

Zacks Kernel-News

Zack Brown

Stabübergabe für Kernel 2.6

Mit Kernel 2.6.0-test11 hat Linus Torvalds die Kontrolle über den 2.6-Tree an Andrew Morton abgetreten. Die Übergabe war schon lange geplant und Andrews Einfluss auf die Test-Releases wuchs ständig. Alle stabilen Kernelserien sind bisher an andere Maintainer übergegangen.

David Weinehall ist immer noch der offizielle Maintainer von Kernel 2.0 (auch wenn es da nicht mehr viel zu veröffentlichen gibt), Alan Cox bringt gelegentlich noch einen neuen 2.2er Kernel heraus und Marcelo Tosatti ent-

wickelt 2.4 nach wie vor aktiv weiter. Die Übergabe der stabilen Serie folgt einem Protokoll, das nach wie vor in Entwicklung ist – wie überhaupt alles bei der Linux-Programmierung.

Diesmal bestand das Experiment darin, dem nachfolgenden Maintainer den Kernel komplett zu übertragen, bevor noch die erste wirkliche 2.6er Release erschien, also Kernel 2.6.0. Bisher hat Linus immer so lange die Verantwortung behalten, bis es sein Stolz erlaubte, den neuen Kernel freizugeben. ■

Cleanroom-Entwicklung für Nvidia-Treiber

Für die Ethernet-Schnittstelle des N-Force-Chipsatzes von Nvidia entsteht derzeit mit »forcedeth« ein Linux-Treiber. Carl-Daniel Hailfinger und Andrew de Quincy haben dafür ein Reverse-Engineering des nur im Binärformat vorliegenden Treibers »nvnet« vorgenommen und daraus eine Spezifikation erstellt, ohne Hilfe von Nvidia in Anspruch zu nehmen. Diese Spezifikation diente Manfred Spraul als Ausgangsbasis für den neuen Treiber.

Die Cleanroom-Entwicklung soll mögliche Klagen wegen Copyright-Verletzungen verhindern und galt bisher als ziemlich normal. Es gibt je-

doch Bestrebungen seitens der Industrie und Lobby-Organisationen, Reverse-Engineering in einigen Ländern zu verbieten. Bisher ist es aber in den USA legal und auch in Deutschland nach Ansicht der meisten Juristen erlaubt, wenn es darum geht, Interoperabilität mit anderen Systemen herzustellen.

Der Code von »forcedeth« wurde bisher erfolgreich an N-Force2-Hardware getestet, für N-Force und N-Force3 lagen zu Redaktionsschluss noch keine Ergebnisse vor. Insgesamt ist der Treiber noch im Alphazustand, Hilfe und zusätzliche Tester sind daher willkommen. ■

Kernel 2.4 friert ein

Da Linux 2.6 inzwischen schon weitgehend benutzbar ist, hat sich Marcelo Tosatti, der Maintainer von Kernel 2.4, entschlossen, nur noch begrenzt Neues in seinen Zweig einfließen zu lassen. Bisher war es akzeptabel, für Kernel 2.6 vorgesehene Technik rückzuportieren, wenn sie dazu stabil genug war. Selbst komplett neuer Code ließ sich bis zu einem gewissen Grad noch unterbringen. Künftig sollen Änderungen an 2.4 jedoch nur noch als Bugfixes oder Security Fixes erfolgen.

So lehnte Marcelo beispielsweise die Libata-Patches von Jeff Garzik ab, obwohl sie in 2.6 schon eine sehr gute Figur machen. Vor ein paar Wochen noch wären sie wohl

auch in Kernel 2.4 gelandet. Andererseits ist der Deep Freeze dennoch nicht vollkommen komplett. Bei dem Journaling-Filesystem XFS zum Beispiel haben dessen Entwickler noch Erweiterungen vorgeschlagen, die Marcelo erst mal ablehnte. Das stieß jedoch auf massiven Widerstand auf der Kernel-Mailingliste. Marcelo ließ sich schließlich überzeugen, nachdem einige wichtige Entwickler die Patches durchgesehen hatten.

Auf jeden Fall zeigen diese beiden Beispiele, dass Marcelo Tosatti versucht einen harten Kurs zu fahren und klarzumachen, dass, wer immer neue Features will, besser Kernel 2.6 statt Kernel 2.4 einsetzen soll. ■

Windows-Treiber im Kernel

In einem radikalen Ansatz hat Pontus Fuchs damit begonnen, bestimmte MS-Windows-Treiber in Linux zu laden. Sein Treiberprojekt »ndiswrapper« versucht, direkt mit der Windows-Network-Driver-API zu kommunizieren, und zwar bis zu einem Grad, der es erlaubt, bestimmte Karten zum Laufen zu bringen.

Für Pontus schien das die einzige sinnvolle Möglichkeit zu sein, solche Hardware un-

ter Linux zu betreiben, für die die Hersteller weder Spezifikationen noch binäre Treiber für Linux bereitstellen. Mit der Broadcom4301-Karte war er bereits teilweise erfolgreich, Pavel Machek berichtete, dass auch Broadcom94306 läuft.

Es ist zwar ein etwas bizarrer Ansatz, um Treiber zu verwenden, die für ein komplett anderes Betriebssystem entstanden sind, aber möglich ist wohl alles. ■

Keine Chance für IDE-SCSI?

Im Dezember kam es zu einer Auseinandersetzung auf der Kernel-Mailingliste, die letztlich zeigte, dass Linux immer noch tief in der praktischen Welt seiner Entwickler verwurzelt ist und dass Änderungen um ihrer selbst willen wenig Chancen haben. Der IDE-SCSI-Treiber wies zwar einige Defekte auf, aber es gab nur wenige Beschwerden und niemand reichte Code ein, um ihn auf den neuesten Stand zu bringen. Zum Redaktionsschluss sah es so aus, dass der Treiber aus 2.6 entfernt wird, statt ihn einfach weiter vor sich hin dümpeln zu lassen. Bill Davidsen hat sich stark dafür eingesetzt, dass der Treiber bleibt, da es ja immerhin einige Zip-Laufwerke und Atapi-Bandlaufwerke

gebe, die auf ihn angewiesen sind. Linus entgegnete, wenn sich keiner findet, dem es wichtig genug ist, Patches zu liefern, müsse der Treiber weiter ruhen. Hier zeigt sich die Abhängigkeit der Open-Source-Entwicklung von dem Interesse der Entwickler, das sie einem bestimmten Projekt entgegenbringen. Linus mag zwar einige wichtige Aspekte steuern können, etwa Code-Qualität oder Release-Zeiten, aber er hat letztlich wenig Einfluss darauf, woran die Entwickler wirklich arbeiten. Zwar schicken einige tausend Programmierer Patches ein, aber das garantiert nicht, dass Projekte nicht doch zeitweise verwaist sind, selbst wenn die betreffende Hardware weit verbreitet ist. ■

Signalbehandlung in 2.4 und 2.6

Ein interessanter Unterschied in der Behandlung von Signalen in Kernel 2.4 und 2.6 hat kürzlich unter Linux-Entwicklern für Aufmerksamkeit gesorgt. Einige Signale sind „thread-synchron“, was bedeutet, dass ein Thread sie behandeln muss, bevor er die Ausführung fortsetzt, und dass er sie nicht blocken kann. Wenn im Kernel 2.4 ein solches Signal trotzdem geblockt wird, ignoriert es die Blockade und kommt beim Thread an. Im Unterschied dazu wird im Kernel 2.6 der Thread durch das Signal getötet. Der Grund für diese Änderung liegt darin, dass das Verhalten in 2.4 dazu führt, Bugs zu ver-

stecken, schließlich gibt es für einen Thread keinen ernsthaften Grund, ein Signal zu blockieren, das nicht blockiert werden kann. Der Entwickler entdeckt dann aber das inkorrekte Verhalten seiner Applikation möglicherweise nicht. Das Verhalten von Kernel 2.6 führt hingegen dazu, dass der Anwender oder Entwickler alarmiert wird, wenn ein Problem vorliegt. Linus Torvalds betrachtet die neue Vorgehensweise als die korrektere von beiden. Es ist aber möglich, zum alten Verhalten zurückzukehren, und zwar durch das Setzen eines »SA_NODEFER«-Flag in der »signal()«-Funktion. ■

Testsoftware für den Kernel

Software für automatisierte Tests des Linux-Kernels gewinnt immer mehr an Bedeutung, vor allem für die Industrie, die von vornherein stabile, einsatzbereite Linux-Versionen braucht. Deshalb werden viele Testprojekte, auch die beiden hier vorgestellten, von der Industrie initiiert oder gefördert. Cliff White hat mit anderen Entwicklern des Open Source Developmet Lab (OSDL) eine so genannte Tinderbox (Zunderdose oder Pulverfass) für den Kernel [<http://tinderbox.osdl.org>] bereitgestellt. Das Projekt leitet sich von der Mozilla-Tinderbox ab und dient dazu, eine Software (also hier den Kernel) bei unterschiedlichen Hardware- und Softwarekonfigurationen zu testen. An den Tests sollen daher unterschiedlich konfigurierte Client-Maschinen teilnehmen, die automatisierte Tests ausführen und die Ergebnisse an einen Server melden. Der Server analysiert die Daten, stellt sie grafisch dar und baut aus den Ergebnissen al-

ler Clients eine Website zusammen, auf der die Probleme beschrieben sind. Die meisten Werkzeuge fürs Debugging erlauben es den Nutzern, bestimmte Fehler zu melden, die der Entwickler dann entsprechend priorisiert und repariert, wenn es seine Zeit erlaubt. Die Tinderbox soll die Basis der zum Test genutzten Geräte verbreitern und gleichzeitig den Organisationsgrad des Testprozesses erhöhen. Die Testsuite des Linux-Test-Projekts [<http://www.linuxtestproject.org>], die im Dezember mit einer neuen Version aufwartete, folgt einem eher klassischen Ansatz. Sie enthält inzwischen über 2000 unterschiedliche Tests für die Kernelfunktionen der Versionen 2.5 und 2.6. Neu hinzugekommen sind Tests für das Dateisystem NFS, Low Level SCSI, asynchrone I/O und für den Umgang mit Zugriffskontrolllisten (ACLs). Das Linux Test Project wird unter anderem von den Firmen IBM, SGI und Bull sowie vom OSDL gefördert. ■

CramFS ohne Maintainer

Die Entwicklung des komprimierten Dateisystems CramFS, das ursprünglich von Daniel Quinlan geschrieben wurde, ist zum Stillstand gekommen. Daniel hat vor einiger Zeit einige – zum Teil offensichtlich sinnvolle – Patches dazu eingereicht, die aber zurückgewiesen wurde. Daraufhin hat Daniel damit aufgehört. Gerüchteweise hat Alexander Viro – seit urdenk-

lichen Zeiten Maintainer des virtuellen Dateisystems – den CramFS-Code neu geschrieben, um das Projekt zu ersetzen. Bestätigt werden konnte das aber bisher nicht. Daniel Quinlan ist auch technischer Leiter der Linux Standard Base (LSB) und betreibt viele andere Projekte, vielleicht fehlt ihm zurzeit nur die Zeit, sich für CramFS zu engagieren. (uwo) ■