

Hard Disk Power Management 2: Implementierung eines energie-gewahren Update-Mechanismus

In den Rechnern akbp1 und akbp2 ist jeweils eine IBM Travelstar Notebook-Festplatte eingebaut; in Linux erscheint sie als hdc.

Seiten, die den Inhalt von Disk-Blöcken enthalten und modifiziert wurden (als *dirty* markiert), werden vom *pdflush*-demon auf die Platte zurückgeschrieben. Der Demon überprüft alle 5 Sekunden (*dirty writeback centisecs*), ob modifizierte Seiten vorliegen und schreibt diese auf Platte, sobald sie mindestens 30 Sekunden alt sind (*dirty expire centisecs*). Damit soll vermieden werden, dass bei kurz aufeinanderfolgenden Schreibzugriffen jedesmal auf die Platte zugegriffen wird. Die Zeiten können unter `/proc/sys/vm/dirty_expire_centisecs` bzw. `dirty_writeback_centisecs` geändert werden.

An diesem Mechanismus sollen folgende Änderungen vorgenommen werden, um einen energie-gewahren Update-Prozess zu implementieren:

1. Es sollen immer alle modifizierten Seiten zurückgeschrieben werden, unabhängig von ihrem Alter. Die *dirty writeback*-Zeit soll auf 60s gesetzt werden.
2. Sobald die Hälfte der *dirty writeback*-Zeit vorbei ist, wird auf den nächsten Disk-Zugriff gewartet. Bei einem Zugriff oder spätestens wenn die *dirty writeback*-Zeit erreicht ist, wird der Update-Prozess aktiviert.
3. Wenn die Platte in den Standby-Modus versetzt wird, wird ein Update erzwungen. Damit soll ein Wieder-Anfahren der Platte durch den Update-Prozess vermieden werden.

Zur Implementierung:

Der Update-Demon ist in `mm/pdflush.c` und `mm/page-writeback.c` implementiert und wird vom Dateisystem mit der Funktion `wb_kupdate()` gestartet.

Als erster Schritt soll über einen Timer einmal pro Sekunde eine Funktion aufgerufen werden, die den Zeitpunkt des letzten Zugriffs auf das Laufwerk mit der aktuellen Zeit (`jiffies`) vergleicht und nach 10 Sekunden Inaktivität ein Update erzwingt (Punkt 3) und das Laufwerk in den Standby-Modus versetzt. Die Struktur `ide_drive_t` (`include/linux/ide.h`) soll dazu um die Variable `unsigned long last_access` erweitert werden. Bei einem Zugriff auf die Festplatte muss diese Variable aktualisiert werden (IDE-Treiber).

Gleichzeitig soll der Update-Demon aktiviert werden (Punkt 2), der überprüfen muss, ob modifizierte Seiten vorliegen und falls diese alt genug sind, ein Update durchführen.

Zum Testen liegen im Verzeichnis `/proj/i4embedded/akbp/harddisk/tests` Programme und Shell-Skripte, die Lese- und Schreibzugriffe durchführen. Als Programm mit periodischen Lesezugriffen liegt unter `/proj/i4embedded/akbp/harddisk/amp` ein einfacher MP3-Player.