

Interview mit Jeff Dike



by Floris Lambrechts
<floris-(at)-linuxfocus-(dot)-org>

About the author:

In ein paar Monaten werde ich Elektrotechniker sein. Ich mag freie Software, offene Standards und nette Benutzerschnittstellen.



Abstract:

User Mode Linux, kurz UML, ist eines der vielen interessanten Features, das der neueste Linux-Kernel (2.6.x) für deinen Desktop, deinen Server oder dein mobiles Gerät mitbringt. Dieses Interview ist eine editierte Abschrift einer e-mail-Diskussion, die ich mit dem Erfinder von UML, Jeff Dike, im Januar 2003 geführt habe.

Zum Ende erwähnt David Coulson kurz ein Werkzeug, um UML-Systeme zu verwalten.

Einführung

Wie immer enthält der Linux-Entwicklungskernel, an dem Entwickler rund um die Welt arbeiten, eine ganze Masse neuer Features. Obwohl UML nur eins davon ist, so ist es doch etwas Besonderes – es hat das Potential, die Art der Verwaltung deiner Linux-Systeme zu ändern. Weil wir glauben, daß es mehr Aufmerksamkeit verdient, als es bisher erhalten hat, präsentieren wir es in einem einleitenden Interview mit keinem Geringeren als dem Projektleiter höchstpersönlich, Jeff Dike.

Jeff persönlich

LinuxFocus: *Du bist Jeff Dike, „der Verantwortliche“ für das UML-Projekt. Kannst du dich kurz vorstellen?*

Jeff Dike: Ja, du hast mich gefunden...

OK, hier ist ein kurzer Satz von Bill Stearns, meinem offiziellen Biographen:

Jeff wurde von seiner Dienstreise nach New Hampshire 1992 entlassen. Seither diente er als Freiheitskämpfer in Idaho für die Potato-Befreiungsfront, als Leiche Nr. 7 in "Tötet Smoochy" und als bleihaltiger Wäschestärker in der größten chinesischer Wäscherei in Deering, New Hampshire. Er schreibt zur Zeit den Linux-Kernel neu als eine DOS-Batchdatei.

LF: *Arbeitest du an UML (semi-)professionell, oder ist es nur ein Hobby?*

JD: Es ist mehr oder weniger ein Hobby. Ich habe viel Unterstützung von Dartmouth ISTS erhalten (www.ists.dartmouth.edu) für die Fliegenfänger-bezogene Arbeit über UML. Ich wurde auch manchmal von Firmen unterstützt, die neue Features oder Bugfixes haben wollten.

LF: *Laß uns zuerst die Grundlagen klarstellen. Nach der Lektüre der Webseite user-mode-linux.sourceforge.net verstehe ich das Konzept so: UML ist ein Linux-Kernel, der auf einem weiteren "Host-Kernel" läuft. Der UML-Kernel lebt also im Userspace und bietet aller Software, die auf ihm läuft, "virtuelle" Hardware. Stimmt das so?*

JD: Ja. UML behandelt ein Linux-System als irgendeine Plattform, auf die Linux vielleicht portiert werden kann, das ist genau wie wenn man eine nackte x86- oder PowerPC-Maschine ansieht und erwägt, Linux darauf zu portieren. Natürlich ist Linux als eine Plattform für ein Betriebssystem ganz anders als eine Hardware-Plattform, und wie UML durch seine Existenz beweist, ist die Schnittstelle der Linux-Systemaufrufe reichhaltig genug, daß ein Betriebssystem damit implementiert werden kann.

Die Hardware von UML ist virtuell in dem Sinn, daß sie alle äußeren Ressourcen durch den Host-Kernel anspricht. Normalerweise bedeutet dies, daß die von einer UML-Instanz gesehenen Geräte tatsächlich virtuell sind, z.B. eine "Festplatte", die in Wirklichkeit eine Datei auf dem Host ist. Allerdings kannst du auch eine physische Festplatte, Partition, CD-ROM oder Diskette über ihre Gerätedatei einem UML-Gerät zuordnen, und UML wird auf die darauf enthaltenen Daten zugreifen können. Es wird jedoch nicht Befehle auf maschinennaher Ebene auf dieselbe Weise an das Gerät schicken können, wie es der Treiber im Host-Kernel macht.

Es hat jedenfalls einige Arbeit in diesem Bereich gegeben, und kürzlich wurde ein Patch veröffentlicht, der es angeblich ermöglicht, herkömmliche PCI-Gerätetreiber in UML einzubauen, die dann die entsprechenden physischen Geräte steuern; ich habe mir das noch nicht angesehen.

LF: *Es läuft also wirklich im Userspace... kannst du es als normaler Benutzer starten? Kannst du es sogar mit chroot benutzen?*

JD: Zweimal ja. Ich rate sogar dringend davon ab, es als *root* zu starten, nicht weil das gefährlich ist, sondern weil es völlig sinnlos ist. Manche Nutzer können sich nicht mit der Idee anfreunden, daß sie UML als normaler Benutzer starten und darin dann zum *root* mit all seinen Rechten werden. Sie glauben, sie müßten UML als *root* starten; ich sage ihnen sehr deutlich, daß das nicht nötig ist, und wenn sie begreifen, daß es stimmt, ist es eine Lernerfahrung für sie. Ich bin erfreut, daß diese Einstellung in der UML-Nutzergemeinschaft Fuß gefaßt hat – ich muß diese Art von Erziehung kaum noch selber machen; andere Nutzer machen das für mich.

Es kann mit *chroot* aufgerufen werden, und das ist auch die empfohlene Vorgehensweise für Menschen, die UML verwenden, um darin möglicherweise feindliche Software oder Nutzer unterzubringen – als eine zusätzliche Schutzschicht für den Host, falls einer es schafft, aus UML auszubrechen.

LF: *Es gibt eine 2.4- und die neuere 2.5-Serie von UML. Kann man UML 2.5 auf einem 2.4-Host-Kernel laufen lassen und umgekehrt?*

JD: Grundsätzlich gibt es keine Abhängigkeiten zwischen der Host- und der UML-Kernelversion. Solange der Host überhaupt UML starten kann, kann er jede Version von UML starten.

LF: *Cool... ist es denn von der x86-Plattform abhängig? Läuft es z.B. auf MIPS oder PowerPC?*

JD: Nein. Es gab einen PowerPC-Port, der einmal fast voll funktionsfähig war, aber er wurde dann nicht mehr gepflegt und ist verfallen. Ports sind sicherlich möglich und sollten nicht so schwer sein. Ich würde gerne welche sehen, vor allem, wenn sie gepflegt werden.

LF: *Solange also keiner diese Arbeit wiederaufnimmt, ist sie ein Teil des Kernels, der nicht portiert ist?*

JD: Nein. UML ist ein Port. Man kann nicht sagen, daß er nicht portiert sei. UML auf eine neue Architektur zu portieren ist so ähnlich wie eine Hardware-Architektur auf einen neuen Chip in derselben Architektur zu portieren.

LF: *Ich verstehe. Wie steht es mit "echten" Ports: Wieviel Arbeit wäre es, UML auf Windows lauffähig zu machen?*

JD: Das ist bereits (fast) zweimal getan worden. Der neueste Versuch hat offenbar eine Woche Hacking gekostet. Der ältere (<http://umlwin32.sf.net>) geschah über Monate hinweg in der Freizeit einiger Leute, daher ist es schwer zu sagen, wieviel kontinuierlichem Zeitaufwand das entspricht.

Ich möchte noch hinzufügen, daß es gut für einen dieser Ports wäre, fertiggestellt und entwanzt zu werden. Es gibt viel Nachfrage und Interesse in UML, das auf Windows läuft.

LF: *Wie gefiel es dir, daß UML im offiziellen Linux-Kernel 2.5 enthalten ist?*

JD: Technisch ist das keine große Sache. Ich habe UML außerhalb der Kernelquellen mehr als drei Jahre lang gepflegt (und ich mache das weiterhin für den UML-Kernel 2.4), und es wäre kein Problem gewesen, so fortzufahren.

Die damit verbundene Anerkennung war nett. Als Linus UML in 2.5 aufnahm, sah die Welt, daß sich UML lohnt (und ein guter Teil der Welt wurde zum ersten Mal auf UML aufmerksam). Dieses "Gütesiegel" ist wohl der bedeutendste Aspekt, UML in Linus' Quellen zu haben, soweit es mich betrifft. Es kann nun nicht mehr als ein verrückter Patch am Rande einer respektablen Kernel-Entwicklung abgetan werden.

LF: *UML ist ein neues Konzept für Systemadministratoren, ein unerforschtes Gebiet. Sind die Tools zur UML-Verwaltung (wie UMLd) bereits reif – sind sie eine effektive Antwort auf die neuen Herausforderungen, die UML den Administratoren schafft?*

JD: Diese Frage sollte Bill Stearns oder David Coulson gestellt werden. Ich beschäftige mich ausschließlich mit dem Kernel und überlasse die Verwaltungswerkzeuge anderen.

LF: *David, was ist mit UMLd?*

David Coulson: UMLd ist ein Daemon, der UMLs für Nutzer und Administratoren verwaltet. Nutzern erlaubt er, ein abgestürztes UML neuzustarten oder den Kernel zu aktualisieren, ohne Zugang zu einer Kommandozeile auf der Maschine zu haben. Administratoren ermöglicht er die Kontrolle und Veränderung von UMLs, sogar die Erzeugung, ohne Kampf mit Dateien auf dem Server.

Idealerweise wäre UMLd fähig, jeden Aspekt des UML zu steuern, was er zum größten Teil tut. Weil UML

jedoch konstant weiterentwickelt wird, muß UMLd zusammen mit allem, was dem UML–Kernel hinzugefügt wird, ebenfalls gepflegt werden.

[...]

Ich habe den UMLd auf einem Netzwerk von zwei Hosts und über 30 UMLs seit einigen Monaten laufen, und ich hatte sehr wenige Probleme mit ihm. Ich baue damit alle meine UMLs, starte sie neu, führe Kernel–Updates durch und so weiter. Um für Endnutzer hilfreich zu sein, braucht UMLd dringend ein hübsches GUI–Frontend, was ich wahrscheinlich anfangen werde zu schreiben, sobald ich mit den Features von UMLd zufrieden bin und das Protokoll, mit dem UMLd mit den Clients kommuniziert, stabilisiert ist.

LF: *Ihr Jungs scheint eine Menge Spaß zu haben. Ein Beispiel ist der Wettbewerb auf UML–Maschinen „Die Administrator–Katastrophe des Monats“; ein anderes ist dieses Zitat: „UML für Spaß und Profit / Tja, vergessen wir das mit dem Profit“; (<http://uml.openconsultancy.com>). Ist UML ein Mittelding zwischen Spielzeug und Werkzeug, oder habt ihr einfach Humor ;–) ?*

JD: Ich glaube letzteres. UML hat einen Haufen Menschen zusammengebracht, die zufällig einen etwas schlagkräftigen Humor haben, daher kommen wir natürlich blendend miteinander aus :–)

Es kann auch sein, daß diese virtualisierte Welt von Universen innerhalb von Universen Menschen mit einer bestimmten Mentalität anzieht, und das könnte etwas von dem umgebenden Humor erklären.

Egal, wie ernst es ist, was wir machen, oder wie ernst die Implikationen und der Gebrauch davon sind, wir werden Spaß haben. Es wäre eine sehr trübselige Welt, wenn einen seine Arbeit nicht irgendwie amüsieren dürfte.

LF: *Habt ihr jetzt, sechzehn Monate nach den ersten Fragen, noch etwas anzumerken?*

JD: Ich möchte erwähnen, daß es einen funktionierenden x86–64–Port von UML gibt, den ich bald freizugeben hoffe.

Ich habe auch ein bißchen mehr über die Neuigkeiten und die anstehenden Änderungen von UML in mein Tagebuch geschrieben.

Fazit

User Mode Linux ist ein netter Zusatz zum Linux–Kernel, ideal zum Testen und Demonstrieren von Applikationen, Kernen und Distributionen. Damit kann auch das „virtuelle Hosting“ flexibler gestaltet werden, indem man Leuten ihr „eigenes“ Linux–Image mit *root*–Zugang gibt, ohne das Host–System zu gefährden.

Verweise [engl.]

- [Die komplette e–mail–Konversation](#) zu diesem Interview (mit zusätzlichen Informationen)
- [Jeff Dikes UML–Tagebuch](#), um zu lesen, was neu ist und in Zukunft kommt
- user-mode-linux.sourceforge.net : Homepage des UML–Projekts
- usermodelinux.org : eine Gemeinschafts–Seite von und für UML–Nutzer, mit Massen weiterer nützlicher Links

Webpages maintained by the LinuxFocus Editor team
© Floris Lambrechts
"some rights reserved" see linuxfocus.org/license/
<http://www.LinuxFocus.org>

Translation information:

en --> -- : Floris Lambrechts <floris-(at)-linuxfocus-(dot)-org>

en --> de: Viktor Horvath <ViktorHorvath/at/gmx.net>

2005-01-11, generated by lfparsr_pdf version 2.51