

Linux für Einsteiger

Teil 6

Programme hinzufügen/entfernen

Wie schon bei der Dateihierarchie am Anfang unseres Kurses besprochen, gibt es eine Systematik, wo welche Programme und Programmteile abgelegt werden sollen. Im Gegensatz zu W* gibt es dabei kein ultimatives Schema, es bleiben einige kleine, manchmal ärgerliche Differenzen bei den verschiedenen Linux Distributionen übrig. Besonders die Handhabung des „/opt“ Verzeichnisses ist verschieden, beispielsweise installiert SuSe das KDE-System in „/opt/kde“, während Debian es unter „/usr“ integriert. Ebenfalls existieren unterschiedliche Programmpaketverwaltungssysteme. Allen gemeinsam ist, dass sie ein vollständig funktionierendes System garantieren, da entweder alle Abhängigkeiten (von anderen Programmen und Bibliotheken) automatisch durch Nachinstallation aufgelöst werden oder die Installation mit Hinweis auf fehlende Teile abgebrochen wird. Natürlich lässt Linux hier manuelle Eingriffe zu, aber auch hier wieder die Warnung : Jeder sollte wissen, was er da tut!

Ein wesentlicher Unterschied zu W* ist die Möglichkeit, verschiedene Versionen von Programmbibliotheken (und auch von Programmen) gleichzeitig zu benutzen. W* Programme sind immer vollständig, d.h. alle benötigten Bibliotheken und Programmteile werden durch das Setup Programm in das System eingespielt, und da es nur ein Windows Systemverzeichnis gibt, werden vorhandene Versionen einfach überschrieben, bzw. neue nicht installiert, sodass entweder ältere Programme eventuell funktionsunfähig werden oder das neu installierte nicht arbeitet. Linux hat eine Namenskonvention, die die Version von Programmen und Bibliotheken erkennen lässt, damit können sie gleichzeitig in einem Verzeichnis stehen, ohne sich gegenseitig zu beeinflussen. In problematischen Fällen kann jedes Linux Programm auch seine eigene Verzeichnishierarchie (.../bin, .../lib, .../man, .../var) benutzen.

Jede der aktuellen Distributionen (SuSe, Mandrake, Fedora, Debian,...) hat ein auf seine speziellen Anforderungen zugeschnittenes Paketverwaltungssystem mit der Möglichkeit, textbasiert oder graphisch darauf zuzugreifen. Hierin ist ein Nachteil offensichtlich, es müssen nämlich für jede Version angepasste Programmpakete bereitgestellt werden.

Was aber, wenn es keine fertigen Pakete für ein benötigtes Programm gibt?

Dann ist die manuelle Installation von entweder passend vorkompilierten binären Programmen oder aus den Quelltexten selbst kompilierten Programmen erforderlich. Diese Programme können nicht von der distributionseigenen Paketverwaltung bearbeitet werden, daher ist

die Installation im Zweig „/usr/local...“ empfehlenswert, weil hier ohne-
hin keine automatische Verwaltung stattfindet.

Es hängt von der Güte der Programmerstellung des Programmierers ab,
ob die Installation problemlos erfolgen kann, meistens ist das Paket als
„.tgz“ oder „.tar.gz“ gepackt. Nach dem Entpacken gibt es mehrere
Textdateien (Readme, Install u.a.) die gelesen werden sollten. Dort sind
Hinweise und Voraussetzungen für die Installation angegeben. Die Über-
setzung aus den Quelltexten und die Installation erfolgt nach dem Linux-
Dreisatz (configure, make, make install). Im Konfigurationsschritt
(Wechsel in das Verzeichnis, dass „configure“ enthält und Aufruf mit
„./configure“) werden benötigte Programme abgeprüft und Systemeinstell-
ungen für die Übersetzung vorgenommen, hier können auch Pa-
rameter zur manuellen Beeinflussung angegeben werden. Wenn die
Konfiguration ohne Fehlermeldung durchgeführt wurde, erfolgt im
nächsten Schritt der Aufruf von „make“, welches den eigentlichen Über-
setzungslauf startet. Er kann je nach Umfang des Programms und Rech-
nergeschwindigkeit einige Zeit benötigen. Auch hier können noch
Fehlermeldungen auftauchen, die nicht durch die vorherige Konfigurati-
on aufgefangen werden konnten. Nach fehlerfreier Abarbeitung erfolgt
im nächsten Schritt die eigentliche Installation der erstellten binären Da-
teien durch Aufruf von „make install“. Dies sollte mit „root“-Rechten er-
folgen, da in Verzeichnisse geschrieben werden soll, die der normale
Nutzer nicht beschreiben darf. Meistens funktionieren auch die Befehle
„make clean“ (löscht alle durch „./configure“ und „make“ erstellten Da-
teien) und „make uninstall“ (macht die Installation rückgängig).

Es gibt auch ein Hilfsprogramm, dass selbstkompilierte Programme in
das Verwaltungssystem integriert, dazu wird im dritten Schritt das Pro-
gramm „checkinstall [options]“ aufgerufen. Mit Hilfe der Optionen
können zum System passende Pakete gebildet werden (auf unserem De-
bian System ist die Option „-D“ nötig).

Zurück zu den vorgefertigten Programmpaketen:

Es gibt zwei verschiedene Packformate mit den Endungen „.deb“ (Debi-
an basierte Distributionen) und „.rpm“ (z.B. SuSe, Mandrake, Fedora...).
Betrachten wir erst einmal die textbasierte Version:

Das Kommando für die Installation eines Pakets (als Beispiel „pro-
gramm1“) lautet:

„dpkg -i programm1.deb“ (für Debian)

„rpm -i programm1.rpm“ (für rpm-basierte Distributionen (rpm=RedHat
Package Manager))

Die Deinstallation erfolgt mit:

„dpkg -r programm1.deb“ bzw. „rpm -e programm1.rpm“

Mit einem derartigen Aufruf zur Installation eines Programms werden nicht automatisch alle Abhängigkeiten aufgelöst, sondern Fehlermeldungen erzeugt. Der Nutzer muss alle Abhängigkeiten selbst auflösen.

Auf die Dauer ist das sehr unbefriedigend, daher gibt es verbesserte Möglichkeiten zur Programminstallation, zumal die Abhängigkeiten bereits in den Paketen beschrieben sind. Unter Debian Linux gibt es den Befehl „apt-get“, der Aufruf „apt-get install programm1“ installiert das Programm und löst dabei alle Abhängigkeiten automatisch auf. Eine derartige Möglichkeit gibt es unter rpm-basierten Systemen noch nicht so perfekt (apt-get wird gerade für SuSe angepasst), hier ist die graphische Paketverwaltung besser, z.B. Yast, dort werden die Abhängigkeiten dadurch gelöst, dass die Pakete in der benötigten Reihenfolge installiert werden (zuerst das Pgm ohne weitere Abhängigkeiten, dann die darauf aufbauenden Pgm). Voraussetzung ist die Verfügbarkeit entsprechender Pakete.

Bei unserem Debian System ist für graphische Bedienung das Programm „Synaptic“ erste Wahl. Wie bei allen derartigen Programmen ist die Graphik nur das Frontend, die eigentlichen Befehle im Hintergrund sind die oben besprochenen.

Woher bekommt das Verwaltungsprogramm Informationen über die installierten und verfügbaren Programmpakete ?

Die installierten Programme werden in einer Datenbank gehalten. Die verfügbaren Programme sind ebenfalls in einer Datenbank, die aus dem Internet aktualisiert wird, damit ist gewährleistet, dass bei der Abhängigkeitsprüfung immer auf die neuesten und verfügbaren Versionen zurückgegriffen werden kann. Hier muss genau auf den passenden Aufruf geachtet werden, ein „update“ ist hier keine Programminstallation, sondern die Auffrischung der Programmdatenbank, erst der Aufruf von „upgrade“ bzw. „dist-upgrade“ aktualisiert das Linux System mit seinen Programmen.