

# Einschlaf-Hilfen

Mit dem neuen Kernel hält ACPI auf relativ breiter Front in Linux Einzug. Die vereinheitlichte Hardware-Konfiguration ist schon aus dem Kernel 2.4 bekannt, beim Powermanagement kommen aber gleich mehrere Schlafmodi hinzu. Timo Hönig

den Advanced Programmable Interrupt Controller (APIC) statt des betagten PIC einsetzen – um Ressourcenkonflikte oder die Mehrfachbelegung von Interrupts zu vermeiden – führt kein Weg an ACPI vorbei. Außerdem erfährt der Kernel über das ACPI-Subsystem, welche PCI-Devices Hotplug-fähig sind und wie diese zu konfigurieren sind. Die Routinen liegen in »drivers/pci/hotplug/acpiphp\_glue.c«.

ACPI dagegen abstrahiert die Hardware-abhängigen Funktionen durch die ACPI Source Language (ASL) und deren Compiler, die ACPI Machine Language. AML beschreibt die Hardware und die Schritte, um auf sie zuzugreifen. Jedes ACPI-kompatible Betriebssystem besitzt einen AML-Interpreter für AML-Byte-Code.

## Zwei-Ebenen-Spezifikation

Die ACPI-Spezifikation [1] gliedert eine ACPI-Architektur in zwei Ebenen. Die erste, Hardware-nahe Ebene (Low Level) der ACPI-Architektur besteht aus:

- ACPI-Tabellen
- ACPI-Bios
- ACPI-Registern

Die ACPI-Tabellen beschreiben die ACPI-Hardware und deren Konfiguration in AML. Die zentrale Datenstruktur jedes ACPI-Systems beschreibt in Definitionenblöcken, wie die Hardware angesprochen wird. Das ACPI-Bios schreibt beim Systemstart die ACPI-Tabellen in den Speicher. Auch beim Schlafenlegen und Aufwecken (Suspend-to-RAM, Suspend-to-Disk) ist das ACPI-Bios involviert.

Die zweite Ebene (High Level) ist Teil des Betriebssystems und stellt mit einem ACPI-Core und den ACPI-Treibern ein API bereit, um auf die Low-Level-Kom-

ponenten von ACPI zuzugreifen. Der AML-Interpreter ist Teil dieser Ebene.

## ACPI unter Linux 2.6

Alle ACPI-Powermanagement-Funktionen [2] fasst die Kernelkonfiguration unter »Power management options (ACPI, APM)« zusammen (siehe **Abbildung 1**). Wer Suspend-to-RAM nutzen will, muss unter »Code maturity level options« die Option »Prompt for development and/or incomplete code/drivers« auswählen, damit bei den ACPI-Optionen der Punkt »Sleep States« erscheint.

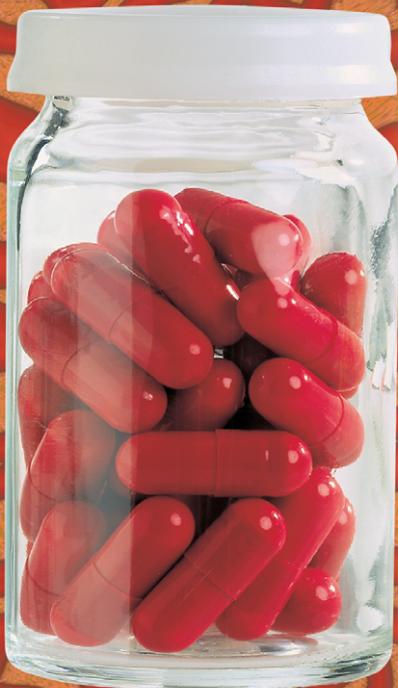
Beim Booten lokalisiert der Kernel mit der Funktion »acpi\_boot\_init()« die ACPI-Tabellen und liest sie über das ACPI-Bios aus. Von besonderer Bedeutung ist die Differentiated System Description Table (DSDT). Sie enthält die Spezifikationen der bereits vorhandenen ACPI-kompatiblen Hardwarekomponenten und deren Konfiguration.

Es ist bekannt, dass manche Hersteller defekte DSDTs in ihre Systeme integrieren. Selbst wenn der Hersteller kein Bios-Upgrade mit einem fehlerfreien DSDT anbietet und auch die Kerneloption »Relaxed AML« (»CONFIG\_ACPI\_RELAXED\_AML«) nicht weiterhilft, ist nicht alle Hoffnung verloren: Für viele Systeme mit defektem DSDT stehen auf [3] fehlerbereinigte DSDTs zum Download bereit. Unter [4] sind die nötigen Schritte nachzulesen.

Wer Pech hat und für sein System kein fehlerloses DSDT findet, darf selbst Hand anlegen: Intel stellt einen ASL-Compiler und den AML-Disassembler »iasl« [5] zur Verfügung. Er kann die defekte DSDT über das Proc-Dateisystem (»cat /proc/acpi/dsdt > dsdt.aml«) kopieren

**Geschafft:** Fast alle Notebooks, Komplett-PCs, Komponenten und Betriebssysteme schmücken sich mit dem Kürzel ACPI. Seit der erste Version des Advanced Configuration and Power Interface im Jahr 1996 ist es das erklärte Ziel der Spezifikation, den APM-Standard (Advanced Power Management) und das unflexible Plug&Play-Bios endlich in Rente zu schicken.

ACPI wird oft als reiner Ersatz für APM angesehen. Falsch: Das C für Configuration ist ein integraler Bestandteil von ACPI, der eine einheitliche und soweit möglich betriebssystemunabhängige Schnittstelle für das Hardwaresetup liefert. So verwendet Linux 2.6 ACPI für das Routing der PCI-Interrupts. Will man hierfür auf einem Ein-Prozessor-System



und anschließend mit dem AML-Compiler disassemblieren. Der Befehl »iasl -d dsdt.aml« disassembliert die DSDT und schreibt den ASL-Source in der Datei »dtds.dsl« (Listing 1).

Nun ist es an der Zeit, den Fehler im ASL-Source zu lokalisieren, zu beheben und erneut zu kompilieren (»iasl -t dtds.dsl«). Warnings und Errors sollte man durch Änderungen des ASL-Source

(»dtds.dsl«) zu eliminieren versuchen, bevor der kompilierten DSDT zum Einsatz kommt. Tipps zum Patchen fehlerhafter DSDTs finden sich auf [6].

Ist Linux 2.6 mit ACPI-Support erst mal gebootet, lohnt ein Blick in die ACPI-Einträge im Proc- und Sys-Filesystem: Über die Einträge in »/proc/acpi« und »/sys« kann das ACPI-Interface angesprochen werden. Schreibzugriffe sind nur Root möglich. Wer »sysfs« noch nicht gemountet hat, sollte dies mit

```
mkdir /sys
mount -t sysfs sysfs /sys
```

nachholen. Der Eintrag »sysfs /sys sysfs defaults 0 0« in der Datei »/etc/fstab« sorgt dafür, dass das Sysfs beim Booten automatisch mountet.

## Sleep States

Die einzelnen Sleep States (Tabelle 1) sind über »/sys/power/state« steuerbar. Welche Zustände das System unter-

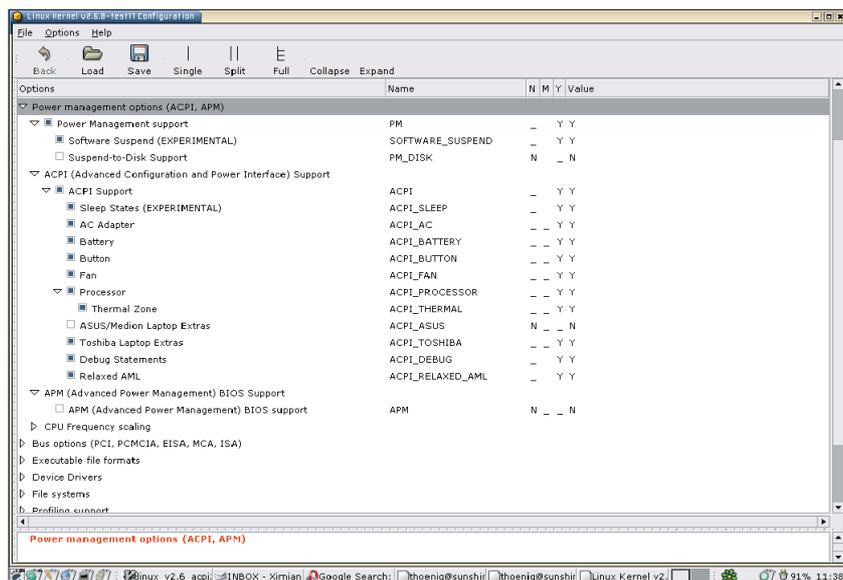


Abbildung 1: Kernel-2.6-Optionen für das Powermanagement per ACPI.

stützt, erfährt »cat /sys/power/state«. Der Befehl »echo -n "Sleep\_State" /sys/power/state« versetzt das System in den entsprechenden Zustand.

Suspend-to-RAM (S3) schaltet bis auf den Hauptspeicher das gesamte System ab: CPU, Caches, Chipsatz und Peripherie (Festplatten, USB und so weiter). Damit der Speicherinhalt nicht verloren geht, bleibt der RAM mit Strom versorgt. Einschlafen und Aufwachen erfolgen bei Suspend-to-RAM in wenigen Sekunden

### Listing 1: Disassemblierung einer DSDT mit »iasl«

```
01 [root@sunshine:~]$ cp /proc/acpi/dsdt ~/dsdt
02 [root@sunshine:~]$ iasl -d ./dsdt
03 Intel ACPI Component Architecture
04 ASL Optimizing Compiler / AML Disassembler version
05 20030918 [Sep 18 2003]
06 Copyright (C) 2000 - 2003 Intel Corporation
07 Supports ACPI Specification Revision 2.0b
08 Loading Acpi table from file dsdt
09 Acpi table [DSDT] successfully installed and loaded
10 Pass 1 parse of [DSDT]
11 Pass 2 parse of [DSDT]
12 Parsing Deferred Opcodes (Methods/Buffers/Packages/
13 Regions)
14 .....
14 Parsing completed
15 Disassembly completed, written to "dsdt.dsl"
16 [root@sunshine:~]$ head dsdt.dsl
17 /*
18 * Intel ACPI Component Architecture
19 * AML Disassembler version 20030918
20 *
21 * Disassembly of dsdt, Thu Dec 11 17:28:16 2003
22 */
23 DefinitionBlock ("DSDT.aml", "DSDT", 1, "TOSHIB",
24 "2000", 537003284)
25 {
26     Name (\_S0, Package (0x04)
27 [root@sunshine:~]$
```

### Listing 2: Suspend-Skript

```
01 #!/bin/sh
02 rmmod usbblp usb-storage hid; # USB Module entfernen
03 rmmod ohci-hcd; # Host-Controller Modul entfernen
04 rmmod usbcore; # USB-Core Modul entfernen
05
06 echo -n "mem" > /sys/power/state; # Suspend to RAM
07
08 modprobe usbcore; # USB-Core Modul laden
09 modprobe ohci-hcd; # Host-Controller Modul laden
10 modprobe usbblp usb-storage hid; # USB Module laden
11
12 exit 0;
```

– theoretisch. Nach den Erfahrungen des Autors funktioniert Suspend-to-RAM nur selten auf Anhieb. Zu Recht führt es der Kernel als »Experimental«.

Suspend-to-Disk (S4) spart von allen Sleep States am meisten Strom, da der aktuelle Speicherinhalt sowie Register und Peripheriestatus auf die Festplatte (unter Linux auf eine genügend große Swap-Partition) geschrieben werden und das System danach komplett stoppt. Der nächste Boot liest den Inhalt des Hauptspeichers und den alten Systemstatus von der Festplatte und stellt den früheren Zustand wieder her. Unter Linux 2.6 funktioniert Suspend to Disk auch unabhängig vom ACPI- oder APM-Support. Ein Rückblick: Linux 2.4 beherrscht Suspend-to-RAM überhaupt nicht; Suspend-to-Disk ist mit den Kernelpatches des Projekts Swsusp [7] aber möglich.

## Software Suspend und Suspend-to-Disk

Verwirrenderweise kennt Linux 2.6 zwei alternative Methoden von Suspend-to-Disk (S4): Software Suspend (»CONFIG\_SOFTWARE\_SUSPEND«) und Suspend-to-Disk Support (»CONFIG\_PM\_DISK«). Die erste kennzeichnet das Kernelsetup als »Experimental«. Suspend-to-Disk Support ist dagegen ein Stable-Zweig von Suspend-to-Disk.

Software Suspend setzt den Bootprompt-Parameter »resume« voraus – ihn in den Kernel einkompilieren ist nicht vorgesehen. Die Option gibt dem Kernel die Swap-Partition bekannt, in die er den Speicherinhalt und den Status sichern soll. Es ist sinnvoll, den Bootprompt-Parameter in der Konfigurationsdatei des Bootmanagers zu verewigen, beispielsweise als »resume = /dev/hda2«.

Bei dem zum Software Suspend alternativen Suspend-to-Disk Support gibt man

die Swap-Partition mit dem Kernelparameter »pmdisk« an. Im Gegensatz zu »resume« lässt sich »pmdisk« statisch in den Kernel einbinden.

## Unruhiger Schlaf

Für Suspend-to-RAM und Suspend-to-Disk ist es unabdingbar, dass die aktuell geladenen Treiber auf einen Suspend- und Resume-Zyklus vorbereitet sind. In Linux 2.6 wurde hierfür das Treibermodell entsprechend verändert. Danach müssen Device-Treiber über eine »suspend«- sowie »resume«-Funktion verfügen, die der Kernel bei Beginn oder Ende eines Sleep State aufruft.

Bleibt das System beim Schlafenlegen oder Aufwachen trotzdem hängen, hilft folgender Work-around: Nach dem Studium der Kernel-Logfiles kompiliert Root den renitenten Treiber – falls er das nicht schon war – als Modul neu. Bevor er nun »echo Sleep\_State > /sys/power/state« aufruft, entfernt er einfach das Treibermodul, um es nach dem Aufwachen wieder zu laden. Der ganze Suspend-Vorgang ist mit einem kleinen Skript gut automatisierbar. Listing 2 zeigt ein Beispiel, das einem System zum Suspend-to-RAM verhalf, dessen USB-Host-Controller (Treibermodul »ohci-hcd.o«) beim Aufwachen eine Kernel-Panic verursachte.

Sind im Zuge von Suspend-to-Disk (sowohl Software Suspend als auch Suspend-to-Disk) Probleme beim Wiederherstellen zu beklagen, können die Kernelparameter von Hand überschrieben werden: Mit »resume = noresume« oder »pmdisk = off« unterlässt der Kernel beim Booten den Versuch, den alten Systemzustand herzustellen. In der Variante Software Suspend muss noch die Swap-Partition neu initialisiert werden, im Beispiel mit »mkswap /dev/hda2«, und an-

Tabelle 1: ACPI Sleep States in Linux 2.6

| ACPI Sleep State  | Sleep_State aus »/sys/power/state« | Beschreibung   |
|---|------------------------------------|--|
| Standby (S1)  | »standby«                          | System hat Strom, die CPU führt jedoch keine Instruktionen mehr aus.               |
| Suspend-to-RAM (S3)   | »mem«                              | System ist bis auf den Hauptspeicher abgeschaltet.                                 |
| Suspend-to-Disk (S4)  | »disk«                             | System ist abgeschaltet, Hauptspeicher und Status werden auf Festplatte gesichert. |
| Anmerkung: Die Zustände S0 (Running, System läuft) und S5 (Soft Off, System ist abgeschaltet) gehören nicht zu den ACPI Sleep States. |                                    |  |

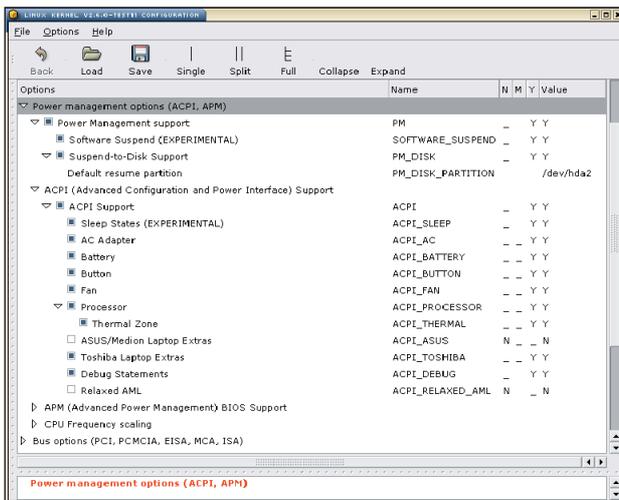


Abbildung 2: »cpufreq«-ACPI-Optionen für die aktuellen Mobile-CPUs.

schließend dem System wieder zugeteilt werden: »swapon /dev/hda2«.

## Dynamische Taktfrequenz für Mobile-CPUs

Mobile-CPUs moderner Bauart können ihre wirksame Taktfrequenz in Abhängigkeit von der ihnen aktuell abgeforderten Rechenleistung runter- und wieder rauffahren (Intel Speedstep und AMD Power Now!). Um mit Linux 2.6 dieses Feature zu nutzen, muss wie in [Abbildung 2](#) »cpufreq« aktiviert und gemäß des Prozessors der richtige »cpufreq«-Treiber ausgewählt sein.

Das Interface von »cpufreq« hat sich während der Entwicklung in Linux 2.5 geändert und erscheint nun endgültig im Sysfs. Wer sich mit einem derart kompilierten Kernel an die Arbeit macht, sollte mit »dmesg | grep cpufreq« prüfen, ob der Kernel das Feature erfolgreich initialisieren konnte (siehe [Listing 3](#)).

Quittiert »cpufreq« dagegen mit einem Fehler, sollte der Admin nochmals sicherstellen, dass er den richtigen »cpufreq«-Treiber gewählt hat. Im Test war – so paradox es klingt – der »cpufreq«-ACPI-Treiber erst mit aktiviertem Plug & Play-Bios-Support im Kernel (»CONFIG\_PNP« und »CONFIG\_PNPBIOS«) zur Mitarbeit zu überreden.

## Eckdaten

Sobald »cpufreq« erfolgreich initialisiert, stehen im Sysfs unter »/sys/devices/system/cpu/cpu0/cpufreq« die folgen-

den Einträge zur Verfügung:

In »cpuinfo\_max\_freq« und »cpuinfo\_min\_freq« stehen der maximale und der minimale Prozessortakt, den die vorhandene CPU verträgt. Die zwei tatsächlichen Frequenzgrenzen – zwischen denen die CPU pendelt – speichern dagegen »scaling\_max\_freq« und »scaling\_min\_freq«. Mit »echo

*Takt* > scaling\_max\_freq« und »echo *Takt* > scaling\_min\_freq« kann Root die Werte neu belegen. »*Takt*« muss dabei im Wertebereich von »cpuinfo\_max\_freq« und »cpuinfo\_min\_freq« liegen.

Der Eintrag »scaling\_available\_governors« gibt preis, welche Strategien für die Anpassung des Takts das System beherrscht: »powersave«, »userspace« oder »performance«. Die aktuell aktive Strategie steht in »scaling\_governor«. Root darf per »echo *Governor* > scaling\_governor« für das System eine andere einschalten, wobei er »*Governor*« durch einen der drei von »scaling\_available\_governors« bekannten Strings ersetzt.

Um ein bisschen von den Echos auf der Kommandozeile wegzukommen, hat das Projekt Cpubyn [\[8\]](#) einen Daemon entwickelt, der das Interface von »cpufreq« benutzt. Cpubyn hilft die Taktraten-Strategie moderner CPUs an die eigenen Bedürfnisse anzupassen.

## ACPI Throttling

Ein Gutteil der Prozessoren, die ihren Takt nicht dynamisch anpassen können, bremst der Kernel dank ACPI auf andere Weise aus. Die gesparte elektrische Leistung ist der Hitze- und Lärmentwicklung vieler Systeme zuträglich – eine gute Idee vielleicht für manchen Server in den Nachtstunden.

Hierfür muss der Prozessor das so genannte Throttling (Drosselung) unterstützen. Davon überzeugt sich der Admin in »/proc/acpi/processor/CPUx/throttling«, das die verfügbaren Thrott-

ling-Zustände listet. Im Zustand »T0« läuft der Prozessor ungebremst. In allen anderen Throttling-Zuständen wird der Prozessor gemäß der Prozentangabe aus »/proc/acpi/processor/CPUx/throttling« langsamer.

## ACPI für Notebooks

Die hier genannten ACPI-Features gehen vor allem Notebook-Benutzer an. In »/proc/acpi/battery/BATx/info« stehen Informationen über die Akkus Ihrer Geräte. Ladezustand, Kapazität und Spannung ist »/proc/acpi/battery/BATx/state« entnehmbar. Unterschreitet der Wert eine gewisse Schwelle – die mittels »echo Schwelle /proc/acpi/battery/BATx/alarm« definierbar ist – warnt ACPI über die »/proc/acpi/events«.

Der Eintrag »/proc/acpi/ac\_adapter/ADPx/state« erteilt Auskunft darüber, ob am Gerät ein Netzteil hängt. Das Ein- und Ausstöpseln der Stromzufuhr hat eine Meldung über »/proc/acpi/events« zur Folge. Denselben Kanal benutzen außerdem der Buttontreiber (Kerneloption »CONFIG\_ACPI\_BUTTON«) und der Temperatortreiber (»CONFIG\_ACPI\_THERMAL«).

Der Buttontreiber teilt mit, sobald der Power-, Sleep- oder Lid-Taster (im Notebook-Display) betätigt wird. Der Temperatortreiber kontrolliert die Wärmeentwicklung der CPU (auszulesen unter »/proc/acpi/thermal\_zone/THRM/temperature«) und reagiert beim Erreichen von in »/proc/acpi/thermal\_zone/THRM/trip\_points« definierten Limits.

Unter Zuhilfenahme des ACPI-Daemon Acpid [9] können Notebookbesitzer mit den Meldungen von »/proc/acpi/events« auf ihre Geräte angemessen reagieren. Acpid sorgt dann beispielsweise beim Schließen des Deckels dafür, dass das Gerät automatisch in den Zustand Suspend-to-Disk geschickt wird.

Trotz der vielen ACPI-Funktionen wird die Begeisterung der Besitzer aktueller Notebooks bald der Ernüchterung weichen. Sie werden feststellen, dass sie un-

ter Linux weder die Helligkeit des Displays mit den vorgesehenen Tasten variieren können noch die Funktionstasten [Fn] ihrem Namen Ehre machen.

Eine gute Nachricht für Asus- und Medion-Notebooks: Der Treiber hinter der Option »ASUS/Medion Laptop Extras« (»CONFIG\_ACPI\_ASUS«) stellt Einträge unter »/proc/acpi/asus«, die die Display-Helligkeit und die LEDs des Systems steuern. Die [Fn]-Kombinationen setzt der Treiber in ACPI-Events um. Für die restlichen Funktionen ist ein Userspace-Daemon verfügbar [10].

Auch Toshiba-Notebooks profitieren von einem speziellen ACPI-Treiber, den »Toshiba Laptop Extras« (»CONFIG\_ACPI\_TOSHIBA«). Leider reicht das Modul die [Fn]-Tasten nur über den Eintrag »/proc/acpi/toshiba/keys« im Proc-Dateisystem weiter, statt Standard-ACPI-Events auszulösen. Die anderen »/proc/acpi/toshiba«-Einträge steuern die Display-Helligkeit und den Lüfter. Auch für die Toshiba-Erweiterung sind Userspace-Daemons verfügbar [11].

## Und was ist mit Doku?

Wer Probleme mit dem Gespann Linux 2.6 und ACPI hat, wird wohl seinem Instinkt folgen und im Netz nach Dokumentationen suchen. Kleiner Tipp: Die gibt's nicht. Auch aktuelle Fachliteratur umschiffet kollektiv gekonnt die Thematik »ACPI als Ersatz für Plug&Play-Bios«. Die Dokumentation zum und im Kernel selbst macht hier keine Ausnahme, neben ein paar Informationen zum Powermanagement via ACPI findet sich nicht viel Greifbares. Die besten Anlaufstellen bleiben neben Google die Acpi4linux-Mailingliste [12] und Newsgroups – hier kann man wenigstens versuchen, mit Leidensgenossen einen Weg zu finden.

## Fazit

Die aktuell verkaufte ACPI-Hardware übt einen Handlungsdruck auf die Entwickler aller Betriebssysteme aus. Mancher wird sich hier an das ACPI-Debakel von Windows 2000 erinnern. Die Ziele, die die ACPI-Spezifikation verfolgt, sind dringlich und sinnvoll: systematische Hardwarekonfiguration und einheitliches Powermanagement.

Als positives Ergebnis der Linux-Implementierung ist dabei zu vermerken, dass seit wenigen Monaten das IRQ-Routing der PCI-Interrupts durch ACPI kaum noch zu Problemen führt. Das herausragend neue Powermanagement-Feature in Linux 2.6 – Suspend-to-RAM – qualifiziert der Kernel jedoch zu Recht mit »Experimental« – hier geht auf vielen Geräten noch zu wenig.

Noch ärgerlicher, dafür kann aber Linux nichts, ist, dass Systeme mit fehlerhaften DSDTs verkauft werden – auch von Herstellern, die selbst im ACPI-Konsortium sitzen. Endanwender dürften damit überfordert sein, ihren Systemen eine gepatchte DSDT unterzujubeln. Es ist viel Feinarbeit zu leisten, bis ACPI wirklich die zentrale Schnittstelle ist, die jegliche Hardware konfiguriert. (jk) ■

### Infos

- [1] ACPI: [<http://www.acpi.info>]
- [2] ACPI4Linux: [<http://acpi.sf.net>]
- [3] ACPI4Linux DSDT Übersicht: [<http://acpi.sf.net/dsdt/index.php>]
- [4] DSDT initrd Patch: [<http://gaugusch.at/kernel.shtml>]
- [5] Intel IASL: [<http://www.intel.com/technology/iapc/acpi/downloads.htm>]
- [6] Defekte DSDT reparieren: [[http://www.cpqlinux.com/acpi-howto.html#fix\\_broken\\_dsdt](http://www.cpqlinux.com/acpi-howto.html#fix_broken_dsdt)]
- [7] Software Suspend für Linux: [<http://swsusp.sf.net>]
- [8] CPU Dynamic Frequency Control: [<http://mnm.uib.es/~gallir/cpudyn/>]
- [9] ACPI Event Daemon: [<http://acpid.sf.net>], [[http://sf.net/project/showfiles.php?group\\_id=33140](http://sf.net/project/showfiles.php?group_id=33140)]
- [10] ASUS/Medion Laptop Extras: [<http://julien.lerouge.free.fr>]
- [11] Toshiba ACPI-Tools: [<http://fnfx.sf.net>], [<http://sourceforge.net/projects/tclkeymon/>]
- [12] ACPI4Linux Mailingliste: [<http://acpi.sf.net/maillinglists.html>]

### Der Autor

Timo Hönig studiert Informatik an der Fachhochschule Augsburg. Seit zehn Jahren bewegt er sich in der Welt von Netzwerken, Desktops, Open Source und Unix. Zurzeit widmet er sich mit Leidenschaft seinem Open-Source-Projekt FnFX, einem Linux-ACPI-Daemon für Toshiba-Laptops. [<http://fnfx.sourceforge.net>]

### Listing 3: Der »cpufreq«-Test

```
01 [root@sunshine:~]$ dmesg | grep cpufreq
02 cpufreq: CPU0 - ACPI performance management activated.
03 cpufreq: *P0: 750 MHz, 22000 mW, 250 uS
04 cpufreq: P1: 350 MHz, 9800 mW, 250 uS
```