

Vorwort zur 1. und 2. Auflage

Obwohl heute Middleware-Systeme zur Verfügung stehen, die in vielen Fällen unabhängig von einem darunter laufenden Betriebssystem sind, ist die Betriebssystemthematik aus der Informatik-Grundausbildung aus verschiedenen Gründen nicht wegzudenken. Ohne Kenntnisse der hauptsächlichen Strukturen, Mechanismen und der Programmierschnittstelle eines Betriebssystems ist es nicht möglich, spezielle Systemdienste zu benutzen oder die Effizienz eines Systems zu optimieren. Es ist ein altes Ideal des Software Engineering, dass die hinter einer Schnittstelle stehende Implementierung als solche unwichtig ist und daher ignoriert werden soll. Auf Betriebssysteme bezogen würde dies heißen, nur die Programmierschnittstelle zu betrachten. Leider sind die dazugehörigen Beschreibungen praktisch immer minimal und befassen sich kaum mit den Konzepten der dahinter stehenden Implementierungen. Dies erschwert nicht nur ein tieferes Verständnis der Systemdienste, sondern kann auch zu einer inadäquaten Nutzung in komplexen Applikationen führen. Dieses Buch will Studierenden der Informatik und weiteren interessierten Personen die Grundlagen der Betriebssystemtheorie aus einer praktischen Perspektive näher bringen. Damit ist gemeint, dass nicht nur die Prinzipien von Betriebssystemen, sondern auch deren Nutzung bei der systemnahen Programmierung aufgezeigt werden. Dieser Ansatz entspricht der Idee des Bachelor-Studiums, das eine Berufsbefähigung nach drei Studienjahren anstrebt und die forschungsorientierte Ausbildung in die Master- und Doktoratstufe verschiebt.

Methodisch wird ein Weg zwischen der Betrachtung anfallender Probleme und ihren Lösungen auf einer theoretischen und einer praktischen Basis beschritten. Der Praxisbezug orientiert sich an den zwei am meisten verbreiteten Systemwelten, nämlich Unix und Windows. Kenntnisse der Prozesstechnik werden keine vorausgesetzt. Wo nötig werden die wichtigsten Prozessorgrundlagen erklärt, soweit sie für das Verständnis des Betriebssystems und der systemnahen Programmierung hilfreich sind. Die zahlreichen Beschreibungen von Systemfunktionen dienen dazu, die Programmbeispiele genau zu verstehen. Dies ist im Zeitalter der Java-Programmierung umso wichtiger, da eine gründliche Ausbildung in

der Programmiersprache C nicht mehr vorausgesetzt werden kann, wenn auch C-Grundkenntnisse für dieses Buch notwendig sind. Im Weiteren verschaffen diese Beschreibungen einen Einstieg in die Systemdokumentationen, wie sie in Form der Unix-Handbuchseiten oder der MSDN-Beschreibungen (Microsoft Developer Network) für Windows vorliegen und erfahrungsgemäß für Neulinge nur schwer verständlich sind. Auf die Betrachtung von Computernetzen wird weitgehend verzichtet, da dieses Thema in einem Betriebssystembuch nur oberflächlich gestreift werden kann und ausgezeichnete Standardwerke zur Verfügung stehen. Dafür werden als Ergänzung die aktuellen Themen Multiprozessorsysteme, Handheld-Betriebssysteme und Virtualisierungstechnologien vorgestellt.

Als Hilfe für Dozierende stehen auf der Buch-Website <http://unix.hsr.ch> Übungsaufgaben mit Lösungen, alle Abbildungen des Buches und etliche Vorlesungsfolien in elektronischer Form zur Verfügung. Abschließend bedanke ich mich bei allen Personen, die mir bei der Realisierung dieses Buchprojektes geholfen haben.

Eduard Glatz
Urdorf, im August 2005

Neuheiten in der zweiten Auflage

Eine neue Auflage erlaubt nicht nur das Aktualisieren schnell veralteter Informationen, sondern auch das Einbringen von Erfahrungen und zusätzlichen Themen. Aktualisiert haben wir die Informationen zu Windows 7, Windows CE und Symbian OS. Die ausführlichen tabellarischen Beschreibungen zu den Systemfunktionen sind nun nicht mehr im Buch, sondern in einer PDF-Datei zusammengefasst auf der Buch-Website (<http://unix.hsr.ch>) verfügbar. Auf vielfältigen Wunsch wurde das Literaturverzeichnis ausgedehnt. Neue Themen betreffen u.a. die Linux-basierten Smartphone-Betriebssysteme Android, WebOS und Maemo, die Systemprogrammierung aus C++, Java und .NET-Sprachen, die Bauweise von SSD (Solid State Disks), spezielle Dateisystemtechnologien (Schattenkopie, Disk Scheduling) und die Vermeidung von Synchronisationsengpässen.

Eduard Glatz
Urdorf, im Februar 2010