carbon tax programs have been instituted in countries around the world, including Sweden, the Netherlands, and New Zealand. Failure to take action now may mean increased costs for your business in the future. Power-aware decision-making helps smooth your business’s compliance with government mandates and may qualify you for future tax incentives.

## The Power/Performance Tradeoff

Design tradeoffs are common in any technical endeavor.

You often must consider a power/performance tradeoff when you make purchasing decisions. Consider a new machine configuration where a cheaper processor model is available at 1.8 gigahertz (GHz) and a more expensive version is available at 2.1 GHz. The performance increase might cost hundreds of additional dollars. Power efficiency involves a similar cost tradeoff—low-power parts can cost more up front, but in this scenario the price increase is offset by savings over the life of the product.

To complicate things further, low-power parts may decrease performance in some manner. “Green” memory might have lower bus speeds and throughput, whereas low-power disks might have reduced capacity or increased latency. These performance penalties can reduce the useful lifespan and versatility of a system or component.

Rarely is absolute power savings possible without complex tradeoffs.

## Efficiency Analysis, Benchmarks, and Metrics

Defining a standard of measurement for server systems is important. Many benchmark measurements seek to quantify only the maximum amount of work that a particular configuration can perform. In terms of power consumption, this information is not as valuable. “Power efficiency” is a much more useful metric. Power efficiency is the ratio of work that is done over time to the power that is consumed over time.

In equation form, this is given as the following:

Over a specific unit of time, a server system can do a specific amount of computational work. This work is known as its *throughput*. Over the same unit of time, the server consumes a particular quantity of power. The time units are the same (generally seconds), so units can be eliminated from the equation. Because power measurements are generally measured in watts, the equation reduces to the following:

Two general mechanisms improve power efficiency:

* Delivering more performance at the same power level.
* Delivering equivalent performance at a lower power level.

The analysis in this paper focuses largely on the second mechanism.

Generating a “load line” for a system configuration is useful. You can do this by measuring the power consumption of a system although throughput varies across the utilization range of a system, from idle up to 100-percent utilization. Figure 1 is an example of the load line concept.

(Lower is better)

Typical Server Utilization Range

Figure 1. Load line example

Figure 1 is a graph of the power consumption of three system configurations across their load lines. Configuration 3 is the most power-efficient configuration at all points except at 10‑percent utilization, where configuration 2 is more power efficient. Configuration 1 has the highest power consumption across the load line, which makes it the least power efficient. All configurations can achieve the same level of peak throughput. By using the load line approach, we can evaluate and compare the performance and energy consumption of different configurations.

The load line is a good measurement approach because measuring power only when hardware is fully utilized does not reflect real-world usage. Production servers on average run at much lower utilizations, typically in the range of 5- to 15‑percent for nonvirtualized servers.

You can use several benchmarks to collect load line data. SPECPower from the Standard Performance Evaluation Corporation (SPEC) is one such benchmark. Internal file and web server workloads can be used. Also, you can modify several traditional server-class benchmarks to vary their workload. We have had success with this approach by using the TPC‑E benchmark from the Transaction Processing Performance Council (TPC).

# ACPI Overview

The Advanced Configuration and Power Interface (ACPI) benefits both hardware and operating system designers by clearly defining an interface between the operating system software and hardware that is to be used for power management. Without providing details about the data structures or communication mechanisms that are involved, this section describes some power management states that exist in practice because of ACPI. For more information, see “[Resources](#_Resources).”

## ACPI Processor States

The processor has traditionally consumed the most power in a server, which makes it a great candidate for power-efficiency optimizations. To add detail and flexibility for processor power management, ACPI defines three sets of states for processors.

You can find comprehensive Microsoft information about PPM in “Processor Power Management in Windows Vista and Windows Server 2008,” on the WHDC website.

### C-States

C-states define incremental levels of processor idle, from C0 (active) to Cn (lowest power idle). The ACPI Specification Revision 2.0 and later does not specify a maximum number of C-states, so Cn is used to refer to the highest-numbered, lowest-power idle state that a processor supports.

C0 is the state at which a processor executes instructions, whereas C1 and greater are nonoperational idle states. C1, C2, … Cn are by definition sequentially lower power (and higher latency) idle states. For example, a processor at C1 idle requires a short time to return to C0. A processor at C2 idle requires more time than C1, but draws less power. A transition from idle state C1 to a deeper idle state C2 is often called *promotion*, whereas a transition from a deep idle state such as C3 to state C2 is often called *demotion.*

#### Implementation

C-state implementations generally involve shutting down successively larger areas of the processor floor plan as the processer enters deeper idle states. A simplified theoretical model might be as follows: the first idle state might involve turning off power to the execution units. No work must be done, so this silicon is already unused, and turning off these components eliminates leakage current. The execution units can be restored to operation very quickly because the processor’s execution context has not changed.

The next idle state might involve a shutdown of the processor’s first-level cache. By shutting down these caches, you can save significant power. Caches also have a hierarchical construction, which lets the processor shut down increasingly larger on‑die caches at each successive idle state and possibly even the entire socket.

### P-States

Processors in operation (state C0) can transition between multiple performance states, or P‑states. P-states define incremental levels of processor performance, from P0 (most performant) to Pn (least performant). The ACPI specification does not specify a maximum number of P-states, so Pn is used to refer to the highest-numbered, lowest-performant P‑state that a processor supports.

Each successively higher numbered P-state consumes less power than the previous P‑state. Processors can dynamically switch between these states during operation to provide only as much computational capacity as is necessary, which saves power during periods of low usage.

Figure 2 shows a hypothetical set of six P-states that would be available to a processor. Note that the maximum P-state (P0) has the highest frequency, whereas successively higher numbered P-states reduce in frequency. In this case, the minimum P-state is P5, so the terms Pn and P5 are interchangeable.



Figure 2. Sample P-states with corresponding frequencies

#### Implementation

Because of transistor physics, you can reduce power consumption on a processor by reducing the frequency or by reducing the operating voltage. Therefore, P-states are implemented as reductions of processor frequency, voltage, or both.[[2]](#endnote-2)

### T-States

You can use a set of throttle states (T-states) to reduce power consumption and processor performance by scaling back the quantity of clock cycles by a percentage value. Assuming one operation per clock cycle, a 2.5-GHz processor at a 50-percent T‑state would perform 1.25 billion operations instead of 2.5 billion operations.

#### Implementation

When a processor enters a T-state, the processor’s clock signal is masked for a specific time period. This eliminates the power that is necessary for clock distribution throughout the processor. It also prevents the processor from doing computational work or changing state, which eliminates the processor transistors’ dynamic power consumption.

# System-Level Power

Servers can differ in size, throughput, and component makeup, but generally they tend to exhibit similar power consumption trends. This section describes those trends and gives a high-level perspective on the power consumption of server-class machines.

## Active System Power Consumption

You should understand how power consumption varies as machine utilization changes. Figure 3 was generated by measuring system power consumption as a scalable workload (the TPC-E benchmark workload) was ramped up.

Figure 3. Power and power efficiency versus throughput (in transactions per second)

The top line on the graph represents power consumption as a percentage of maximum power. As shown in Figure 3, idle power consumption is approximately 65 percent of maximum, even though no work is done. This is not unusual behavior. It is common for server idle power to be approximately two-thirds of fully utilized system power consumption.

The bottom line tracks efficiency relative to the maximum, given in transactions per watt. The important takeaway is that an idle server is 0‑percent efficient and the most efficient server (currently) is one at full utilization.

Both findings are strong rationale for increasing overall utilization in the data center, especially through machine consolidation. Combining two boxes into one eliminates the over‑60‑percent idle power overhead that is necessary to keep one of the boxes running and moves the remaining server further along its power efficiency curve.

## Power Consumption of Idle Systems and Components

In a Microsoft test lab, we connected a dual-socket, dual-core server that was shipped in 2005 (Server A) and a quad-socket, quad-core server that was shipped in 2008 (Server B) to power meters at the wall. We measured power consumption for the servers at idle and as components were removed. Components installed in each system are shown in Table 1.

Table 1. Components in 2005 and 2008 Test Systems

|  |  |  |
| --- | --- | --- |
| Component | Server A (shipped in 2005) | Server B (shipped in 2008) |
| Processors | 2 dual-core processors | 4 quad-core processors |
| Memory | 32 2-GB double data rate (DDR) PC-2100 dual inline memory modules (DIMMs) | 32 4-GB DDR-2 PC-5300 DIMMs |
| Network | 1 PCI-X network adapter (single-port, 1‑GB) | 2 PCIe network adapters (quad‑port, 1-GB) |
| Disk controllers | 4 PCI-X host bus adapters—HBAs (dual-port) | 2 PCIe HBAs (2x dual port) |
| Disks | 4 36‑GB, 15,000-RPM, 3.5-inch SCSI | 4 72-GB, 15‑k RPM, 2.5-inch Serial Attached SCSI (SAS) |

In some cases, it was necessary to extrapolate power consumption for components that were required to keep the system running. For example, the system must have one processor and one memory stick. The power measurements for Server B are shown in Figure 4.

Figure 4. Server B system power during device removal experiment

Note the following are important points:

* The processors on this machine could not be removed because of the fragile connection points. Instead, we obtained the idle power data from the manufacturer’s technical specification sheet for the product.
* The memory power that the sticks consumed on the daughterboards (9 W) differs from that consumed on the motherboard (12.5 W). We do not have an explanation for this behavior, so we assume that it is related to the platform design.
* The system consumed 27 W of power when the server was fully powered off. This may seem excessive, but this is not anomalous behavior. The other system consumed 30 W when powered off. Unplugging systems is currently the only way to eliminate this power waste; you can automate this process if you deploy remote-controlled power strips.

The remaining power that is consumed when all possible components were removed or otherwise accounted for was added to an “others” section. Consumers of power in this category include power supply, motherboard and chipsets, fans, and other miscellaneous items.

With all components in place, idle power consumption was measured at 568 W on the 2005 system and 635 W on the 2008 system. Figure 5 and Figure 6 detail the power consumption of individual components on these systems relative to overall system consumption.

TOTAL: 568W

Figure 5. Component power distribution, 2005 two-socket dual-core server

TOTAL: 635W

Figure 6. Component power distribution, 2008 four-socket quad-core server

In the newer server, Server B, the processors are not the largest consumers of power in the system—32 sticks of memory can easily consume as much or more power than other system components at idle.

The cause of the increase in memory power usage is explained in more detail in ”[Hardware Component Effects on Power](#_Hardware_Component_Effects)” later in this paper, but generally a doubling in bus speed or capacity can double the memory power consumption. Because both bus speed and capacity increased between systems, the fourfold increase in memory power consumption is understandable.

Unfortunately, the processors in the 2008 machine could not be removed. From manufacturer specifications, we estimated the idle power consumption to be approximately 136 W. In this scenario, even though the newer system has more cores, the advanced PPM in the newer processor family saves 78 W at idle.

The 48‑percent decrease in disk power consumption is because of the switch from 3.5-inch to 2.5-inch disk drives. For the small quantity of disks in this system, the absolute savings is only 5 percent of overall system power, but for large disk arrays the savings can accumulate quickly.

The network adapter and HBA power variation is due more to quantity and feature set than to any power optimizations.

The large quantities of power that are attributed to the “others” category demonstrate the significant effect that frequently ignored items such as power supplies and fans have on power consumption.

# Operating System Effects on Server Power

Power efficiency cannot be achieved by hardware alone. The operating system plays an important role in increasing a system’s power efficiency. This section details how the Windows Server 2008 R2 operating system contributes to server power efficiency.

## Windows Server 2003 vs. Windows Server 2008 R2

Figure 7 shows that Windows Server 2008 R2 can deliver more throughput than Windows Server 2003 SP2 and can achieve power savings of up to 18 percent at comparable throughput levels. Such savings are possible because of new power management features that were introduced in Windows Server 2008 and Windows Server 2008 R2, many of which are specifically related to ACPI PPM support.

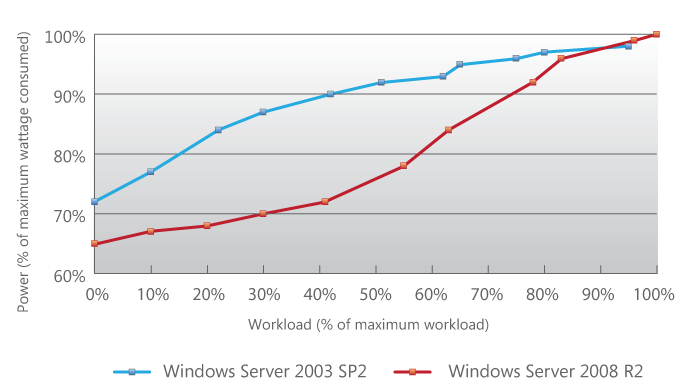


Figure 7. Improved power efficiency in Windows Server 2008 R2

The new power management features include the following:

* Support for all ACPI 3.0 objects.
* Multiprocessor PPM support.
* Noncontiguous idle state support.
* Better in-box driver support.
* Various algorithmic improvements and background task reductions.
* Core parking support.
* Storage power management features.
* Intelligent timer tick distribution (tick skipping).

For more information, see “Improve Energy Efficiency and Manage Power Consumption with Windows Server 2008 R2” on the WHDC website.

## Power Plan Selections

Windows Server 2008 R2 has three power plans that can be accessed through the Control Panel Power Options application:

* **High Performance**

This power plan has few power-saving features enabled. Parameters heavily favor performance over power, so processors and devices constantly consume maximum power. This might be appropriate for a machine where minimizing latency is critically important.

* **Balanced (Default)**

This power plan enables most power management features. Under this plan, Windows determines an optimal state for the processors and devices that delivers necessary throughput while maximizing power savings. The Balanced plan has several options that can be configured in Control Panel and even more that are available through in-box tools such as *powercfg.exe*. These parameters are covered in detail in ”Power Policy Configuration and Deployment in Windows” on the WHDC website.

* **Power** **Saver**

This power plan restricts processors to 75 percent of their maximum performance state and takes the most advantage of low-power device states. This option is most useful for controlling thermal conditions or placing a system under a strict power budget. Workloads might experience Quality-of-Service degradations in the Power Saver plan.

If saving power is a concern, you should choose between the Balanced and Power Saver plan. The Balanced plan takes the best advantage of operating system power management by striking the most optimal balance between power and performance.

## Advanced Processor Power Management Concepts

PPM algorithms in Windows Server 2008 R2 are complex. However, PPM is an important part of server power management. Understanding the concepts that are involved can help you understand how tuning PPM parameters can increase efficiency and how poorly designed hardware, drivers, or application configurations can reduce the effectiveness of power management.

### Performance and Idle State Transitions: Single Processor

Workloads on servers are transient. Workload patterns change throughout the day and are subject to instantaneous spikes. Windows Server 2008 R2 uses ACPI performance states and idle states to match available computational resources to the current system demand.

Under the Balanced power plan that was described earlier, the operating system reevaluates processor performance states at 100-millisecond (ms) intervals, which is called a time check. At each time check, the operating system determines the appropriate P-state or C-state for the processor for the next time interval.[[3]](#endnote-3)

### Performance and Idle State Transitions: Multiple Processors

In today’s market, single-core, single-processor servers are rare. PPM becomes much more complex on multicore processors and multisocket servers.[[4]](#endnote-4) To demonstrate how P-states and C‑states change as system utilization changes, we tracked the cumulative time that a multiprocessor system’s cores spent in various P-states and the C1 idle state on a machine in which a workload was incrementally decreased every 10 minutes. We used this information to create Figure 8.

Figure 8. Distribution of P-states and C1-state as workload decreases over time

As Figure 8 shows, minutes 30 through 40 are typical of full machine utilization. All processors spend almost all their time in performance state P0, and C1 residency is at a minimum. As system utilization decreases, processors are distributed into higher numbered P-states and idle state C1. Minutes 130 through 140 represent the idle scenario and, as expected, we see almost all the time is spent in C1 and P4.

Note that after minute 60, processors enter the C1 state without necessarily entering higher P-states. This is because performance and idle states do not depend on each other. A processor is not required to go through Pn to enter an idle state, and P‑states are considered irrelevant to a processor in an idle state.

# Driver and Application Effects on Power Efficiency

Achieving excellent power efficiency is a delicate task that requires cooperation from all software elements, not just the operating system. The following sections describe how third-party drivers and applications can affect system power efficiency.

## Maintaining Idle States

Dropping a processor into an idle state has associated latency and performance costs.[[5]](#endnote-5) To achieve real power savings, processors must enter the lowest possible idle states and remain there for long periods of time.[[6]](#endnote-6)

A key strategy for power efficiency is to maintain residency in low-power states. Unfortunately, accomplishing this is difficult. Any piece of hardware or software, including drivers and user-mode applications, can break idle state residency and ruin the effectiveness of power management.

### Interrupts

Frequent interrupts are detrimental to power management. Processors must service hardware interrupts quickly regardless of the state they are in, so interrupts pull a processor out of an idle state.

Most devices use interrupts to perform I/O operations.[[7]](#endnote-7) Each time that a computer receives a network packet or a keystroke, the processor receives an interrupt. Although handling interrupts is usually desirable, unnecessary interrupts from third-party device drivers are identified as a common disruptive system activity for PPM.

### Timers

At a predetermined interval on any computer system, a hardware interrupt—which is colloquially called a *timer tick*—is generated to provide clock functionality for the system. Because this is a hardware interrupt, the timer is also a guaranteed “wake‑up” for any idle processors that receive the timer tick. The default for a Windows machine is one timer tick every 15.6 ms.

Applications can adjust the tick rate by using a Windows API. User-mode applications can reduce the tick rate to 1 ms. Short timer intervals adversely affect power management by reducing the chances that processors remain in low-power states long enough to realize nontrivial power savings.

### Processor Affinity

To increase performance, you can set *interrupt affinity—*or a tendency to interrupt a specific processor*—*by using the IntPolicy tool. For more information about this tool, see “Interrupt-Affinity Policy Tool“ on the WHDC website. However, the PPM engine cannot see interrupt affinity and might target affinitized processors for entry into idle states. Constant interrupts to these processors reduce the chance of idle state power savings. If you need to set interrupt affinity, be aware of the potential impact to power efficiency and measure the impact, if you can.

For software applications, you can configure *thread affinity—*or a preference for code to be run on a particular processor. Setting thread affinities to a particular processor can result in better cache hit performance and therefore better application throughput, but may also have harmful effects on power. For example, if you use affinities to spread out work among all processors on a system, throughput might increase but less time will be spent in idle states than in a scenario in which all work is affinitized to one socket or core.

You should consider your goals for power and performance and configure thread affinity appropriately.

## Measuring PPM Effectiveness

A simple approach to diagnosing bad driver and application behavior is to examine the processor idle and C1 residency time and the interrupt-per-second count when the system is at or near idle. You can do this by using the Windows Performance Monitor tool (*perfmon.exe*), which is included in Windows. A walkthrough of this tool is given in the appendix.

## Preventing Problems

Identifying and determining the cause of some power management problems can require advanced analysis, but you might not have the time to perform such an analysis. Instead, this section gives some simple approaches that can be effective for finding and preventing problems.

### Removing Unnecessary Software

A pragmatic administrator can reduce inefficient behavior by removing all unnecessary server roles and applications from a system. Less unnecessary code executing on a system means more power efficiency.

### Turning Off or Unplugging Unnecessary Hardware

Removing devices from the system and turning off unnecessary chipset features in the BIOS mean that fewer drivers are loaded on the system. Repeating from the previous section, less code executing on a system means less potential for bad behavior.

### Using In-Box Drivers Where Possible

In-box drivers are part of Windows Server 2008 R2 and are tested to ensure compatibility with power management schemes. However, we do not necessarily verify third-party drivers, which can lack power management features or can generate excessive interrupt traffic.

A poorly written USB device driver is a frequent offender in this category. Some third-party USB peripheral and controller drivers can generate enough interrupt activity that processors cannot enter states C2 or lower, even when the system is otherwise completely idle.

Using in-box drivers where possible reduces the risk that a driver will compromise system power efficiency through lack of support or bad behavior.

# Increasing Data Center Utilization

Idle server power overhead is expensive. It requires significant power but gives your organization zero computational output in exchange. Reducing machine count and increasing deployed system utilization levels better amortize this cost and increase overall data center efficiency.

## Overprovisioning

Data centers are generally overprovisioned and underutilized. The most convincing explanation is the tendency for IT administrators to be risk averse. They might provision as much hardware as is necessary to meet service level agreements continuously during the day, even if this means most of the boxes are at low utilization or idle most of the time.

Another cause of overprovisioning is lack of monitoring. Almost one-third of the IT administrators surveyed for one Forrester report did not know their current systems’ utilizations.

## Monitoring

Monitoring system utilization is easy to do and can help you identify candidate areas for power savings. Even though it does not provide a comprehensive picture of overall system utilization, a good starting point is to measure CPU and disk load by using the Performance Monitor in Windows Server 2008 R2. For an example of CPU monitoring, see “[Appendix. Viewing PPM Counters by Using Perfmon.exe.](#_Appendix_CA._Viewing)”

## System Consolidation

If servers are poorly utilized, you can sometimes consolidate deployed systems onto fewer physical machines to eliminate the significant overhead of idle server power. This approach frees capacity, increases overall efficiency, and saves your organization money.

Consolidating deployed systems onto fewer physical machines is possible in several ways:

* Virtualization
* Multirole servers
* Dynamic provisioning

### Virtualization

Virtualization lets you consolidate the workloads of underutilized servers onto a smaller number of more heavily utilized servers. Fewer physical machines can lead to reduced costs through lower hardware and energy costs and reduced management overhead. Virtualizing workloads by using Hyper-V in Windows Server 2008 R2 is an excellent way to reduce the overall power footprint in your data center. For more information, see “Improve Energy Efficiency and Manage Power Consumption with Windows Server 2008 R2” on the WHDC website.

### Multiple Roles in Windows Server 2008 R2

The simplest ways to consolidate servers are to install multiple roles in Windows Server 2008 R2 or to add additional server application software to an existing machine.

The key to successfully using this strategy is to combine roles that have complementary workload resource requirements. For example, a payroll system that is used only at night and a business application that is used only during the day might peacefully coexist on a single machine. Combining low-usage or complementary applications on existing hardware can be a simple and effective approach to saving power.

### Dynamic Provisioning

A third solution to overprovisioning is to dynamically allocate systems as required. For applications that require many servers to meet peak demand, high-usage patterns might exist during business hours but systems remain near idle for the rest of the day. If your enterprise has a good understanding of load patterns, you can develop automation to shut down excess capacity during off-peak hours.

Windows Server 2008 R2 supports live migration of virtual machines with the Hyper-V role. By using live migration, you can easily consolidate workloads on underutilized servers and shut down excess capacity. If you are not using Hyper-V, you can still achieve the benefits of dynamic provisioning if your systems have specific, fixed usage periods. For example, on nightly enterprise backup or build servers, you can implement scripts to turn off the machines and turn them on at certain times.

## Virtualization and Power Management in Windows Server 2008 R2

When you install the Hyper-V role on Windows Server 2008 R2, the dynamics of power management change slightly.

### Guest Operating System Enlightenments

Windows Server 2008 R2 contains several optimizations, called enlightenments, that enhance power efficiency when they are deployed as a guest operating system. Deep C-state support and intelligent timer tick distribution are two such enlightenments that can improve the power efficiency of virtual machines (VMs) that run Windows Server 2008 R2 as compared to VMs that run other operating systems.

### Performance

Performance tuning can improve system responsiveness and power efficiency. Minimizing background work such as synthetic I/O and timer ticks for VMs reduces interrupt traffic and ensures that PPM effects are maximized. You should follow the performance tuning steps for virtualized systems in “Performance Tuning Guidelines for Windows Server 2008 R2” on the WHDC website.

# Hardware Component Effects on Power

Each hardware component in a server consumes power. You should understand the optimizations that are available within each component and select efficient components to maximize power efficiency at a system level.

## Processors

Modern processors include advanced power management features such as performance and idle states that operating systems can use to save power. Some processors take a different approach and use low operating voltages to increase power efficiency. Regardless of the specific implementation, choosing power-efficient processors for your data center is a good investment.

### Performance States

According to The Green Grid article “Five Ways to Reduce Data Center Power Consumption,” enabling processor performance states on server platforms can reduce power consumption up to 20 percent.

Processor performance states can vary in number and implementation for each processor generation. For example, Intel’s Xeon line of processors includes four P‑states on the Harpertown family versus six P-states on the Tigerton and nine P‑states on the Nehalem (see Table 2).

Table 2. P-State Support in Intel Processor Models

|  |  |  |
| --- | --- | --- |
| Processor family | Release date | P-state support |
| Harpertown | September 18, 2007 | P0 @ 3,165-MHz (100%); P1 @ 2,666-MHz (84%);  P2 @ 2,332-MHz ( 73%); P3 @ 1,999-MHz (63%) |
| Tigerton | September 5, 2007 | P0 @ 2,931-MHz (100%); P1 @ 2,665-MHz (90%);  P2 @ 2,398-MHz ( 81%); P3 @ 2,132,MHz (72%);  P4 @ 1,865-MHz ( 63%); P5 @ 1,599-MHz (54%) |
| Nehalem | March 30, 2009 | P0 @ 2,793-MHz (100%); P1 @ 2,660-MHz (95%);  P2 @ 2,527-MHz ( 90%); P3 @ 2,394-MHz (85%);  P4 @ 2,261-MHz ( 80%); P5 @ 2,128-MHz (76%);  P6 @ 1,995-MHz ( 71%); P7 @ 1,862-MHz (66%);  P8 @ 1,596-MHz ( 57%) |

These processors were chosen only to exemplify P-state differences between processor lines. More P‑states do not always guarantee increased efficiency, especially when you compare processors from two architectures or families. Benchmarks or manufacturer’s datasheets are usually necessary to determine this information.

### Idle States

Idle states present the best opportunity for overall power savings on a processor. The latest quad-core processors might see as much as a 60-W difference between C0 and Cn power. Again, availability is specific to processor families and architectures, so refer to datasheets or benchmark tests to determine coverage and efficiency.

### Low-Voltage Processors

A different variation of power-efficient processor also exists on the market today. These processors, known as “low voltage” or “ultra-low voltage,” are exactly what their name implies. Some of these processors are designed to be power efficient without using performance states. Again, you should review datasheets and use benchmarks to compare processors.

## Memory

Memory is becoming an increasingly important element of server power consumption, especially in servers that have fully populated memory banks. As the section “[Power Consumption of Idle Systems and Components](#_Power_Consumption_of)” demonstrated earlier in this paper, on a server in our lab, memory required 54 percent of total system power.

You must consider several parameters when you optimize memory purchases for power:

* Memory family (DDR, DDR2, or DDR3)
* Bus speed
* Memory capacity
* Chip density
* Additional features, such as buffering stages and reduced chip counts

### Memory Family

Generally, if all else is equal, newer RAM families consume less power than older families because of new features that promote power efficiency. If bus speed, capacity, and density are the same, a DDR2 RAM module—or physical stick of memory—should consume less power than a DDR1 module.[[8]](#endnote-8)

### Bus Speed

Bus clock frequencies generally double each generation. Even within the same generation, bus speed increases correspond to a significant increase in power consumption. Generally, doubling bus speed doubles the power consumption.[[9]](#endnote-9)

### Memory Capacity

RAM capacity is the “size” of a module (such as 512 MB or 2 GB). Generally, higher capacity RAM consumes more power.[[10]](#endnote-10)

### Chip Density

Dynamic RAM (DRAM) chip density refers to the amount of data that can be stored in each chip. Each memory module has multiple chips. Generally, the higher the chip density, the lower the overall power consumption. If overall capacity is constant, doubling the density of the memory chips reduces the number of chips on the module by half. This significantly reduces the power consumption.[[11]](#endnote-11)

Additional Features

Some memory modules have additional features such as buffering stages and reduced chip counts. These features have a significant effect on memory power consumption.

#### Fully Buffered DIMMS

Instead of writing to memory directly, fully buffered DIMMs (FBDIMMs) introduce an intermediate buffering stage between the memory controller and the memory modules. This buffering stage can improve reliability and throughput, but has an associated increase in latency, and the power consumption of such DIMMs is generally double that of a comparable DDR2 module.[[12]](#endnote-12) If you do not require the additional reliability or throughput, consider a platform that does not use FBDIMMs.

#### Low Chip Count and Low-Voltage Memory

Low chip count (LCC) and low-voltage (LV) memory are recent innovations for DDR2 and FBDIMM memory technologies that reduce the number of chips by half and reduce the operating voltage. According to manufacturers, these modules can reduce power consumption by 20 to 40 percent over standard modules.

## Storage

You can achieve significant power reductions by selecting power-efficient hard disks for your data center. Although the absolute power savings per disk may seem small, quantities of hard disks that are deployed in a data center can easily number in the thousands. At this scale, choosing power-efficient hard disks can save many kilowatts (kW) of power.

### Size Reduction

The simplest way to save power for storage is to migrate from 3.5-inch to 2.5-inch disk drives. This is one of the few power-saving changes that does not generally lead to a decrease in performance, and the investment cost can be repaid by the long-term savings that the new platform provides.

The reduction in platter size and weight means that the disk can use smaller actuators and more power-efficient motors, which results in a large power savings. In our labs, 2.5-inch drives consume 50 percent less power than 3.5-inch drives that have similar rotational speed.

In addition, the performance of 2.5-inch drives is generally equal to or better than that of 3.5-inch drives, with a few caveats for specific workloads and data layouts. The complicating factor is that the sheer area of a 3.5-inch drive allows for higher capacity per spindle, which 2.5-inch drives have not yet achieved. Currently, it may be unfeasibly expensive or impossible to use 2.5-inch drives to build low-power, high-capacity installations.

### RPM Reduction

Enterprise-class, 15,000-RPM disks offer the highest performance from rotational media disk drives, but they are not always necessary. In many scenarios, you can deploy lower RPM disk drives (such as 10,000 or even 7,200 RPM) in a storage array and still accomplish your business goals. If large sequential reads and writes are not part of your workload, this power optimization has a small effect on performance.[[13]](#endnote-13)

### Solid-State Disks

Solid state disks (SSDs) have no magnetic platters to rotate, eliminating power costs for disk motors and actuators—which are the most significant items on the power budget for rotational drives. Although client rotational drives spin their media down at idle periods to save power, enterprise drives do not. Therefore, SSDs generally use much less power in an enterprise server than hard disk drives (HDDs). However, SSD technology is changing rapidly. If you are considering SSDs for your IT environment, you must consider not only their power usage, but also their expected lifetime and how much your application workload can take advantage of solid-state storage technology.

### RAID Selection

In the past, power was not generally a factor in choosing a RAID configuration. However, power capacity can influence this decision. If your infrastructure is approaching its maximum power capacity, the power budget of large RAID arrays can restrict the expansion of existing RAID configurations or prevent the purchase of new configurations.

If replacing existing 3.5-inch disks with 2.5-inch disks of identical capacity is not possible and your organization is at its power capacity limit, one way to free some capacity is to change your RAID configuration. For example, you might move from a high degree of reliability and performance (RAID 10) to a lower degree (RAID 5). Of course, you should evaluate whether the performance, redundancy, and availability of different RAID setups can still achieve your business requirements. For detailed information on these tradeoffs, see “Disk Subsystem Performance Analysis for Windows” on the WHDC website.

## Network Adapters

To minimize network adapter power consumption, purchase only as much capacity as you need. If a server has low utilization or does not require a large amount of bandwidth, purchasing the highest throughput network adapters is unnecessary and consumes more power. Typical power consumption in our lab for a 1‑Gbps PCIe network adapter was 10 W, whereas a 10‑Gbps PCIe card consumed about 17 W.

Another thing to consider is the number of ports on the card. If four connections are required, a single four-port card consumes significantly less power than four individual single-port cards. Current quad-port 1‑Gbps cards consume around 17 W of power, whereas four 1‑Gbps single-port cards typically consume almost 40 W.

## Remote Power Strips

As demonstrated in “[Power Consumption of Idle Systems and Components](#_Power_Consumption_of)” earlier in this paper, servers that have been shut down can still consume 27 to 30 W of power. Physically unplugging the servers is one way to fix this problem, but you likely lack the resources to unplug and plug in servers each day.

Power strips that can be controlled remotely by networking or other means let you automate this process. Again, this introduces an up-front cost that can be recouped over time.

Table 3. Annual Remote Power Strip Energy Savings

|  |  |  |  |
| --- | --- | --- | --- |
| Strip size  (outlets) | Energy savings1  (W) | Annual energy savings2  (kWh) | Annual savings  ($) |
| 4 | 120 | 612.8 | 64.65 |
| 8 | 240 | 1225.7 | 129.31 |
| 16 | 480 | 2451.4 | 258.62 |

1Assumes 30-W savings per outlet.

2Assumes servers are powered off for 14 hours per day.

## Cooling

A commonly disregarded or forgotten element of server power consumption is on‑board fans. Most servers use multiple in-box fans to generate the required airflow.

Variable-speed fans let the platform reduce the fan RPM rate when the server is not under peak load. The paper “Data center TCO benefits of reduced system airflow,” which was submitted to the ITHERM conference, found that fans in some 1U rack-mounted servers consume 15 to 20 percent of overall system power. This can translate to a significant power saving during off-peak periods in the data center. Variable-speed fans typically require additional support from the platform, so this might not be a simple power savings option for existing installations.

## Power Supply Units

Power supply units (PSUs) perform an AC/DC conversion. This process has built-in inefficiencies, so much so that PSUs have received increased attention from industry and government.

### Power Supply Unit Efficiency

PSUs have two key measures of efficiency:

* Power factor
* Ratio of input/output power

Power factor measures how in-phase the input voltage and current waveforms are. To reduce inefficiency, ensure that your power supplies use *active* power factor correction (PFC).

The ratio of input-to-output power is the key determinant of power supply quality. Power supplies must be less than 100-percent efficient in this category because of the conversion from AC to DC power. With the recent change in power trends, ENERGY STAR standards, and the institution of the 80 PLUS program, new designs are becoming available that raise the overall efficiency significantly.

Whereas default supplies were typically about 70‑percent efficient, *The Register* reports that new models from vendors such as Dell, SGI, and Rackable are over 86‑percent efficient across the entire load spectrum. This translates to a significant percentage reduction in power waste and excess heat in your data center—a real candidate for cost savings.

Consider Table 4, which details the effects of power supply efficiency on a 4‑U machine that requires 500 W of power.

Table 4. Effects of Power-Supply Efficiency on 4‑U Server Consuming 500 W of   
 Power 24x7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Efficiency | Output power (W) | Required input power (W) | Waste power (W) | Annual waste power cost ($) |
| 70 (default) | 500 | 714 | 214 | 197.02 |
| 80 (near 80 plus bronze) | 500 | 625 | 125 | 115.52 |
| 85 (80 plus silver) | 500 | 588 | 88 | 81.33 |
| 90 (above 80 plus gold) | 500 | 555 | 55 | 50.83 |

On a 70‑percent efficient power supply, 714 W of wall power is required to supply 500 W to the system. The additional 214 W is waste power, much of which is converted into heat and requires additional costs for cooling and airflow infrastructure. A 90‑percent efficient supply requires only 555 W, a saving of 159 W for one machine. If this is a 24x7 server, you will realize annual savings of over $146 on your power bill. If cooling costs are added, this might be as high as $200 or more. For a farm of 1,000 such servers, efficient power supplies could save your organization $200,000 a year.

### Efficiency Programs

Government programs such as ENERGY STAR in the United States and industry certification programs such as 80 PLUS are working to test and ensure the efficiency of power supplies and server systems. These organizations publish results that might help you find the most ideal power supply for your particular configuration.

You can find more information on these programs in “[Resources](#_Resources).”

# Data Center Infrastructure

Data center real estate, high-capacity power and HVAC equipment, network routing and UPS systems, maintenance for all these items, and the power that is required to operate them can easily equal or exceed the costs of computing equipment and computing equipment power year over year.

You can directly reduce these costs by using innovative management technologies and intelligent planning. One such tool is the Microsoft Assessment and Planning (MAP) Toolkit. For more information, see “[Resources](#_Resources).”

External resources also exist. Many companies specialize in planning for infrastructure efficiency, and we are a participating member of the Green Grid consortium, a group that is dedicated to improving energy efficiency in data centers.

For links to Web sites that can provide more information, see “[Resources](#_Resources).”

# Resources

#### Microsoft documentation and white papers

Disk Subsystem Performance Analysis for Windows

<http://www.microsoft.com/whdc/archive/subsys_perf.mspx>

Interrupt-Affinity Policy Tool

<http://www.microsoft.com/whdc/system/sysperf/intpolicy.mspx>

Performance Tuning Guidelines for Windows Server 2008 R2

[http://www.microsoft.com/whdc/system/sysperf/Perf\_tun\_srv-R2.mspx](http://www.microsoft.com/whdc/system/sysperf/Perf_tun_srv.mspx)

Power Metering and Budgeting

<http://msdn.microsoft.com/en-us/library/ff543910(v=VS.85).aspx>

Power Policy Configuration and Deployment in Windows

<http://www.microsoft.com/whdc/system/pnppwr/powermgmt/PMpolicy_Windows.mspx>

Processor Power Management in Windows Vista and Windows Server 2008

<http://www.microsoft.com/whdc/system/pnppwr/powermgmt/ProcPowerMgmt.mspx>

Recommendations for Power Budgeting with Windows Server

<http://www.microsoft.com/whdc/system/pnppwr/powermgmt/Svr_PowerBudget.mspx>

Using PowerCfg to Evaluate System Energy Efficiency

<http://www.microsoft.com/whdc/system/pnppwr/powermgmt/PowerCfg.mspx>

Windows Server 2008 Power Savings

<http://www.microsoft.com/downloads/details.aspx?FamilyID=61d493fd-855d-4719-8662-3a40ba3a0a5c&displaylang=en>

#### Microsoft Tools and Web Sites

Improve Energy Efficiency and Manage Power Consumption with Windows Server 2008 R2

<http://download.microsoft.com/download/5/B/D/5BD5C253-4259-428B-A3E4-1F9C3D803074/WS08%20R2%20Power%20Savings%20White%20PaperTDM.docx>

Microsoft Environmental Sustainability web page

<http://www.microsoft.com/environment>

Microsoft Assessment and Planning Toolkit

<http://www.microsoft.com/downloads/details.aspx?FamilyID=67240b76-3148-4e49-943d-4d9ea7f77730&DisplayLang=en>

*TechNet Magazine*, “Sustainable Computing: Is It Time to Turn Off Your Servers?” by Dave Ohara

<http://technet.microsoft.com/en-us/magazine/cc700341.aspx>

Windows Server Catalog, servers meeting Additional Qualification for Enhanced Power Management

<http://www.windowsservercatalog.com/results.aspx?bCatID=1333&cpID=0&avc=10&OR=1>

*Windows Server Performance Team Blog*, “Configuring Windows Server 2008 Power Parameters for Increased Power Efficiency”

<http://blogs.technet.com/b/winserverperformance/archive/2008/12/04/configuring-windows-server-2008-power-parameters-for-increased-power-efficiency.aspx>

#### United States Government

Energy Information Administration, “Average Retail Price of Electricity to Ultimate Customers by End-Use Sector, by State”

<http://www.eia.doe.gov/cneaf/electricity/epm/table5_6_a.html>

Environmental Protection Agency (EPA), “Report to Congress on Server and Data Center Energy Efficiency,” August 2, 2007

<http://www.energystar.gov/ia/partners/prod_development/downloads/EPA_Report_Exec_Summary_Final.pdf>

EPA, ENERGY STAR® Program Requirements for Computer Servers

<http://www.energystar.gov/ia/partners/product_specs/program_reqs/computer_server_prog_req.pdf>

#### Organizations

80 PLUS Program

<http://www.80plus.org>

ACPI Specification, Revision 3.0b

<http://www.acpi.info/spec30b.htm>

Applied Power Electronic Conference (APEC)  
Details on power supply efficiency testing, metrics, power supply design standards and proposed ENERGY STAR requirements

<http://www.apec-conf.org/2006/APEC_2006_SP4_2.pdf>

Forrester research, “Firms Still Struggle To Predict Capacity Utilization Accurately,” by Laura Koetzle, 17 May 2006

[http://apps.bmc.com/USA/Corporate/attachments/Forrester\_Capacity\_Utilization\_Survey\_May.pdf](http://apps.bmc.com/USA/Corporate/attachments/Forrester_Capacity_Utilization_Survey_May.pdf%20)

Standard Performance Evaluation Corporation, “SPECpower\_ssj2008”

<http://www.spec.org/power_ssj2008/>

The Green Grid, “Five Ways to Reduce Data Center Power Consumption,” by Mark Blackburn, 2008

http://www.thegreengrid.org/en/Global/Content/white-papers/Five-Ways-to-Save-Power

Transaction Processing Performance Council, “TPC Benchmark E (TPC-E)”

<http://www.tpc.org/tpce/default.asp>

#### Articles and Papers

“Data center TCO benefits of reduced system airflow,” by C. G. Malone, W. Vinson, and C. E. Bash, *Thermal and Thermomechanical Phenomena in Electronic Systems*. ITHERM 2008 11th Intersociety Conference, 28‑31 May, 2008, pp. 1199-1202

<http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4544397&isnumber=4544243>

*Electronics Cooling*, “In the data center, power and cooling costs more than the IT equipment it supports,” by Christian Belady, February 2007

<http://electronics-cooling.com/articles/2007/feb/a3/>

*The Register*, “Dell develops ultra-efficient server power supplies,” by Austin Modine

<http://www.theregister.co.uk/2008/06/25/dell_80_plus_gold_power_supply/>

# Appendix. Viewing PPM Counters by Using Perfmon.exe

This walkthrough guides you through the process of adding PPM-related performance counters to the Performance Monitor tool (sometimes called Perfmon) in Windows Server 2008 R2. It also demonstrates how the counters change under different machine loads and power policies. You can use this information to identify and fix issues that affect power management on your servers.

First, open the Performance Monitor tool by opening a new command windows and typing “perfmon”. Under **Monitoring Tools**, double-click **Performance Monitor**.

Performance Monitor initially shows a graph of % Processor Time. Clear this counter by right-clicking the graph window and selecting **Remove All Counters.**

Next, right-click the graph window and select **Add Counters**. After the counter sets finish loading, locate the **Processor Information** performance counter group in the **Available Counters** frame and expand the group by clicking the arrow to the right of the group name.

Next, add the following counters to the performance monitor. You can press and hold the CTRL key to select multiple counters at a time:

* % C1 Time
* % C2 Time
* % C3 Time
* % Idle time
* % of Maximum Frequency
* Interrupts / Sec
* Processor Frequency

After you have selected these counters, a set of instances appear in the lower-left pane that are labeled **Instances of selected object**. Each logical processor generally counts as one instance for processor performance counters, as well as an overall **Total** instance. For each counter, select the **All Instances** option, and then click the **Add** button below the pane to add the counters to Performance Monitor.

When you are finished, your screen should look similar to Figure A-1.

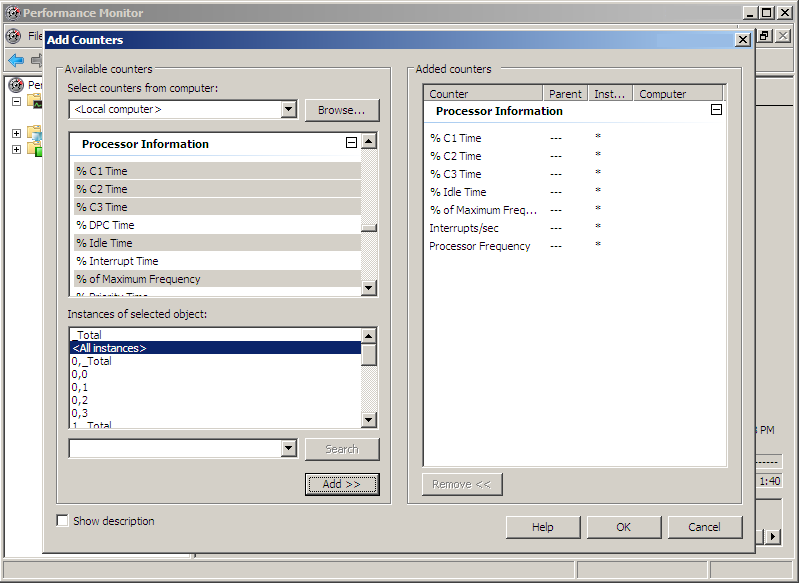


Figure A-1. Adding selected instances of the Processor Information performance   
 counters

To return to the performance monitor graph with your new counters loaded, click **OK**.

The next step is to configure this data to a more readable format. You can do this in three ways:

* Click the **Change Graph Type** button two times.

-or-

* Use the drop-down arrow to the right of the **Change Graph Type** button, and then select **Report** (see Figure A-2).

-or-

* Press Ctrl-G two times to change the graph format to **Report**.

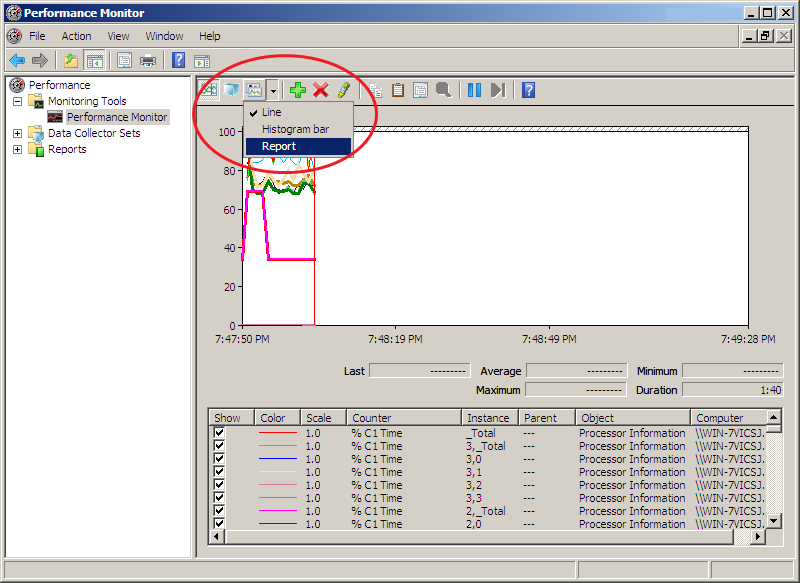


Figure A-2. Changing the performance monitor to Report mode

In **Report** format, you now see easy-to-read tables of data for your system. First, we will review what each row and column of the table means. Each row corresponds to a particular performance counter, which is given in the left-most column.

The columns correspond to individual logical processors, processor packages, or totals. In this example, the machine that was used to generate these screenshots has 4 processor sockets. Each processor socket has 4 processor cores, for a total of 16 logical processors.

The naming convention for these processors in Performance Monitor is <*socket*>,<*core*>. As shown in the rightmost two columns of Figure A-3, the column titled 0,0 contains the counter data for socket 0, core 0, whereas 0,1 contains the counter data for socket 0, core 1. This convention continues for a 16-logical-processor configuration up to socket 3, core 3.

Performance Monitor also displays special *Total* instances for each socket and for all logical processors overall. The 0,\_Total column represents the counter totals for all the cores on socket 0, whereas 3,\_Total represents the counter totals for the cores on socket 3. The second column in Figure A-3, which is labeled “\_Total”, displays the counter totals for all 16 cores on the server.

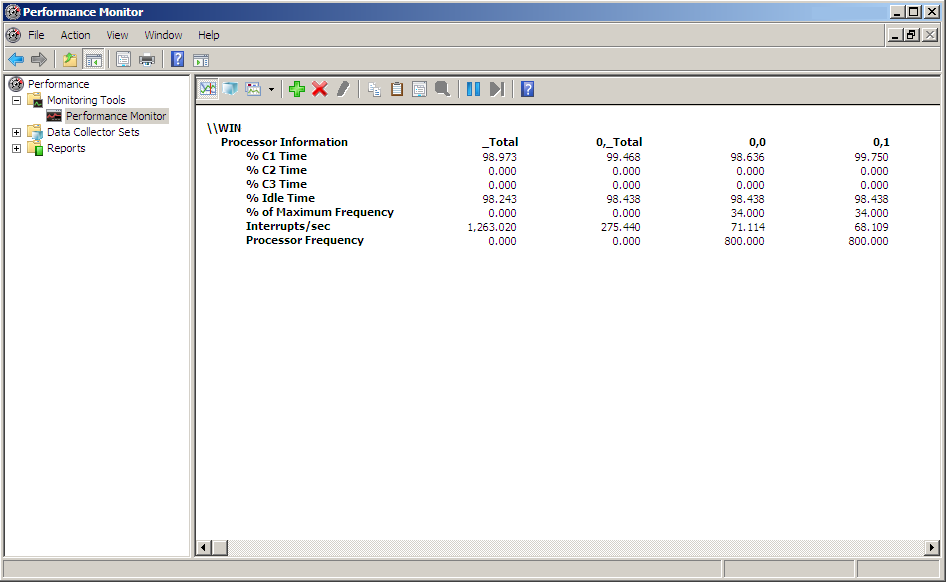


Figure A-3. Processor Information counters active on a 16-processor system in   
 Balanced mode

The \_Total column is an average (for the percentage-based counters and Processor Frequency counter) or a sum (for the Interrupts/Sec counter) of the values across all processors in the system. Each column to the right of the \_Total column that does not contain “\_Total” in the header represents a logical processor, so the \_Total column is an average or a sum of all these columns.

A simple example of this behavior can be seen in the Interrupts/sec counter. The \_total interrupts per second counter counts all interrupts on the system, whereas the individual processor columns show the number of interrupts per second that occurred on each logical processor.

Performance Monitor counters fluctuate with the load on your system. As load increases, you should see drastically different values in some of these counters. These counters also indicate whether multimedia timers are enabled.

For example, examine Figure A-3, which shows the counters for a 16-core server at idle with the Balanced power policy selected and no multimedia timer. The system is spending 98 percent of its time at idle and approximately 99 percent of the time in idle state C1. Cores 0,0 and 0,1 are at 34 percent of their maximum frequency, a mere 800 hertz (Hz). Interrupt count is low—approximately 300 per socket, 75 per processor, for a total of 1,263 per second.

Now compare that to the same system with the High Performance power policy and a 1‑ms multimedia timer enabled, as shown in Figure A-4. No work is actually being done by the system, and the % Idle Time and % C1 Time counters are still at 99 percent. Yet the % of maximum frequency for Cores 0,0 and 0,1 jumped to 100 percent, and the frequency increased to 2,300 MHz. Interrupt counts have skyrocketed to over 1,000 per processor, for a total of 16,555 per second across the system.

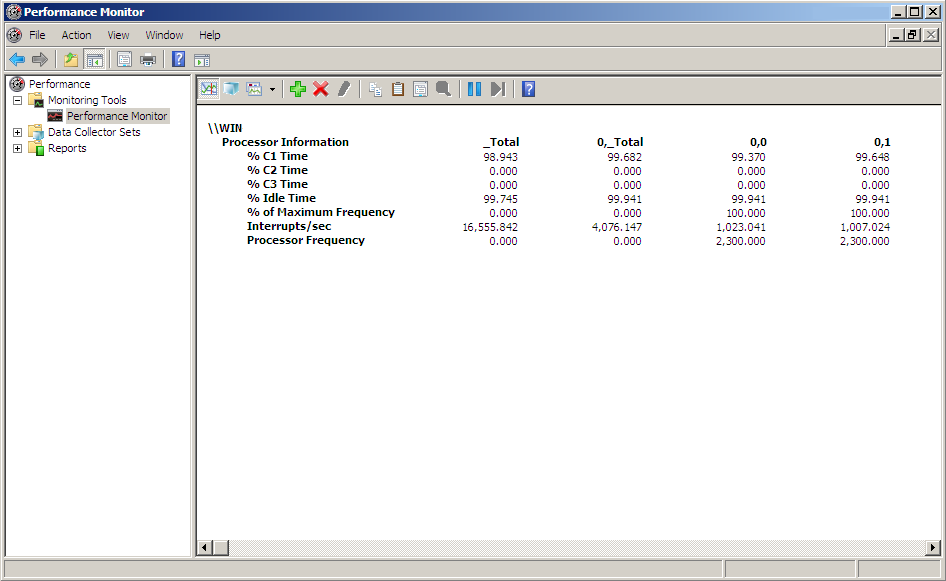


Figure A-4. Counter values with High Performance power policy and a   
 multimedia timer enabled

A system with a 15.68-ms timer resolution has a minimum 63 interrupts per second per processor. Expect a system at or near idle to have more than this amount, generally somewhere in the range of 150 to 400 per second per processor. Although your results will vary, the key point is that at idle, interrupt counts of over 1,000 interrupts per second per processor are typically generated by a 1‑ms application timer or a misbehaving driver. Remember that systems under moderate or heavy load can easily generate thousands of interrupts per second per processor.

We reiterate that these changes in Interrupts/sec and % of Maximum Frequency were not caused by any change in workload. In both scenarios, these servers were almost completely idle. This highlights the importance of correct configuration and system monitoring to ensure that your servers are achieving maximum efficiency.

The final scenario demonstrates how the counters change when workloads are introduced onto the system. We ran a small application that allocated memory from the system in a tight loop and captured the performance counter changes in Figure A‑5.

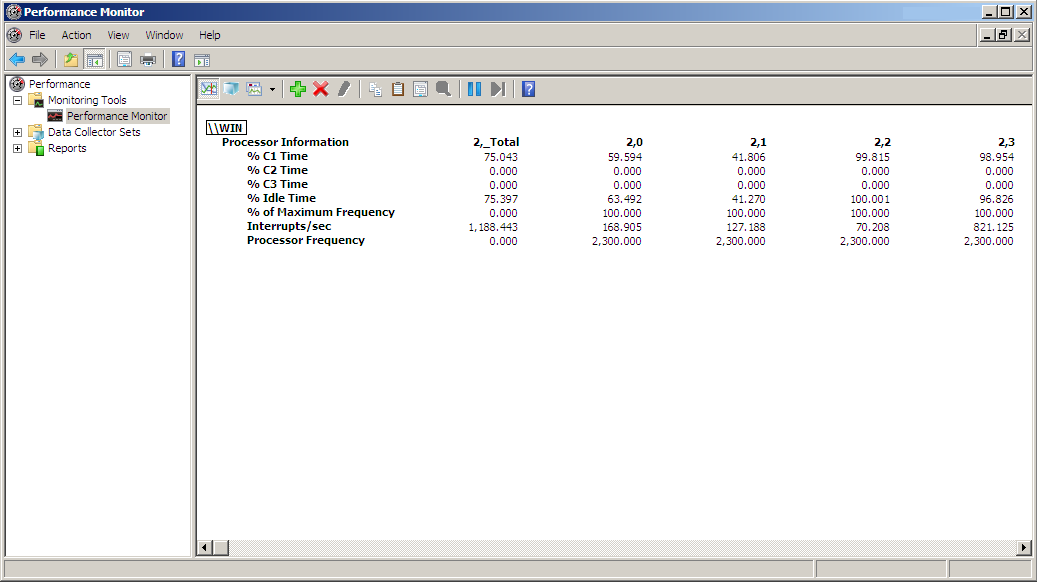


Figure A-5. Counter values during a memory allocation workload

The counter data for the socket (column 2,\_Total) shows that the %C1 Time and %Idle Time counters have reduced significantly under this workload from 99 to 75 percent. The PPM engine also raised the processor frequency to 100 percent to handle the load. The number of interrupts per second on the socket increased from their idle value of 300 to over 1,000, but this is still far from the 4,000 interrupts per socket that were caused by the 1‑ms timer.

You may notice that the counters in Figure A-5 were not from socket 0, but instead from socket 2. You might also notice that cores 2,2 and 2,3 have a much higher idle time than cores 2,0 and 2,1. This demonstrates how workloads are not always evenly distributed among all logical processors on a system. Often, one processor in a socket is utilized while others are almost completely idle. It is best to use the \_Total column to obtain an accurate picture of your overall server CPU utilization.

# End Notes

1. The first transistor circuits were built by using passive components that consumed power while the device was inactive. The complementary metal-oxide semiconductor (CMOS) transistor eliminated this static power consumption from digital circuits many years ago, which reduced transistor power consumption to that required for switching events. As transistor size passed the 180‑nanometer (nm) mark, leakage currents reintroduced static power consumption. The result has been a substantial year-to-year increase in power consumption and density. [↑](#endnote-ref-1)
2. For example, if P0 is 2,500 MHz at 1.8 volts, P1 could be 2,200 MHz at 1.8 volts, 2,500 MHz at 1.75 volts, or 2,200 MHz at 1.75 volts. All of these are valid performance reductions. Frequency is exposed to the operating system through a performance counter, so it is relatively easy to determine whether a reduction in performance state affects frequency or voltage. [↑](#endnote-ref-2)
3. The operating system can do this effectively by using the historic CPU utilization. The operating system uses a predictive model that is based on historic utilization data to calculate the expected throughput that is necessary for the next time interval and chooses to increase, decrease, or maintain P-state or possibly enter an idle state accordingly. [↑](#endnote-ref-3)
4. The combinatorial complexity of the problem becomes apparent after a bit of thought. If you have four processors that can do 400 cycles of possible work each over a specified unit of time at full throughput, which P-state and C-state combinations lead to the most power efficient setup for calculating 200 operations? 400 operations? 1000? Also consider that servers do not know exactly how much work they must perform in the next time quantum. The algorithm must consider historic workload data and make an accurate prediction. [↑](#endnote-ref-4)
5. Instruction and data caches are not essential for correct execution, so the data in them can be discarded without affecting functionality. However, these caches improve performance significantly, so the system suffers a large performance loss if they must be refilled during execution. An alternative option is to write the cache data to the next highest level of cache memory, shut down, and read the data back in when the system resumes from the idle state. Of course, this significantly increases the latency of a state transition. [↑](#endnote-ref-5)
6. The rationale for this is as follows: If the processor enters a low idle state and is called back to C0 immediately, this causes a net efficiency loss for the platform. However, if the core enters C1 and remains there for tens of milliseconds, the power savings can outweigh any performance loss, which results in a net efficiency gain. [↑](#endnote-ref-6)
7. Some drivers and devices employ a technique called *polling* to carry out I/O operations. Rather than stop execution and let the processor enter an idle state, drivers that poll use a tight loop to repeatedly check if I/O has completed. Although this does not require a later interrupt that can potentially wake a processor from an idle state, polling requires the processor to remain active and consume power during the entire operation. The best approach to use depends on the latency of the I/O request and can employ both strategies. [↑](#endnote-ref-7)
8. Original DDR ran at a nominal voltage of 2.5 volts. DDR2 lowered this to 1.8 volts, and DDR3 specifies a nominal voltage of only 1.5 volts. Although the voltage drop alone saves power, each generation of DDR RAM also includes power management features that previous generations lacked. DDR2 introduced self-refresh states, lower activate and standby power consumption, and 4-bit prefetching, which allowed it to save 65 percent on power consumption during its highest active operating condition. DDR3 introduces 8-bit prefetching and dual-gate transistors to lower leakage current. [↑](#endnote-ref-8)
9. For example, according to our test results, a 2‑GB DDR2 stick of PC2-4200 RAM that has a bus speed of 533 MHz consumes 12.3 W of power. A 2‑GB DDR2 stick of PC2-8500 RAM (1,066-MHz bus speed) that has the same density consumes 23.8 W, which is almost double the amount. [↑](#endnote-ref-9)
10. This is true if all other parameters are held equal. Doubling the capacity at the same density requires doubling the number of chips on the module and approximately doubles the power consumption. However, increases in capacity can be offset by an accompanying increase in chip density, and in some cases, this can actually result in power savings. For example, our test results show a 1‑GB stick of PC2-8300 memory that has a chip density of 512 Mbit consumes 12.1 W of power whereas a 2‑GB stick of the same RAM speed that has a 1‑Gbit chip density consumes 11.0 W. [↑](#endnote-ref-10)
11. For example, a 2‑GB DIMM with a chip density of 512 Mbit requires 36 memory chips. According to our test results, a module that has a bus speed of 1,066 MHz consumes 23.8 W. A module with 1‑Gbit chip density requires 18 chips and consumes only 10.9 W [↑](#endnote-ref-11)
12. The buffering stage in an FBDIMM can act as signal conditioning and error correction logic, which increases overall reliability. This stage also lets a memory controller issue reads and writes in parallel, which increases memory bandwidth. However, instead of directly reading from or writing to memory, the memory controller writes to the buffer, which then writes to memory, which adds an extra amount of latency to each request. [↑](#endnote-ref-12)
13. In random access scenarios, our tests have shown that the additional rotational latency from stepping down to a lower RPM class (for example, from 15,000 to 10,000 or from 10,000 to 7,200) can be as low as 1 ms. [↑](#endnote-ref-13)