

Zeitschrift: Comtec : Informations- und Telekommunikationstechnologie = information and telecommunication technology

Herausgeber: Swisscom

Band: 79 (2001)

Heft: 3

Artikel: Echtzeitbetriebssysteme auf dem Vormarsch

Autor: Stautner, Heike

DOI: <https://doi.org/10.5169/seals-876526>

Nutzungsbedingungen

Die ETH-Bibliothek ist die Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Zeitschriften und ist nicht verantwortlich für deren Inhalte. Die Rechte liegen in der Regel bei den Herausgebern beziehungsweise den externen Rechteinhabern. [Siehe Rechtliche Hinweise.](#)

Conditions d'utilisation

L'ETH Library est le fournisseur des revues numérisées. Elle ne détient aucun droit d'auteur sur les revues et n'est pas responsable de leur contenu. En règle générale, les droits sont détenus par les éditeurs ou les détenteurs de droits externes. [Voir Informations légales.](#)

Terms of use

The ETH Library is the provider of the digitised journals. It does not own any copyrights to the journals and is not responsible for their content. The rights usually lie with the publishers or the external rights holders. [See Legal notice.](#)

Download PDF: 26.04.2025

ETH-Bibliothek Zürich, E-Periodica, <https://www.e-periodica.ch>

Desktop oder Embedded?

Echtzeitbetriebssysteme auf dem Vormarsch

Jeder ist es wohl gewohnt, dass sein Windows-System mindestens einmal in der Woche abstürzt. Dabei sind heute offene, nicht gespeicherte Dateien kein allzu grosses Drama mehr, da es für die Sicherung inzwischen die richtigen Lösungen gibt. Doch wie sieht es mit Non-PC-Devices aus? Kann man es Anwendern wirklich zumuten, ihr Handy, die Set-Top-Box oder das Navigationssystem im Auto ständig neu zu booten und dabei auf nicht gesicherte Dateien zu verzichten?

Ganz sicher nicht, denn es geht auch anders. Vor allem im Bereich Consumer Electronics und Telekommunikation denken Hersteller inzwischen um und prüfen den Einsatz so

HEIKE STAUTNER

genannter Echtzeitbetriebssysteme, die ihre Zuverlässigkeit bereits in einer Reihe von Embedded-Systemen wie in der Medizintechnik, industriellen Automatisierung und Automobilindustrie unter Beweis gestellt haben.

Die Wahl des richtigen Betriebssystems

Angesichts der Anforderungen seitens der Hersteller von Internet Appliances und Konsumelektronik sowie der Anwender scheint die Zeit nicht nur reif für Echtzeitbetriebssysteme, sondern auch für komplette Echtzeitplattformen, wie sie vor kurzem QNX vorgestellt hat. Die Frage ist, inwieweit eignen sich herkömmliche Betriebssysteme für die Embedded-Welt und welche Voraussetzun-

gen muss ein Echtzeitbetriebssystem haben, um sich in Zukunft neue Märkte zu erschliessen. Die Entwicklung wird dabei nicht durch die Wünsche seitens der Anwender vorangetrieben, sondern eindeutig von den Herstellern diktiert. Der Wettbewerbsdruck ist gross, sodass sie gezwungen sind, Systeme mit immer mehr Features in immer kürzeren Abständen auf den Markt zu bringen. Mehr Funktionalität bei gleichbleibender Benutzerfreundlichkeit und demselben Look-and-Feel sowie eine schnellere Time-to-Market bestimmen derzeit den Trend in der Embedded-Entwicklung. Drahtlose Kommunikation, Web-Zugang, Spracherkennung, E-Commerce und andere Online Services werden in den kommenden Monaten und Jahren zum Stan-

dard werden, was Hersteller von Konsumelektronik- und Mobilfunkgeräten über kurz oder lang mit einem ernsthaften Problem konfrontieren wird: Die Wahl des richtigen Betriebssystems.

Wer macht das Rennen?

Der Markt ist eindeutig zweigeteilt, wobei die Frage ist: Wer macht das Rennen? Die Desktop-Betriebssysteme wie Windows, BeOS oder Linux, die in den Embedded-Markt drängen oder die angestammten Echtzeitbetriebssysteme wie QNX? Herkömmliche Betriebssysteme haben einen entscheidenden Nachteil, da sie nicht speziell für die Anforderungen von Embedded-Systemen wie PDAs, Web Pads oder Mobilfunkgeräten entwickelt worden sind. So sind sie weder

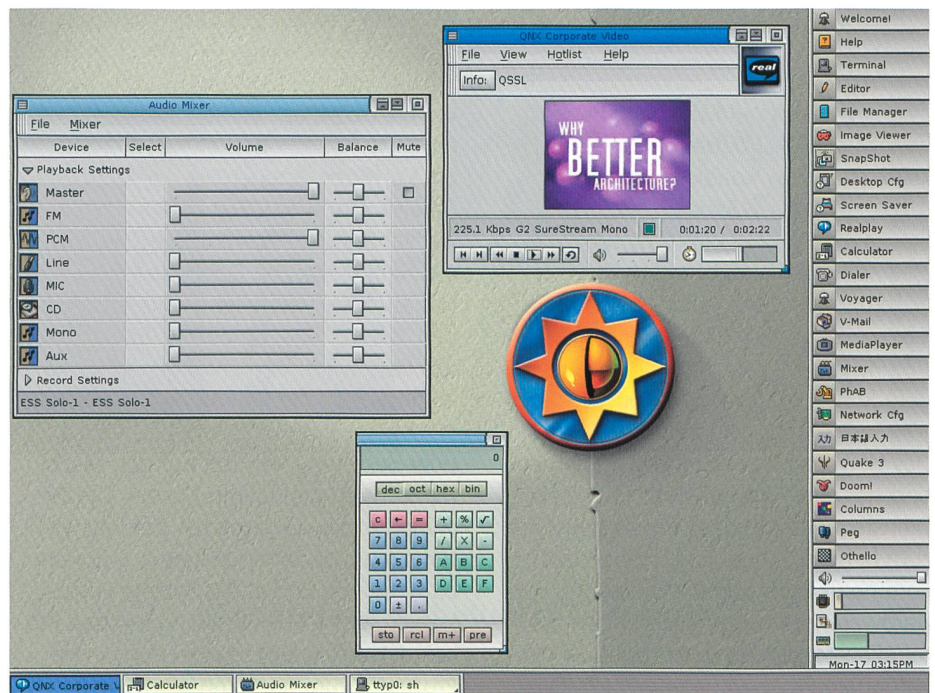


absolut zuverlässig noch echtzeitfähig, noch kommen sie mit begrenztem Speicherplatz aus. Ihre APIs lassen sich weder kundenspezifisch anpassen, noch helfen sie OEM-Herstellern bei der Verkürzung der Time-to-Market. Microsoft hat dies bereits vor längerem erkannt und versucht, mit Windows CE gegenzusteuern, was bis jetzt nicht sehr erfolgreich gelungen ist. Linux ist in aller Munde, doch sind sich die Hersteller noch nicht im Klaren darüber, wie Linux zu verkleinern ist, um für die Embedded-Entwicklung in Frage zu kommen. Eine Standardisierung wird hier sicherlich noch etwas auf sich warten lassen.

Doch gerade im Bereich Consumer Electronics geht die Entwicklung rasant voran. Immer neue Geräte drängen auf den Markt und buhlen um die Gunst des kaufkräftigen Anwenders. Produktlebenszyklen werden gerade in diesem Markt immer kürzer und als Folge damit auch die Entwicklungszyklen. Dan Dodge, CEO von QNX Software Systems, kennt die Wünsche und Probleme der Gerätehersteller: «Jeder Konsumelektronik-Hersteller, der zu uns kommt, hätte sein Produkt am liebsten bereits gestern. Inzwischen ist die Zeit zwischen Produktentwurf und Marktreife so kurz geworden, dass unsere Kunden nicht mehr in eigene, grosse Entwicklungsabteilungen investieren wollen. Sie sind auf der Suche nach einer fertigen, umfassenden Lösung, die ihnen die Funktionalitäten und Möglichkeiten eröffnet, die sie brauchen. Mit einem «nackten» Betriebssystem kommt man gerade in diesem Markt nicht mehr weiter.»

Alles aus einer Hand

Der Trend geht eindeutig hin zur One-Stop-Lösung. Hersteller haben einfach nicht mehr die Zeit, die entsprechenden Komponenten bei einzelnen Anbietern einzukaufen und dann viel Mühe und Geld zu investieren, um sie aufeinander abzustimmen. Gefragt ist eine komplette Lösung, die sich bereits im Embedded-Bereich bewährt hat und die alle Funktionalitäten bietet, um Internetzugangsgeschäfte, Home und Information Appliances schnell zu entwickeln. Dabei ist der Ansatz inzwischen ein völlig anderer als noch vor ungefähr ein bis zwei Jahren. Damals konzentrierten sich alle Anstrengungen auf Set-Top-Boxen und die Zusammenführung von Fernsehen und PC. Internetzugang über das TV-Gerät sollte um jeden Preis realisiert werden. Inzwi-



Die umfassende Multimedia-Ausstattung der QNX Realtime Platform enthält unter anderem Plug-Ins für DVD, Dolby Digital (AC-3), MP3, RealPlayer 7, Macro Media Flash 4, MPEG IIII und CD Audio. Bild: QNX Software Systems

schen hat sich gezeigt, dass dieser Ansatz der falsche und in der Praxis nicht zu realisieren ist. Der Grund dafür ist einfach: Während TV eine Push-Technologie ist, handelt es sich beim Internet um eine Pull-Technologie, was sich nicht vereinbaren lässt. Hersteller haben daraus gelernt. Inzwischen versuchen sie nicht mehr nur Devices für das «Surfen im Web» fit zu machen, sondern das Internet als Tool zu nutzen.

Gefragt ist inzwischen die komplette Funktionalität und Vielfältigkeit des Internets, sodass Grenzen zwischen den einzelnen Geräten verschwimmen. Das bedeutet beispielsweise, dass aus einer Set-Top-Box auch ein MP3-Player werden kann. Herkömmliche Produkte sollen nicht mehr nur um Web-Funktionalität ergänzt werden, sondern echte Service und Entertainment Appliances werden, wobei Web Browsing und E-Mail zunächst sicherlich eine der häufigsten Funktionen sein werden. Ziel ist es, die Integration dieser Geräte zu Hause zu erreichen und die Abstimmung mit Handheld-Geräten, mit denen der Anwender arbeitet. Dabei steht für die Zukunft das «Share» von Informationen und die Interoperabilität mit der Hauselektronik im Vordergrund.

Die Realisierung dieser multimedialen Projekte setzt nicht nur ein Echtzeitbetriebssystem voraus, das alle Funktionen

und Applikationen des Systems simultan kontrolliert und steuert, sondern eine umfassende Echtzeitplattform einschliesslich Media-Player für DVD, MP3, MPEG1 (VCD & VCD2), MPEG2, Real-Audio/Video und CD Audio. Integriert sein sollten Internetapplikationen wie Web-Browser, Plug-Ins und E-Mail-Client ebenso wie 2-D- und 3-D-Hardwarebeschleuniger für Spiele, Spracherkennungs- und Video-Conferencing-Software sowie Treiberunterstützung für USB-Geräte, High-End-Audio-Geräte und Drucker. Entscheidend dabei ist, dass alle diese Komponenten aufeinander abgestimmt sind und der Gerätehersteller die Möglichkeit hat, die einzelnen Interfaces kundenspezifisch anzupassen. In jedem Fall soll das Produkt sein eigenes Marken-Design tragen. Ein Problem, mit dem derzeit Lizenznehmer von Windows CE kämpfen, da mit dem Start des Device unfreiwillig auch das Microsoft-Logo erscheint. Eine Werbung, die sicherlich nicht im Sinne aller OEM-Hersteller ist. Wichtig für Entwickler von Embedded-Systemen ist die offene Architektur des Betriebssystems, sodass der Support von Standard APIs problemlos möglich ist. Dies zeichnet natürlich auch eine Reihe herkömmlicher Betriebssysteme aus, doch ist es schwierig beziehungsweise unmöglich, dem Embedded-System Treiber und Applikationen im laufenden

Betrieb hinzuzufügen, upzugraden oder zu entfernen, ohne das System zu re-booten oder neu zu testen. Hier liegt der Vorteil eindeutig auf der Seite POSIX-basierter Echtzeitbetriebssysteme wie QNX, das zudem dank seiner UPM (Universal Process Modell)-Architektur Speicherschutz für sämtliche Software-Komponenten einschliesslich Betriebssystemmodule und Treiber gewährleistet. Der grosse Vorteil liegt in der Rationalisierung des Entwicklungszyklus, was gerade für Hersteller von Konsumelektronik-Geräten entscheidend ist.

Das Universal-Prozessmodell implementiert nur einen kleinen Satz essenzieller Dienste innerhalb des Kernels selbst wie zum Beispiel Scheduling, Interprozesskommunikation und die Umleitung von Hardware-Interrupts. Alle übrigen Systemdienste werden durch optional hinzufügbare Prozesse bereitgestellt. Das führt dazu, dass jeder Treiber, jedes Protokoll, jedes Dateisystem, jeder I/O-Manager und jedes Grafiks subsystem in seinem eigenen speichergeschützten Adressraum laufen kann. Dass diese Architektur die Zuverlässigkeit erhöht, ist offensichtlich. Erstens enthält der Betriebssystemkernel nur wenig Programmcode, der Fehler verursachen könnte. Zweitens ist es sehr unwahrscheinlich, dass irgendein Modul – auch ein schlecht programmiertes mit höchstem Zugriffsrecht – den Kernel korrumpieren könnte. Darüber hinaus läuft nun jedes Modul als unabhängiger Prozess. Das bedeutet, dass sich jeder Teil des Softwaresystems beliebig starten, stoppen, verändern oder upgraden lässt, ohne einen Neustart durchführen oder den Kernel neu erstellen zu müssen.

Das Device als kritische Grösse

Gerade bei Geräten des täglichen Lebens oder der Unterhaltungselektronik ist Zu-

verlässigkeit das entscheidende Kriterium, das über Erfolg und Misserfolg entscheidet. Anders als der Desktop-Bereich, wo die Dominanz von Microsoft ungebrochen ist und nur wenige Alternativen zur Verfügung stehen, ist der Embedded-Bereich sehr offen. Erst in den nächsten vier Jahren wird sich entscheiden, wer das Rennen macht. Dabei werden sicherlich auch die Hersteller die Nase vorn haben, die auf ein Betriebssystem setzen, das Portabilität und Erweiterbarkeit gewährleistet. Ständig kommen neue Prozessoren mit einer höheren Performance oder besseren Integrität auf den Markt, meistens zu geringeren Kosten. Die Architektur des eingesetzten Betriebssystems muss es daher erlauben, Software zu schreiben und in das System einzuspielen, ohne über den eingesetzten Prozessor nachzudenken oder von einem Prozessor zum nächsten wechseln zu können, ohne dabei die Software gross anpassen zu müssen. Java kann in diesem Fall einen gewissen Grad an Unabhängigkeit bieten. Jedoch sind heutzutage viele Komponenten, gerade Treiber oder Multimedia-Applikationen, zugunsten der Performance in C geschrieben. Die Betriebssystemarchitektur sollte also eine einfache Portabilität auf andere Prozessoren zulassen.

Ein anderes Dilemma beanspruchen speziell «All-in-one»-Entertainment-Systeme für sich. Einerseits sollen Applikationen wie Internet-Browser oder E-Mail-Client regelmässig auf den neusten Stand gebracht werden, um den aktuellen Standards zu genügen. Andererseits sind gerade die Benutzer solcher Systeme keine Computerspezialisten, das heisst, sie sind abgeneigt oder einfach nicht in der Lage, neue Software selbst zu installieren. Die Erweiterung oder das Updaten des Systems muss automatisch passieren. Dem Hersteller muss es möglich sein, auf

Wunsch den neuesten Web-Browser zu installieren und den alten zu ersetzen oder der Benutzer muss sich einfach den neuesten Browser von einer Webseite laden können, wobei die Installation völlig automatisch abläuft. Dasselbe gilt für neue Hardware. Nehmen wir an, der Benutzer schliesst einen neuen Drucker an. Ist noch kein Treiber für diesen Drucker installiert, so sollte das System diesen Drucker erkennen und von einer Remote-Webseite einen neuen Treiber laden und installieren. Dies alles ohne Eingreifen des Benutzers.

Anwenderfreundlichkeit und Flexibilität sind für den Erfolg kommender Geräte entscheidend. Kommunikation darf nicht mehr nur eingeschränkt möglich sein, sondern muss auch über die unterschiedlichen Plattformen und Systeme verschiedener Hersteller hinweg funktionieren. Im Idealfall sollte der Anwender beispielsweise über sein Mobilfunkgerät auch seinen PDA, Fernseher oder Kühlschrank ansprechen können, ohne dabei mühsam bestimmte Konfigurationen vorzunehmen. Dieses Szenario kann allerdings nur Realität werden, wenn das eingesetzte Betriebssystem die Möglichkeit zu einem verteilten und transparenten Netzwerk mitbringt, sodass jede neue Applikation oder Plattform existierende Ressourcen automatisch erkennt. Damit sind die Anforderungen an zukünftige Betriebssysteme identifiziert: Zuverlässigkeit, Stabilität, Portabilität, Erweiterbarkeit, Netzwerktransparenz und kundenspezifische Anpassungsfähigkeit. Voraussetzungen, die sicherlich auch bei der Implementierung zukünftiger Technologien wie Wireless Connectivity, Spracheingabe und Video-Conferencing in Embedded-Systeme von Nutzen sein werden. Device-Hersteller werden in Zukunft genau abwägen müssen, über welche Funktionalitäten ihre Geräte verfügen sollen, wie lange die Entwicklungszeiten sein sollen und zu welchem Preis sie ihr Produkt anbieten wollen. Bestimmte Appliances werden mit dem einen oder anderen herkömmlichen Betriebssystem sicherlich gut beraten sein. Doch gerade bei Konsumelektronikgeräten geht der Trend eindeutig hin zur Echtzeitplattform. 7

Summary

Desktop or embedded? Real-time Systems take off

Windows users are accustomed to their systems crashing at least once a week. But thanks to back-up solutions, open, unsaved files no longer pose such a major problem. What about non-PC devices? Can users really be expected to repeatedly reboot their mobile phone, set-top box or car navigation system and accept the loss of unsaved files?

Heike Stautner, freie Journalistin,
München, E-Mail: hmstau@gmx.de



Convincing cabling solutions

Welche steht Ihnen am besten?



Gebäudeverkabelungs-Lösungen für Ihre Bedürfnisse

Das R&M freenet LAN-Verkabelungssystem für Sprach-, Daten- und Videoanwendungen erfüllt höchste Ansprüche. Es ist in drei verschiedene Lösungen unterteilt, die jeweils für das entsprechende

Kundenbedürfnis das optimale Angebot darstellen. Das

R&M CLASSICsystem erfüllt alle heutigen Standards auf Basis Kat. 5e und bietet ein optimales Preis-

Leistungs-Verhältnis. Das R&M STARsystem basiert auf

dem Kat. 6 Standard und ist für anspruchsvolle Kundensegmente interessant, die höchsten Wert auf die Zukunftssicherheit ihrer IT-Infrastruktur legen. Mit dem R&M VISIONsystem

gehen wir absolut neue Wege. Als Alternative zur klassischen universellen Gebäudeverkabelung bieten wir eine «Single Point of Administration»-Infrastruktur, die Platz sparend auf Etagenverteiler verzichtet und alle Daten mit Glasfaser- oder Kupferleiter von einem

zentralen Punkt im LAN direkt an jeden Arbeitsplatz bringt. Interessiert? Dann rufen Sie uns doch einfach an!

zentralen Punkt im LAN direkt an jeden Arbeitsplatz bringt. Interessiert? Dann rufen Sie uns doch einfach an!

Interessiert? Dann rufen Sie uns doch einfach an!

R&M *freenet*

The LAN solution

Reichle & De-Massari AG

Binzstrasse 31

CH-8622 Wetzikon

Telefon +41 1 931 97 77

Telefax +41 1 931 93 29

www.rdm.ch

Info 01 931 97 77