

## Projektarbeit



**Gruppe :** Widmer Christian

Peter Saredi

**Ort, Datum:** TSU Uster, 2003

*Dok. Version:* V 1.00

## Inhaltsverzeichnis

<b>0</b>	<b>EINFÜHRUNG.....</b>	<b>3</b>
<b>1</b>	<b>WAS IST WINDOWS CE.....</b>	<b>4</b>
1.1	UNTERSCHIEDE ZWISCHEN DEN VERSIONEN .....	5
1.2	BEISPIEL EINGESETZTER PRODUKTE.....	6
<b>2</b>	<b>ARCHITEKTUR.....</b>	<b>7</b>
2.1	AUFBAU .....	7
2.2	SPEICHERVERWALTUNG.....	8
2.3	PROZESSVERWALTUNG.....	9
2.4	GUI.....	9
2.5	NETZWERKANBINDUNG .....	9
2.6	PERIPHERIE .....	10
2.7	WINDOWS CE BAUKASTEN.....	10
2.8	VERSIONEN.....	10
<b>3</b>	<b>ENTWICKLUNGSOBERFLÄCHEN .....</b>	<b>11</b>
3.1	EMBEDED VISUAL STUDIO 3.0 .....	11
<b>4</b>	<b>ZUKUNFT.....</b>	<b>13</b>

## 0 Einführung

Im Rahmen der Projektarbeit im Fach Betriebssysteme haben wir uns für das Windows-CE Betriebssystem entschieden.

Der Grund für diesen Entscheid liegt darin das uns dieses Thema interessiert und wir in betracht ziehen die abschliessende Diplomarbeit im Zusammenhang mit einem PDA zu schreiben.

Darum haben wir auch die Entwicklungsoberflächen unter die Lupe genommen.

Um den Umfang in Grenzen zu halten haben wir uns in der Architektur nur auf das notwendigste beschränkt.

Diese Arbeit hat uns weitere Möglichkeiten in Bezug auf die Anwendungsmöglichkeiten vom Windows-CE aufgezeigt.

Wir sind der Meinung, dass es in Zukunft mehr Personal Digital Assistent (PDA) und andere mobile „Alleskönner“ die auf Windows CE .Net basieren geben wird.

Denn die Menschheit möchte immer mehr von Unterwegs aus alles erledigen können. Ebenso möchte man nur ein Gerät verwenden, dass immer kleiner und vielseitiger wird.

## 1 Was ist Windows CE

Microsoft® Windows CE ist ein modulares 32-Bit-Betriebssystem und wurde vor allem für mobile Endgeräte wie z.B. PDAs, Webpads und Handhelds, aber auch Spielekonsolen, Mobilfunkgeräte und Smartphones entwickelt.

Dank seiner Multiple-Platform- und Echtzeitfähigkeit wird es in Geräten mit unterschiedlichen Hardware-Plattformen und somit auch im Embedded-Bereich eingesetzt.

Microsoft Windows CE gibt es derzeit in verschiedenen Ausprägungen wie z.B. den Pocket PC (Windows CE 3.0), den Handheld PC 2000 (Windows CE 3.0) und Windows CE embedded.

Der wesentliche Unterschied zu anderen Betriebssystemen der Windows-Familie, sind die geringen Anforderungen an die Hardwareressourcen. So sind für einen Betrieb von Windows CE in der Minimalkonfiguration nur 28kB RAM und 330kB ROM notwendig.

Windows CE ist ein Betriebssystem für mobile Endgeräte. Nach mehreren erfolglosen Versuchen, im Bereich der PDAs (**P**ersonal **D**igital **A**ssistent) Fuß zu fassen, gelang es Microsoft nun, Windows CE auf dem Markt zu positionieren und sich auch im Bereich des Mobile Computing zu etablieren. Der Name CE, der für **C**onsumer **E**lectronics steht, zeigt schon die Zielrichtung: neben den schon erwähnten PDAs auch Spielkonsolen, Settop-Boxen und Kommunikationsgeräte.



PDAs, oder **pocketPCs**, wie es bei Microsoft heißt, sind meistens stiftgesteuerte Organizer. Die Eingabe geschieht durch Schrifterkennung oder eine virtuelle Tastatur auf dem Touchscreen. Aufgrund der grafischen Oberfläche ist ein Farbbildschirm beinahe unabdingbar.



Als **handheldPC** bezeichnet Microsoft Geräte mit einer Tastatur. Man bedient sie entweder wie die pocketPCs über einen Touchscreen, der mit einem Stift bedient wird, oder ein Touchpad. Die Geräte verfügen über keine Festplatte.



Der Einsatz von Win CE in **Spielkonsolen** ist angestrebt, bis jetzt aber selten. Microsofts eigenes Gerät in diesem Marktsegment, die Xbox (Bild), wird nicht mit Windows CE, sondern mit der embedded-Version von Windows 2000 betrieben.

Windows CE ist weiterhin für den Gebrauch in **embedded-Systemen** gedacht. Embedded-Systeme sind Ein-Chip-Geräte, bei denen alle Komponenten wie Hardware-Timer, I/O-Ports, Konverter etc. auf einem Chip integriert sind. Sie arbeiten autonom, also unabhängig vom Datenfluss anderer Komponenten (Grafik, Sound, o.ä.), und zeichnen sich durch ihre Rechengenauigkeit aus. Die Chips sind meistens eher gering getaktet und recht günstig in der Herstellung.

Mögliche Einsatzgebiete sind Videorekorder, Waschmaschinen, Mobiltelefone oder in der Medizintechnik. Siemens hatte sich sogar den Quellcode von Windows CE gesichert, um es als Plattform für alle Hausgeräte von der Kaffeemaschine bis zum Kühlschrank zu nutzen. Diese Pläne wurden aber noch nicht umgesetzt. In Kürze soll Windows CE sogar in Autoradios mit Routenplaner- und Organizerfunktionalität eingesetzt werden. Microsoft versucht, mit Windows CE einen weiteren Schritt in die Richtung „windows everywhere“ zu gehen. So soll auch im tragbaren Bereich ein Betriebssystem von Microsoft DIE Standardlösung werden. Die besonderen Vorteile gegenüber anderen mobilen Geräten liegen zum einen in der einheitlichen Windows-Entwicklungsumgebung, zum anderen sind die Geräte von vorneherein für die Zusammenarbeit mit den Office-Anwendungen von Microsoft vorgesehen und werden



## 1.2 Beispiel Eingesetzter Produkte



Casio E-200



IPAQ H3150



ASUS MyPal A620 Pocket PC



Toshiba e750 Bluetooth



Yakumo Delta 400



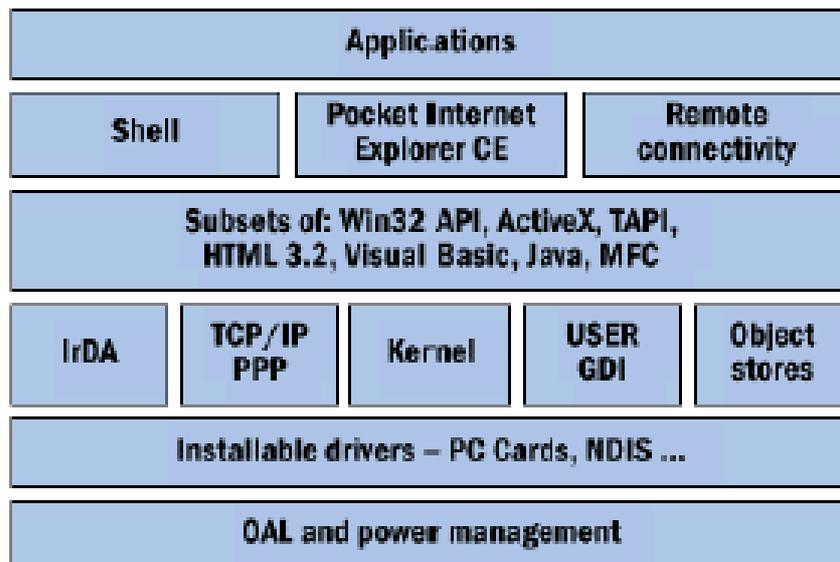
T-Mobile MDA II

## 2 Architektur

### 2.1 Aufbau

Windows CE ist ein preemptives 32-Bit-Multitasking-Multithreaded-Echtzeit-Betriebssystem und kann bis zu 32 Prozesse gleichzeitig ausführen. Die Anzahl der Threads pro Prozeß wird nur durch den vorhandenen Arbeitsspeicher begrenzt.

Windows CE wird durch eine dünne Schicht von Code, die zwischen dem Kernel und der Hardware-Plattform liegt, an die spezifische Hardware angepaßt. Diese Schicht wird als OEM Adaption Layer (OAL) bezeichnet (siehe Abbildung). Der OAL isoliert die gerätespezifischen Hardware-Features vom Kernel. Der Windows CE Kernel auf der anderen Seite enthält dem Prozessor angepaßten Code, der die Kernprozesse des Prozessors verwaltet. Der OAL ist immer spezifisch für eine bestimmte CPU (SH3, 486, etc.) und Hardware-Plattform. Die Verantwortung des OAL liegt darin, daß die Zielplattform durch den Kernel erreicht und gesteuert werden kann. Diese Steuerung umfaßt die zeitgesteuerten Prozesse der Hardware, Geräte-Interrupts und die Implementierung des Power-Managements für die an die Plattform angeschlossene Peripherie.



Windows –CE Architektur

Bei Windows CE handelt es sich um eine völlige Neuentwicklung, also nicht etwa um eine abgespeckte Version von Windows 95/98 oder Windows NT. Es ist sehr kompakt, was aufgrund der beschränkten Speicherkapazität der Geräte zwingend ist. Minimalsysteme kommen schon mit 512 KB ROM und 350 KB RAM aus. Für einen PDA wird schon mehr benötigt, hier sind die Mindestanforderungen 4 MB ROM und 2 MB an RAM.

Die weitreichende Modularisierung des Systems ist besonders hervorzuheben, denn jedes Subsystem kann nach Bedarf weggelassen bzw. hinzugefügt werden. Es sind sogar Systeme ohne Anzeigesystem möglich. Windows CE ist ein 32-Bit OS, es beinhaltet Multitasking/Multithreading und ist per ActiveX und Javascript erweiterbar. Das Powermanagement ist ein fester Bestandteil des Betriebssystems. Die übliche Hardwarebasis sind verbrauchsarme 32-Bit RISC-Prozessoren, wie etwa Hitachis SuperH-Familie,

diverse MIPS-Prozessoren mit R3000- oder R4000-Kernel, Motorolas PowerPC 821, Intels i486-Familie, ARM- und StrongARM-Prozessoren. Die Architektur von Windows besteht im wesentlichen aus vier großen Hauptmodulen, die z.T. aufeinander aufbauen: OEM Adaption Layer, Kernel, Graphics, Windowing and Event Subsystem (GWES) und Object Store.

**OEM Adaption Layer** Die OEM Adaption Layer (OEM = Original Equipment Manufacturer), kurz OAL, ist eine dünne hardware-spezifische Schicht. Sie sitzt direkt zwischen der Hardware und dem Kernel und vermittelt zwischen diesen beiden. Die OAL umfasst alles, was ein OEM implementieren muss, um Windows CE auf seine Hardware anzupassen. Sie ist daher auf ein Minimum reduziert. So enthält das Modul nur die Power-Management-Funktionen und die lebensnotwendigen Funktionen des Schedulers, des timer-gesteuerten Umschalters für die Threads, der ein Bestandteil des Kernels ist. Ebenso findet dort die Interruptbehandlung statt.

**Kernel** Der Kernel (deutsch: der Kern) ist der Hauptbestandteil eines jeden Betriebssystems. Bei CE sind in ihm die wichtigsten Systemfunktionalitäten implementiert. So übernimmt er die Verwaltung des Speichers und der Prozesse. Er ermöglicht präemptives Multitasking und Multithreading. Dabei stellt Multithreading auch kleinere, leichtgewichtige Prozesse bereit, die aufgrund von weniger Overhead schneller abgearbeitet werden, aber auch über weniger Funktionalität verfügen. Das Umschalten zwischen den Threads übernimmt ein prioritätsgesteuerter Scheduler. Der Kernel ist weiterhin auch für das Exception Handling (die Ausnahmebehandlung) zuständig.

## 2.2 Speicherverwaltung

Windows CE-Geräte verfügen normalerweise über zwei Arten von Speicher: ROM (read-only memory) und RAM (random access memory). Im ROM sind das Betriebssystem und die mitgelieferten Anwendungen gespeichert. Der RAM ist unterteilt in Programmspeicher und Datenspeicher. Der Programmspeicher wird für zusätzliche Programme und Heaps und Stacks verwendet, während im Datenspeicher Dateien und Datenbanken gespeichert werden. Auf diesen Datenspeicher setzt der Object Store auf, das Dateisystem von Windows CE. Er ist zuständig für die dauerhafte Speicherung aller Daten. Dabei werden drei Möglichkeiten unterstützt:

- Das File-Handling wie bei den Desktop-Varianten von Windows. So kann das Modul Object Store im (an den PC) angedockten Zustand auf dessen Festplatte lesend und schreibend zugreifen. Es dient zum Ablegen von beliebigen Dateien auf der Festplatte des PCs oder im Speicher des CEGerätes.
- Die Registry von Windows CE ist identisch mit der der Desktop-Versionen. Wie dort auch werden in ihr die Einstellungen von Anwendungen und des Betriebssystems sowie Benutzerdaten abgelegt. Windows CE bietet eine spezielle API (**A**pplication **P**rogramming **I**nterface) für Datenbanken. Sie nimmt strukturierte Daten auf, wie sie in der Adress- und Terminverwaltung oder in Excel auftreten. Sie erlaubt die einfache Indizierung und damit auch den einfachen Zugriff auf die Daten.

Um dem Verlust von Daten bei Spannungsausfall oder Systemabstürzen o.ä. zu vermeiden, bietet Windows CE ein transaktionssicheres FAT-Dateisystem. Die Sicherheit wird dabei durch eine partielle Spiegelung der Daten erreicht. Zusätzlich werden alle Änderungen aufgezeichnet, so dass sie bei Bedarf rückgängig gemacht werden können.

## 2.3 Prozessverwaltung

Es sind maximal 32 Prozesse mit beliebig vielen Threads möglich. Jeder Thread besitzt eine eigene Message Que. Dort können Nachrichten von Systemressourcen oder anderen Threads hinterlegt werden. Scheduling ist präemptiv, prioritätsbasiert auf Thread-Ebene mit maximal sieben Prioritätsstufen. Priorität 0-1 für Echtzeitanwendungen, 2-4 Kernel und andere Anwendungen, 5-7 hier können die Anwendungen jederzeit verdrängt werden. Zur Interprozesskommunikation und Synchronisation werden folgende Konzepte unterstützt: Critical Section, Mutex, Event, Memory und Mapped File. Es sind maximal 32 MB Speicher pro Prozess möglich. Die Adressräume der einzelnen Prozesse sind voreinander geschützt. Ein Interrupt Request (IRQ) wird von einer Interrupt-Service-Routine (ISR) im Kern registriert und einem Interrupt-Service-Thread (IST) zur weiteren Bearbeitung übergeben. Dieses Model erlaubt kurze Interrupt-Sperrzeiten, jedoch keine verschachtelten Interrupts.

## 2.4 GUI

Die Benutzerschnittstelle von Windows CE bietet dem Anwender aus der Desktopwelt vertraute Bedienkonzepte und das gewohnte Windows Look and Feel. Es wurden lediglich Anpassungen vorgenommen um die Oberfläche auf kleinere Displays zu optimieren. So wurde zum Beispiel auf 3D Buttons ganz verzichtet. Zuständig für die grafische Oberfläche ist das GWES.

**Graphics, Windowing and Event Subsystem** Das GWES vereint in sich die Benutzer- und GDI-Subsysteme (GDI = Graphical Device Interface, steuert die Anzeige von Text und Grafik), wie sie aus den Desktopvarianten bekannt sind, und deren Funktionalitäten. Es bildet die Schnittstelle zwischen dem Benutzer auf der einen Seite und Anwendung und Betriebssystem auf der anderen. Das zentrale Merkmal des GWES ist das Fenster. Jede Anwendung benötigt ein Fenster, um mit dem Betriebssystem kommunizieren zu können. Für Anwendungen, die im Hintergrund laufen, wird ein unsichtbares Fenster erzeugt. Wenn eine Anwendung gestartet wird, erstellt CE ein Fenster und eine dazugehörige Nachrichtenwarteschlange. Macht der Benutzer an dem Gerät Eingaben durch Tastatur, Maus oder Stift, werden diese als Nachrichten (siehe Abb.1) an das Anwendungsfenster und den Fenstermanager weitergeleitet. Ein Nachricht könnte beispielsweise sein „OK-Button angeklickt“. Das Anwendungsfenster meldet daraufhin dem Fenstermanager, was zu tun ist, also z.B. einen Dialog schließen. Der Fenstermanager meldet dies an das Nachrichtensystem, an die Steuerelemente und an die Grafikgeräteschnittstelle. Die Grafikgeräteschnittstelle steuert den Anzeigentreiber und damit die Ausgabe auf dem Bildschirm. Das GWES wird sogar von Systemen benutzt, die eigentlich über keine grafische Benutzeroberfläche verfügen, da auch dort Anwendungen Nachrichten empfangen und mit dem System kommunizieren müssen.

## 2.5 Netzwerkanbindung

Windows CE unterstützt zwei Kommunikationsmodelle. Die serielle- und die Netzwerk – Kommunikation. Serielle Kommunikation wird ermöglicht über ein serielles Kabel, USB oder Infrarot (über IrDA Protokoll). Für Netzwerk- Kommunikation unterstützt Windows CE die http, ftp, TCP/IP, PPP und SLIP Protokolle, sowie RAS (Remote Acces Service) Client support.

## 2.6 Peripherie

Es werden prinzipiell verschiedene Arten von Treiber unterschieden.

**Native Driver** sind integrierte Treiber (z.Bsp. Tastatur) sie werden zum Kernel gelinkt.

**Stream Interface Driver** werden als Standalone DLL's geladen. Sie unterstützen streambasierte Peripherie, ihre Treiberschnittstelle stellt Funktionen zum öffnen und schließen von Datenströmen bereit.

**NDIS based Driver** erlauben die unabhängige Implementation der Netzwerkprotokolle von den Hardwaretreibern.

**USB Driver** zur Unterstützung des Universalen Seriellen Busses.

## 2.7 Windows CE Baukasten

Windows CE ist kein fertiges Betriebssystem im herkömmlichen Sinne, sondern er als Baukasten zu verstehen. Es ist durch die modulare Architektur an die verschiedensten Anforderungen der Zielhardware anpassbar. Um das eigene Windows CE zu kreieren, muss als erstes die OEM HAL implementiert werden. Dort wird zum Beispiel das Display festgelegt und Treiber für die Eingabegeräte implementiert. Als nächstes wird mit dem Windows CE Toolkit der Prozessor und die OS Komponenten gewählt. In diesem Schritt können auch eigene „Built In Anwendungen“ für das OS Image hinzugefügt werden. Jetzt kann das Image erzeugt werden und in den ROM des Zielgerätes geladen werden. Da ein Debugging direkt auf dem Gerät nicht möglich ist, kann dies über eine Remoteverbindung mit Hilfe von Tools aus dem Toolkit realisiert werden.

## 2.8 Versionen

**Windows CE 1.0** kam 1996 auf den Markt und lag nur in einer englischsprachigen Version vor. Außerdem unterstützte es noch keine Farbbildschirme und die zugehörigen Programme hatten nur eingeschränkte Funktionalität.

**Windows CE 2.0** erschien 1997, nun auch in deutscher Sprache und unterstützte auch Farbbildschirme. Eingesetzt wurde es auf Pocket PCs und Handheld PCs. Desweiteren wurde die Funktionalität der mitgelieferten Programme erweitert.

Anfang 1998 erschien dann **Windows CE 2.11** anfangs nur auf dem Handheld PC Pro, später auch für den Handheld PC und den Pocket PC.

Die aktuellste Version vor .NET – **Windows CE 3.0** – wurde im ersten Quartal 2000 der Öffentlichkeit präsentiert und wurde gegenüber der Vorgängerversion deutlich abgespeckt und optimiert.

Die aktuelle Version – **Windows CE .NET 4.2** – Die erste Version von Windows CE .NET wurde im Januar 2001 veröffentlicht. Folgende Verbesserungen wurden seit der ersten Version von Windows CE .NET eingebracht:

- Bessere Feinabstimmung bei den Komponenten
- Schnellere Leistung
- Erweiterte integrierte Browsing und Multimedia Fähigkeiten
- Verbesserte Echtzeitunterstützung
- Umfassende Treiberunterstützung

### 3 Entwicklungsoberflächen

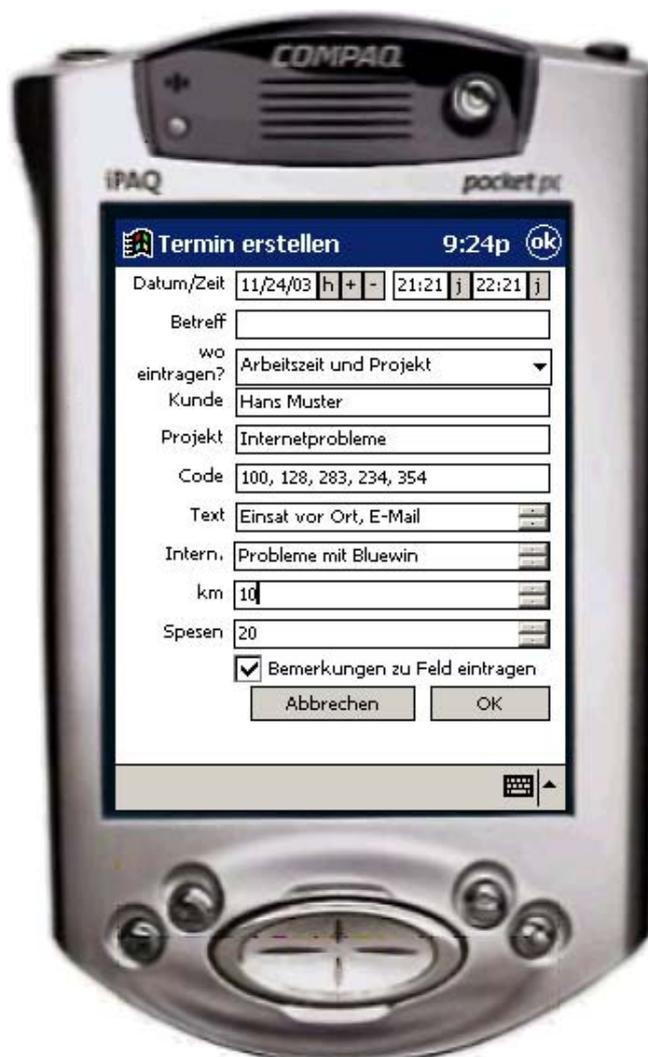
Zur Programmierung von eigenen Windows CE Anwendungen steht die gesamte Windowswelt offen. So kann die Win32 API, die MFC Klassen, Visual Basic / C++ und DirectX genutzt werden. Zur einfachen Entwicklung bietet das Visual Studio eigens dafür bestimmte Windows CE Projekte.

#### 3.1 Embeded Visual Studio 3.0

Microsoft Embedded Visual C++® 3.0 (kurz eVC) ist Teil der **Microsoft eMbedded Visual Tools 3.0** (eVT). Diese beinhalten außerdem noch Embedded Visual Basic® 3.0 (kurz eVB) sowie Software Development Kits (SDKs → Ein SDK enthält Header-Files, Libraries, DLLs, Beispielprogramme und einen Emulator für die jeweilige Plattform). Für die verschiedenen Windows CE Gerätetypen. Einigermaßen erstaunlich für ein Microsoft-Produkt ist die Tatsache, daß die eVT frei erhältlich ist.

##### 3.1.1 Anwendungsbeispiele

Projekt-Verwaltung mit MS Outlook



Terminverwaltung auf dem PDA

Nach dem die Arbeitszeit / das Projekt auf dem PDA beim Kunden eingetragen wurde, muss es nur noch mit OK bestätigt werden. Somit ist es nun im MS Outlook verfügbar und kann dann bei der Synchronisation des PDA's in die Projekt-Verwaltung eingetragen werden.

Damit sparen wir Zeit bei der täglichen Rapportierung der Arbeitszeit.

Code in Microsoft eMbedded Visual Basic 3.0

```

PV_OLTermin - frmTerminErstellen (Code)
(General) (Declarations)
Option Explicit

Const vHeightGross = 720
Const vHeightNormal = 240

Function ZeitRunden(vZeitTxt As String, vFeld As String)
Dim vStunden As Long, vMinuten As Long
Dim vNeueZeit As Date, vAntwort As Long

vStunden = Hour(CDate(vZeitTxt))
vMinuten = Minute(CDate(vZeitTxt))

vMinuten = Int(vMinuten / 5 + 0.5000001) * 5
If vMinuten >= 60 Then
vStunden = vStunden + 1
vMinuten = 0
End If

vNeueZeit = FormatDateTime(CDate(vStunden & ":" & vMinuten), vbShortTime)
If vMinuten = Minute(CDate(vZeitTxt)) Then
If MsgBox(vNeueZeit & " eintragen?", vbYesNo + vbQuestion) = vbYes Then
Controls(vFeld) = vNeueZeit
End If
Else
vAntwort = MsgBox(vZeitTxt & " runden auf " & vNeueZeit & "?" & vbCrLf, vbYesNo + vbQuestion)
If vAntwort = vbYes Then
Controls(vFeld) = vNeueZeit
ElseIf vAntwort = vbNo Then
Controls(vFeld) = FormatDateTime(CDate(vZeitTxt), vbShortTime)
End If
End If
End Function

```

```

PV_OLTermin - frmTerminErstellen (Code)
(General) (Declarations)

vBody = vBody & vbCrLf & vTrenn & fctlText.Text
vBody = vBody & vbCrLf & vTrenn & fctlInternertext.Text
vBody = vBody & vbCrLf & vTrenn & fctlKm.Text
vBody = vBody & vbCrLf & vTrenn & fctlSpesen.Text
End If

Set vOL = CreateObject("PocketOutlook.Application")
vOL.Logon

Set vItem = vOL.CreateItem(olAppointmentItem)

vItem.Subject = vTrenn & fctlBetreff.Text

vItem.Body = vBody

vDatum = CDate(fctlDatum)
vVon = CDate(fctlVon)
vBis = CDate(fctlBis)

fctlDatum.Text = FormatDateTime(vDatum, vbShortDate)
fctlVon.Text = FormatDateTime(vVon, vbShortTime)
fctlBis.Text = FormatDateTime(vBis, vbShortTime)

vItem.Start = vDatum + vVon
vItem.End = vDatum + vBis

vInTagen = CDate(Int(vItem.Start)) - Date

Select Case vInTagen
Case 0
vMeldung = "Heute"
Case 1
vMeldung = "Morgen"

```

## 4 Zukunft

Wir vermuten das in Zukunft Microsoft vom eigentlichen Windows-CE abkommen wird um ein „normales“ abgespecktes Betriebssystem einzusetzen. In der heutigen Zeit wo Speicherplatz und Speichergrösse keine grosse Rolle mehr spielen wir es in nächster Zukunft möglich sein ein Betriebssystem zu haben in verschiedenen Varianten davon erhältlich sind.

Zukunftsvision:

