Recommendations for Power Budgeting with Windows Server

July 10, 2008

Abstract

Inefficient allocation of power and cooling resources in data centers can result in unused power capacity. This increases costs and artificially limits the number of servers that can be deployed. To address the overallocation of power budgets, many server vendors have introduced power management solutions that try to enable the capping of power consumption on a per-machine basis to reduce the unused capacity. However, some of these solutions can have negative or unintended consequences when they overlap, conflict, or otherwise interfere with the power management capabilities that are supported in Windows Server® operating systems.

This paper summarizes common approaches to power budgeting in the server marketplace and recommends best practices to ensure that these solutions work as intended and interoperate with Windows Server power management capabilities. The information in this paper is intended for product planners, developers, and system designers.

This information applies for the following operating systems:  
 Windows Server 2008  
 Windows Server 2003

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

For the latest information, see:   
 <http://www.microsoft.com/whdc/system/pnppwr/powermgmt/svr_powerbudget.mspx>

Disclaimer: This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

© 2008 Microsoft Corporation. All rights reserved.

Microsoft, Win32, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Document History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Change |  |  |  |
| July 10, 2008 | First publication | | | |

Contents

[Introduction 3](#_Toc203464875)

[Limitations of Current Solutions 3](#_Toc203464876)

[Throttling Mechanisms 4](#_Toc203464877)

[Adjusting Windows Power Policy 6](#_Toc203464878)

[Using ACPI Notifications 6](#_Toc203464879)

[Using CPU Platform Thermal Control Circuits 7](#_Toc203464880)

[Combining Power Throttling Mechanisms 7](#_Toc203464881)

[Windows Processor Throttling Policies 8](#_Toc203464882)

[Dynamic Processor Throttling Policies on Windows Server 2003 8](#_Toc203464883)

[Processor Performance States on Windows Server 2008 9](#_Toc203464884)

[Best Practices and Recommendations 12](#_Toc203464885)

[Resources 13](#_Toc203464886)

# Introduction

Server hardware and other information technology (IT) equipment in the data center consume power and cooling resources. To ensure the uninterrupted operation of the servers and the data center itself, data center managers must allocate these resources in a very predictable manner. To estimate the power and cooling requirements for their systems, many IT organizations rely on the plate rating, some form of estimation, or lab measurements (usually in a test or benchmark environment). They then use this value to allocate—or *budget*—power to equipment within the data center.

Although this approach appears straightforward, actual server power consumption typically varies significantly from these estimates. The actual power draw of a server is often much lower than the budgeted amount. According to a study from REPP, many servers consume power at a level between 25 and 30 percent of their plate rating[[1]](#footnote-2). System usage—including CPU, I/O, and other hardware—in the data center also significantly affects power consumption. IDC estimates that current system usage averages between 10 and 20 percent[[2]](#footnote-3) and that, at that usage level, the nominal system power draw is often measurably lower than the maximum or budgeted amount. The difference between the actual power usage and the budgeted usage is called *stranded capacity*.

Stranded capacity increases operational costs and limits how much IT equipment can be deployed into the data center. Instead, to accommodate additional servers, data center managers are often forced to either install additional infrastructure or build new facilities, both of which come at a significant cost. Frequently this is done even though sufficient capacity is present in the existing facility.

Enabling data centers to better use the capacity that they already have by helping them to more accurately quantify the needs of their servers provides significant customer value. Again, this must be done in a predictable manner. Data center managers must ensure the availability of the data center, so they must allocate power and cooling conservatively. They require a guarantee that all the equipment that is provisioned in the data center can correctly operate within the budget without adversely affecting data center uptime.

# Limitations of Current Solutions

To address these requirements, many server vendors have developed power management solutions that try to ensure that servers operate within a fixed power budget that the IT administrator chooses. After configuration, the platform ensures that the power consumption of the server does not exceed the allocated budget. The primary power management control that most solutions use today is to restrict the speed of the processor. This restriction limits the processor power consumption and keeps the machine below the allocated budget.

Today, these solutions are typically controlled by a Baseboard Management Controller (BMC) on the server. That is, they operate out-of-band (OOB) from the operating system that runs on the server. Frequently, the solutions disable or override processor speed and thermal controls that the operating system uses. Therefore, conflicts with operating system power management can occur and can adversely affect system management, performance, and reliability. Such conflicts can occur with Windows Server®, which actively manages these same hardware controls. Conflicts here have a direct and adverse effect on customers.

Platforms that manage processor power management controls without the knowledge of the operating system can cause negative side-effects. The following are some examples:

* Windows Server power management features can be disabled, which adversely affects the power efficiency of the server.
* The actual processor speed may differ from that reported by Windows®.
* The information that Windows uses in certain calculations may be invalid, which leads to incorrect decisions. In particular, this can result in the system making unsuitable processor performance state (P‑state) transitions.
* Timing-related problems can occur during state transitions. Many power management controls are associated with timing components such as transition delays. If the platform adjusts the controls without notifying Windows, conflicts can occur when both the platform and the system try to adjust operation based on different assumptions about timing.
* Accurate information on the state of the system is not visible to the end user or to applications. Therefore, diagnostics and troubleshooting efforts can report misleading or incorrect results. Additionally, workload management tools cannot determine the actual state of the machine, so they cannot accurately adjust the workload.

# Throttling Mechanisms

To avoid wasting infrastructure capacity, power budgets for servers in a data center must be accurate. At the same time, however, care must be taken to prevent servers from exceeding their allocated budget. These two goals are somewhat at odds with each other because it is difficult to optimize for both requirements.

Today third-party power budgeting solutions address these goals by enabling customers to set a power budget on each server. The hardware platform is responsible for ensuring that power consumption never exceeds the allocated budget. Server vendors typically implement such solutions in the BMC, which communicates directly to a management console by using proprietary mechanisms.

Appropriate Windows power policy settings can often keep a server running within its budget. As power consumption becomes very close to the budgeted limit, however, active management of the power controls might be required to prevent exceeding the power budget. This paper refers to such management as ”throttling.” Throttling a server’s power usage ensures that the budget will not be exceeded.

Although the throttle helps enforce the budget, in a typical server deployment, engagement of throttling should be the exception instead of the rule. Throttling effectively limits the server’s performance capability, and a server that operates in a throttled state for extended periods of time may indicate a poorly configured power budget. Management tools can detect and report this condition so that the administrator can adjust the budget. However, if performance is secondary, an administrator might permanently limit the performance of a machine in order to stay below a strict budget.

We recommend three mechanisms for throttling power usage on servers that are running Windows Server 2003 or Windows Server 2008. In order of preference, these mechanisms are as follows:

1. Adjusting power policy.

2. Using ACPI notifications.

3. Using CPU platform thermal control circuits.

Figure 1 shows an overview of these three approaches.



Figure 1. Power Throttling Mechanisms

Adjusting Windows power policy and using ACPI notification are appropriate for most scenarios, because a small amount of latency is tolerable. However, in extreme cases in which the power demands increase quickly, these mechanisms alone might not be adequate. If the system must react quickly to reduce power consumption to avoid exceeding the power budget, the use of CPU platform thermal controls might be required.

## Adjusting Windows Power Policy

The preferred mechanism for controlling power usage is to appropriately adjust Windows power policy. As power consumption of the system must be changed, the value of the maximum processor performance state can be adjusted up or down accordingly. By adjusting the power policy settings for the processor, this mechanism can effectively manipulate processor performance (P‑states) and throttle (T‑states) states.

In a typical configuration, a vendor-written or data center–specific software agent monitors the power consumption through a proprietary mechanism and adjusts policy accordingly. In this approach, the Windows power manager optimizes system operation for maximum performance while staying within the bounds of the specified power policy. This mechanism requires code to be running on Windows. It has the advantage of working with Windows power policy to maintain the budget for the system and can be used for any component that integrates with power policy.

## Using ACPI Notifications

The second mechanism to throttle the system is to use ACPI notifications on the processor object to adjust the processor performance states and throttle states that are available for use. In this case, the platform monitors system power consumption and notifies Windows through standard ACPI mechanisms when changes are required.

When consumption approaches the allocated budget, the platform takes the following steps:

1. The platform adjusts the Performance Present Capabilities (\_PPC) and/or Throttling Present Capabilities (\_TPC) objects, respectively, to limit the P‑states and T‑states that are available to the operating system.

2. The platform sends an ACPI Notify on the processor object to signal the operating system that these values have changed.

Windows reevaluates the P‑state and T‑state usage according to the new capabilities. When power consumption falls below the budget, the platform can use the same mechanism to readjust P‑states and T‑states.

Because of this mechanism, the operating system reevaluates the ACPI objects. Such reevaluation can affect system performance, so throttling by using ACPI notifications should not be used frequently. Generally, these objects should not be adjusted more than two times each second. With this mechanism, Windows can still efficiently manage power to the requirements of the workload while keeping power usage below the budget that was allocated for the server. However, note that Windows cannot determine why the performance was limited; it is only able to report that the platform capabilities have changed. In Windows Server 2008, the kernel power manager logs each change in system capabilities in the System Event Log to help with diagnosis of any performance issues that might occur because of too much throttling.

For more information about Windows support for ACPI objects, see “Processor Power Management in Windows Vista and Windows Server 2008.”

## Using CPU Platform Thermal Control Circuits

If the system must react immediately to reduce power consumption to avoid exceeding the power budget, adjusting Windows power policy and using ACPI notification might incur too much latency. In the third approach, the BMC might directly access the CPU’s platform thermal control circuits to throttle the machine.

The invocation of these controls is undetected and unsynchronized with operating system power management functionality in Windows Server 2003 and Windows Server 2008. Therefore, Windows does not consider the state of the processor and thermal controls as it tries to efficiently manage the system’s power management controls. The use of platform thermal controls adversely affects performance. Therefore, it should be considered a temporary solution to limit the system power budget until one of the other approaches can take effect.

Frequently, the use of platform thermal controls can be limited by using a guard band—a threshold set below the actual budget to provide a safety margin. When power consumption approaches the guard band, the first two mechanisms can be used first. If these do not achieve the required effect in sufficient time and consumption nears the budget, platform thermal controls can be engaged until the other mechanisms that are described in this section take effect.

# Combining Power Throttling Mechanisms

We recommend that vendors configure systems that combine these three throttling mechanisms. The goal is to build solutions that interoperate cleanly with Windows Server and avoid an effect on performance and reliability but remain within the assigned power budget.

Under ordinary operating circumstances, Windows power policy is in effect to manage the power consumption of the system efficiently to the workload. By default, Windows Server 2008 dynamically manages processor performance states according to the system power policy. This is an option in Windows Server 2003, although it is not enabled by default. Power policy can be adjusted to limit the power consumption of the system and therefore its performance.

When practical, Windows power policy should be used to control power consumption, especially in situations that require a static power setting. For example, if a server requires a low-power mode that limits the processor to the lowest performance state, the preferred approach is to adjust the power policy settings. For example:

* In Windows Server 2003, set the dynamic processor throttling policy to Degrade.
* In Windows Server 2008, set the Maximum performance state to 50 percent or the lowest applicable percentage.

The use of Windows power policy to control power consumption enables a proactive approach to workload management. Adjusting power policy affects how Windows performs work and thereby affects how power is consumed. Frequently, this approach alone can prevent the BMC from having to engage hardware controls. Windows power policy is visible to applications, so workload management systems can consider this when they determine the placement of workloads on servers and so on. This helps avoid conflicts between the workload that is assigned to a machine and that machine’s power budget.

Another advantage is that Windows power policy is dynamically controllable at runtime in any of several ways:

* By using Win32® APIs on the local machine.
* By using Windows Management Instrumentation (WMI) scripting from a remote machine.
* By using Group Policy on a network.

Dynamic control enables the centralized management of policy across a wide number of machines. If a local server management agent is running on the host, power policy can be integrated with other management points. This enables both third-party applications and hardware to cooperate with Windows to manage power.

When power policy alone cannot manage power within the power budget, the platform can use the ACPI notification mechanism to adjust the P‑states and/or T‑states that are available to Windows. In extreme cases that require near real-time latency, the platform can use platform thermal controls while it waits for the ACPI notification mechanism to take effect. Platform thermal controls should not be used for extended periods. For further details on your processor’s thermal constraints, consult your CPU vendor.

To provide reliable and dependable operation within the assigned budget, the platform must provide a hardware failsafe mechanism. If Windows does not respond quickly enough—or does not respond at all—the platform must take action to enforce the budget, such as using platform thermal controls. Hardware and platform-specific measures also might apply if an operating system crash occurs or if the operating system is not present, such as before boot, during provisioning or during deployment. Generally, a hardware failsafe mechanism should be used only until Windows can take action.

# Windows Processor Throttling Policies

*Processor throttling* refers to the operating system–directed use of processor performance states. Processor power policy has evolved from Windows Server 2003 to Windows Server 2008. This section briefly describes the processor throttling policies that are implemented in these systems.

## Dynamic Processor Throttling Policies on Windows Server 2003

Windows Server 2003 uses four *dynamic processor throttling policies* that determine how the Windows power manager chooses a processor performance state. These policies are shown in Table 1. On hardware platforms that do not support processor performance states, these policies have no effect.

Table 1. Windows Dynamic Processor Throttling Policies

|  |  |
| --- | --- |
| Policy | Description |
| None | Always run at the highest performance state. |
| Constant | Always run at the lowest performance state. |
| Adaptive | Run at a performance state that is chosen automatically according to demand. |
| Degrade | Always run at the lowest performance state, and use additional linear performance reduction as battery discharges.  This policy applies only to laptop systems that expose an ACPI long-term battery and support ACPI 1.0 linear stop-clock throttle states. On a server system with no battery or linear clock throttle states, this policy acts the same as the Constant policy and simply uses the lowest performance state. |

In Windows Server 2003, the processor throttling policy is linked to the power scheme setting. Setting the power scheme to “Server Balanced Processor Power and Performance” enables the Adaptive processor throttling policy.

To view the processor throttling policy, use the Powercfg.exe command-line tool as in the following example:

C:\>powercfg /query "Server Balanced Processor Power and Performance"

Field Description Value

----------------- -----

Name Server Balanced Processor Power and Performance

Numerical ID 4

Turn off monitor (AC) Not Supported

Turn off monitor (DC) Not Supported

Turn off hard disks (AC) Never

Turn off hard disks (DC) After 15 mins

System standby (AC) Never

System standby (DC) After 5 mins

System hibernates (AC) Never

System hibernates (DC) Never

Processor Throttle (AC) ADAPTIVE

Processor Throttle (DC) ADAPTIVE

Powercfg.exe can also be used to manually adjust the throttling policy for any power scheme. The following example sets the processor throttling policy to Constant:

C:\>powercfg /change "Server Balanced Processor Power and Performance" /processor-throttle-ac constant

For detailed information about how to use powercfg.exe on Windows Server 2003, see the KB article “How to use Powercfg.exe in Windows Server 2003.”

## Processor Performance States on Windows Server 2008

In Windows Server 2008, each processor power policy includes an upper and lower limit, which are known as the Maximum processor state and the Minimum processor state, respectively. These limits establish the range of currently available P‑states that Windows can use. These values can be set independently to define the bounds for any contiguous range of performance states, or they can be set to the same value to force the system to remain at a specific state.

Windows Server 2008 implements adaptive power policy by using a dynamic-based switching (DBS) algorithm. The algorithm uses all available performance states that fall within the range that is bounded by these upper and lower limits. To choose a new target performance state, Windows Server 2008 chooses the closest match between the current power policy setting and the states that are available on the system, rounding up if it is necessary.

The use of a maximum and minimum policy value provides for a flexible yet simple expression of processor policy. For example, the policy can provide for the static use of any single performance state within the range of states that the hardware supports. Power policy can otherwise constrain the use of available states to a subset of the full range that the platform supports. This enables locking at either the highest or lowest state, or allows the use of DBS on either the full range of P‑states or a subset of the available P‑states.

To display the current processor power policy settings on Windows Server 2008, use powercfg.exe as follows:

C:\>powercfg /q scheme\_current sub\_processor

Power Scheme GUID: 381b4222-f694-41f0-9685-ff5bb260df2e (Balanced)

Subgroup GUID: 54533251-82be-4824-96c1-47b60b740d00 (Processor power management)

Power Setting GUID: 68f262a7-f621-4069-b9a5-4874169be23c (Processor power CState settings)

Possible Setting Index: 000

Possible Setting Friendly Name: Power saver

Possible Setting Index: 001

Possible Setting Friendly Name: Power saver

Possible Setting Index: 002

Possible Setting Friendly Name: Balanced

Possible Setting Index: 003

Possible Setting Friendly Name: Balanced

Possible Setting Index: 004

Possible Setting Friendly Name: High performance

Possible Setting Index: 005

Possible Setting Friendly Name: High performance

Current AC Power Setting Index: 0x00000002

Current DC Power Setting Index: 0x00000003

Power Setting GUID: 893dee8e-2bef-41e0-89c6-b55d0929964c (Minimum processor state)

Minimum Possible Setting: 0x00000000

Maximum Possible Setting: 0x00000064

Possible Settings increment: 0x00000001

Possible Settings units: %

Current AC Power Setting Index: 0x00000005

Current DC Power Setting Index: 0x00000005

Power Setting GUID: bbdc3814-18e9-4463-8a55-d197327c45c0 (Processor power PerfState settings)

Possible Setting Index: 000

Possible Setting Friendly Name: Power saver

Possible Setting Index: 001

Possible Setting Friendly Name: Power saver

Possible Setting Index: 002

Possible Setting Friendly Name: Balanced

Possible Setting Index: 003

Possible Setting Friendly Name: Balanced

Possible Setting Index: 004

Possible Setting Friendly Name: High performance

Possible Setting Index: 005

Possible Setting Friendly Name: High performance

Current AC Power Setting Index: 0x00000002

Current DC Power Setting Index: 0x00000003

Power Setting GUID: bc5038f7-23e0-4960-96da-33abaf5935ec (Maximum processor state)

Minimum Possible Setting: 0x00000000

Maximum Possible Setting: 0x00000064

Possible Settings increment: 0x00000001

Possible Settings units: %

Current AC Power Setting Index: 0x00000064

Current DC Power Setting Index: 0x00000064

Powercfg.exe can also be used to manually adjust the processor policy settings for any power plan. The following example sets the processor throttling policy to limit the maximum processor performance state to 50 percent:

C:\>powercfg -setacvalueindex scheme\_current sub\_processor procthrottlemax 50

C:\>powercfg -setactive scheme\_current

Windows Server 2008 does not use a dynamic algorithm with ACPI linear clock throttle states. When the system does not support any processor performance states, Windows Server 2008 can use linear throttle states to statically cap the maximum processor speed through the Maximum processor state power policy value.

By default, when the system supports both ACPI processor performance states and linear clock throttle states, Windows Server 2008 does not reduce the processor speed below that of the lowest performance state. The kernel power manager enforces this limitation when choosing a target state, regardless of the value of the Maximum processor state power policy value.

**Note:** ACPI thermal zones are an exception to this constraint. For details, see “Processor Power Management in Windows Vista and Windows Server 2008.”

An administrator can configure Windows to allow the use of linear throttle states when performance states are present. This allows the Maximum processor state policy value to be set to a value below the speed of the lowest performance state, so that the processor can be capped at this value. However, Windows does not dynamically use states below the value of the lowest performance state.

The power policy value that controls this behavior is marked with the Hidden attribute by default and so is not shown in the Power Options user interface. To change this setting, follow these steps:

1. Open an elevated command prompt.

2. View the Allow Throttle States processor power policy by using the following command:

C:\>powercfg -qh scheme\_current sub\_processor

Power Scheme GUID: 381b4222-f694-41f0-9685-ff5bb260df2e (Balanced)

Subgroup GUID: 54533251-82be-4824-96c1-47b60b740d00 (Processor power management)

Power Setting GUID: 3b04d4fd-1cc7-4f23-ab1c-d1337819c4bb (Allow Throttle States)

Possible Setting Index: 000

Possible Setting Friendly Name: Off

Possible Setting Index: 001

Possible Setting Friendly Name: On

Current AC Power Setting Index: 0x00000000

Current DC Power Setting Index: 0x00000000

3. Copy the globally unique identifier (GUID) for the Allow Throttle States processor power policy.

4. Set the Allow Throttle States policy value to 1 by using the following command:

C:\>powercfg -setacvalueindex scheme\_current sub\_processor

3b04d4fd-1cc7-4f23-ab1c-d1337819c4bb 1

5. Apply the new policy setting by using the following command:

C:\>powercfg -setactive scheme\_current

For detailed information about how to use powercfg.exe on Windows Server 2008, see “PowerCfg Command-Line Options.”

# Best Practices and Recommendations

Windows power policy should be the primary mechanism for throttling power in Windows Server. This maximizes performance and power draw to load within the set budget. Hardware throttling mechanisms must be present to provide a reliable failsafe. The cooperation of Windows power policy and hardware mechanisms ensures the best and reliable operation of servers.

The following are best practices for throttling power on Windows Server systems:

* Enable Windows to dynamically manage processor performance states.
* Adjust Windows power policy to control power consumption where practical.
* Use ACPI Notify() events to cap processor(s) as a preventive safeguard if power consumption approaches the allocated power budget despite the Windows power policy.

For Windows Server 2003 or 2008 with a \_PSS object, use the   
Notify(*cpu*, 0x80) event.

For Windows Server 2008 with a \_TSS object, use the Notify(*cpu*, 0x82) event.

These events are suitable to be invoked at approximately 1- to 2‑second intervals. Use them only as needed to maintain the power ceiling as power consumption approaches the budget.

* Use thermal control circuits as a hardware failsafe if power consumption further encroaches on the guard band of the power cap.

Thermal control circuits are suitable for sub-millisecond power correction and only for very short duration application, such as in abnormal conditions.

* Do not bypass or overload specified ACPI and OSPM interfaces to implement external control.
* Do not engage thermal control circuits for extended periods of time.

# Resources

#### White Papers on the WHDC Web Site

Processor Power Management in Windows Vista and Windows Server 2008

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

How to Enable Processor Power Management in Windows Server 2003

<http://www.microsoft.com/whdc/system/pnppwr/powermgmt/w2k3_ProcPower.mspx>

Power Policy Configuration and Deployment in Windows Vista

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

Windows Native Processor Performance Control

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

#### KB Articles

How to use Powercfg.exe in Windows Server 2003

<http://support.microsoft.com/kb/324347>

#### Specifications

ACPI – Advanced Configuration and Power Interface Specification version 3.0a

<http://www.acpi.info/spec30a.htm>

#### Tools

PowerCfg Command-Line Options

<http://technet2.microsoft.com/WindowsVista/en/library/1d58b934-f56a-4796-b2df-7be2eb9c03bc1033.mspx?mfr=true>

1. Beck, Fred, *Energy Smart Data Centers: Applying the Energy Efficient Design and Technology to the Digital Information Sector*, REPP Research Report No. 14, Page 8. [↑](#footnote-ref-2)
2. *Server Power Consumption Reemerges as a Critical Cost Factor in Datacenters*, IDC, Insight #33937, August 2005. [↑](#footnote-ref-3)