**How Intel Power Management Works**

**ABSTRACT:** It is worth noting that the Intel® Xeon® processor 5500 series-based servers offer the capability to allow the operator to actually set power consumption and trade off the power consumption level against performance. This capability is valuable from two perspectives. First, power capping brings predictable power consumption within the specified power capping range, and second, servers implementing power capping offer actual power readouts as a bonus: their power supplies are PMBus*-enabled and their historical power consumption can be retrieved through standard APIs. With actual historical power data, it is possible to optimize the loading of power limited racks, whereas before the most accurate estimation of power consumption had to be derived from derated nameplate data. The nameplate estimation for power consumption is a static measure that requires allowing a considerable safety margin. This conservative approach to power sizing leads to over-provisioning of power. This was acceptable when energy costs were a second order consideration. Not today anymore.

Integration with the Intel® Data Center Manager SDK allows dialing the power to be consumed by groups of over a thousand servers, allowing a power control authority of tens of thousands of watts in data centers. How does power capping work? The foundation for power management resides in CPU voltage and frequency scaling implemented by the Intel® Xeon® processor 5500 series architecture. More likely than not, CPUs represent the most energetic components in a server. If we can regulate the power consumed by the CPUs we can have an appreciable effect on the power consumed by the server as a whole. Multiply this control over the thousands of servers in a data center. Through this mechanism, we can alter the power consumed in that data center in significant ways.

**Nesting Technologies for Scaling Power Management**

Figure 1 below depicts a series of abstractions for a large deployment of servers in a data center, and cloud storage appliances in particular. Chipsets are used to bind together the CPUs and the memory in a server. A server carries direct-attached (DASD) storage. Servers are organized in racks, and racks are organized in rows. The aggregation of rows encompasses all the servers in a data center.

Today our main lever for power control is by throttling the power consumed by the CPUs up and down. In the near future we can expect to see memory power control added to the mix. The basic mechanism of CPU voltage and
frequency scaling allows moving the power consumption of a CPU by a few tens of watts up or down. Aggregating this capability over the tens of thousands of CPUs in a data center expands the range of attainable power control to tens of kilowatts or even hundreds.

There is also a potential synergistic effect captured by the PUE (power usage effectiveness) factor as defined by The Green Grid* industry group. If servers use less power, the power allocated to the cooling equipment can be ratcheted down as well.

---

**Figure 1**  Power control hierarchy.

Power control for groups of servers is attained by composing power control capabilities of power control of each server. Likewise, power control for a server is attained by composing CPU power control as illustrated in Figure 1.

Conceptually, power control for thousands of servers in a data center is implemented through a series of coordinated set of nested mechanisms.

**Managing CPU Power Consumption**

The lowest level is implemented through frequency and voltage scaling: laws of physics dictate that for a given architecture, power consumption is proportional to the CPU’s frequency and to the square of the voltage used to power the CPU.
There are mechanisms built into the CPU architecture that allow a certain number of discrete combinations of voltage and frequency. Using the ACPI standard nomenclature, these discrete combinations are called P-states, the highest performing state is nominally identified as P0, and the lower power consumption states are identified as P1, P2 and so on.

An Intel® Xeon® processor 5500 series supports over ten states, the actual number depending on the processor model. For the sake of an example, a CPU in P0 may have been assigned a voltage of 1.4 volts and 3.6 GHz, at which point it draws about 100 watts.

As the CPU transitions to lower power states, it may have a state P4 using 1.2 volts running at 2.8 GHz and consuming about 70 watts. Table 1 illustrates this example. Actual CPUs may actually support over ten P-states. Please consult the CPU specifications for the actual values.

<table>
<thead>
<tr>
<th>Table 1</th>
<th>P-State Table Example.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power State</td>
<td>CPU Voltage (V)</td>
</tr>
<tr>
<td>P0</td>
<td>1.4</td>
</tr>
<tr>
<td>P1</td>
<td>1.35</td>
</tr>
<tr>
<td>P2</td>
<td>1.3</td>
</tr>
<tr>
<td>P3</td>
<td>1.25</td>
</tr>
<tr>
<td>P4</td>
<td>1.2</td>
</tr>
</tbody>
</table>

The P-states by themselves can't control the power consumed by a server. The CPU itself has no mechanisms to measure the power it consumes. This mechanism is implemented by firmware running in the Intel® Xeon® processor 5500 series chipset. This firmware implements the Intel® Intelligent Power Node Manager technology.

Managing Server Power Consumption

If what we want is to measure the power consumed by a server, looking only at CPU consumption does not provide the whole picture. For this purpose, the power supplies in Intel® Intelligent Power Node Manager-enabled servers are instrumented to provide actual power readings for the whole server through the PMBus* standard.

This process is shown in Figure 2. With these components in place, it is now possible to establish a classic control feedback loop where we compare a target power against the actual power indicated by the power supplies. The
Intel® Intelligent Power Node Manager code manipulates the P-states up or down until the desired target power is reached.

If the desired power lies between two P-states, the Intel® Intelligent Power Node Manager code rapidly switches between the two states until the average power consumption meets the set power. This is an implementation of another classic control scheme, affectionately called bang-bang control for obvious reasons.

![Power Control Loop in an Intel® Xeon® Processor 5500 Series Server.](image)

**Figure 2** Power Control Loop in an Intel® Xeon® Processor 5500 Series Server.

Figure 3 provides an expanded detail of the control loop. Here we can see that the Intel® Intelligent Power Node Manager firmware implements the difference engine. Intel® Intelligent Power Node manager directs the operating system or the hypervisor to change to a target P-state.

![Expanded view of the Intel® Xeon® processor 5500 series power control loop.](image)

**Figure 3** Expanded view of the Intel® Xeon® processor 5500 series power control loop.

Note that the target P-state is set through an in-band mechanism, that is, through the operating system. Intel® Intelligent Power Node Manager does not set P-states directly. It sends requests to the operating system or hypervisor through an API. This is necessary to coordinate power policies.
with other power policies that the operating system or hypervisor might be carrying out.

Changes in processor P-state induce changes in the level of power consumption registered by the PMBus enabled power supplies.

Managing Power Consumption in Server Groups

From a data center perspective, the ability to regulate power consumption of just a single server has a small impact and is not intrinsically useful. Harnessing the “power of the masses” represents a key capability. We need the means to control servers as a group, and just as we were able to obtain power supply readouts for one server, we need to monitor the power for the group of servers to allow meeting a global power target for that group of servers. This function is provided by the Intel® Data Center Manager software development kit and shown in Figure 4.

![Power control loop in a group managed by Intel® Data Center Manager.](image)

**Figure 4**  Power control loop in a group managed by Intel® Data Center Manager.

Note that Intel® Data Center Manager implements a feedback control mechanism very similar to the mechanism that regulates power consumption for a single server, but at a much larger scale. Instead of watching one or two power supplies, Intel® Data Center Manager oversees the power consumption of multiple servers or “nodes” whose number can range up to thousands. At this level Intel® Data Center Manager is in charge of implementing the difference engine.

Figure 5 depicts an expanded view of the Intel® Data Center Manager control loop as well as the relationship with the NM control loop underneath. No specific agents need to run in each node. Intel® Data Center Manager communicates with the board management controller (BMC) in each node for setting power targets and for doing readouts of the actual power consumed. Intel® Intelligent Power Node Manager firmware takes care of ensuring that the individual server meets the assigned power consumption target.
Intel® Data Center Manager was purposely architected as an SDK as a building block for industry players to build more sophisticated and valuable capabilities for the benefit of data center operators. One possible application is shown in Figure 6, where Intel® Data Center Manager has been integrated into a cloud storage appliance application. Some Intel® Intelligent Power Node Manager-enabled servers come with inlet temperature sensors. This allows the application to monitor the inlet temperature of a group of servers, and if the temperature rises above a certain threshold, it can take a number of measures, from throttling back power consumption to reducing thermal stresses. At this level the application code is in charge of the difference engine.

**Figure 5** Intel® Data Center Manager power control loop, detailed view.

An application interfacing with Intel® Data Center manager no longer “sees” individual server nodes; the application code’s power policy engine designates a power consumption target to Intel® Data Center Manager through the Intel® Data Center Manager API. Intel® Data Center Manager in turn breaks down the power target into power targets to the individual nodes under its command.
Cloud Storage Application Power Management

The application code also needs to mind the power consumed by the storage subsystem. Less sophisticated implementation may have a monitor-only capability; the application code reads out the power consumed by the storage subsystem by querying the intelligent PDU that feeds it and then sets the server power in a way that the total consumption for the appliance matches the overall set power.

Figure 6  Storage appliance control loop.

Figure 7 provides a more detailed view of this process. In this figure we see the appliance control algorithm implemented by the application’s power policy engine. The constituent components are loosely coupled through Web services APIs.

One potential concern with three nested loops operating concurrently is the potential for oscillatory or unstable behaviors. This issue does not arise in practice; the time constants across the three levels are pretty spread out: the time constant involved with the Intel® Intelligent Power Node Manager loop is in the order of milliseconds, whereas the Intel® Data Center Manager constant is in the order of a few seconds. Finally, the time constant associated with power management for an appliance is in the order of tens of seconds. Because of the time constant spreads, it is possible to optimize each loop individually to ensure that no under-damped, oscillatory behaviors occur.

**About the Authors**

**Enrique Castro-Leon** is an enterprise and data center architect and technology strategist for Intel Digital Enterprise Group working in OS design and architecture, software engineering, high-performance computing, platform definition, and business development. His work involves matching emerging technologies with innovative business models in developed and emerging markets. In this capacity he has served as a consultant for corporations large and small, nonprofits and governmental organizations on issues of technology and data center planning.

He has taken roles as visionary, solution architect, project manager, and has undertaken the technical execution in advanced proof of concept projects with corporate end users and inside Intel illustrating the use of emerging technologies. He has published over 40 articles, conference papers, and white papers on technology strategy and management as well as SOA and Web services. He holds PhD and MS degrees in Electrical Engineering and Computer Science from Purdue University.

**Bernard Golden** has been called “a renowned open source expert” (IT Business Edge) and “an open source guru” (SearchCRM.com) and is regularly featured in magazines like Computerworld, InformationWeek, and
Inc. His blog “The Open Source” is one of the most popular features of CIO Magazine’s Web site. Bernard is a frequent speaker at industry conferences like LinuxWorld, the Open Source Business Conference, and the Red Hat Summit. He is the author of Succeeding with Open Source, (Addison-Wesley, 2005, published in four languages), which is used in over a dozen university open source programs throughout the world. Bernard is the CEO of Navica, a Silicon Valley IT management consulting firm.

**Miguel Gomez** is a Technology Specialist for the Networks and Service Platforms Unit of Telefónica Investigación y Desarrollo, working on innovation and technological consultancy projects related to the transformation and evolution of telecommunications services infrastructure. He has published over 20 articles and conference papers on next-generation service provisioning infrastructures and service infrastructure management. He holds PhD and MS degrees in Telecommunications Engineering from Universidad Politécnica de Madrid.

Copyright © 2010 Intel Corporation. All rights reserved.


No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4744. Requests to the Publisher for permission should be addressed to the Publisher, Intel Press, Intel Corporation, 2111 NE 25 Avenue, JF3-330, Hillsboro, OR 97124-5961. E-mail: intelpress@intel.com.