

PC NEWS

Club**PocketPC** Club**DigitalHome** Club**System** Club**Dev** Club**Education**



CLUB COMPUTER.AT

Club**System**

SecLookOn

Genius Connect

Club**PocketPC**

Scotty

Xperia

OperaMini

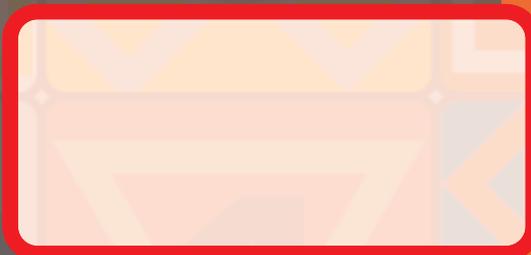
Club**DigitalHome**

Unterschriftsfont

Club**Dev**

Datenbankaufbau

Workflow



Inhalt

Liebe Leserinnen und Leser!

Franz Fiala

LIESMICH

- 1  Cover
Werner Krause
- 2  Liebe Leserinnen und Leser
Franz Fiala
- 2  Inhalt
- 4  Autorinnen und Autoren
- 4  Impressum
- 4  Inserenten

CLUBS

- 4  CC|Skripten
Martin Weissenböck

METATHEMEN

- 6 **META** UNICODE 5.1.0 meets Or-Om-code∞.1.0
OR-OM
- 8  Studien...
Oskar Wagner

CLUBDIGITALHOME

- 13  Asus Eee-PC 4G mit Linux
Christian Schneider
- 14  Unterschrift als Font
Franz Fiala

CLUBPOCKET-PC

- 9  Scotty mobil
Walter Riemer
- 10  Sony Ericsson Xperia X1
Paul Belcl
- 11  Opera Mini auf einem WM6.1 Smartphone
Walter Riemer

CLUBSYSTEM

- 3  Einfaches Passwort mit 340 Bit
Helmut Schluderbacher, Peter Trykar
- 5  Genius Connect
Paul Belcl

CLUBDEV

- 17 α Datenbankimplementierung
Christian Zahler
- 28 α Windows Workflow Foundation
Thomas Reinwart

LUSTIGES

- 2  Beichtstuhl
Christian Berger

PCNEWS-112

Lassen Sie sich bei der Passworteingabe ruhig über die Schultern schauen...

... denn es nützt einem Beobachter nichts, wenn Ihr System das von **Helmut Schluderbacher** und **Peter Trykar** vorgestellte, revolutionäre Passwortverfahren verwendet. Nicht Passwörter sondern Regeln merkt man sich. Ein Angreifer kann Ihre Eingaben ruhig mitschreiben, denn beim nächsten Login werden es wieder andere sein. Die Anzahl der Antwortmöglichkeiten auf eine grafisch gestellte Frage ist so groß, dass die Logik hinter der Eingabe nicht erkannt werden kann. Ideal für Umgebungen, "wo die Wände Augen haben" wie zum Beispiel beim Unterricht, wenn Lehrer von Schülern bei der Arbeit am Computer beobachtet werden.

Datenbanken

PCNEWS-110 SQL-Server 2005/2008

PCNEWS-112 Datenbankimplementierung-Teil 1

- 1 Datenbank-Grundlagen
- 2 Transact SQL
- 3 Sichten (*Views*)
- 4 Gespeicherte Prozeduren (*Stored Procedures*)
- 5 Trigger

PCNEWS-113 Datenbankimplementierung-Teil 2

- 6 Client-Programmierung von MS SQL Server 2005
- 8 ActiveX Data Objekts (ADO) und ADO.NET

Google

Google-Dienste, Google-Suchkommandos, eigene Suchmaschinen, Dienste für Webmaster und andere Google-Goodies werden in unseren nächsten Ausgaben vorgestellt. Damit auch die Gefahren der Omnipräsenz von Google verdeutlicht werden, laden wir alle Autoren und Die-es-noch-werden-wollen-Autoren Ihre Erfahrungen zum Thema "Google" via PCNEWS darzustellen.

Dazu kommen Microsoft Server 2008 und Entwicklung mit Visual Studio Express-Versionen.

Franz Fiala

Beichtstuhl 2000



http://pcnews.at/?id=pcn112

Autoren

Belcl Paul 1966 5,10
 EDV-Koordinator, Trainer für Windows Mobile, Promotor bei Microsoft Österreich, Direktor des Clubpocketpc
Firma BELCL EDV-Koordination & Systemberatung
Club CCC
Hobbies Fotografieren, Reisen, Segway
 E: pbelcl@ccc.at
 http://www.belcl.at/

Berger Christian 2
 Karikaturist und Comiczeichner für verschiedene Kärntner Zeitungen
Firma Karicartoons
 E: karicartoons@aon.at
 http://www.bergercartoons.com/

Fiala Franz Dipl.-Ing. 1948 2,14
 Leitung der Redaktion und des Verlags der PCNEWS, Obmann des PCC; Lehrer für Nachrichtentechnik und Elektronik.i.R.
Firma TGM-N
Werdegang BFPZ-Arsenal
Club CCC/MCCA/PCC/VIT
Absolvent TU-Wien, Nachrichtentechnik
Privates verheiratet, 1 Kind
 E: franz.fiala@clubcomputer.at
 http://franz.fiala.cc/

Krause Werner Mag. 1955 1
 Lehrer für Bildnerische Erziehung
Schule GRG 23 Alterlaa, Bundesgymnasium Wien 23
Absolvent Hochschulf. Angewandte Kunst, Gebrauchsgrafik
Hobbies Fotografieren, Computergrafik (CorelDraw/Photoshop u.a.) Videoschnitt, Coverbilder für PCNEWS
Privates 2 Kinder
 E: w.krause@chello.at

Pflegerl Siegfried Dr. 6
 Schriftsteller Philosophie, Kunst- und Evolutionstheorie sowie Soziologie
Club PCC
 E: siegfried.pflegerl@chello.at
 http://portal.or-om.org/

Reinwart Thomas 1973 28
 Softwareentwickler, MCAD, MCSD, MCDBA, MCSA, MCSE Zertifizierungen
Firma Reinwart
 E: office@reinwart.com
 http://www.reinwart.com/

Riemer Walter Dipl.-Ing. 1940 9,11
 Autor mehrerer Lehrbücher für den Unterricht, Ingenieurkonsulent für Elektrotechnik, früher Lehrer für Informatik und Leiter des Rechenzentrums am TGM
Club PCC
Hobbies Musik und Sport
Privates verheiratet, 3 Kinder
 E: wriemer@A1.net
 http://niederfellabrunn.at/

Schluderbacher Helmut Dipl.-Ing. 1959 3
 Informatiker
Club CCC
Absolvent TU Wien, Informatik
Interessen User-Interface
Hobbies Musik, Katzen
 E: schlud@ccc.at

Schneider Christian Mag. 13
 AHS-Lehrer für Deutsch
Club CCC
 E: schneider@gmx.at

Trykar Peter 1964 3
 Sales und Marketing Spezialist
Firma MERLINnovations
Absolvent BFS Luftfahrzeugtechnik
Hobbies Musik, Motorräder, Schriftstellerei, Tischlern
 E: ptrykar@hotmail.com

Wagner Oskar Anton M.Sc.PhD EdD 8
 Nach 38 Jahren Telekommunikation, davon 34 Jahre im Unterrichtswesen im Ruhestand
Club CCC
 E: wago@aon.at
 http://members.aon.at/oe1-100470/

Impressum

Impressum, Offenlegung
Richtung Auf Anwendungen im Unterricht bezogene Informationen über Personal Computer Systeme. Berichte über Veranstaltungen der Herausgeber.
Erscheint 5 mal pro Jahr, Feb., Apr., Jun., Sep., Nov
Verleger PCNEWS-Eigenverlag
 Siccardsburggasse 4/1/22 1100 Wien
 ☎ 01-6009933-210 FAX: -9210
 E: pcnews@pcnews.at
 http://www.pcnews.at/
Herausgeber ClubComputer
 Fernkornogasse 17/1/6 1100 Wien
 ☎ 01-6009933-11 FAX: -12
 E: office@clubcomputer.at
 http://www.clubcomputer.at/
Druck, Versand FriedrichVDV
 Zamenhofstraße 43-45, 4020 Linz
 ☎ 0732-669627-0 FAX: 669627-5
 E: office@friedrichvdv.com
 http://www.friedrichvdv.com/

ClubComputer
Leitung, CCC Werner Illsinger
 ☎ 01-6009933-220 FAX: -9220
 E: werner.illsinger@clubcomputer.at
PCNEWS, PCC Franz Fiala
 ☎ 01-6009933-210 FAX: -9210
 E: franz.fiala@clubcomputer.at
Marketing Ferdinand De Cassan
 ☎ 01-6009933-230 FAX: -9230
 E: ferdinand.de.cassan@clubcomputer.at
ClubPocketPC Paul Belcl
 ☎ 01-6009933-288 FAX: -9288
 E: paul.belcl@clubcomputer.at
ClubDigitalHome Christian Haberl
 ☎ 01-6009933-240 FAX: -9240
 E: christian.haberl@clubcomputer.at
cc|Akademie Georg Tsamis
 ☎ 01-6009933-250 FAX: -9250
 E: georg.tsamis@clubcomputer.at

PCNEWS-112
Kenzeichnung ISSN 1022-1611, GZ 02Z031324M
Layout Corel-Ventura 10, Corel-Draw 12.0
Herstellung Bogenoffset, 80g
Erscheint Wien, Februar 2009
Texte http://pcnews.at/?id=PCN112
Kopien Für den Unterricht oder andere nicht-kommerzielle Nutzung frei kopierbar. Für gewerbliche Weiterverwendung liegen die Nutzungsrechte beim jeweiligen Autor. (Gilt auch für alle am PCNEWS-Server zugänglichen Daten.)
Werbung A4: 1 Seite 522,- EURO U2, 3,4782,- EURO Beilage: bis 50g 138,- EUR pro 1000 Stück
Bezug 1 Heft: 5,- EURO (zuzüglich Versand)
 5 Hefte: 20,- EURO (1 Jahr, inklusive Versand) kostenlos für Mitglieder von ClubComputer
Hinweise Druckfehler und Irrtümer vorbehalten. Alle erwähnten Produktnamen sind eingetragene Warenzeichen der entsprechenden Erzeuger.

Inserenten

●MTM-Systeme 32

 ☎ Hadrawagasse 36 1220 Wien
 ☎ Ing. Gerhard Muttenthaler
 ☎ 01-2032814 FAX: 2021303
 ☎ 0664-4305636
 E: g.muttenthaler@mtm.at
 http://www.mtm.at/
Produkte uC/UP-Entwicklungswerkzeuge, Starterkits, Industriecomputer, Netzqualitätsanalyzer, USV-Anlagen
Vertretung Tasking, PLS, Infineon, TQ-Components, Kontron, Dranetz-BMI, Panasonic, Dr. Haag, HT-Italia, Dr. Kaneff
 U1-Kagran, 26A bis Englisch-Feld-Gasse
Erreichbar

Weissenböck Martin Dir. Dr. 1950 4
 Direktor der HTL Wien 3 Rennweg, Leiter der ADIM und Autor von ADIM-Skripten, Leiter der ARGE Telekommunikation
Schule HTL Wien 3R, ADIM
Club ADIM/CCC/PCC
 E: martin@weissenboeck.at
 http://www.weissenboeck.at/

Zahler Christian Mag. 1968 17
 Gewerbetreibender, Autor von ADIM-Skripten, Erwachsenenbildung, Lektor für Informatik, MCSE
Firma WIFI St. Pölten, FHS Steyr
Club ADIM/PCC
 E: office@zahler.at
 http://www.zahler.at/

CC|skripten

Martin Weissenböck
ADIM, Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik
 1190 Wien, Gregor Mendel Straße 37
 Tel.: 01-314 00 288 FAX: 01-314 00 788

EDV-Skripten	
Nr	Titel
38	Turbo Pascal (Borland)
39	RUN/C Classic
40	Turbo-C (Borland)
41-3	Turbo/Power-Basic
43-2	DOS
43-3	DOS und Windows
47	Turbo-Pascal (Borland)
49	Quick-Basic (Microsoft)
50	C++ (Borland)
53-3	AutoCAD I (2D-Grafik)
53-5	AutoCAD I (2D-Grafik)
54	AutoCAD II (AutoLisp+Tuning)
55	AutoCAD III (3D-Grafik)
56	Grundlagen der Informatik
61	Visual Basic (Microsoft)
63	Windows und Office
81	Linux
191,192	Angewandte Informatik I + II
201,202	Word I+II
203	Excel
205,206	Access I+II
221	HTML
222	HTML und CSS
223	JavaScript,
227	VB.NET
231,232	Photoshop I+II
237, 238	Dreamweaver, Interaktive und animierte Webseiten

CDs	
Nr	Titel
110	Best Of VoIP (CD)
111	All About VoIP (DVD)

Bestellhinweise, Download
<http://www.adim.at/>
<http://adim.at/download/>
<http://www.adim.at/dateien/BESTELL.pdf>

Genius Connect

–kurzer Erfahrungsbericht

Paul Belcl

Das Problem welches mit dieser Software gelöst wird, habe vielleicht nicht viele; wenn man es allerdings hat, dann hilft Genius Connect wirklich hervorragend dabei!

Problemstellung

Man hat seine Kontaktdaten nicht im Outlook, oder Exchange, sondern in einer anderen Datenbank, will aber trotzdem die Vorteile von Outlook nutzen, beispielsweise die Synchronisation mit einem PDA, oder andere Funktionen. Allerdings sollen die Originaldaten in der bestehenden Datenbank bleiben!

Dann ist Genius Connect die ideale Software!

Was kann Genius Connect

Mit Genius Connect ist es möglich, nahezu alle Bereiche von Outlook als auch Exchange Folder mit einer beliebigen ODBC-fähigen Datenbasis zu synchronisieren. Unterstützt werden folgende Typen:

Outlook 2002/2002/2007

- Kalender, Kontakte, Mail, Aufgaben, Notizen, Journal

Outlook 2000

- Kontakte

Genius Connect ist sehr leistungsfähig und kann auch mit SQL-Befehlen umgehen. Es ist auch möglich, während der Synchronisation Felddatentypen zu ändern, z.B. ein Datumsfeld in ein Textfeld zu ändern u.s.w.

Ich habe mir Genius Connect angesehen, um meine Adressdatenbank von Access mit den Outlook-Kontakten zu synchronisieren. Ich werde daher diese Vorgehensweise näher erklären, unabhängig davon, dass die Software natürlich viele weitere Funktionen hat!

Vorbereitung – ODBC Datenquelle



Bevor eine Synchronisation stattfinden kann, muss die Datenbank als ODBC-Quelle eingetragen werden. Dazu geht man in die *Systemsteuerung* unter *Verwaltung* und sucht *Datenquellen (ODBC)*.

In meinem Fall füge ich eine MS-Access-Datenquelle hinzu. Im nächsten Fenster kann man dann die Datenbank auswählen, die als ODBC Quelle dienen soll. (Abb01)

Dies ist die Grundvoraussetzung, um eine Synchronisation überhaupt machen zu können.

Danach wird Genius Connect installiert.

Datenbank Synchronisation einrichten

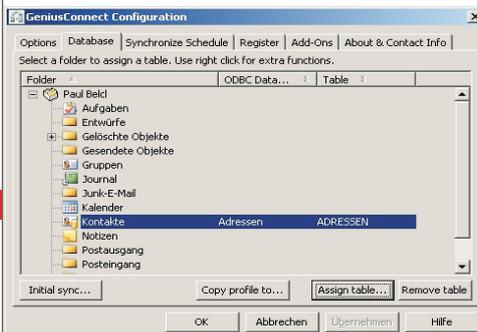
Nach Installation von Genius Connect bekommt man im Outlook eine zusätzliche Menüleiste (Abb02).



Zuerst muss man die Grundeinstellungen machen und daher wählt man „Genius Connect Setup“.

Ich wähle dort den „Database“-Karteikartenreiter und suche mir die *Kontakte* aus.

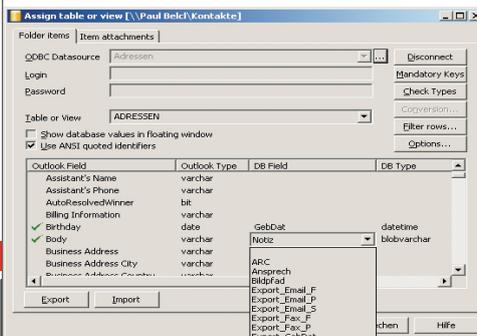
Danach muss ich eine Datenbasis auswählen, die ich vorher als ODBC-Quelle auf meinem Rechner eingetragen habe. In meinem Fall „ADRESSEN“. Nun muss ich noch eine Tabelle in der Datenbank auswählen. Diese heißt bei mir ebenfalls *ADRESSEN*. (Abb03)



Im nächsten Schritt müssen alle Felder zugewiesen werden, die zu synchronisieren sind.

Dazu muss man natürlich nicht nur die Datenbank gut kennen, sondern sich auch mit den Funktionen der Outlook-Felder vertraut machen.

In meinem Test habe ich lange gesucht bis ich herausfinden konnte, dass das Notizfeld im Outlook den Namen „Body“ hat. (Abb04).



Natürlich gibt es auch die Möglichkeit diverse Filter über so genannte „Select Statements“ zu verwenden. Damit kann man beispielsweise nur Datensätze synchronisieren, die gewissen Regeln entsprechen.

Hat man alle Definitionen festgelegt, kann die Synchronisation losgehen.

Synchronisation

In der Menüleiste gibt es vier Schaltflächen um zu synchronisieren. (Abb02)

Mit „Load“ kann man einen oder alle Datensätze aus der Datenbank laden. Mit „Store“ lassen sich die Daten vom Outlook in die Access-Datenbank zurückspeichern.

Ich habe für meinen Fall beschlossen, die Daten von der Access-Datenbank nur zu laden, aber nie zurückspeichern, denn die Access-Datenbank ist die Basis und Outlook soll nur als Anzeigeprogramm dienen. Genius Connect kann

natürlich auch eine zeitgesteuerte Synchronisation automatisch durchführen.

Warum nicht die Daten ins Outlook übersiedeln?

Ich bin in der Beziehung etwas altmodisch! Meine Datenbasis und auch die Datenbank sind über Jahre gewachsen. Ich habe unzählige Felder und Funktionen eingebaut, oder einbauen lassen, die ich im Outlook niemals finden werde! Zum Beispiel: Automatischer Daten Export für Endgeräte (Schnurlostelefone, Anrufbeantwortersoftware); Druckvorlagen für Adressetiketten, Einlageblätter fürs Time System meiner Frau; Automatisches Ausfüllen der Adresse in Briefvorlagen; Postleitzahl-Automatik; und viele andere Funktionen!

Außerdem lässt sich meine Access-Datenbank ohne Probleme jederzeit in ein anderes Datenformat umwandeln, falls es nötig ist.

Warum nicht einfach nur importieren?

Ich könnte in meinem Anwendungsfall die Daten ja auch nur importieren, wird sich mancher denken. Das stimmt, aber dann gehen mir bei der Synchronisation von Outlook mit meinem Pocket-PC einige nette Funktionen verloren; zum Beispiel die Möglichkeit, Bilder zu meinen Kontakten zu verknüpfen!

Beim Löschen und Wiederimportieren der Daten ändern sich die Datensatz-IDs der einzelnen Kontakte. Das bedeutet: nach jedem Import muss ich alle Bilder neu zuweisen.

Mit Genius Connect werden die Daten EINMAL synchronisiert und die Datensatz IDs bleiben gleich, solange der betreffende Datensatz in der Datenbank existiert. Daher ist es möglich, die Daten zu aktualisieren OHNE dass sich die Datensatz IDs verändern. Tja, das klingt zwar etwas kompliziert, aber es funktioniert!

Ergebnis: was ist nun besser?

In meinem Fall ist es nun möglich, im Outlook Bilder zu meinen Kontakten hinzuzufügen, deren Verknüpfungen auch dann bestehen bleiben, wenn sich Felder im bestehenden Datensatz ändern (z.B. eine Telefonnummer)

Um die Daten zu aktualisieren, muss ich jetzt nicht mehr alle Daten löschen und neu importieren, was meinen Pocket PC meist dazu veranlasst hat, sehr lange zu brauchen, die Daten danach wieder auf Gleich zu bringen, sondern ich drücke einfach auf den Knopf „load ALL“ und eine Minute später sind meine Daten aktuell.

Auch die Verknüpfungen von Personeneinträgen in diversen Oberflächen (z.B. SPB-Mobile Shell) bleiben bestehen, wenn ich die Daten aktualisiere. Früher mussten diese nach einem Neuimport immer wieder mühsam hinzugefügt werden.

Resümee

Genius Connect hat viele Anwendungsmöglichkeiten!

Ich habe in meinem Fall nur einen sehr kleinen Teil davon beschrieben der nötig war meine Anforderungen zu lösen.

Dieser Artikel soll auch nur als Anregung dienen in einem ähnlichen Fall vielleicht mal einen Blick auf Genius Connect zu werfen. Die Software kann mit großer Wahrscheinlichkeit nahezu jede Synchronisations-Anforderung mit Outlook oder Exchange erfüllen

Nähere Infos, sowie eine Testversion gibt es auf der Webseite von Genius Connect:

<http://www.geniusconnect.com/>.

tage ausgebildeter Mathematiker oder Logiker nicht **verstehen** kann, oder dass diese Gedanken eben nicht in der heutigen Sprache der Mathematiker und Logiker abgefasst sei, ist nur bedingt gültig. In den beiden Aufsätzen wird sehr wohl auf die philosophischen Grundlagen der heutigen Mathematik und Logik (vor allem auf die **Grundlagenkrise**) genau eingegangen und dort werden auch die heutigen Bezeichnungen und Formeln benützt. Und vor allem, wenn eine neue Philosophie der Logik und Mathematik wirklich neu ist, dann muss sie eben ihre eigenen neuen sprachlichen Ausdrücke und Relationen erst erfinden, um das Neue erklären zu können. Eine wirklich kompetente Kritik unseres Ansatzes kann daher erst dann erfolgen, wenn man auf die Argumente in den beiden Aufsätzen **konkret** eingeht. Dieses Eingehen kann allerdings auch nur darin bestehen, dass der Kritiker meint, ihm sei das alles zu abstrus oder zu unverständlich. Dann darf er aber auch nur behaupten, dass es **für ihn** unbrauchbar ist. Er darf nicht ausschließen, dass diese Argumente **für andere** sehr wohl anregend und zugänglich sind. Unsere Ausführungen sind daher nur dann als falsch zu bezeichnen, wenn man nicht berücksichtigt, dass sie von völlig anderen, neuen Voraussetzungen ausgehen, wenn man also über die Argumente "drüberfährt".

Was haben aber letztlich solche hochabstrakten philosophischen Kalküle über Sprache, Mathematik und Logik mit **Kunst** zu tun. Die Gruppe Or-Om präsentiert ja hier ein **Kunst**projekt? Nun: der Kunstbegriff hat sich in den letzten 50 Jahren sehr erweitert. Ein wichtiger neuer Zweig derselben ist die **Konzeptkunst**:

- <http://de.wikipedia.org/wiki/Konzeptkunst>
- http://dreher.netzliteratur.net/3_Konzeptkunst_Titel.html
- <http://tas1.uni-muenchen.de/links/NAKS.html>

Kunst wird sowohl in Richtung auf das konkrete Leben als auch auf andere Bereiche der Gesellschaft (z.B. auch auf Wissenschaft und Wirtschaft[Kunstmarkt]) hin erweitert. Kunst ist daher hier nicht Mantel für Philosophie sondern als erweiterte Kunst selbständig tätig. Andererseits ist es natürlich auch möglich, heikle philosophische Probleme im Medium der Kunst zu präsentieren, um in diesem Bereich eine Anregung zum Nachdenken zu geben. Die genauen wissenschaftlichen Argumente werden ja in den beiden Aufsätzen flankierend mitgeliefert.

Das Kunstprojekt, welches die Gruppe Or-Om derzeit im Museumsquartier in Wien installiert, **ist also in diesen argumentativen Zusammenhängen zu beurteilen.**

- <http://qdk.blogspot.com/2008/11/15/p75/>
- <http://programm.mqw.at/programmdatenbank/index.phtml?von=2008-11-26&bis=&OID=&keyword=unicode&cc=mq&tmp=q21-det&TID=4615>

Hier seien die Basics zusammengefasst:

Motto:

A: 'Was verstehst Du unter "Semantischer Incodierung"?'

B: 'Unicode 5.1.0 ist der Turm von Bilbam. in dem langfristig für jedes endliche sinntragende Schriftzeichen bzw. Textelement aller bekannten und künftigen Schriftkulturen und Zeichensysteme ein digitaler Code festgelegt wird. Über diesem Turm schwebt die Pyramide Or-Omcode ∞.1.0 das Codesystem der künftigen unendlichen Universalsprache (Or-Om-Sprache). Mit der Integration (Incodierung) in den unendlichen Or-Omcode ∞.1.0 erfährt jedes endliche Zeichen

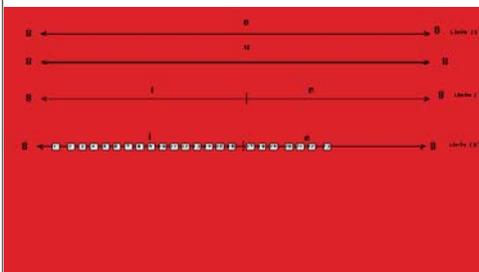
in Unicode 5.1.0 eine Änderung seines Sinnes und seiner semantischen Dimension.

A: Kann man sagen, durch die Berührung mit dem Unendlichen verwandelt sich die Semantik des Zeichens im neuen universalen Kontext, der für alle möglichen Zeichen aller Zeiten und Welten gilt?

B: 'Korrekt!'

Grand Fert

Or-Om-Code ∞.1.0



Semantic Encoding

Unicode 5.1.0

Das Bild auf der vorigen Seite oben zeigt die derzeit definierten Codepages des Unicode.

Semantische Incodierung

Die Bezeichnung "Incodierung" wird aus dem Englischen "Encoding" übernommen. Diesen Begriff benützt UNICODE selbst für die Integration aller Zeichen aller Sprachen usw. der Welt in sein Codesystem.

Das purpurne Viereck □ repräsentiert einen Buchstaben (C∞1) im **Or-Omcode ∞.1.0** der Universalsprache (Or-Om-Sprache). Dieser Buchstabe (C∞1) enthält in seiner Linie (3) die 125 Blöcke des Unicode 5.1.0 als innere, begrenzte Segmente. Alle Buchstaben, Schriftzeichen, Symbole, Operatoren, Glyphen, Ideographie und Ideogramme in den 125 Blöcken des Unicode 5.1.0 sind als begrenzte formale und semantische Elemente in die universelle grammatische und semantische Or-Om-Struktur des unendlichen Zeichens (C∞1) incodiert.

Durch diese semantische Incodierung werden alle semantischen Dimensionen aller Zeichen des Unicode, alle ihre Kombinationen in Begriffen, Wörtern, Sätzen und deren grammatischen Regeln irgendeiner auch künftigen Sprache in den unendlichen Grundlagen des *Universal Language Codes (ULC)* und des Zeichens (C∞1) **relativiert.**

Die bisherigen semantischen Dimensionen aller Zeichen und ihrer Beziehungen zueinander erfahren eine evolutionäre Überschreitung und Wandlung innerhalb des Or-Om-Code ∞.1.0. Der neue Code bildet auch die Grundlage für die Herstellung bisher nicht erkannter Gesetze der Balance und **Harmonie.**

Was ist Or-Omcode ∞.1.0?

- <http://de.wikipedia.org/wiki/Universalsprache>

In den umfangreichen Schriften des Gottfried Wilhelm Leibniz finden sich Überlegungen hinsichtlich einer konstruierten Universalsprache. Leibniz verstand darunter eine *characteristica universalis* (auch Universalmathematik), als einer Algebra, die in der Lage wäre, alle denkbaren Begriffe auszudrücken. Diese Algebra sollte nach seinen Vorstellungen Regeln für eine symbolische Handhabung, die er *calculus ratiocinator* nannte, umfassen. Sein Plan war es, das Den-

ken auf eine sichere Basis zu stellen, indem viele Teile desselben zur Funktion einer Kalkulation würden, welche leicht verständlich wäre. Die *characteristica* sollten auf einem Alphabet des menschlichen Denkens aufbauen.

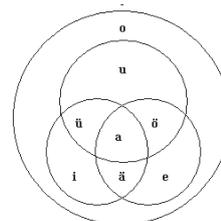
Dieser Leibniz'sche Plan wurde durch die Wesenlehre des Philosophen Karl Christian Friedrich Krause (1781 bis 1832)

http://de.wikipedia.org/wiki/Karl_Christian_Friedrich_Krause realisiert. Die Wesenlehre ergibt eine evolutiv neue Stufe der Erkenntnistheorie, welche das menschliche Bewusstsein bis zur wissenschaftlichen Erkenntnis der unbedingten und unendlichen Göttlichen Wesenheit führt, einer neuen Grundlage aller Wissenschaften. Alle begrifflichen Erkenntnisse aller Wissenschaften, besonders der Logik, Mathematik, Physik, Biologie, Theologie (Religion), Ethik, Ästhetik und alle Grundlagen neuer Sozialformen, die zu einer harmonisierten Menschheit führen, werden aus dieser Wesenheit abgeleitet. Diese strukturellen Ableitungen in der Göttlichen Wesenheit ergeben neue unendliche und unbedingte Begriffe (Kategorien). Diese neuen Begriffe müssen in einer neuen Universalssprache dargestellt werden, und diese Sprache muß Buchstaben, Figuren und Symbole benötigen, - den *Universal Language Codes (ULC)*.

Diese Sprache und ihr Code - abgeleitet aus der unendlichen und unbedingten Wesenheit - überschreiten alle früheren traditionellen Sprachen, ihre Grammatik und Semantik. Alle bisherigen Buchstaben, Schriften, Symbole, Operatoren, Glyphen und Ideographie können und sollten in den Or-Om-Code ∞.1.0 inkodiert werden, weil dieser Code die höchste Stufe sozial sedimentierter und wissenschaftlicher Sprachbildung darstellt.

Wie in Form der obigen Pyramide angedeutet, beschäftigen sich etwa die modernen Systeme der Logik und Mathematik und ihre Sonder-sprachen nur mit Beziehungen in Linie (3). Keine moderne Logik oder Mathematik berücksichtigt die Implikationen und Ableitungen von Linie (1) über Linie (2) zu Linie (3). Das Fehlen der fundamentalen logischen Beziehungen zwischen o, u, i und e und ihren internen semantischen Implikationen in den modernen Systemen führt zu schweren Mängeln, Unvollständigkeiten und Begrenzungen der Wissenschaftssprachen.

Die logischen Beziehungen zwischen den Linien (1), (2) und (3) sind im folgenden Zeichen des **Or-Om-Code ∞.1.0** dargestellt. Hier dürfen aber zum genauen Verständnis nicht die Grundlagen der modernen Mengenlehre für das Verständnis herangezogen werden, sondern nur die Ableitungen, die in den beiden obigen Aufsätzen erwähnt sind!



- <http://portal.or-om.org/science/NeueLogik/tabid/6066/Default.aspx>

- <http://portal.or-om.org/science/GrundlagenderMathematik/tabid/6294/Default.aspx>

http://portal.or-om.org/art/UnicodemeetsOrOmCode/tabid/6319/Default.aspx

PCNEWS.AT

Lachen müsst' man, wenn's nicht zum Weinen wär!

Studien...

Oskar Wagner

Die „**Umweltepideziologische Untersuchung der Krebsinzidenz in den Gemeinden Hausmannstätten und Vasoldsberg**“ von Dr. Gerd Oberfeld veröffentlicht Ende Jänner 2008, sorgte national und international für große Aufregung. Beiträge wie die nachfolgenden in den namhaften Printmedien erregten Aufsehen und verunsicherten erhebliche Teile der Bevölkerung enorm.

Studie: Handy-Masten verursachen Krebs (Presse, 1. Februar 2008)¹

Im Umkreis von 200 Metern ist das Krebsrisiko besonders hoch, bestätigt nun eine Grazer Studie.

Neue Studie: Krebsrisiko durch Mobilfunk (Wiener Zeitung, 2. Februar 2008)²

Graz. Ein „signifikant erhöhtes Krebsrisiko im Umkreis von 200 Metern durch Mobilfunkstrahlung“ hat der Salzburger Umweltmediziner Gerd Oberfeld in einer vom Gesundheitsressort des Landes Steiermark im Jahr 2005 in Auftrag gegebenen Studie festgestellt.

Die „**Schweizer Interessensgemeinschaft Elektromog-Betroffener**“ führt mit Datum 21. Februar 2008 im Internet³ aus: „... die erste echte Langzeitstudie mit über 1300 Teilnehmern über eine Zeitspanne von 13 Jahren zeigt erschreckende Resultate. Ein C-Netz-Sender (434,2MHz) verursachte bei vergleichbarer Sendeleistung mit einem heutigen Mobilfunksender für die Anwohner das 23-fache Brustkrebsrisiko und das 121-fache Gehirntumorrisiko. Das allgemeine Krebsrisiko für alle Arten betrug das 8,5-Fache...“ (Anmerkung O.Wagner: Die angegebene Frequenz 434,2 MHz entspricht Kanal 46 des Industriestandards für die 69 Sprechfunkkanäle im LPD-Bereich. Im C-Netz wurden die Frequenzbereiche 451,30–455,74 MHz und 461,30–465,74 MHz verwendet.)

Durchgeführt wurde diese Studie, die sich auf den Raum Vasoldsberg/ Hausmannstätten (Bezirk Graz-Umgebung) bezieht, weil die Bevölkerung einen Verdacht auf eine Häufung von Krebserkrankungen äußerte. Untersucht wurde das Gebiet in einem Umkreis von 1.200 Metern rund um das Festnetzährlant in Hausmannstätten. Das Ergebnis der Studie war: Ein „signifikant erhöhtes Krebsrisiko im Umkreis von 200 Metern durch Mobilfunkstrahlung“. Bei der Ursachenermittlung war man davon ausgegangen, dass eine Sendeanlage für das 1997 abgeschaltete C-Netz der Verursacher sei. Das Foto im veröffentlichten Bericht zeigt eine provisorische D-Netz-Anlage, die am Standort im Jahr 1994 für 6 Monate installiert war. Danach wurde diese D-Netz-Antenne auf einem 33 m hohen Rohrmasten am selben Grundstück aufgestellt⁴. Das D-Netz, der vorübergehende Nachfolger des C-Netzes, wurde mit Ablauf des 28. Februar 2002 eingestellt und arbeitete in den Frequenzbereichen 890 MHz bis 905 MHz und von 935 MHz bis 950 MHz.

Die „mobikom austria“ - Eigentümerin des betroffenen Standortes - forderte Dr. Oberfeld gemeinsam mit allen Mobilfunkunternehmen wiederholt zum öffentlichen Widerruf dieser Studie auf, da an dem betreffenden Standort niemals eine Mobilfunkanlage für das C-Netz betrieben wurde. Aufgrund der Beweislage sah sich das Land Steiermark veranlasst, am 13. März 2008 die Studie von der Website des Gesundheitsressorts zu entfernen.⁵ Die Studie selbst wurde aber vom Autor nicht widerrufen, sodass sich die „mobikom austria“ genötigt sah, den Rechtsweg zu beschreiten. Der Geschäftsführer des „Forum Mobilkommunikation“ kritisiert: „Krebs auf eine nicht-existent Mobilfunkanlage zurückzuführen lässt jegliche objektive und

sorgfältige Herangehensweise vermissen und hat leider auch negative Auswirkungen auf die Mobilfunk-Diskussion.“⁶ Durch einen gerichtlichen Vergleich wurde nunmehr die Nichtexistenz der Mobilfunkanlage festgestellt. Darin wird ausgeführt:

„1. Der Beklagte, Herr Dr. Gerd Oberfeld, nimmt zur Kenntnis, dass sich im Bereich des Wählamtes „Schemmerlstraße“ in Hausmannstätten, Steiermark, zu keinem Zeitpunkt eine „C-Netz“-Mobilfunkanlage befunden hat. Der Beklagte verpflichtet sich, es ab sofort zu unterlassen, die Tatsachenbehauptung, dass sich am vorgenannten Ort eine solche Mobilfunkanlage befunden hätte, welche auf einen kausalen Zusammenhang mit dem gehäuftem Auftreten von Krebserkrankungen schließen lässt, oder eine Behauptung sinnigen Inhaltes, insbesondere auch im Rahmen seiner „Umweltepideziologischen Untersuchung der Krebsinzidenz in den Gemeinden Hausmannstätten & Vasoldsberg“ aus Jänner 2008, zu verbreiten.

2. ...Salzburg, am 3. November 2008“

Die Reaktionen darauf waren bisher eher verhalten! Es wäre sehr zu wünschen, dass die Printmedien die Unrichtigkeit dieser Studie ebenfalls so markant veröffentlichten würden wie deren Verlautbarung! Die wohl wichtigste Aussage trifft das Bundesministerium für Verkehr, Innovation und Technologie, nämlich: „Zu dieser Studie wird festgestellt: Die Expositionsangaben sind falsch – zur angegebenen Expositionszeit war kein C-Netz-Sender vorhanden. Die Ergebnisse sind daher irrelevant.“⁸

Dr. Martin Rössli vom Institut für Sozial- und Präventivmedizin der Universität Bern untersuchte das Gutachten Dr. Oberfeld's im Auftrag der Krebsliga Schweiz und kam am 25. Februar 2008 zum Schluss: „... Dass Mobilfunkbasisstationen das Krebsrisiko in diesem Ausmaß erhöhen, ist jedoch unplausibel. Wäre dies nämlich der Fall, müsste in den letzten Jahren, parallel zum Aufbau der Mobilfunknetze, eine drastische Zunahme der Krebserkrankungen beobachtet worden sein. Dies war nicht der Fall. Die Hauptschwäche der Studie liegt darin, dass nur ein Gebiet untersucht wurde, bei dem schon vor Beginn der Untersuchung eine auffällige Häufung von Krebsfällen bestand. ...“⁹

Das „UHF-70cm-Band“, das insgesamt den Bereich von 400 bis 470 MHz umfasst, steht national und international einer Menge von Anwendern zur Verfügung. Zwischen 410 und 430 MHz befinden sich analoge und digitale Bündelfunksysteme für private Nutzung und für Anbieter von Mobilfunkdiensten, sowie Richtfunk bei stationären Anwendungen. 430 bis 440 MHz sind neben den lizenzierten Funkamateuren auch den „Industrial, Scientific & Medical“ - Anwendungen (ISM), „Low Power Devices“ (LPD) bzw. „Short Range Devices“ (SRD) von 433,05 bis 434,79 MHz und für Personrufanlagen von 439,11 bis 439,99 MHz (die auch in Krankenhäusern verwendet werden) vorbehalten. Anschließend folgt der Sprechfunk „Private Mobile Radio“ (PMR) von 446,0 bis 446,1 MHz. Der Bereich 449,775 bis 449,850 MHz dient Funkfernsteuerungsanlagen zur Übertragung medizinischer Messwerte(!) und zur Fernsteuerung von Verkehrssignalanlagen.¹⁰ Nachfolgend einige Anwendungen des ISM-Bereichs: „Babyphon“, drahtlose Rufanlagen, Funkfernsteuerer für Beleuchtung und Funksteckdosen, Fernthermometer, Funkfernsteuerung der Zentralverriegelung beim Kraftfahrzeug und vieles andere mehr. Diese Aufzählung ist keineswegs vollständig!

Über einen anderen, aber doch zum Obigen passenden Fall berichtet das Forum Mobilkommunikation am 14. November 2008¹¹:

„Klage gegen T-Mobile und mobikom austria wurde abgewiesen

Ein seit mehreren Jahren laufendes Gerichtsverfahren gegen zwei österreichische Mobilfunknetzbetreiber hat vor dem Landesgericht Eisenstadt ein Ende gefunden. Basierend auf mehreren Fachgutachten

wies das Gericht die Klage des Gemeindefarztes Dr. Reinhold Jandrisovits ab. Der Arzt hatte die Netzbetreiber in seinem wie auch im Namen einiger Gemeindeglieder auf Unterlassung des Betriebs der Mobilfunkanlagen sowie auf Schadenersatz geklagt und im Laufe des Verfahrens verschiedenste Theorien als Gründe für die vermuteten gehäuftem Erkrankungen in seiner Gemeinde präsentiert.

Die im Auftrag des Gerichts eingeholten umweltmedizinischen sowie elektrotechnischen Fachgutachten konnten jedoch keinen Zusammenhang mit dem Betrieb der Sendeanlagen erkennen. Das Gericht befand abschließend, dass nicht festgestellt werden konnte, dass die von den Basisstationen in Müllendorf ausgehenden Immissionen die Beschwerden des Klägers und die damit in Zusammenhang stehenden Schäden verursacht haben oder verursachen. Das Urteil ist rechtskräftig.“

Weltweit gibt es über 1,4 Millionen Basisstationen und ihre Zahl nimmt mit der immer intensiveren Nutzung der mobilen Kommunikation ständig zu. Die Weltgesundheitsorganisation (WHO)¹² hat sich bereits mehrfach mit gesundheitlichen Fragen rund um den Betrieb von Mobilfunk-Basisstationen beschäftigt und im Mai 2006 ein neues Faktenblatt - Nr. 304 - unter dem Titel „Basisstationen und drahtlose Technologie“ herausgegeben. Fazit der WHO: „In Anbetracht der geringen Expositionsstärke und der bisher erschlossenen Forschungsergebnisse gibt es keinen überzeugenden wissenschaftlichen Nachweis, dass schwache HF Signale von Basisstationen und drahtlosen Netzwerken negative gesundheitliche Auswirkungen haben.“¹³

Nach der Verkehrsunfallstatistik geschehen in Österreich die meisten Unfälle mit Fußgängern auf Schutzwegen! Der Logik der beiden in diesem Beitrag oben genannten Medizinern folgend, müssten diese Schutzwege für Fußgänger als besonders gefährlich verboten werden. Das Oberfeld-Gutachten kommentierte ein Leser im Online-Standard schon am 9. März 2008 noch bissiger: „Ich kenne eine Gegend im Burgenland dort ist die Krebsrate fast doppelt so hoch wie im österreichischen Schnitt. Und dort haben die Weinbauern fast alle rote Traktoren mit dem Kennzeichen ND... Ergo: Rote Traktoren mit Kennzeichen ND...erhöhen das Krebsrisiko enorm.“¹⁴

1 <http://www.wienerzeitung.at/DesktopDefault.aspx?TabID=3932&Alias=WZ0&cob=325772&currrentpage=8>

2 <http://diepresse.com/home/panorama/oesterreich/359856/index.do?from=simarchiv>

3 <http://www.gigahertz.ch/1278/>

4 <http://www.presetext.at/pte.mc?pte=080225022>

5 http://www.fmk.at/content.php?id=366&cb=229_1042, <http://www.fmk.at/media/pdf/pdf799.pdf>

6 <http://www.presetext.at/pte.mc?pte=080623035>

7 <http://www.fmk.at/content.php?id=366>

8 <http://www.bmvit.gv.at/telekommunikation/funk/mobiltelefonie/downloads/wbfgesundheit2008.pdf>

9 http://www.mobile-research.ethz.ch/var/Kommentar_Roeesli_oberfeldstudie.pdf

10 siehe Frequenznutzungsverordnung 2005, BGBl. II Nr. 307/2005

11 http://www.fmk.at/content.php?id=249&cb=166_1097

12 Die Weltgesundheitsorganisation, engl. World Health Organization, ist eine Sonderorganisation der Vereinten Nationen mit Sitz in Genf, Schweiz. Sie wurde am 7. April 1948 gegründet, zählt 193 Mitgliedsstaaten und ist die Koordinationsbehörde der Vereinten Nationen für das internationale öffentliche Gesundheitswesen

13 <http://www.who.int/mediacentre/factsheets/fs304/en/> die deutsche Version findet sich auf der Homepage des schweizerischen Forum Mobil unter http://www.forummobil.ch/de/service_content_detail.php?id=306&lan=de

14 <http://derstandard.at/?id=3242334>

Scotty mobil

Walter Riemer

Seit kurzer Zeit bieten die ÖBB unter dem Schlagwort "scotty" verschiedene Dienste im Internet an, zum Beispiel den jetzt auf "Routenplaner" erweiterten, jetzt SCOTTY genannten Fahrplan auf http://fahrplan.oebb.at/bin/query.exe/dn?L-vs_addr, aber auch das Gegenstück "Scotty mobil" für Mobiltelefone, insbesondere auch Smartphones, sofern sie javafähig sind.

Natürlich kann man mit jedem Browser aus dem Smartphone heraus auch die normale Fahrplanabfrage aufrufen

(<http://www.oebb.at/de/> oder auch das "nicht-mobile" SCOTTY auf

http://fahrplan.oebb.at/bin/query.exe/dn?L-vs_addr). Dann hat man aber ein auf normale Computer-Displays zugeschnittenes Bild, in dem man entweder nichts lesen kann oder aber eben nur einen Ausschnitt sieht, sodass man leicht etwas übersieht. So war zum Beispiel aus dem neben-

stehenden Bild nicht direkt zu erkennen, dass die Verbindung um 14:39 eine Bahnverbindung, die um 14:44 aber eine Busverbindung war, weil das in der nicht mehr dargestellten nächsten Spalte rechts stand. Konsequenz: Weder der Bus noch der Zug wurde „erwischt“.

Scotty mobil greift offensichtlich zur selben Datenbank zu, liefert aber die Informationen in einem an das kleine (hochformatige) Display eines Mobiltelefons sehr gut angepassten Layout. Dazu kommen noch Möglichkeiten, Kartenmaterial herunterzuladen, sodass man sich auch auf Fußwegstrecken nicht mehr verirren sollte.

Die Installation läuft weitestgehend automatisch ab, sofern man mit dem Smartphone-Browser <http://scottymobil.oebb.at> angesteuert hat. Man sollte vorher ein Java-Runtimesystem (MIDlet-Manager, in meinem Fall Jbed) installiert haben; im Installationsprozess gibt es zwar einen (recht unpräzisen) Hinweis, dass zusätzliche Software des Mobiltelefonherstellers nötig sein wird (eben das Java-Runtimesystem; das hätte man schon deutlicher machen können!). Was allerdings der wenig geschulte Handy-Benutzer dann konkret macht, ist natürlich nicht Scottys Angelegenheit.

Scotty gibt Auskunft über die aktuellsten Fahrpläne in Echtzeit, sodass zum Beispiel sogar geänderte Abfahrts- und Ankunftszeiten oder Verspätungen abgefragt werden können, und das nicht nur für die ÖBB, sondern für alle öffentlichen Verkehrsmittel (da die ÖBB ja ein Muster an Pünktlichkeit sind, hat der Autor diese Behauptung noch nicht verifizieren können). Sogar Fußwege werden auf Karten dargestellt-Verirren also unmöglich!

Näheres fährt man auf

<http://www.oebb.at/pv/de/Servicebox/Fahrplanabfrage/Bedienungshinweise/index.jsp>.

Eingegebene Daten (zum Beispiel Start- und Zielpunkte) werden gleich online geprüft und, sofern neu, auch offline (für eigene spätere Abfragen) gespeichert.

Hinfahrt	Bahnhof/Haltestelle	Datum	Zeit	Prognose	Gleis	Verkehrsmittel
Korneuburg	Wien Handelskai (Bahnsteige 1-2)	13.12.08	ab 09:10		2	R 2211
			an 09:22		2	
Bemerkungen: Regionalzug Thayatal Richtung: Payerbach-Reichenau nur 2. Klasse; Fahrradmitnahme begrenzt möglich; Rollstuhlstellplatz; Rollstuhlstellplatz - Voranmeldung unter 05-1717 erforderlich; Niederflurfahrzeug						
Wien Heiligenstadt	Wien Handelskai (Bahnsteige 11-12)		ab 09:29		12	S 45
			an 09:33		3	
Bemerkungen: S-Bahn Richtung: Wien Hütteldorf nur 2. Klasse; Fahrradmitnahme begrenzt möglich; Rollstuhlstellplatz; Rollstuhlstellplatz - Voranmeldung unter 05-1717 erforderlich; Niederflurfahrzeug; bedingt rollstuhlgängliches WC						
Wien Nußdorf	Wien Heiligenstadt		ab 09:38		1	S 40
			an 09:40		1	
Bemerkungen: S-Bahn Richtung: Tulln/Donau nur 2. Klasse; Fahrradmitnahme begrenzt möglich						
Dauer: 0:30; Verkehrstage						
17 Tarif km Ticket und Preise Selbstbedienungsstrecke. Bitte den Fahrausweis vor Fahrtantritt kaufen.						
Zwischenhalte einblenden Karte Druckansicht Seitenanfang						
Weitere Informationen						

Beispiel für eine ÖBB-Fahrplanabfrage mit SCOTTY über einen PC-Browser

Wer seit Jahren die ÖBB-Fahrplanabfrage benutzt, wird erfreut sein über die wesentlich übersichtlicher gewordene Bedienbarkeit von Scotty (sowohl im normalen Internet http://fahrplan.oebb.at/bin/query.exe/dn?L-vs_addr als auch insbesondere am Handy).

Bleibt noch als Letztes zu klären, wie man sein schon installiertes Scotty bequem aufruft (<http://scottymobil.oebb.at> ist keine gute Wahl, das will nämlich jedes Mal neu installieren). Außerdem ist ja SCOTTY mobil eine Java-Applikation, also von Browsern völlig unabhängig.

Der MIDlet-Manager `jbed.exe` ist mit dem eigentlich hervorragenden Dateimanager Total Commander nicht zu finden (wohl wegen eines "hidden"-Attributs). Ausnahmsweise ist das WM6-Bordmittel Datei-Explorer diesbezüglich überlegen:

Im Windows-Verzeichnis findet man `jbed`, `jbed.exe`, `jbed.0409.mui` und `jbed-he1p.0409.htm` (letzteres sollte man lesen, wenn man endlich Näheres über MIDlet = *Mobile Information Device applications (applet)* erfahren will).

Im spb Pocket Plus findet man `jbed.exe` leider nicht unter "Programme", wohl aber wenigstens unter "Custom"-Add-File, sodass man einen Shortcut anlegen kann. Im Start-Menü, wo ja generell alle zuletzt aufgerufenen Programme stehen (sollten), verbirgt sich Jbed (wohl wegen Microsofts Abneigung gegen Java) auch.

Wenn man so Jbed aufruft, kann man (bei mir) zwischen Opera Mini und SCOTTY mobil wählen und wird tatsächlich verbunden. Wahnsinnig praktisch ist diese zweistufige Art, SCOTTY aufzurufen, natürlich nicht. Eine einstufige Lösung wie bei Opera Mini ist sicher möglich, aber das muss noch erforscht werden.

Etwas störend ist noch, dass man, wenn man eine neue Route abfragt (die alten werden ja im Smartphone gespeichert!) immer wieder die gleichen "Permissions" geben muss; daran ist noch zu arbeiten.

Über das zu erwartende (relativ bescheidene) Datenvolumen erfährt man folgendes:

Inhalt	Datenmenge
Download der Applikation	ca.210k
Fahrplandaten aktualisieren	ca.2-10k
Karten (pro Zoomstufe/ Maßstabsänderungen)	ca.100k
Echtzeitinformationen abfragen	ca.0,2-3k
News,RSS-Feed aktualisieren	ca.0,2-3k

Wenn man also zum Beispiel 4 Cent je 32 kB-Datenblock zahlt (Tarif bob-Vierer), kostet eine Fahrplanabfrage bzw. -aktualisierung ungefähr 1 Cent (oder weniger); Karten allerdings sind teurer, etwa 12 Cent je Aktualisierung.

In der Praxis schaut eine Abfrage so aus:



So meldet sich SCOTTY am Anfang der „Session“. Man wird meist Fahrplanauskunft wählen.



Landeshauptstädte und eigene frühere Ziele stehen dann zur Auswahl. Natürlich kommt auch „Neuer Ort“ in Frage.



Nach wenigen Eingaben und Klicks kann man die gewünschte Verbindung auswählen



und sich alle Details übersichtlich darstellen lassen.

Was sich hinter „Bahnsteige 1-“ in der vierten Zeile verbirgt, konnte am Smartphone nicht eruiert werden, auch nicht in Landscape-Darstellung (Display auf Querformat umgeschaltet), da wird nämlich das SCOTTY-Bild auch nicht breiter. Am PC zu Hause kann man lesen: 1-2 (was immer das wieder heißen soll?) bzw. 4 Zeilen weiter unten: 11-12.

Immerhin: Eine große Hilfe kann die Auskunft schon sein!

http://www.operamini.com/

CLUB PocketPC.AT

Sony Ericsson Xperia X1

– frischer Wind am Pocket PC Markt

Paul Belci

Dank Jürgen von A1, darf ich das Sony Ericsson Xperia, welches ich für einen Clubabend ausgeliehen habe, schon einige Tage vorher abholen, und so kommt es zu einem Kurzttest

Technische Daten

Die technischen Daten des Xperia lesen sich gut!

- Qualcomm-Prozessor mit 528 MHz Taktfrequenz
- 3 Zoll Bildschirm: 800 X 480 Bildpunkte, TFT mit 65.536 Farben
- ausschiebbarer QWERTZ-Tastatur mit Hintergrundbeleuchtung
- 512 MB ROM, 256 MB RAM
- Maße: 110,5 x 52,6 x 17,0 mm
- Gewicht: 158,0 g
- Steckplatz für MicroSD Karten (SDHC)
- Quadband, mit HSDPA (7,2 Mbps) und HSUPA (2 Mbps)
- 3,2 Megapixel-Kamera mit Auto-Focus und Fotoleuchte
- Kopfhöreranschluss (3,5 mm Klinenstecker)

Erster Eindruck

Das XPERIA X1 macht einen sehr massiven Eindruck und wirkt sehr gut verarbeitet. Dies liegt sicher auch an dem Aluminiumgehäuse. Der Schiebemechanismus der eingebauten Tastatur ist einer der stabilsten, die ich je gesehen habe.

Im Vergleich zu meinem HTC Touch HD ist das Gerät um ca. 1cm schmaler und ca. 0,5mm kürzer. Dafür hat es nur eine Bildschirmdiagonale von 3 Zoll! (im Vergleich 3,8 Zoll beim HD).

Die Auflösung ist mit 480 x 800 Bildpunkten sehr hoch! Das führt allerdings dazu, dass man den Stift sehr häufig benutzen muss, wenn man die fingerbedienbare Oberfläche verlässt. Denn bei dieser hohen Auflösung wirkt das Bild zwar brillant und gestochen scharf, aber auch etwas klein!

Tastatur

Die Tastatur macht einen sehr brauchbaren Eindruck. Die Tasten haben ausreichend Abstand und sind gut zu drücken. Es sind alle wichtigen Tasten vorhanden, bis auf die STRG Taste. Aber das kennen wir ja schon von diversen HTC-Geräten.

Das äußere Bedienfeld mit den dreieckigen Tasten ist etwas gewöhnungsbedürftig, man ist versucht, die Tasten eher mit dem Fingernagel zu drücken, um keine benachbarte Taste zu treffen, aber dieser Eindruck ist eher unbegründet, denn die Tasten lassen sich gut drücken ;-)

Es gibt keine Tasten am Gerät, die mit eigenen Funktionen belegt werden können, selbst die „Panel“-Taste, welche die Xperia Panel Funktion aufruft, kann in der Systemsteuerung nicht umbelegt werden.

Schade, denn sie würde sich dafür anbieten!

Oberfläche und Software

Bei der Installation meiner Software machte das Xperia einen schnellen Eindruck. Man hat das Gefühl, als hätte Sony Ericsson ausreichend Zeit in die Optimierung des Betriebssystems an die Hardware investiert. Es gibt auch eine neue Oberfläche, die aus so genannten Panels besteht. Man kann diese sehr flexibel nach eigenen Wünschen mit Inhalt füllen. Angefangen von Programmlinks, über Kalender, Uhrzeit und Terminen über RSS-Feeds und einiges mehr!

Die Software reagiert allerdings manchmal etwas träge, wenn man sie mit zu viel Inhalt befüllt, da diese Daten natürlich alle auch geladen und aufbereitet werden wollen.

Leider ist dieses Panel gerade mal zum Starten für gewisse Dinge gemacht, das heißt man kommt sehr bald wieder in das Windows Mobile Menü und dann ist es vorbei mit der Fingerbedienung und man muss zum Stift greifen.

Es haben sich allerdings schon einige Anhänger dieser Oberfläche gefunden und ich gehe davon aus, dass sich hier noch einiges tun wird.

Telefon, GPS und Sync

Der Telefonteil funktioniert problemlos. Ich habe sogar das Gefühl als würde der Telefonteil etwas besser funktionieren als bei den HTC Geräten. Interessant ist auch die Signalisierung für ankommende Anrufe. Vier seitlich eingebaute grüne Leuchtdioden blinken unübersehbar. Wird der Anruf nicht entgegengenommen, blinken die LEDs gelb weiter, um anzuzeigen dass ein Anruf verpasst wurde. Nette optische Spielerei...

Auch das eingebaute GPS hat in meinem Test keine Probleme gemacht. Einzig beim Datentransfer mit HSDPA habe ich festgestellt, dass das Xperia relativ oft auf 3G (UMTS) verbunden ist, statt H (für HSDPA). Allerdings hatte ich das Gerät zu kurz im Test, um genaue Erfahrungen zu sammeln. Die Verbindung funktionierte trotzdem meist ordnungsgemäß.

Zum Syncen via Kabel gibt es einen Mini-USB-Anschluss, der sich inzwischen bei den meisten Herstellern schon als Standard durchgesetzt hat.

Video, Audio und Foto

Die 3,2 MP Kamera im Xperia ist brauchbar. Eine Digitalkamera wird sie allerdings nicht ersetzen!

Sony Ericsson hat im Xperia angeblich einen schnellen Grafikchip verbaut, der für eine tolle Videoausgabe sorgen soll. Dies bestätigt



auch ein Demovideo des neuen James Bond Filmes im MP4-Format, welches auf dem Gerät vorinstalliert ist. Das Video läuft sauber und ruckelfrei ab, und die Qualität ist dabei auch noch ausreichend gut. Ich habe das Video allerdings auch auf einem Touch HD abgespielt und dort lief es ebenso toll.

Auch die Audioausgabe ist bestens. Durch den eingebauten 3,5"-Klinenstecker kann man jeden beliebigen Kopfhörer anschließen.

Fazit

Das Sony Ericsson Xperia hat mich beeindruckt. Speziell deshalb, weil es das erste Windows Mobile Gerät von Sony Ericsson ist und dafür läuft es wirklich stabil! Speziell die Verarbeitung wirkt sehr hochwertig. Mir persönlich ist der Bildschirm schon etwas zu klein für die hohe Auflösung, die das Gerät hat. Allerdings ist der Formfaktor dadurch unschlagbar. Ein wirklich ernst zu nehmender Konkurrent für das HTC Touch Pro!



Opera Mini auf einem WM6.1 Smartphone

Walter Riemer

Nachdem sich der Autor nach jahrelangem hin- und herhaltendem Widerstand doch entschlossen hat, von der herkömmlichen Terminverwaltung mittels faltbaren Taschenkalenders ab- und auf eine zeitgemäße Methode zuzugehen (Pocket LooX C550, VGA-Display), war der nächste nicht unlogische Schritt, sich vom betagten Siemens S55-Handy zugunsten eines Smartphone zu verabschieden. Warum nicht irgendein anderes modernes Telefon?

Erstens, weil inzwischen eine gewisse Vertrautheit mit dem Windows Mobile-Betriebssystem entstanden war.

Zweitens, weil die Firma pearl.de ein erstaunlich günstiges und sehr kompaktes derartiges Gerät anbot - der Autor gehört nicht zu jenen Leuten, die gern Hunderte EURO für solche Sachen ausgeben. Das vertragsfreie "simvalley X-25" ist um rund EUR 150 zu haben; das Modell XP-25 ist dabei nur das einfachste einer Serie, es kann GPRS 10; die -45 und -65 haben zusätzlich WLAN bzw. das Topmodell auch noch GPS eingebaut. Das Gerät hat auch die Erwartungen durchaus erfüllt.

Das Verwalten von Terminen und Kontakten ist damit auf drei Rechnern (Notebook, LooX und simvalley) synchron problemlos möglich; alle Offline-Kontakte stehen auch im Telefon (unter der Software Pocket Informant) zur Verfügung; auch dass man am Morgen beim Einschalten des Telefons gleich an gültige Termine erinnert wird, erscheint sehr angenehm.

Eine Ergänzung des Betriebssystems WM6.1 durch einen "Personal Information Manager" (bei mir Pocket Informant) und hinsichtlich des Telefons mit "spb Phone Suite" und einigen anderen Dienstprogrammen, zum Beispiel insbesondere auch spb Pocket Plus, ohne das man ja kaum leben möchte, ist sicher anzuraten, aber dann hat man schon ein ziemlich komplettes, übersichtliches und leistungsfähiges System.

Das alte Handy S55 war zwar durchaus WAP-fähig, aber die mannigfachen Unbequemlichkeiten hielten den Autor verlässlich davon ab, dies wirklich zu benutzen. Hurra - so ein Smartphone kann natürlich viel mehr. Dank GPRS-Internetzugang kann man auch einen Kontakt finden, den man nicht offline in der Datenbank hat. Und dann noch dazu der auf Handys zugeschnittene Browser "Opera Mini", was will man mehr (ausgedehnt surfen oder recherchieren macht man sowieso zu Hause). Eine Adresse oder Telefonnummer von Herold jederzeit bekommen zu können, wäre schon schön!

Was kann das Opera Mini?

Abgesehen von Display- bzw. Layout-Anpassungen an den kleinen Handy-Bildschirm (beim simvalley 240x320, aber in hervorragender Qualität), reduziert Opera Mini vor allem den Datenverkehr auf unglaublich schlaue Weise (und erhöht dadurch natürlich auch die Geschwindigkeit, auch wenn entsprechend verkleinerte Bilder geladen werden):

Sobald vom Handy aus eine URL aktiviert wird, sendet Opera Mini einen Request an einen Server im Hause Opera. Der holt sich die Originalseite, codiert sie in ein viel sparsameres Format um und lässt das Ergebnis in Opera Mini am Handy anzeigen.

Also: Opera Mini muss her! Leider ist das nicht so einfach, wie man es sich wünschen würde. (Es ist schon relativ einfach, wenn man weiß, wie!)

Opera bietet nämlich eine ganze Palette von Telefonen, mit denen man das passende Opera Mini herunterladen kann - WM6-Smartphone ist nicht dabei, es wird nur eine "Generic-Version" angeboten.

Opera Mini ist ein Java-Programm; Microsoft ist ein Java-Feind, weshalb es in WM6.1 kein Java Runtime System gibt. Also muss ein Fremdprogramm her.

Die diversen Foren geben verlässliche Informationen nur sehr ungern her, und so muss man verschiedenes erfolglos probieren, bis man bei zu hoch gewordenem Frustrations-Level aufgibt. Die MIDlet-Manager (= Java Runtime System), die da empfohlen wurden (dVZ i300 Java Installer, IntentMIDlet Manager von Risidoro usw.) weigerten sich jedenfalls, das generische Opera Mini angemessen zu betreuen, oder besser: sie sagten nicht einmal "das geht nicht", sondern schlummerten nur vor sich hin.

Allein, manchmal findet auch eine blinde Henne ein Körnchen (Wahrheit), und so stieß der Autor endlich einmal doch auf kompetente Informationen: <http://www.smartphonemag.com> (Smartphone & Pocket PC magazine) hat sie und gibt sie auch bereitwillig her. Dort gibt es insbesondere sehr ausführliche Artikel darüber, wie man vorgehen sollte, was zu beachten ist und was dahinter steckt; der entscheidende Kopf dafür ist offensichtlich Herr Werner Ruotsalainen:

http://www.smartphonemag.com/cms/blogs/3/opera_mini_4_1_full_multiplatform_tutori

<http://www.smartphonemag.com/cms/blogs/3/2302>

http://www.smartphonemag.com/cms/blogs/3/tutorial_control_issues_of_java_midlets

Das Wichtigste, das man dort erfährt, ist, kurz zusammengefasst:

1. Woher bekommt man eine geeignete Java Runtime Umgebung?

Die vorstehenden angeführten Unterlagen schwören auf Jbed. Das kostet bei <http://www.handango.com> \$ 49.95. Gratis (und noch dazu eine jüngere Version) erhält man es auf <http://winmobiletech.com/092007MidletBible/Jbed.zip>.

Also das "Esmertec Jbed 20080222 3.1.cab" auf das Handy kopieren und installieren, vorzugsweise (aber angeblich nicht mehr zwingend) im Speicher, nicht auf der Speicherkarte. 2,6 MB kann man schon riskieren!

2. Woher bekommt man eine geeignete Opera mini-Version?

Auf <http://www.operamini.com/> kann man theoretisch die Version 4.2 "generic version" holen; zur Auswahl stehen Basic MIDP 1 und Advanced MIDP 2 (was immer sich dahinter verbirgt; diesbezüglich ist das vorerwähnte Tutorial schon wieder nicht ganz am letzten Stand). Welche wählen? Alles zeitraubend probieren? Wo erfährt man etwas - wieder das alte Problem!

Besser ist es, von <http://mini.opera.com> Opera Mini direkt auf das Handy zu laden (das geht auch mit dem Internet Explorer, den man ja sowieso auf dem Smartphone hat). Dann geht alles fast von selbst (die etwas mehr als 100 kB kann man auch mit GPRS riskieren; es geht alles blitzschnell!). Man wird an der Hand geführt, bis zum Annehmen der Lizenz und einigen Grundeinstellungen.

Und schon funktioniert die Sache, mit ein paar Einschränkungen, zum Beispiel nicht hundertprozentige Javascript-Unterstützung. Aber, siehe oben: Surfen oder Recherchieren macht man sowieso besser zu Hause:

Sobald der Autor Opera Mini zum ersten Mal verließ, war es allerdings nachher unauffindbar; ebenso wie das Java Runtime Jbed (bei versuchsweisem Neuinstallieren beider Komponenten wurde allerdings gemeldet, man müsse erst die alten Versionen deinstallieren, also waren sie offensichtlich vorhanden - aber wo? Weder im Programm-Ordner noch in spb Pocket Plus war irgendeine Spur zu sehen; selbst Durchsuchen, etwa nach jbed.exe oder opera, ergab nichts.

Die Lösung, auf die man wohl nie allein käme, stammt von (wie könnte es anders sein?) http://www.smartphonemag.com/cms/blogs/3/tutorial_make_opera_mini_4_1_your_default (das fehlende „t“ ist kein Fehler!)

Man holt sich MortScript, entpackt es und installiert MortScript-4.1-PPC.cab auf das Smartphone. Unter "Programme" erscheint ein Ordner mit drei Textdateien, die Mortscript beschreiben, was einem noch nicht sehr viel weiter hilft.

Dann braucht man noch **0.11b_StartOperaMini.zip** (erhält man auf <http://forum.xda-developers.com/showthread.php?t=384271>, wo man sich registrieren muss). Die Vielzahl enthaltenen Dateien kopiert man in einem Ordner irgendwohin auf sein Smartphone; kann auch die Speicherkarte sein; der Autor hat den ganzen Ordner einfach nach Application Data kopiert). Diese enthaltene Datei **StartOperaMini.msccr** führt man dann durch Anklicken aus einem Dateimanager aus und alles Nötige wird erledigt: Es findet Jbed (sehr tüchtig!), den zugehörigen Index von Opera Mini und Wunder über Wunder: Man hat die gewünschten Icons, insbesondere "Start Opera Mini" im Programme-Ordner.

Warum man bei so einer für jeden interessanten oder wichtigen Angelegenheit ein gelernter Informatiker höchsten Intelligenzgrads und unbegrenzter Ausdauer (und unbegrenzter Zeit: im konkreten Fall ungefähr 6 Stunden) sein muss, findet man im ganzen Internet allerdings nicht; aber ohne dieses würde man das Problem überhaupt nicht lösen können!

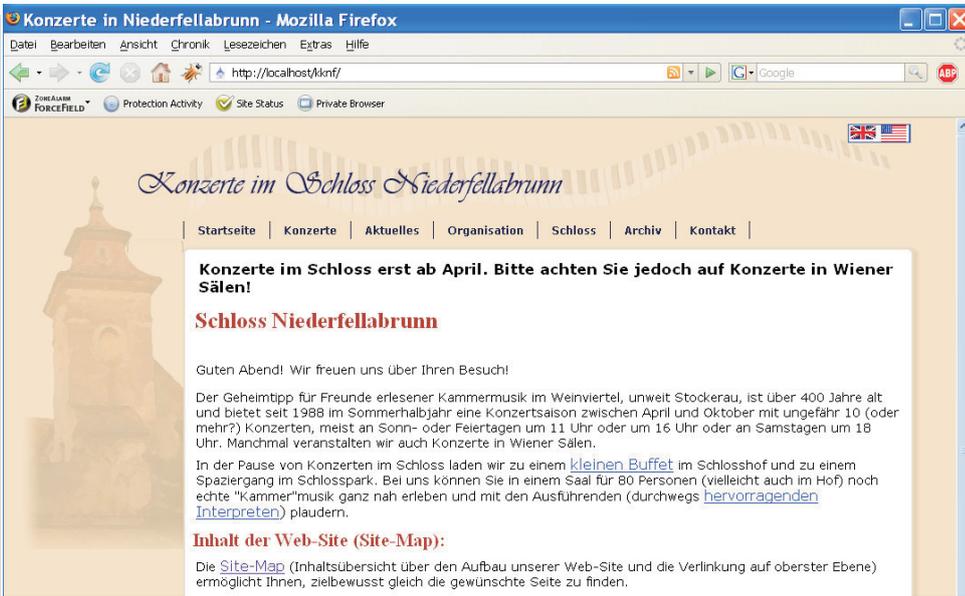
Trotzdem lohnt sich der Aufwand: Dass auch der (normale) Opera-Browser für den Pocket-PC viel besser funktioniert als der Internet Explorer, ist ja bekannt. Angeblich wurde der IE für WM6 auch verbessert. Wie sehr, kann man durch Vergleich mit Opera Mini selbst beurteilen:

Die Startseite der Niederfellabrunn-Website sieht man in der nebenstehenden Abbildung.

Der Internet Explorer stellt sie auf dem Smartphone so dar:



Wenn man die Darstellung verkleinert, schaut es so aus:



Website niederfellabrunn.at

Opera Mini zeigt dagegen folgendes (und das Laden geht fast so schnell wie mit einem Breitbandanschluss, trotz GPRS):



Wenn man die Darstellung vergrößert, um den Text lesen zu können, hat man folgendes Bild:



Bravo, Microsoft! Eigentlich wollte ich schreiben: Bravo Opera!

Zur nebenstehenden Vorstellung des EEE-PC; Kommentar eines Spezialisten

Linux mit Pocket-PC & Handy

1. Alle Handies mit Windows als Betriebssystem (MDA, PDA) funktionieren unter Linux nicht bzw. nur mit einem speziellen Programm eines russischen Programmierers. Davon habe ich die Finger gelassen.
2. Viele Handies kann man mit Xandros-Linux als Modem benutzen, als "Ersatz" für einen UMTS-Stick. Ich benutze dazu ein SE 350 i, auch das SE W580i und das SE 770 funktionieren so, des weiteren viele Nokia-Modelle. Liste auf eeepc.de - Forum.
3. Synchronisieren kann man nur einige wenige Palm-Modelle mit Symbian-BS. aber die sind am Aussterben.
4. Mittels der Linux-Programme Gammu und Wammu (das ist die GUI dazu) soll es zumin-

dest teilweise gehen, aber das habe ich noch nicht ausprobiert.

Insgesamt gesehen denke ich, ist dies der größte Mangel am 701 4G unter Xandros-Linux. Wenn man Windows installiert, funktioniert natürlich alles. Deshalb habe ich noch XP als Zweitsystem installiert (auf dem 901, aber das geht auch auf dem 701, notfalls über USB-Stick).

P.S.: Selbst unter ubuntu-Linux (das ist das windows-ähnlichste Linux) ist dieses Synchronisieren mit dem Handy fast unmöglich. Allerdings sind jetzt schon die ersten Linux-Handies auf dem Markt. Wenn die sich ebenso schnell verbreiten wie die Linux-Netbooks, ist auch dieses "Problem" bald gelöst.

Asus Eee-PC 4G mit Linux

Netbook in Reinkultur

Christian Schneider



Angezogen durch den äußerst attraktiven Preis von nur ca. 210€ habe ich mir bei www.amazon.at einen Asus Eee-Minilaptop gekauft, und ich bin sehr zufrieden damit.

Das Gerät ist trendy, weiß und sehr klein; zum Vergleich habe ich auf dem Bild mein UMTS-Modem beigelegt. Beim ersten Mal Einschalten muss man nur dem Lizenzvertrag zustimmen, Namen und Root-Passwort eingeben und kann sofort arbeiten. Ein umfangreiches Softwarepaket inklusive OpenOffice ist bereits installiert. Die installierten Programme kann man allerdings nicht entfernen, man sollte sich bei „Einstellungen“ die Update-Funktion übers Internet anschauen.

Der Bildschirm verfügt über sechs Desktops (*Internet, Arbeiten, Lernen, Spielen, Einstellungen, Favoriten*), die oben über die Karteireiter durch Anklicken zugänglich sind.

Hier der „Internet“-Desktop, der reichhaltig gestaltet ist.



Internet-Desktop

Nur das installierte Mailprogramm „Thunderbird“ ist bei „Arbeiten“ zu finden:

Hier findet sich auch der Datei-Manager, der ähnlich dem Windows-Explorer funktioniert.



Der Arbeiten-Desktop

Die Internet-Programme sind Firefox2 zum Browsen und Thunderbird für Mail:

Hier sieht man den „Firefox“ mit geöffneter Seite der Eee-Gemeinde www.eee-pc.de, auf der sich viel Material und ein Hilfeforum befinden.



Firefox

Im Vordergrund ist die von Asus bereitgestellte Software für den mobilen Internetzugang, die

einwandfrei funktioniert und sehr einfach zu bedienen ist. Man steckt ganz einfach das UMTS-Modem (ich habe ein Huawei E220 für Teling „Willi“) in einen der drei USB-Anschlüsse, und das Programm startet automatisch. Auch die Druckerinstallation ist sehr einfach. Man steckt alles an, und schon funktioniert es.

Mails kann man mit dem Programm Thunderbird Mail einfach und effektiv verfassen.



Thunderbird

Die Unterhaltung kommt auch nicht zu kurz, speziell für Kinder sind Spiele und Lernprogramme installiert, und mit der eingebauten 0,9 Megapixel Webcam kann man Fotos machen und kurze Filme drehen. Auch ein Malprogramm speziell für Kinder findet man.

Die Office-Aufgaben erledigt OpenOffice 2, das aus zahlreichen Modulen besteht, etwa dem Writer, einer Tabellenkalkulation, einer Präsentationssoftware und vieles mehr... Die Bedienung orientiert sich an Microsoft Office, so dass man gleich loslegen kann:



Open Office Writer

Die Ausstattung des Laptops überzeugt: Die Tastatur ist nach kurzer Eingewöhnungszeit gut zu bedienen, ein Touchpad mit Scrollfunktion erleichtert das Navigieren. Durch die geringe Größe ist der Eee ein immer-und-überall-dabei-Laptop. Der Bildschirm ist klein und hat die relativ geringe Auflösung von 800x480, ist aber ausreichend. Der Akku ermöglicht mehrstündigen Betrieb ohne aufzuladen. Es handelt sich hier um eine vollwertige Arbeitsmaschine, bei der man nichts mehr installieren muss sondern sofort loslegen kann. Selbst einen Beamer kann man problemlos anschließen und mit höherer Auflösung betreiben. Fazit: ein echter Einsteiger-, Kinder- und Immer-dabei-Laptop.

Unterschrift als Font

Franz Fiala

Fallweise langen in der PCNEWS-Redaktion Anfragen über ältere Artikel ein, die in den meisten Fällen auch besorgt werden können. Doch manchmal sind die seinerzeit angewendeten Verfahren überholt.

Eine solche Anfrage war es etwa, eine Unterschrift als True-Type-Font zu speichern. Dieser Vorgang wurde in der PCNEWS-33 im Jahr 1993 mit dem Programm Font-Monger beschrieben.

(<http://pcnews.at/pdf/n330029.pdf>)

Es ist teuer, wegen eines einzigen Fonts gleich ein Programm anzuschaffen. Daher ergab sich die Frage, ob man einen solchen Font nicht auch mit vorhandenen Programmen herstellen könnte. Gefunden wurde eine alternative Herstellungsmethode mit dem verbreiteten Programm CorelDraw (aktuelle Version "Graphics Suite X4").

Herstellung eines Unterschrift-Fonts

Die Schritte zur Herstellung eines Unterschrift-Fonts sind:

1. Scannen der Unterschrift
2. Umrissbildung
3. Exportieren in einen True-Type-Font
4. Test in einem Word-Dokument
5. Verbesserungen

1. Scannen der Unterschrift

Erste Versuche, eine Unterschrift zu Papier zu bringen, zeigen, dass man unbedingt einen gut deckenden Stift und ein glattes Papier verwenden sollte. Gewöhnliches Kopierpapier ist etwas zu rau und daher sind die Umrisslinien nicht glatt. Bleistiftstriche haben stark veränderliche Grauwerte und neigen zum Ausfransen.

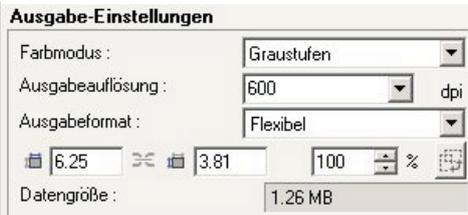


Auswahl der Unterschriftsprobe

Die Einstellungen beim Einscannen sollten sein:

- Grauwerte, 600 dpi, Helligkeit und Kontrast entsprechend einstellen

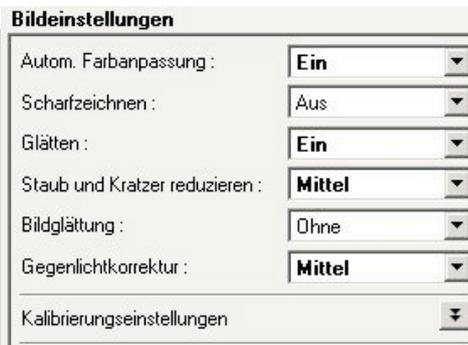
Wenn man die Unterschrift "Schwarz-Weiß" (also ohne Grauwerte) scannt, hat man später nur sehr wenig Spielraum bei der Gestaltung der Umrisslinie. Insbesondere, wenn die Strichfarbe sich ändert oder gar der Strich unterbrochen ist, kann man mit dem Schwarz-Weiß ge-



Scanner-Einstellungen: 600 dpi, Graustufen



Um Grauzonen zu vermeiden und um klare Übergänge zu erreichen, sollten Kontrast und Helligkeit etwas angehoben werden.



Die sonstigen Einstellungen haben eher weniger Einfluss.

scannten Bild ohne händische Nachbearbeitung nicht viel anfangen. Man muss daher in diesem Fall mit Gradation, Helligkeit und Kontrast bereits beim Einscannen das gewünschte Ergebnis bekommen. Flexibler ist es daher, mit Grauwerten zu arbeiten. **(siehe Bilder unten)**

Die entstandene Grafikdatei zieht man auf ein leeres Dokument in der Corel-Oberfläche.

2. Umrissbildung



Bitmap der Unterschrift in Corel-Draw

In früheren Versionen von CorelDraw gab es das getrennte Programm CorelTrace, mit dem es möglich war, aus einer Bitmap-Datei eine Vektor-Datei herzustellen. Das ist mit der aktuellen CorelDraw-Version nicht mehr nötig, die Trace-Funktion ist im Menüpunkt *Bitmaps* integriert.

Es genügt, das Bild zu markieren und mit *Bitmaps* -> *Outline Trace* -> *Line Art...* den Umwandlungsvorgang einzuleiten.

Wenn die Größe des importierten Bildes zu groß ist, bekommt man den Hinweis, dass das Bild auf die maximal zulässige Größe reduziert wird. Man erhält nach einer kurzen Wartezeit folgendes Bild:



Oben: Originalbild, unten: Vektorisierter Umriss, der grau-karierte Hintergrund ist nicht mehr Teil des Schriftzugs.

Mit Grauwerten gescannter Schriftzug

Schwarz-Weiß-gescannter Schriftzug



Der Trace-Dialog bietet mehrere Einstellungen. Es lohnt sich, hier zu experimentieren. Wenn man nicht mehr weiterkommt, kann man mit dem Button *Reset* alle Änderungen rückgängig machen.

Die Einstellungen sind folgende:

Delete original image: Wenn diese Checkbox aktiviert ist, wird das ursprüngliche Bitmap-Bild gelöscht. Solange man experimentiert, sollte man diese Box nicht anklicken.

Remove Background: Der weiße Hintergrund wird entfernt

Automatically choose color: Das Trace-Programm wählt automatisch die Randfarbe als Hintergrund und entfernt sie aus dem Ergebnis.

Specify color: Hier könnte man mit der Pipette eine eigene Farbe als Hintergrundfarbe definieren. Das ist aber nur notwendig, wenn sich das Traceprogramm bei einer komplexeren Vorlage bezüglich des Hintergrunds geirrt hätte.

Remove color from entire image: Aktivieren, um die weißen Flächen in den Schlingen wie den Hintergrund zu behandeln. (Man kann diese Flächen auch noch später in CorelDraw löschen.)

Trace result details: Hier sieht man, aus wie vielen Kurven und Knoten der Linienzug zusammengesetzt ist. Vor allem aber sieht man hier die Anzahl der Farben aus denen sich die Vektorgrafik zusammensetzt (im Beispiel sind es sechs Grauwerte). Das Resultat darf aber nur aus einer Farben bestehen.

Detail, Smoothing, Corner smoothness: Mit diesen drei Reglern kann man den Umriss im Detail beeinflussen. Mit *Detail* wird die Komplexität der Kurve beeinflusst. Wird *Detail* reduziert, wird der Umriss stark vereinfacht. *Smoothing* stellt die Weichheit der Übergänge ein und *Corner smoothness* füllt eventuelle Kanten besser aus. Mit diesen Einstellungen kann man das Aussehen des Schriftzugs in weiten Grenzen beeinflussen.

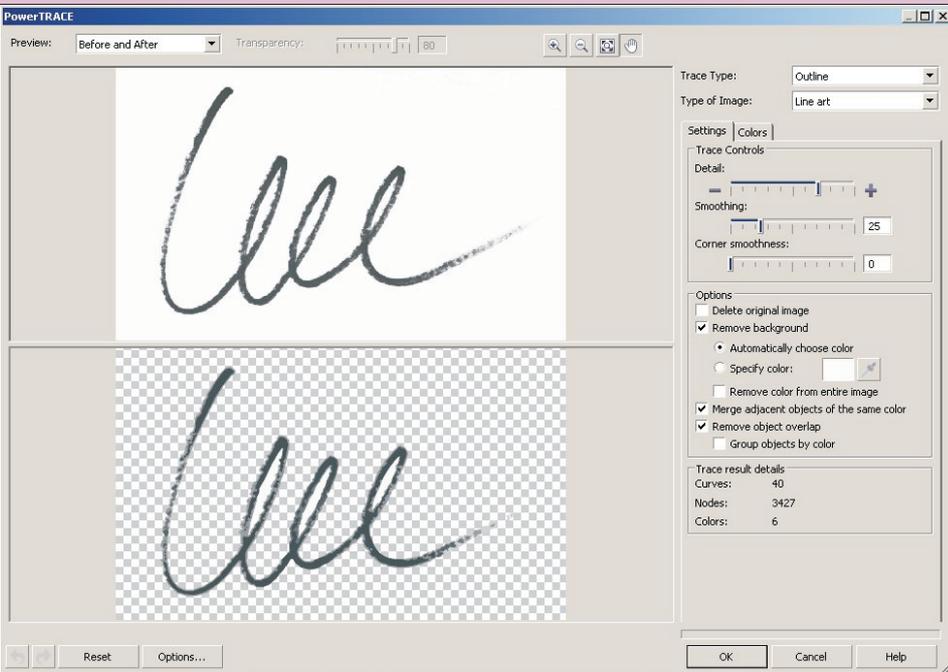
Farbreduktion

Die Einstellungen für *Detail, Smoothing* und *Corner smoothness* beeinflussen auch die Anzahl der Farben im Ergebnis. Es kann daher sein, dass bei ausreichend geringer *Detail*-Einstellung die Farben auf nur mehr zwei reduziert sind. Genau das muss nämlich für die spätere Umwandlung in einen Font gegeben sein.

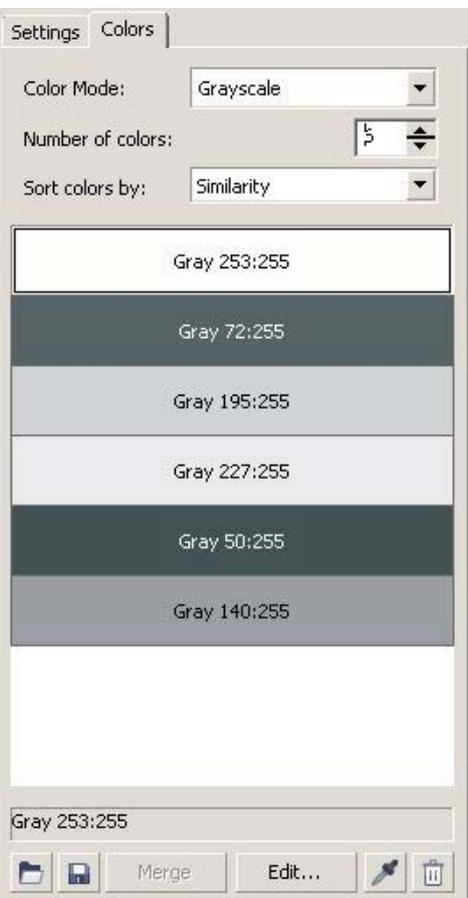
Wenn also die Anzahl der Farben wie im Beispiel noch sechs ist, dann muss man in der Karteikarte *Colors* die Anzahl der Farben auf zwei reduzieren (siehe nebenstehendes Bild). Diese Grauwerte kann man zu einem einzigen Grauwert (egal welchen) vereinheitlichen, indem man mit gedrückter Shift-Taste \square alle Grauwerte markiert und dann auf den Button *Merge* klickt. Im Normalfall wird auf einen mittleren Grauwert eingestellt. Man kann aber unter *Options* auch einstellen, dass alle Grauwerte auf die zuerst angeklickte Farbe reduziert werden. (Das ist aber für das spätere Ergebnis ohne Bedeutung.)

Die Checkbox *"Delete original image"* ist in der Anfangseinstellung nicht angeklickt, daher kommt zu dem ursprünglichen Bitmap-Objekt nach Abschluss das Umrissbildung die Objektgruppe des Umrisses dazu. Man schließt die Operation mit *OK* ab und kehrt zu CorelDraw zurück.

Man hat jetzt zwei übereinander liegende Objekte: das ursprüngliche Bitmap-Bild und das

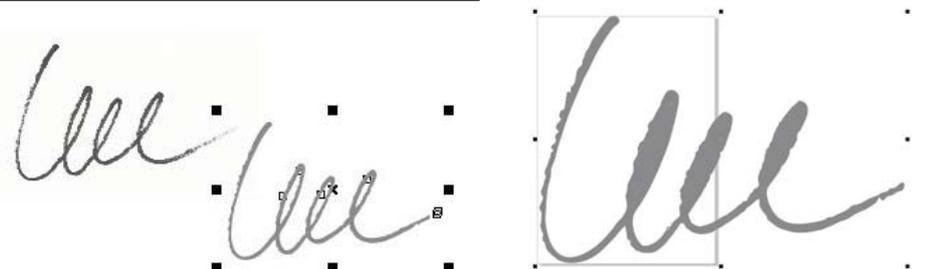


Ergebnis des Tracelaufs auf Grund der Voreinstellungen



Reduktion der Grauwerte auf einen einzigen durch Markieren der Grauwerte und einen Klick auf Merge.

Ursprüngliches Bitmap (links) und vektorisierter Schriftzug (rechts). Man kann die unterschiedlichen Grauwerte durch die Mit-telung mit Merge erkennen



vektorierte Ergebnis. Das Bitmap-Bild zieht man zu Seite und markiert das Vektor-Bild.

In der Statuszeile sieht man, aus wie vielen Teilen das Objekt besteht. Für die Umwandlung in

Group of 6 Objects on Layer 1

Das Umrissbild besteht im Allgemeinen aus mehreren Objekten.

einen Font darf es aus nur einem einzigen Objekt bestehen.

Man untersucht jetzt das vektorisierte Ergebnis in einer vergrößerten Darstellung. Kosmetische Änderungen des Umrisses könnte man mit dem *Shape tool* aus CorelDraw korrigieren.

Jetzt muss man aus den aus möglicherweise mehreren Objekten bestehenden Schriftzug zu einem einzigen Objekt zusammenführen. Dazu löst man mit *Arrange -> Ungroup All* die gruppierten Teilelemente auf und kombiniert die Vektorteilbilder mit *Arrange -> Combine* zu einem einzigen Objekt. Die Statusanzeige muss sein *"Curve on Layer 1"*.

3. Exportieren in einen True-Type-Font

Die Dokumentation dieses Vorgangs in der Corel-Hilfe ist recht dürftig ausgefallen, die Hinweise in diesem Abschnitt ergaben sich erst nach zahlreichen erfolglosen Versuchen mit dem Font-Export.

Bevor man das Vektorbild in einen Font exportiert, muss man es skalieren und zwar so, dass jener Punkt, der auf der *Baseline* (Grundlinie für Kleinbuchstaben) aufsitzen soll, die Unterkante der A4-Seite darstellt; und jener Punkt, der der Höhe der Kleinbuchstaben entsprechen soll die Oberkante der A4-Seite, etwa so:

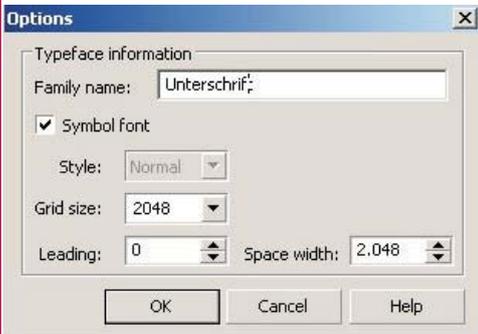
A4-Seite entspricht der Höhe der Kleinbuchstaben

Um das vektorisierte Zeichen zu exportieren, genügt es, den Menüpunkt *Datei -> Export* anzuklicken und als Ausgabeformat True-Type-Font und einen Dateinamen anzugeben.

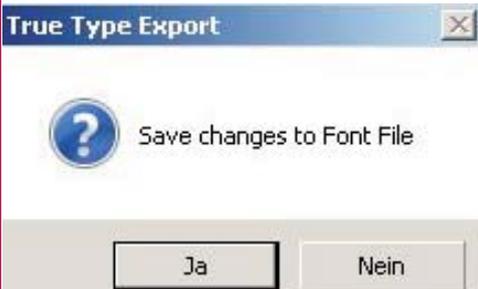
Dateiname:
 Dateityp:

Man wählt einen neuen Fontdateinamen, kann aber auch in einer bestehenden Fontdatei ein beliebiges Zeichen überschreiben.

Im folgenden Dialog wählt man den Fontnamen:



Mit *Symbol font* wird eingestellt, ob der Font ein Alphabet enthält oder spezielle Symbole, die nicht in einem textuellen Zusammenhang verwendet werden können. Bei Symbol-Fonts wird in eine Vorschau ein anderer Zeichensatz zur Benennung verwendet.

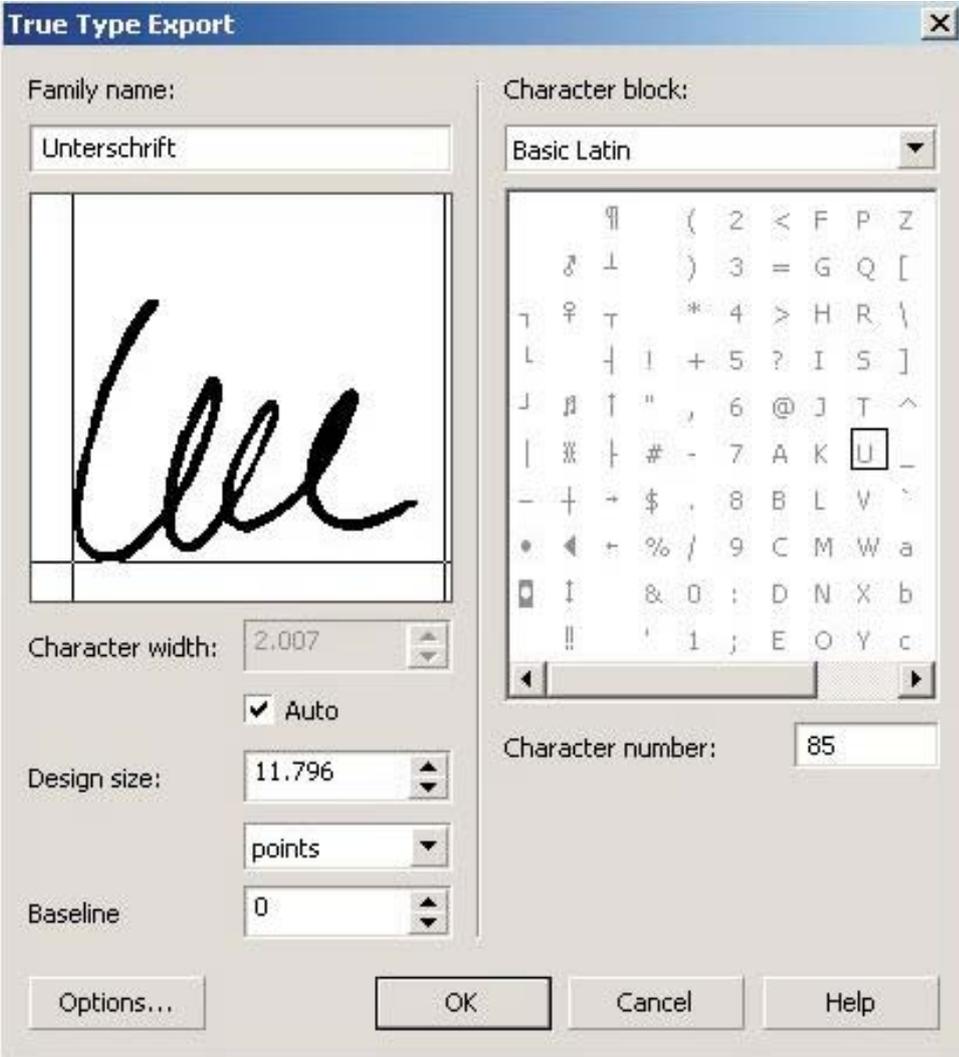


Man wird darauf hingewiesen, dass in dem gewählten Font jedes Zeichen überschrieben wird, das man in dem folgenden Dialog auswählt.

Das Ergebnis schaut scheinbar gut aus. Man hat den Eindruck, als könnte man kann die *Baseline* (die Grundline, auf der das Zeichen aufsitzt) der Zeichen-Position anpassen, die Zeichenposition wählen und auch den Namen festlegen, unter dem der Font im Schriftkatalog aufscheint.

4. Test in einem Word-Dokument

Jetzt installiert man die entstandene TTF-Datei; mit der rechten Maustaste die Datei anklicken und über das Kontextmenü den Punkt *Installieren* wählen. Jetzt ist die Schrift im Schriftkatalog verfügbar. Man schreibt in Word das Zeichen, unter dem man den Schriftzug gespeichert hat, markiert das Zeichen ordnet die Schriftart *Unterschrift* zu.



Font-Dialog von CoreDraw

5. Verbesserungen

Ausgefüllte Schlingen

Bei der verwendeten Beispielvorlage ist nicht gleich aufgefallen, dass die Schlingen im Font ausgefüllt und nicht weiß sind. Die entsprechende Checkbox im Trace-Dialog war nicht angeklickt. Das entstehende Vektorgebilde besteht daher aus einer zusammenhängenden Außenlinie und so vielen weißen Flächen als es solche Schlingen gibt. Es genügt, wenn man im Vektorbild jede einzelne Schlingenfläche anklickt und löscht.



Was man sich jetzt noch wünschen könnte, ist, dass die Unterschrift nicht in einem eigenen Font untergebracht ist sondern in einem bestehenden Font, der vorzugsweise im Schriftverkehr verwendet wird.

Im folgenden Beispiel wird der Font *"Lucida Handwriting"* verwendet.

Man ist geneigt anzunehmen, dass man für ein persönliches Zeichen irgend eine Position aus dem privaten Bereich des Unicode verwenden könnte. Leider dürfte das bei bestehenden Fonts nicht funktionieren, jedenfalls ist es mir nicht gelungen. Es war daher nötig, ein Zeichen zu suchen, das durch die Unterschrift ersetzt wird. Ich habe mich für die Zeichen 126="Tilde"='~' und 161/'i' entschieden.

Zusammenfassung

Die Herstellung eines Fonts aus beliebigen Linienzügen mit CoreDraw hat sich als recht einfach herausgestellt. Auch die Zuordnung mehrerer solcher Linienzüge oder spezieller Grafiken ist problemlos möglich.

Problematisch ist dagegen die Bearbeitung eines bestehenden Fonts, weil es CoreDraw nicht erlaubt, diesen zu importieren. Dazu benötigt man dann weiter spezialisierte Programme wie Font Monger oder Font Creator, die wir in einer der nächsten Ausgaben der PCNEWS vorstellen wollen.



Datenbankimplementierung

Sofern Ausführungen auf SQL Server 2005 Bezug nehmen, gelten sie gleichermaßen auch für SQL Server 2008.

Christian Zahler

1 Datenbank-Grundlagen

Eine Datenbank ist eine Sammlung von Daten aus der Realität.

1.1 Arten von Datenbanken

1.1.1 Sequenzieller Zugriff

Älteres Datenzugriffsverfahren (Speicherung auf Magnetbändern!). Sequenziell = „hintereinander“ (vgl. Videokassette).

- Datensätze haben Trennzeichen (etwa ANSI-13 = Zeilenumbruch); Datenfelder haben ebenfalls Trennzeichen (etwa Semikolon)
- Daten können nur sequentiell (nacheinander) gelesen werden. Daher ist dieses System extrem langsam beim Suchen und Sortieren (Man stellt am Vorabend eine Abfrage, die erst am nächsten Tag ausgewertet wird.)
- keine fixe Datensatzlänge, daher speicherplatzsparend

Beispiel

Das CSV-Dateiformat (*Comma Separated Value*) kann von Excel gelesen werden und wird oft als Schnittstelle zu Großdatenbanksystemen verwendet.

KNr;Nachname;Vorname;PLZ;Strasse

- 1;Camino;Alejandra;28001;Gran Via, 1
- 2;Feuer;Alexander;04179;Heerstr. 22
- 3;Trujillo;Ana;05021;Avda. de la Constitución 2222
- 4;Domingues;Anabela;05634-030;Av. Inês de Castro, 414
- 5;Fonseca;André;04876-786;Av. Brasil, 442
- 6;Devon;Ann;WX3 6FW;35 King George
- 7;Roulet;Annette;31000;1 rue Alsace-Lorraine
- 8;Moreno;Antonio;05023;Mataderos 2312
- 9;Cruz;Aria;05442-030;Rua Orós, 92
- 10;Braunschweiger;Art;82520;P.O. Box 555
- 11;Batista;Bernardo;02389-673;Rua da Panificadora, 12
- 12;Schmitt;Carine;44000;54, rue Royale
- 13;González;Carlos;3508;Carrera 52 con Ave. Bolívar #65-98 Llano Largo
- 14;Hernández;Carlos;5022;Carrera 22 con Ave. Carlos Soublette #8-35
- 15;Dewey;Catherine;B-1180;Rue Joseph-Bens 532

Beispiel

INI-Dateien

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINNT
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows NT Server, Version 4.0"
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Windows NT Server, Version 4.0 [VGA-Modus]" /basevideo /sos
```

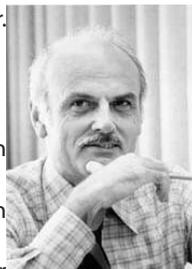
1.1.2 Index-sequentieller Zugriff

- Speicherung der Daten so wie beim sequentiellen Zugriff
- Zusätzlich wird eine „schlanke“ Index-Datei angelegt, in der zum Beispiel ein „indiziertes Feld“ (etwa der Nachname) und die Nummer des Bytes, an dem der Datensatz beginnt, gespeichert wird. Eine Suche nach Nachnamen ist somit wesentlich schneller möglich, da nur die Indexdatei durchsucht wird und nicht die gesamte Datenbank.

1.1.3 Relationales Konzept

Das relationale Datenbankmodell wurde 1970 von Dr. Edgar Frank CODD (1923 – 2003) entwickelt.

- Die Daten sind generell in Relationen gespeichert.
- Relationen sind Tabellen, wobei
 - die Reihenfolge der Spalten ("Felder") egal sein muss
 - die Reihenfolge der Zeilen (Datensätze) egal sein muss
 - es ein Feld geben muss, über dessen Wert jeder Datensatz eindeutig identifiziert werden kann ("Primärschlüssel")
- Relationen bestehen aus Feldern („Spaltenüberschriften“), deren konkrete Ausprägungen als „Attribute“ (in Excel= Zelle) bezeichnet werden.
- Der Wertebereich eines Attributs kann eingeschränkt sein.



Dr. Edgar Frank "Ted" CODD

Marktübersicht für relationale Datenbank-Management-Systeme (RDBMS)

1. Dateibasierende Datenbanksysteme ("Klein-Datenbanken")

Bei diesen Datenbanksystemen befinden sich alle Datenbank-Objekte (Tabellen, Abfragen etc.) alle in einer einzigen Datei (zum Beispiel in Access: *.MDB-Datei).

- Microsoft Access, aktuelle Version Access 2003 (für "experience", intern Version 11)
- Microsoft FoxPro
- MySQL (Linux Open Source)

2. Client-/Server-Datenbanksysteme

Hier sind die Datenbankobjekte auf mehr als eine Datei verteilt. Typischerweise gibt es keine Berichts- und Formularobjekte. Der Server stellt benötigte Daten meist als "Datensatzgruppen" (*Recordsets*) den Clients zur Verfügung, die Darstellung wird meist am Client von Frontend-Software übernommen.

- Microsoft SQL Server
- Oracle
- PostgreSQL (Linux Open Source)
- Sybase Adaptive Server
- Informix-Systeme

1.1.4 Hierarchische Datenbanken, XML-Datenbanken

Hier hat sich in den letzten Jahren XML einen fixen Platz in der Datenspeicherung erobert.

```
<?xml version="1.0" encoding="UTF-8"?>
<tKunden>
<datensatz>
  <KdNr>23</KdNr>
  <Vorname>Helmut</Vorname>
  <Nachname>Gruber</Nachname>
</datensatz>
<datensatz>
  <KdNr>47</KdNr>
  <Vorname>Maria</Vorname>
  <Nachname>Gschwandtner</Nachname>
</datensatz>
<datensatz>
  <KdNr>19</KdNr>
  <Vorname>Peter</Vorname>
  <Nachname>Maier</Nachname>
</datensatz>
</tKunden>
```

DTD

```
<!DOCTYPE tKunden [
  <!ELEMENT tKunden (datensatz+) >
  <!ELEMENT datensatz (KdNr,Vorname*,Nachname) >
  <!ELEMENT KdNr (#PCDATA) >
  <!ELEMENT Vorname (#PCDATA) >
  <!ELEMENT Nachname (#PCDATA) >
]>
```

XML-Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
  <xsd:element name="tKunden">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="datensatz" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="generated" type="xsd:dateTime"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="datensatz">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="KdNr" minOccurs="0" od:jetType="longinteger" od:sqlSType="int" type="xsd:int">
          <xsd:annotation>
            <xsd:appinfo>
              <od:fieldProperty name="ColumnWidth" type="3" value="-1"/>
              <od:fieldProperty name="ColumnOrder" type="3" value="0"/>
              <od:fieldProperty name="ColumnHidden" type="1" value="0"/>
              <od:fieldProperty name="DecimalPlaces" type="2" value="255"/>
              <od:fieldProperty name="Required" type="1" value="0"/>
            </od:appinfo>
          </xsd:annotation>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
```

```
<od:fieldProperty name="DisplayControl" type="3" value="109"/>
<od:fieldProperty name="TextAlign" type="2" value="0"/>
<od:fieldProperty name="AggregateType" type="4" value="-1"/>
</xsd:appinfo>
</xsd:annotation>
</xsd:element>
<xsd:element name="Vorname" minOccurs="0" od:jetType="text"
od:sqlType="nvarchar">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:maxLength value="255"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="Nachname" minOccurs="0" od:jetType="text"
od:sqlType="nvarchar">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:maxLength value="255"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

1.2 Datenbankplanung

1.2.1 Planung von Datenbanken; Entity-Relationship-Modell

● Welche Informationen gehören in die Datenbank und sollen gespeichert werden?

● Datenbankstruktur

Grafische Unterstützung beim DB-Design bietet das **Entity-Relationship-Modell** (Dr. Peter Chen, 1976).

In diesem Modell sind folgende Begriffe wesentlich:

● **Entity:** Ein real existierendes Objekt, das in einer DB abgebildet werden soll. Wird durch ein Rechteck gekennzeichnet.

● **Relationship:** gibt an, wie zwei Entitäten miteinander verknüpft sind. Relationships werden durch eine Raute symbolisiert.

● **Attribut:** "Feld", wird durch ein Oval dargestellt

● **Primärschlüsselattribut:** Der Attributname wird zusätzlich unterstrichen.



Dr. Peter Chen

Beispiel



Kardinalität von Beziehungen: Sie gibt an, wie viele Elemente der einen Entität mit wie vielen Elementen der anderen Entität in Beziehung stehen.

a) 1:1-Beziehung

Jedem Element der linken Entität kann nur genau ein Element der rechten Entität zugeordnet werden und umgekehrt.

Beispiel



b) 1:n-Beziehung

Jedem Element der linken Entität können beliebig viele Elemente der rechten Entität zugeordnet werden. Jedem Element der rechten Entität kann nur genau ein Element der linken Entität zugeordnet werden.

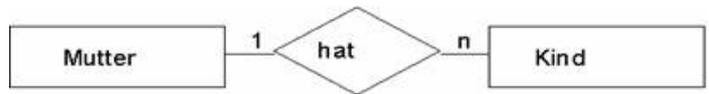
Beispiel



c) m:n-Beziehung

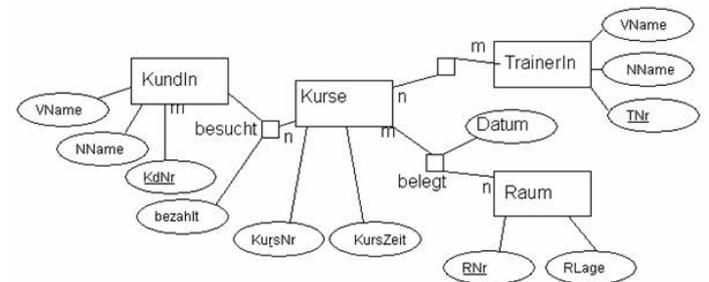
Beliebig vielen Elementen der linken Entität können beliebig viele Elemente der rechten Entität zugeordnet werden. Dieser Verknüpfungstyp kommt in der Realität am häufigsten vor.

Beispiel



Hinweis: m:n Beziehungen können nicht direkt in ein relationales Modell übertragen werden.

Beispiel: ER-Diagramm für ein Schulungsinstitut



1.2.2 Umsetzung des ER-Diagramms in das relationale Modell

Hier sind nur einige Grundregeln zu beachten:

1. Aus jeder Entität wird eine Relation. Relationen dieser Art werden oft als "Stammdaten-Tabelle" bezeichnet.
2. Bei 1:1-Beziehungen ist zu überprüfen, ob die beiden Entitäten nicht in einer Tabelle zusammengefasst werden können.
3. 1:N-Beziehungen können direkt in ein relationales Modell umgesetzt werden; in die N-Tabelle muss ein Fremdschlüsselfeld eingefügt werden.
4. M:N-Beziehungen sind nicht direkt in ein relationales Modell umsetzbar; hier ist eine Zwischentabelle notwendig.

Die Umsetzung wird zu einem späteren Zeitpunkt an Hand des praktischen Beispiels näher erläutert.

1.3 Normalisierung von Datenbanken

Ziele bei der Realisierung von Datenbanken ist die Vermeidung von:

- **Redundanz:** Die Daten in einer Datenbank sind dann redundant, wenn Teile der Daten mehrfach vorkommen!
- **Inkonsistenz:** mehrere Schreibweisen für ein und dasselbe Objekt: zum Beispiel St. Pölten, Sankt Pölten, St. Poelten, St Pölten, ...

Zur Vermeidung von Redundanzen und Inkonsistenzen gibt es die so genannten Normalformen. Wenn die Tabellen einer DB den Normalformen genügen, ist ein wichtiger Beitrag zur Redundanzvermeidung geleistet (noch keine Garantie, dass überhaupt keine Redundanz!)

- **1. Normalform:** Keine Listen als Wertebereiche
- **2. Normalform:** Attribute dürfen nicht von einem Teil eines Schlüssels abhängen
- **3. Normalform:** Attribute dürfen nicht voneinander ableitbar sein.

1.3.1 1. Normalform

- Jedes Feld besitzt einen eindeutigen Namen, kein Feld kommt mehrfach vor.
- Jedes Datenelement ist atomar und nicht weiter zerlegbar.
- Jede Tabelle besitzt einen Primärschlüssel.
- Beziehungen zwischen Entitäten werden ausschließlich über Schlüsselfelder hergestellt (keine absoluten Adressen).

Probleme bei Datenbanken in 1. Normalform

- Identische Attributwerte werden mehrfach gespeichert (Redundanz)
- Einfügeanomalien (Es kann kein Student angelegt werden, der sich noch nicht für ein Seminar entschieden hat)
- Löschanomalien (Student muss gelöscht werden, falls er alle gebuchten Seminare absagt)
- Änderungsanomalien (Eine nachträgliche Änderung der Attribute (Namensänderung bei Heirat) führt zu Änderung an mehreren Datensätzen)

Als Attributwerte sind nur atomare Werte (**integer, string**) erlaubt, keine Listen oder Mengen.

tblBuch

Buchnr	Buchtitel	Autor
184	Sozialstaat Österreich	Ernst, Federspiel, Langbein

Lösung

tblBuch

Buchnr	Buchtitel	Autor
184	Sozialstaat Österreich	Ernst
184	Sozialstaat Österreich	Federspiel
184	Sozialstaat Österreich	Langbein

tblKunden

Name	Adresse
Harrer, Heinrich	Bahnhofplatz 3, 3100, St. Pölten

Lösung: Zerlegung in mehrere Felder

Nachname	Vorname	Straße	PLZ	Ort
Harrer	Heinrich	Bahnhofplatz 3	3100	St. Pölten

1.3.2 2. Normalform

- Die Entität liegt in 1. Normalform vor.
- Jedes Feld, das nicht Bestandteil des Primärschlüssels ist, ist voll funktional abhängig vom Primärschlüssel (alle Teile des Primärschlüssels werden benötigt, um die restlichen Felder zu bestimmen), d.h. Beseitigung der nicht voll funktionalen Abhängigkeiten.

Andere Formulierung

Eine Tabelle befindet sich in der 2. Normalform, wenn

- sie sich in der 1. Normalform befindet und wenn
- alle Nichtschlüsselattribute von allen Attributen des Primärschlüssels abhängen.

Beispiel

tblEntLehnung

Kundennr	Nachname	Vorname	Buchnr	EntLehndatum
23	Müller	Aloisia		770182	02.05.2001
23	Müller	Aloisia		912341	02.05.2001
109	Giger	Brunhilde		891021	30.04.2001
176	Huber	Herbert		NULL	NULL

entspricht nicht der 2. Normalform:

Buchnr hängt nicht vom Primärschlüssel **Kundennr** ab

EntLehndatum hängt nicht vom Primärschlüssel **Kundennr** ab

Anomalien

1. Löschanomalie

Bei Rückgabe aller Bücher werden auch die Informationen über den/die Entleiher/in gelöscht.

2. Einfügeanomalie

Will man Informationen über einen Kunden einfügen, der noch kein Buch ausgeliehen hat, dann müssen alle Felder, die sich auf das Ausleihen von Büchern beziehen, mit NULL-Einträgen bzw. (noch schlimmer) mit Dummy-Einträgen gefüllt werden (z.B. **Buchnr** = 999999 bedeutet "noch kein Buch ausborgt"). Setzt man Primärschlüssel auf **Buchnr**, so können Kunden, die noch kein Buch entlehnt haben, gar nicht angelegt werden. Ist allerdings **Kundennr** Primärschlüssel, so kann jeder Kunde nur ein Buch ausborgen.

3. Änderungsanomalie

Bei Änderung von Personendaten (neuer Name, neue Adresse, neue Telefonnummer) müssen diese Änderungen in mehreren Datensätzen durchgeführt werden. Wird ein betroffener Datensatz nicht geändert, so enthält die Datenbank widersprüchliche Informationen.

Lösung

Tabelle muss in mehrere Tabellen aufgespalten werden.

1.3.3 3. Normalform

- Jede Entität liegt in 2. Normalform vor.
- Jedes Feld, welches nicht Bestandteil des Kandidatenschlüssels ist, hängt nicht transitiv von einem Kandidatenschlüssel ab.

2 Transact SQL

Die Programmiersprache SQL (*Structured Query Language, Standard Query Language*) wurde von IBM entwickelt ? "Projekt R". Um 1975 kam man auf die Idee, für Datenbankabfragen eine eigene Programmiersprache zu entwickeln.

ca. 1977 – 79: Entwicklung von SQL (Programmiersprache der vierten Generation)

- Generation 1: reine Maschinensprachen (binäre Programmierung)
- Generation 2: Assembler
- Generation 3: höhere Programmiersprachen
- Generation 4: Datenbankabfragesprachen

SQL-Normen (ANSI = *American National Standards Institut*, ISO = *International Standardization Organization*)

- SQL-89 (ältere Version)
- SQL-92: ab Access 97 bzw. SQL Server 7.0
- SQL-99 (aktuelle Version): Kaum praktische Implementierungen vorhanden.

Viele Hersteller verwenden zusätzlich zu den ANSI-kompatiblen Basis-SQL-Befehlen produktspezifische Erweiterungen:

- TSQL (*Transact SQL*): SQL-Erweiterung für Microsoft SQL Server
- SQL*Plus: Erweiterung für Oracle-Datenbanken

2.1 Erstellen einer SQL Server-Datenbank

Wiederherstellungsmodell (Recovery Model)

SQL 2000/2005 Bedeutung

Full	Bedeutung
Full	Log enthält alle Transaktionen seit dem letzten Backup; Log-File wird kontinuierlich wachsen
Simple	nur aktive Transaktionen sind im Log; Logfile sehr klein; kein Point-in-Time-Recovery, keine vollständige Datenwiederherstellung möglich
Bulk Logged	erlaubt unprotokollierten Massenimport; andere Transaktionen werden jedoch protokolliert; kein Point-in-Time-Recovery

Verkleinern der Datenfiles

DBCC SHRINKDATABASE
DBCC SHRINKFILE

Optionen

- NOTRUNCATE – Datenfile wird bis zur "Hochwasserlinie" verkleinert
- EMPTYFILE – Alle Daten dieses Files werden in andere Datenfiles verschoben
- TRUNCATEONLY – Datenfile wird verkleinert, ohne die Daten intern zu verschieben

Dateigruppen

Werden verwendet, um die Flexibilität und Performance zu erhöhen. Tabellen werden am besten zunächst Dateigruppen zugeordnet, erst die Dateigruppe wird mehreren Datendateien zugeordnet.

Dateigruppe in den Datenbankeigenschaften anlegen; im Karteireiter "Data Files" können die einzelnen Datendateien einer Dateigruppe zugeordnet werden.



TSQL-Code

```

/* Anlage einer neuen Datenbank
Skript Version 1.0
11.05.2007 */
create database Verkauf
on primary -- Dateigruppe primary
(name = 'verkauf1', filename = 'E:\verkauf1.mdf',
size=10 MB,maxsize=unlimited,filegrowth=10 %),
filegroup daten2006 -- weitere Dateigruppe, optional!
(name = 'verkauf2', filename = 'E:\verkauf2.ndf',
size=5 MB,maxsize=100 MB,filegrowth=10 MB)
log on -- Transaktionsprotokoll
(name = 'verkauf_log', filename = 'F:\verkauf_log.ldf',
size=2 MB,maxsize=unlimited,filegrowth=1 MB);
    
```

Datenbankeigenschaften ändern

```
ALTER DATABASE SampleDBTsql
MODIFY FILE
(NAME = 'SampleDBTsql_Log',
MAXSIZE=20MB)
GO
```

Datenbanken löschen

```
USE master
DROP DATABASE SampleDBTsql, SampleDBWizard
GO
EXEC sp_he1pdb
GO
```

2.2 Dateimäßiger Aufbau einer SQL Server 2005-Datenbank:

- **Hauptdatendatei** (Endung *.MDF = *main data file*): enthält die konkreten Datenbankobjekte, zum Beispiel Tabellen, Sichten, gespeicherte Prozeduren etc.; enthält Systemtabellen
- **weitere Datendateien** (*.NDF = *non-main data file*)
- **Transaktionsprotokoll** (*.LDF = *Transaction Log*): Alle Änderungen der Daten seit dem letzten Backup werden im Transaktionsprotokoll gespeichert. Dadurch werden Wiederherstellungen bis zum aktuellen Datenbestand möglich.

Die Datendateien und Transaktionsprotokolle sollten auf unterschiedlichen Laufwerken gespeichert werden.

Das Transaktionsprotokoll wird in einem internen Format gespeichert.

Ein "Checkpoint"-Prozess löst (etwa ein Mal jede Sekunde) die konkrete Aktualisierung der Datenbank auf der physischen Festplatte aus.

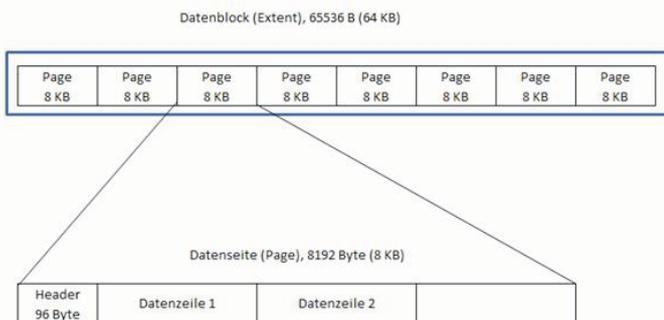
2.3 Interner Aufbau einer Datendatei:

Eine SQL Server-Datendatei (*.mdf, *.ndf) besteht grundsätzlich aus 8 KB großen Seiten (engl. *Pages*), in denen Daten gespeichert sind. Jeweils 8 aufeinanderfolgende Seiten bilden einen Block (engl. *Extent*).

Man unterscheidet:

- **Einheitlicher Datenblock:** alle 8 Seiten gehören zum selben Datenbankobjekt (etwa zur selben Tabelle)
- **Gemischter Datenblock:** die 8 Seiten gehören zu unterschiedlichen Datenbankobjekten.

Jede Seite beginnt mit einem 96 Byte großen Header und enthält dann einen oder mehrere Datenzeilen. Am Ende jeder Seite befindet sich die Zeilenoffsettabelle (32 Byte), die den „Abstand“ jeder Datenzeile vom Beginn der Seite enthält, und zwar in umgekehrter Reihenfolge der Datenzeilen. Für die eigentliche Datenspeicherung stehen pro Seite max. 8060 Byte zur Verfügung.



Man unterscheidet nach dem Inhalt der Pages:

Datenseiten: Diese enthalten unterschiedliche Arten von Daten:

- **Datenseiten:** enthalten Daten (ausgenommen solche vom Datentyp `text`, `ntext`, `image`, `varchar(max)`, `nvarchar(max)`, `varbinary(max)`)
- **LOB-Seiten** (LOB = *Large Objects*): enthalten Daten vom Datentyp `text`, `ntext`, `image`, `varchar(max)`, `nvarchar(max)`, `varbinary(max)`
- **Indexseiten**

Verwaltungsseiten: Diese enthalten Informationen über die Struktur und den Aufbau der Datendatei.

- **GAM/SGAM** (*Global Allocation Map, Secondary GAM*)

GAM/SGAM-Seiten enthalten Informationen über den Zustand von 64000 Extents mit folgender Belegung:

	GAM-bit	SGAM-bit
Freier Datenblock (nicht in Verwendung)	1	0
Einheitlicher Datenblock oder vollgemischter Block	0	0
Gemischter Datenblock mit freien Seiten	0	1

- **PFS** (*Page Free Space*)

PFS-Seiten enthalten Informationen darüber, wie viel Platz auf den nächsten 8000 Blöcken noch frei ist.

- **IAM** (*Index Allocation Map*)

IAM-Seiten verwalten Indizes.

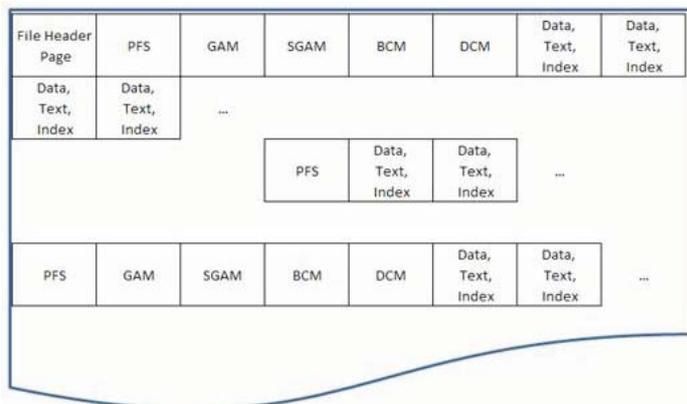
- **BCM** (*Bulk Changed Map*)

BCM-Seiten enthalten Informationen darüber, welche der nächsten 64000 Blöcke mit bcp seit dem letzten BACKUP LOG-Vorgang verändert wurden (wenn ein Bit auf 1 gesetzt ist, dann wurde der entsprechende Block geändert).

- **DCM** (*Differential Changed Map*)

DCM-Seiten enthalten Informationen darüber, welche der nächsten 64000 Blöcke seit dem letzten BACKUP LOG-Vorgang verändert wurden (wenn ein Bit auf 1 gesetzt ist, dann wurde der entsprechende Block geändert).

Jede Datendatei ist folgendermaßen aufgebaut:



Aufbau eines Datensatzes:

- **Datensatz-Header**
 - 4 Byte
 - 2 Byte Datensatz-Metadaten (record type)
 - 2 Byte, die auf das NULL-Bitmap zeigen
- **Spalten mit fixer Länge** (Datentypen zum Beispiel `bigint`, `char(10)`, `datetime`)
- **NULL-Bitmap**
 - 2 Byte, die die Anzahl der Spalten im Datensatz speichern
 - variable Anzahl von Byte; je ein Bit pro Spalte wird verwendet, um anzugeben, ob die Spalte NULL enthält oder nicht (SQL Server 2000 verwendete nur Bits für Spalten, die NULL-Werte enthalten durften; in SQL Server 2005 wird ein bit für jede Spalte verwendet, egal, ob sie NULL-Werte annehmen darf oder nicht)
 - this allows an optimization when reading columns that are NULL
- **Offset-Struktur für Spalten mit variabler Länge**
 - 2 Byte, die die Anzahl der Spalten mit variabler Länge angeben
 - 2 Byte pro Spalte mit variabler Länge, die die Anzahl der Byte vom Beginn des Datensatzes bis zum Beginn der Spalte angibt (Offset)
- **Versionsangabe** (nur SQL Server 2005)
 - 14 Byte-Struktur, die einen Zeitstempel sowie einen Zeiger auf den Versionspeicher in `tempdb` enthält.

2.4 Tabellen anlegen

Da die Daten in Tabellen gespeichert werden, werden als nächster Schritt neue Tabellen erstellt.

2.4.1 Regeln für Feldnamen, Tabellennamen und anderen Datenbank-Objekten

Feldnamen und andere Objektnamen dürfen **maximal 128 Zeichen** enthalten.

Verboten sind: Rufzeichen, eckige Klammern, Punkte und Akzentzeichen

Dringend abzuraten ist von der Verwendung von Leerzeichen, Umlauten und Sonderzeichen.

Erfüllen Feldnamen diese Regel, so werden sie „reguläre Bezeichner“ genannt.

Dringend abzuraten ist von der Verwendung von Bezeichnungen, die bereits Access-intern verwendet werden, zum Beispiel „Name“.

Wenn Sie einen Feldnamen wählen, der mit „-nummer“ endet, so schlägt Access automatisch eine Indizierung „Ja (Duplikate möglich)“ vor.

Tipps für das Speichern von Tabellen: Beginnen Sie den Namen der Tabelle mit einem kleingeschriebenen `t` oder `tb1` (also beispielsweise `tKunden`, `tb1Kunden`); bei Abfragen verwenden Sie `q` (für "query"). Damit können Sie beim Erstellen von Formularen und Berichten Tabellen sofort von Abfragen unterscheiden.

2.4.2 Felddatentypen

a) Ganzzahl: int, smallint, bigint, ...

Mit ganzzahligen Werten kann exakt (ohne Ungenauigkeiten) gerechnet werden.

b) Dezimalzahl

`float()` *approximate numeric*

- Damit kann nicht exakt gerechnet werden. Es treten bei jedem Rechenvorgang Ungenauigkeiten und Rundungsfehler auf (z.B. $2.0 + 3.0 = 4.999999542$)

`decimal()` *exact numeric*

- Exaktes Rechnen möglich, da skalierte Ganzzahl gespeichert wird.

c) Alphanumerisch

`char()` fixe Länge (max. 8000 Zeichen)

`varchar()` variable Länge (max. 8000 Zeichen)

`text()` lange Textfelder (max. 2 Mio. Zeichen)

`nchar()`, `nvarchar()`, `ntext()` ... *national character support*; hier wird UNICODE (2 Byte/Zeichen) verwendet, dies reduziert natürlich die Anzahl der verwendbaren Zeichen auf die Hälfte.

d) Datum/Zeit

`datetime` ab der gregorianischen Kalenderreform bis 9999

`smalldatetime` ab 1.1.1900 bis etwa 2090

`money` *Currency* - skalierte Ganzzahl; intern wird die Zahl mit 10000 multipliziert, für die Darstellung wieder dividiert und auf zwei Stellen gerundet.

2.4.3 Tabellen anlegen im Management Studio

Spaltenname	Datentyp	NULL zulassen
KdNr	int	<input type="checkbox"/>
Vorname	nvarchar(50)	<input checked="" type="checkbox"/>
Nachname	nvarchar(50)	<input type="checkbox"/>

Column Name	Data Type	Allow Nulls
DepartmentID	smallint	<input type="checkbox"/>
Name	nvarchar	<input type="checkbox"/>
GroupName	nvarchar	<input type="checkbox"/>
ModifiedDate	datetime	<input type="checkbox"/>

Column Properties

(General)

(Name) Name

Allow Nulls No

Data Type nvarchar

Default Value or Binding

Length 50

Table Designer

Collation <database default>

Computed Column Specification

Condensed Data Type nvarchar(50)

Description

Full-text Specification No

Has Non-SQL Server Subscriber No

Identity Specification No

Is Deterministic Yes

Is DTS-published No

Is Indexable Yes

(General)

2.4.4 Tabelle anlegen mit TSQL-Kommandos

```
use Auftrag;
create table dbo.tArtikel
( ArtNr int identity(1,1) primary key,
  ArtBez nvarchar(50) NOT NULL,
  Einzelpreis money NOT NULL
);
```

```
use Auftrag;
create table dbo.tAuftrag
(
  AuftrNr int identity(1,1) primary key,
  KdNr int not null,
  Datum datetime not null
)
```

2.5 Primärschlüssel und Indizes

Als Primärschlüssel wird ein Feld oder eine Kombination von Feldern verwendet, über die jeder Datensatz eindeutig identifizierbar ist. Die Werte des Primärschlüssels müssen also eindeutig sein, sodass aus der Kenntnis des Primärschlüsselwertes auf genau einen Datensatz rückgeschlossen werden kann.

Beispiele für Primärschlüsselfelder

- Kundennummer
- Artikelnummer
- Buchungsnummer

Es gibt auch mehrteilige Primärschlüssel. Dieser wird beispielsweise aus der Kombination von Geburtsdatum und Sozialversicherungsnummer gebildet wird. Dazu markiert man beide Zeilen mit gedrückter `[STRG]`-Taste. In diesem Fall Achtung: Die beiden Schlüsselssymbole sind irreführend; auch in dieser Tabelle gibt es nur **einen** Primärschlüssel!

Anmerkung: Jedes indizierte Feld kann als „Sekundärschlüssel“ bezeichnet werden. Der Begriff wird aber im Zusammenhang mit Datenbanken praktisch nicht verwendet, da – außer den bereits erwähnten Zeitvorteilen beim Suchen und Sortieren – ein Sekundärschlüssel keine weiteren Vorteile bringt.

2.6 Auswahlabfragen

```
use AdventureWorks
select loginid,gender from HumanResources.Employee
```

2.6.1 Formulieren von Kriterien

Im der `WHERE`-Klausel werden nun Kriterien formuliert, nach denen die Daten gefiltert werden.

Textfelder

Suchmuster für Textfelder werden in Access immer mit einem doppelten Anführungszeichen gekennzeichnet.

`= 'Müller'` exakte Übereinstimmung wird gefordert (es werden alle Datensätze im Abfrageergebnis ausgegeben, deren Eintrag im Nachnamen exakt dem Wort „Müller“ entspricht); das `=`-Zeichen kann weggelassen werden

`LIKE 'S%'` Wie `=` Ähnlichkeitsoperator; es sind auch Jokerzeichen im Suchmuster zugelassen. `*...0` bis beliebig viele Zeichen (laut Norm: %)

`Wie 'M_er'` ? ... exakt ein unbekanntes Zeichen (laut Norm: _)

`<'S%'` A bis R

`>'S%'` S bis Z; eigentlich `>=` (`=` gibt es bei Texten nicht)

`Between 'B%' And 'S%'` B bis R

```
use AdventureWorks
select ProductID, Name
from Production.Product
where Name Between 'L%' And 'T%';
/* Ergebnis: L - S */
```

Datumsfelder

Einträge in Datumsfeldern werden in SQL Server so wie Texte gekennzeichnet.

`Between '01/01/1970' And '31/03/1970'`

`<Date()`

Zahlenfelder

Werte werden immer ohne spezielle Kennzeichnung (Anführungszeichen bzw. Nummernzeichen) eingetragen.

Between 100 And 500 <100 And >500
 >34
 <150
 >=56,3

Beim Between-Operator werden bei Zahlenfeldern beide Grenzen mit einbezogen.

Leere bzw. nicht-leere Felder

Nicht benutzte Felder werden intern durch die symbolische Konstante <NULL> gekennzeichnet. <NULL>-Einträge können mit keinem anderen Wert verglichen werden, nicht einmal mit anderen <NULL>-Werten. Daher darf der LIKE-Operator in diesem Fall nicht verwendet werden, es gibt eine eigene Syntax:

Is NULL
 Is NOT NULL

Kombination mehrerer Kriterien

Diese erfolgt mit BOOLEschen Operatoren (AND, OR, NOT)

a) Verknüpfung mit AND:

Dabei stehen Kriterien in derselben Zeile nebeneinander. Ein Datensatz erscheint nur dann im Abfrageergebnis, wenn beide Kriterien wahr sind. Übliche Darstellung: „Wahrheitswerte-Tabelle“

A	B	A And B
wahr	wahr	wahr
wahr	falsch	falsch
falsch	wahr	falsch
falsch	falsch	falsch

b) Verknüpfung mit OR (nicht ausschließendes ODER):

Der Datensatz erscheint dann im Abfrageergebnis, wenn mindestens ein Kriterium oder auch beide wahr sind. Achtung: Es entspricht nicht dem üblichen Sprachgebrauch. Dabei steht das erste Kriterium steht in der Kriterienzeile und das zweite Kriterium steht in der Oder-Zeile.

A	B	A Or B
wahr	wahr	wahr
wahr	falsch	wahr
falsch	wahr	wahr
falsch	falsch	falsch

2.6.2 Berechnete Felder in Abfragen

```
select anzahl, einzelpreis, anzahl * einzelpreis as gesamtpreis
From dbo.Artikel
```

2.6.3 Aggregatfunktionen und Gruppierung:

Beispiel

```
use adventureWorks
select ProductLine, sum(StandardCost) as Gruppensumme
from Production.Product
group by ProductLine;
```

Ergebnis

ProductLine	Gruppensumme
NULL	2115,0796
M	41125,5062
R	59497,1694
S	858,0118
T	26740,1255

(5 Zeile(n) betroffen)

Beispiel

```
use adventureWorks
select ProductLine, sum(StandardCost) as Gruppensumme
from Production.Product
where ProductLine = 'M'
group by ProductLine;
```

Ergebnis

ProductLine	Gruppensumme
M	41125,5062

(1 Zeile(n) betroffen)

Beispiel

Soll nach aggregierten Werten gefiltert werden, darf WHERE nicht verwendet werden. Statt dessen muss eine HAVING-Klausel nach der GROUP BY-Klausel angefügt werden.

```
use adventureWorks
select ProductLine, sum(StandardCost) as Gruppensumme
from Production.Product
group by ProductLine
having sum(StandardCost) > 20000;
```

Ergebnis

ProductLine	Gruppensumme
M	41125,5062
R	59497,1694
T	26740,1255

(3 Zeile(n) betroffen)

Beispiel

```
use adventureWorks
select ProductLine, sum(StandardCost) as Gruppensumme
from Production.Product
group by ProductLine
with rollup;
```

Ergebnis

ProductLine	Gruppensumme
NULL	2115,0796
M	41125,5062
R	59497,1694
S	858,0118
T	26740,1255
NULL	130335,8925

(6 Zeile(n) betroffen)

Hier kann man nicht eindeutig erkennen, welche Zeile die Gesamtsumme über alle Gruppensummen darstellt.

```
use adventureWorks
select ProductLine, sum(StandardCost) as Gruppensumme,
grouping(ProductLine) as IstSumme
from Production.Product
group by ProductLine
with rollup;
```

ProductLine	Gruppensumme	IstSumme
NULL	2115,0796	0
M	41125,5062	0
R	59497,1694	0
S	858,0118	0
T	26740,1255	0
NULL	130335,8925	1

(6 Zeile(n) betroffen)

Im folgenden Beispiel werden die Gruppensummen für ProductLine und ProductSubCategoryID berechnet. Der CUBE-Operator berechnet zusätzlich die Summen pro ProductLine und pro ProductSubCategoryID.

```
select
productline,
ProductSubCategoryID,
sum(StandardCost) as Gruppensumme,
grouping(productline) as IstAggrProductLine,
grouping(ProductSubCategoryID) as IstAggrSubCat
from production.product
group by productline ,ProductSubCategoryID
with cube;
```

productline	ProductSubCategoryID	Gruppensumme	IstAggrProductLine	IstAggrSubCat
NULL	NULL	1060,89	0	0
NULL	5	122,8638	0	0
NULL	6	94,572	0	0
NULL	7	8,9866	0	0
NULL	8	371,6148	0	0
NULL	9	94,5498	0	0
NULL	10	245,6209	0	0
NULL	11	115,9817	0	0
NULL	NULL	2115,0796	0	1
M	1	29880,6408	0	0
M	4	100,6682	0	0
M	12	10218,2052	0	0
M	13	81,5052	0	0
M	15	52,7917	0	0
M	17	542,1108	0	0
M	20	47,0127	0	0
M	22	78,5289	0	0
M	23	6,7926	0	0
M	27	59,466	0	0
M	28	3,7363	0	0
M	30	8,2205	0	0
M	36	10,3084	0	0
M	37	35,5189	0	0
M	NULL	41125,5062	0	1
R	2	42721,6278	0	0
R	4	100,6682	0	0
R	13	81,5052	0	0
R	14	15852,432	0	0
R	15	35,4135	0	0
R	17	625,6048	0	0

R	23	6,7246	0	0
R	28	3,3623	0	0
R	33	38,7627	0	0
R	37	31,0683	0	0
R	NULL	59497,1694	0	1
S	18	111,3627	0	0
S	19	6,9223	0	0
S	20	27,4779	0	0
S	21	320,2584	0	0
S	22	98,9836	0	0
S	24	92,8002	0	0
S	25	71,247	0	0
S	26	44,88	0	0
S	28	1,8663	0	0
S	29	2,9733	0	0
S	31	39,2589	0	0
S	32	20,5663	0	0
S	34	10,3125	0	0
S	36	8,2459	0	0
S	37	0,8565	0	0
S	NULL	858,0118	0	1
T	3	19490,5544	0	0
T	4	61,1211	0	0
T	13	35,9596	0	0
T	15	70,1699	0	0
T	16	6812,4686	0	0
T	17	205,5808	0	0
T	35	51,5625	0	0
T	37	12,7086	0	0
T	NULL	26740,1255	0	1
NULL	NULL	130335,8925	1	1
NULL	NULL	1060,89	1	0
NULL	1	29880,6408	1	0
NULL	2	42721,6278	1	0
NULL	3	19490,5544	1	0
NULL	4	262,4575	1	0
NULL	5	122,8638	1	0
NULL	6	94,572	1	0
NULL	7	8,9866	1	0
NULL	8	371,6148	1	0
NULL	9	94,5498	1	0

....
(99 Zeile(n) betroffen)

Die Klausel **COMPUTE** liefert Ergebniszeilen in einem nicht-relationalen Format. Nicht empfehlenswert!

```
select name, productline, StandardCost, ProductSubCategoryID
from production.product
order by productline, ProductSubCategoryID
compute sum(StandardCost) by productline
compute sum(StandardCost);
```

Ergebnis

....				
Touring Rear Wheel	T	108,7844		17
Touring Front Wheel	T	96,7964		17
Touring-Panniers, Large	T	51,5625		35
Touring Tire Tube	T	1,8663		37
Touring Tire	T	10,8423		37
sum				

26740,1255				
sum				

130335,8925				
(510 Zeile(n) betroffen)				

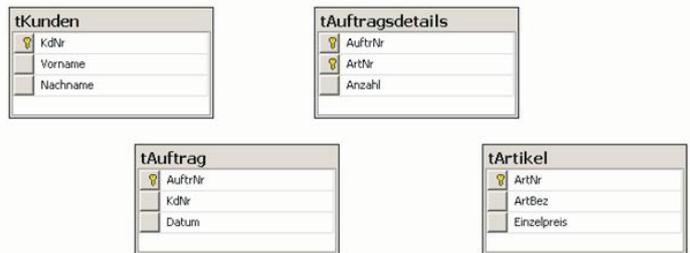
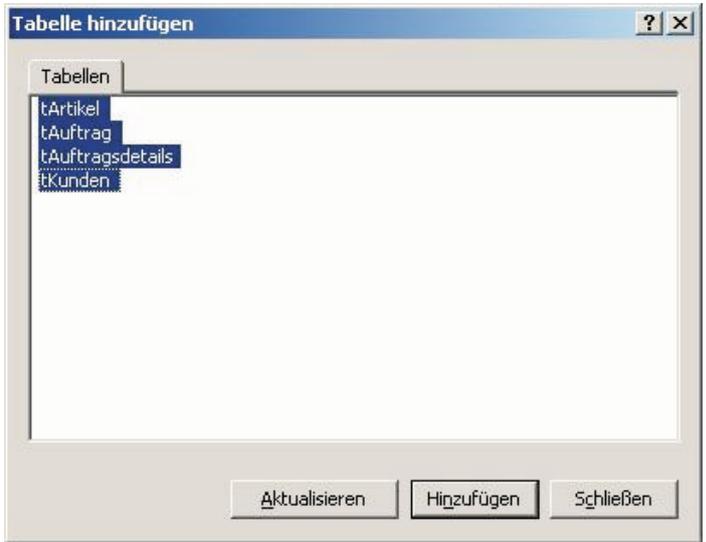
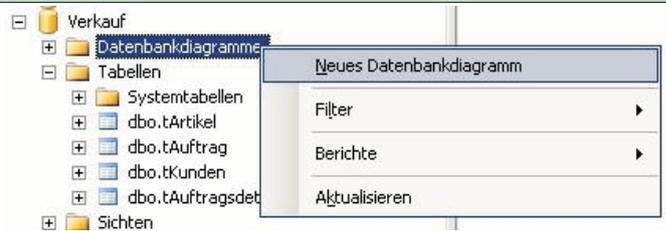
2.7 Beziehungen in Diagrammen erstellen

Die Erstellung von Fremdschlüsseleinschränkungen (Beziehungen) aktiviert einen Mechanismus im SQL Server, der die Integrität der eingegebenen Daten prüft (referentielle Integrität).

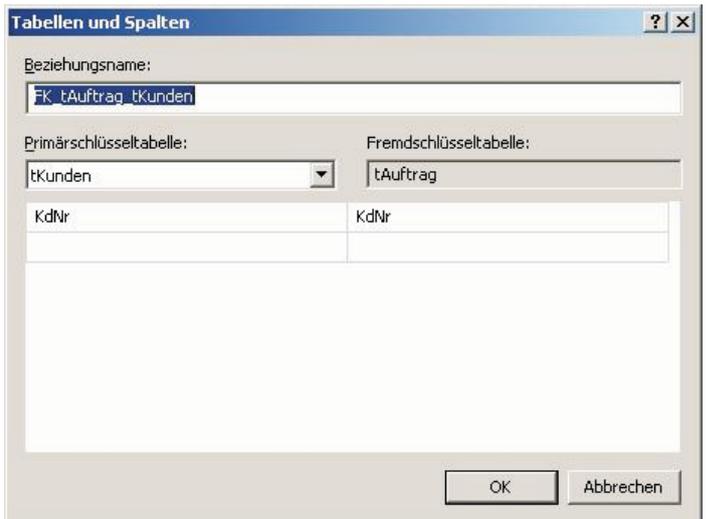
Referenzielle Integrität: Beispielsweise dürfen in einer Verkaufstabelle nur Kunden enthalten sind, die auch in der Kundentabelle angelegt sind

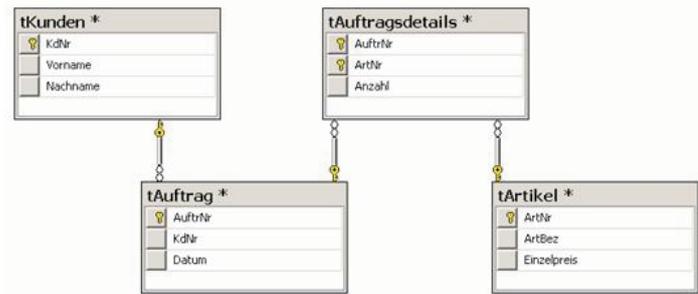
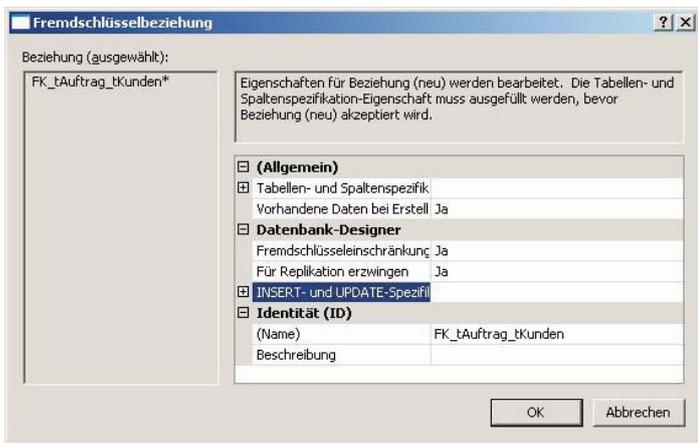
- Prüfung, ob Fremdschlüsselfelder vorhandenen Primärschlüsselfeldern entsprechen
- aus Mastertabelle können Datensätze (Tupel) erst dann gelöscht werden, wenn die verknüpften Datensätze in der Detailtabelle gelöscht werden

Im SQL Server 2005 Management Studio müssen dafür Datenbankdiagramme erstellt werden.

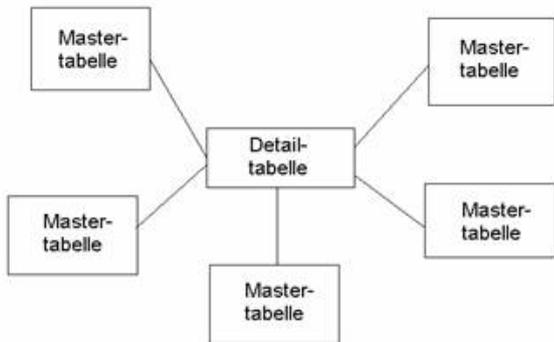


Die Beziehung wird erstellt, in dem die verknüpften Felder mit Drag and Drop verbunden werden.





In einem Diagramm ist es meist günstig, wenn man die Mastertabellen (Stammdatentabellen mit Primärschlüsseln) sternförmig um die Detailtabellen (Fremdschlüsselstabellen) anordnet:



2.8 Auswahlabfragen basierend auf mehreren Tabellen

```
select
tAuftragsdetails.AuftrNr,
tAuftrag.Datum,
tAuftrag.KdNr,
tKunden.Vorname,
tKunden.Nachname,
tAuftragsdetails.ArtNr,
tArtikel.ArtBez,
tAuftragsdetails.Anzahl,
tArtikel.Einzelpreis,
Anzahl * Einzelpreis AS Zeilenpreis
from
tKunden inner join tAuftrag
on tKunden.KdNr = tAuftrag.KdNr
inner join tAuftragsdetails
on tAuftrag.AuftrNr = tAuftragsdetails.AuftrNr
inner join tArtikel
on tAuftragsdetails.ArtNr = tArtikel.ArtNr
where
tAuftragsdetails.AuftrNr = 1;

Aliasnamen:

select
tAuftragsdetails.AuftrNr,
tAuftrag.Datum,
```

```
tAuftrag.KdNr,
tKunden.Vorname,
tKunden.Nachname,
tAuftragsdetails.ArtNr,
tArtikel.ArtBez,
tAuftragsdetails.Anzahl,
tArtikel.Einzelpreis,
Anzahl * Einzelpreis AS Zeilenpreis
from
tKunden as k inner join tAuftrag as a
on k.KdNr = a.KdNr
inner join tAuftragsdetails as d
on a.AuftrNr = d.AuftrNr
inner join tArtikel as art
on d.ArtNr = art.ArtNr
where
d.AuftrNr = 1;
```

CROSS JOINS

```
use Verkauf
select tKunden.Nachname, tArtikel.ArtBez
from tKunden, tArtikel
```

liefert CARTESISCHES Produkt von tKunden X tArtikel

Nachname	ArtBez
Fröschl	Socken schwarz
Achatz	Socken schwarz
Zahler	Socken schwarz
Wehba	Socken schwarz
Thor	Socken schwarz
Keil	Socken schwarz
Moser	Socken schwarz
Fröschl	T-Shirt rot
Achatz	T-Shirt rot
Zahler	T-Shirt rot
Wehba	T-Shirt rot
Thor	T-Shirt rot
Keil	T-Shirt rot
Moser	T-Shirt rot
Fröschl	Socken blau
Achatz	Socken blau
Zahler	Socken blau
Wehba	Socken blau
Thor	Socken blau
Keil	Socken blau
Moser	Socken blau

(21 Zeile(n) betroffen)

Verknüpfungen mit derselben Tabelle:

```
use Auftrag
select
a.Vorname, a.Nachname, 'Chef:', b.Vorname, b.Nachname
from
tMitarbeiter as a left outer join tMitarbeiter as b
on a.Vorgesetzter = b.PersNr

select tKunden.vorname, tKunden.nachname from tKunden
union
select tMitarbeiter.vorname, tMitarbeiter.nachname from tMitarbeiter
order by nachname

select artnr, artBez, Einzelpreis,
(select avg(Einzelpreis) from tArtikel) as Durchschnittspreis,
Einzelpreis-(select avg(Einzelpreis) from tArtikel) as Unterschied
from tArtikel
```

/* Alle Kunden- und Auftragsnummern, die mehr als 20 Stück des Artikels mit der Artikelnummer 2 bestellt haben */

```
select AuftrNr, KdNr
from tAuftrag
where 20 <
(select anzahl from tAuftragsdetails
where tAuftrag.AuftrNr = tAuftragsdetails.AuftragsNr
and tAuftragsdetails.ArtNr = 2)
```

/* alle Artikel, deren Einzelpreis größer als der Durchschnittspreis ist */

```
select ArtNr, ArtBez, Einzelpreis
from tArtikel as Art1
where Art1.Einzelpreis >
(select avg(Art2.Einzelpreis) from tArtikel as Art2)
```

2.9 Einfügen, Ändern und Löschen von Daten

Einfügen einzelner Datenzeilen:

```
use verkauf
insert dbo.tKunden
values (815, 'Helmut', 'Keil')
go

/*
Fügt neuen Datensatz ein
Version: 1.5
```

```
Datum: 08.05.2007
*/
use verkauf
insert dbo.tKunden
(Nachname, Vorname, KdNr) -- Reihenfolge wird festgelegt
values ('Moser', 'Stefan', 1201)
go

insert dbo.tArtikel
values ('Socken schwarz', 1.5); -- Identity-Werte nicht explizit festlegen

insert dbo.tArtikel (ArtBez, Einzelpreis)
values ('T-Shirt rot', 13.62);

insert dbo.tArtikel (ArtBez, Einzelpreis)
values ('Socken blau', 1.21);
```

Einfügen mehrerer Datenzeilen, die aus anderen Tabellen selektiert werden:

```
use Auftrag
insert tKunden -insert into tKunden
select PersNr, Vorname, Nachname
from tMitarbeiter
go

select vorname, nachname into #TempKunden from tKunden

select * from #TempKunden

update tArtikel
set ArtBez='Hose'
where ArtNr=2;
```

2.10 Arbeiten mit vordefinierten Funktionen

Beispiel: Es soll der in der Spalte ContactName der Tabelle Customers der Datenbank Northwind befindliche Text in Vor- und Nachname geteilt werden.

```
select * into Customers2 FROM Customers

-- Findet die Länge des Nachnamens
SELECT ContactName, PATINDEX('%%', REVERSE(ContactName)) - 1 FROM Customers

-- Schneidet den Nachnamen aus
SELECT ContactName, Nachname = RIGHT(ContactName, PATINDEX('%%', REVERSE(ContactName)) - 1) FROM Customers

-- Schneidet den Teil vor dem Nachnamen aus
SELECT ContactName, Vorname =
RTRIM(SUBSTRING(ContactName, 1, LEN(ContactName) - PATINDEX('%%', REVERSE(ContactName)))) FROM Customers

-- Liefert eine Zahl, wenn im Rest noch ein Blank vorkommt
SELECT ContactName, Nacharbeit = PATINDEX('% ', RTRIM(SUBSTRING(ContactName, 1, LEN(ContactName) - PATINDEX('%%', REVERSE(ContactName)))) FROM Customers

UPDATE Customers2
SET
LastName = RIGHT(ContactName, PATINDEX('%%', REVERSE(ContactName)) - 1),
FirstName = RTRIM(SUBSTRING(ContactName, 1, LEN(ContactName) - PATINDEX('%%', REVERSE(ContactName)))) ,
BearbeitungsPosition = PATINDEX('% ', RTRIM(SUBSTRING(ContactName, 1, LEN(ContactName) - PATINDEX('%%', REVERSE(ContactName))))))

select ContactName, FirstName, LastName, BearbeitungsPosition, * from Customers2 where BearbeitungsPosition > 0
```

Funktion (Syntax)	Bedeutung
Len(String)	Berechnet die Anzahl der Zeichen eines Strings.
LTrim(String)	Entfernt führende Leerzeichen in einem String
RTrim(String)	Entfernt Leerzeichen am Ende eines Strings
PatIndex(Muster, String)	Ermittelt die Position, an der ein Muster das erste Mal im angegebenen String auftritt
Reverse(String)	Kehrt einen Text zeichenweise um (aus 'nebel' wird 'leben')

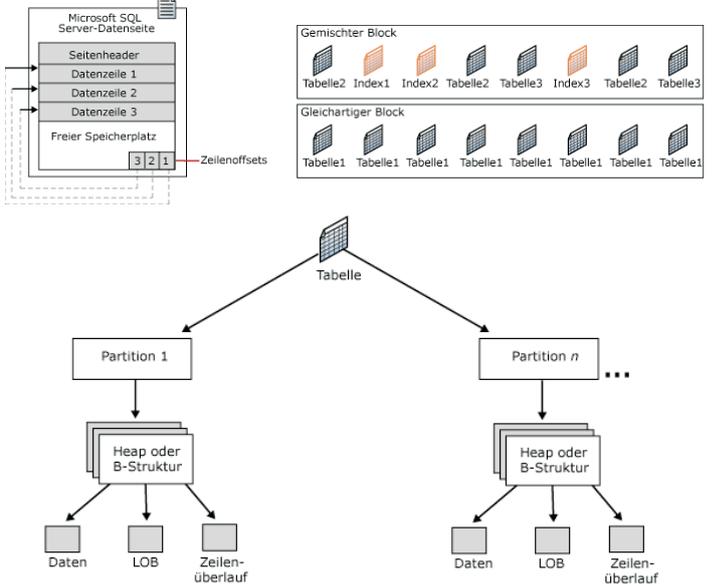
SubString(String, Beginn, Anzahl) Schneidet einen Teil des Strings heraus, beginnend vom Zeichen "Beginn" werden "Anzahl" Zeichen herausgeschnitten

Left(String, Anzahl) Liefert eine Anzahl von Zeichen, von links beginnend, von einem String

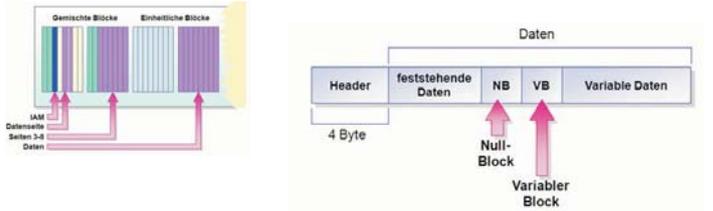
Right(String, Anzahl) Liefert eine Anzahl von Zeichen, von rechts beginnend, von einem String

2.11 Indizes

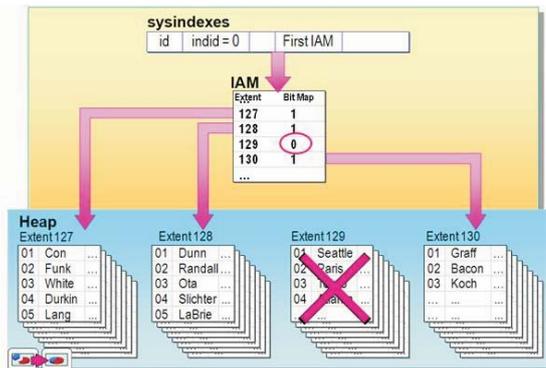
Interne Datenorganisation von SQL Server:



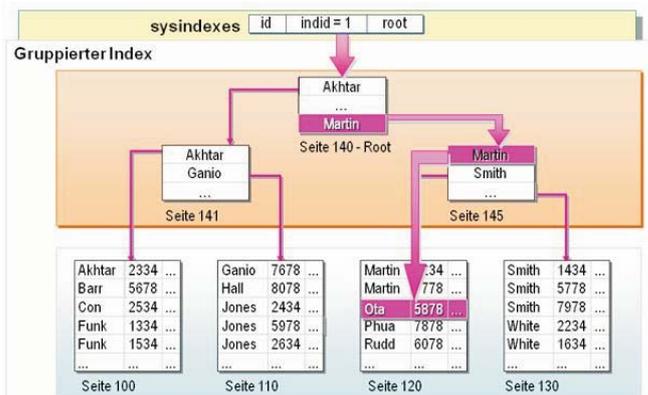
Eine Datenseite ist wie folgt aufgebaut:



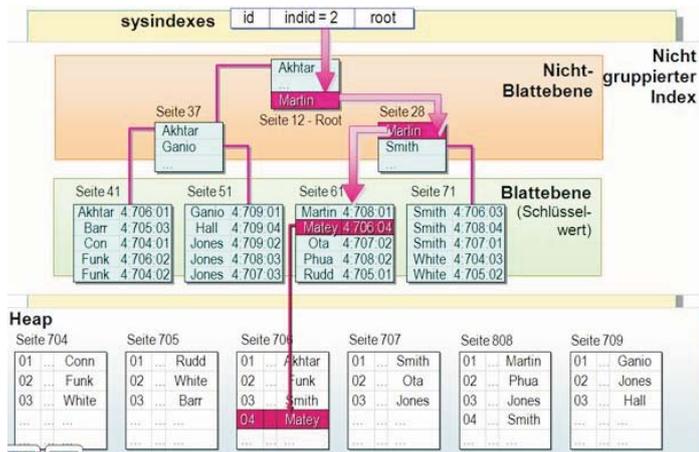
Suchen in einer Tabelle ohne Indizes (=Heap) (Wird auch als "table scan" bezeichnet):



Suchen mit gruppiertem Index:



Suchen mit nicht gruppierten Indizes:



Neu ist die Möglichkeit, während der Indexerstellung die Tabellen online zu halten.

```
CREATE INDEX ix_Employee_ManagerID on HumanResources.Employee(ManagerID) WITH (ONLINE=ON,MAXDOP=1)
```

„Covered Query“: Abfrage, bei der alle Spalten Teil eines Index sind.

Mit der INCLUDE-Funktion können nun auch Spalten aufgenommen werden, die nicht Teil des Indexes sind:

```
CREATE INDEX ix_AddressDetails on Contact.Address (AddressID) INCLUDE (AddressLine1, AddressLine2)
```

2.12 Einschränkungen(Constraints)

Spalten- und Tabelleneinschränkungen

Einschränkungen können Spalten- oder Tabelleneinschränkungen sein:

- Eine Spalteneinschränkung wird als Teil einer Spaltendefinition angegeben und gilt nur für diese Spalte.
- Eine Tabelleneinschränkung wird unabhängig von einer Spaltendefinition deklariert und kann für mehr als eine Spalte in einer Tabelle gelten. Tabelleneinschränkungen müssen verwendet werden, wenn mehr als eine Spalte in eine Einschränkung eingeschlossen werden muss.

Primary Key-Einschränkungen, Default-Einschränkungen

```
use Auftrag;
create table dbo.tArtikel
( ArtNr int identity(1,1),
  ArtBez nvarchar(50) NOT NULL,
  Einzelpreis money NOT NULL constraint DF_tArtikel_Einzelpreis default (0.0),
  constraint PK_tArtikel_ArtNr primary key nonclustered (ArtNr ASC) with (ignore_dup_key = off)
);
```

Wenn in einer Tabelle z. B. zwei oder mehr Spalten für den Primärschlüssel verwendet werden, müssen Sie eine Tabelleneinschränkung verwenden, um beide Spalten in den Primärschlüssel einzuschließen. Stellen Sie sich eine Tabelle vor, die Ereignisse aufzeichnet, die für einen Computer in einer Fabrik eintreten. Nehmen Sie weiterhin an, dass unterschiedliche Ereignistypen gleichzeitig eintreten können, dass jedoch nie zwei Ereignisse desselben Typs gleichzeitig eintreten. Dieser Sachverhalt kann in der Tabelle erzwungen werden, indem Sie die type- und die time-Spalte in einen Primärschlüssel einschließen, der zwei Spalten umfasst.

```
CREATE TABLE factory_process
(event_type int,
 event_time datetime,
 event_site char(50),
 event_desc char(1024),
 CONSTRAINT event_key PRIMARY KEY (event_type, event_time) )
CREATE TABLE tPLZ
(
 PLZ char(5) not NULL,
 Ort varchar(50) not NULL
 CONSTRAINT PK_tPLZ PRIMARY KEY (PLZ, Ort)
);
```

Foreign Key-Constraints

```
alter table dbo.tAuftrag
add MitarbeiterNr int null;

alter table dbo.tAuftrag
with check -- vorhandene Datensätze werden überprüft
add constraint FK_MitarbeiterNr_tMitarbeiter
foreign key(MitarbeiterNr)
references dbo.tMitarbeiter(MitarbeiterNr);
```

```
alter table dbo.tAuftrag -- beginnen Sie immer mit der Detailtabelle,
-- das ist die Tabelle, die den Fremdschlüssel
enthält
with check -- vorhandene Datensätze werden überprüft
add constraint FK_KdNr_tKunden
foreign key(KdNr) -- Fremdschlüssel
references dbo.tKunden(KdNr) -- Bezug auf Primärschlüssel der anderen
Tabelle
on update cascade; -- Kaskadierungsoptionen
```

3 Sichten (Views)

Man sollte nie direkt mit den Tabellen, sondern immer mit Abfragen arbeiten.

```
use Auftrag
go

create view dbo.Auftragszicht
as
select
tAuftragsdetails.AuftrNr,
tAuftrag.Datum,
tAuftrag.KdNr,
tKunden.Vorname,
tKunden.Nachname,
tAuftragsdetails.ArtNr,
tArtikel.ArtBez,
tAuftragsdetails.Anzahl,
tArtikel.Einzelpreis,
Anzahl * Einzelpreis AS Zeilenpreis
from
tKunden as k inner join tAuftrag as a
on k.KdNr = a.KdNr
inner join tAuftragsdetails as d
on a.AuftrNr = d.AuftragsNr
inner join tArtikel as art
on d.ArtNr = art.ArtNr
where
d.AuftrNr = 1;
```

WICHTIG: Niemals verknüpfte Primärschlüsselfelder in der Abfrage verwenden! Verknüpfte Felder in der Detailtabelle MÜSSEN in der Abfrage enthalten sein!

4 Gespeicherte Prozeduren (Stored Procedures)

Grundsätzliche Syntax:

```
create proc prKunden
as
select * from tKunden
```

Gespeicherte Prozeduren mit Eingabeparametern

Im folgenden Beispiel wird mit Hilfe eines Eingabeparameters eine Parameterabfrage realisiert:

```
create proc dbo.pSucheKdNr
@KdNr int
as
select
dbo.tKunden.KdNr,
dbo.tKunden.Vorname,
dbo.tKunden.Nachname
from
dbo.tKunden
where dbo.tKunden.KdNr = @KdNr;
```

```
exec pSucheKdNr 210
```

```
exec pSucheKdNr @KdNr=88
```

Verwendung von Rückgabewerten (return values)

```
alter proc dbo.pKundeEinfuegen
@KdNr int,
@Vorname nvarchar(50),
@Nachname nvarchar(50)
as
if (exists (select dbo.tKunden.KdNr from dbo.tKunden
where dbo.tKunden.KdNr = @KdNr))
begin
return -1 --Prozedur wird abgebrochen, Rückgabewert von -1
end
insert dbo.tKunden
values (@KdNr, @Vorname, @Nachname)
return 0
```

```
-- Beispiel für Rückgabewert: existiert der Kunde -> -1, sonst 0
declare @ret int
exec @ret = pKundeEinfuegen 37, 'Matthias', 'Gruber'
select @ret
```

Fehlerbehandlung mit TRY-CATCH-Strukturen

Konzept: Der TSQL-Code innerhalb der TRY-Anweisung wird testweise ausgeführt. Tritt ein Fehler auf, so wird sofort zum CATCH-Block verzweigt und die dort angeführten Anweisungen ausgeführt.

Wichtig: Im CATCH-Block muss standardmäßig unbedingt ein ROLLBACK TRAN durchgeführt werden, sonst bleibt die Transaktion im Fehlerfall "hängen".

Funktion	Bedeutung
ERROR_NUMBER()	Fehlernummer
ERROR_LINE()	Zeilennummer, in der der Fehler aufgetreten ist
ERROR_PROCEDURE()	Name der gespeicherten Prozedur, in der der Fehler aufgetreten ist
ERROR_SEVERITY()	Schweregrad: 0-10... Informationsmeldungen, 11-15... benutzerdefinierbare Fehler, 16-20... schwere Fehler, 21-25... kritische Fehler
ERROR_MESSAGE()	Fehlermeldung
ERROR_STATE()	Statuswert des Fehlers (normalerweise immer 0; sollte derselbe Fehler an mehreren Stellen des Programms auftreten können, kann dies über den Statuswert mitgeteilt werden)

```
create proc dbo.InsertAuftragsdetails
    @AuftrNr int,
    @ArtNr int,
    @Anzahl int
as
begin try -- wir versuchen folgenden TSQL-Code
    begin tran
        insert dbo.tAuftragsdetails
            values (@AuftrNr, @ArtNr, @Anzahl)
        commit tran
    end try

begin catch -- wenn obiger Code fehlerhaft ausgeführt
    rollback tran
    select ERROR_NUMBER() as Fehlernummer,
           ERROR_MESSAGE() as Fehlermeldung
end catch
```

```
- Testfälle:
exec InsertAuftragsdetails 1,99,3
-- Ergebnis: Fehler 547 / Beziehung mit tArtikel verletzt
exec InsertAuftragsdetails 2,2,3
-- Ergebnis: Fehler 2627 / Primärschlüssel bereits vorhanden
exec InsertAuftragsdetails 1,3,NULL
-- Ergebnis: Fehler 515 / NULL-Wert in Anzahl-Spalte verboten
```

Weiteres Beispiel

```
create proc pKundeInsert
    @KdNr int,
    @Vorname nvarchar(50),
    @Nachname nvarchar(50)
as
begin try
    insert tKunden
        values (@KdNr, @Vorname, @Nachname)
end try
begin catch
    select ERROR_NUMBER() Fehlernummer, ERROR_MESSAGE() Fehlermeldung
end catch
```

```
-test
exec pKundeInsert 123, 'Max', 'Muster'
(1 Zeile(n) betroffen)
exec pKundeInsert 123, 'Maria', 'Muster'
(1 Zeile(n) betroffen)
```

Gespeicherte Prozeduren mit Ausgabeparametern

Ausgabeparameter haben den Vorteil, dass sie im aufrufenden Programm weiterverwendet werden können. Während eine gespeicherte Prozedur nur genau einen Rückgabewert haben kann, können beliebig viele Ausgabeparameter verwendet werden.

```
/* Insert in die tArtikel-Tabelle
Hinweis: ArtBez ist IDENTITY und Primärschlüssel!!
*/
alter PROCEDURE dbo.InsertArtikel
    @ArtBez nvarchar(50)=NULL,
    @Einzelpreis money=NULL,
    @NeueArtNr int OUTPUT
AS
SET NOCOUNT ON
if (isnull(@ArtBez, '')='') or
(isnull(@Einzelpreis, '')='')
begin
    raiserror(50011,1,16)
    return
end
```

```
insert tArtikel
    (ArtBez, Einzelpreis)
values
    (@ArtBez, @Einzelpreis)

SET NOCOUNT OFF
-- SELECT @NeueArtNr = @@IDENTITY
set @NeueArtNr = SCOPE_IDENTITY()
GO
```

Benutzerdefinierte Fehlermeldungen müssen definiert werden; dazu steht die gespeicherte Systemprozedur sp_addmessage zur Verfügung:

```
/* Hinzufügen neuer benutzerdefinierter Fehlermeldung */
exec sp_addmessage 50011, 16,
    'Datensatz nicht hinzugefügt, da ArtBez NULL ist', 'us_english'

/* Test der Stored Procedure */
declare @neuenummer int
exec insertArtikel 'Schuhe blau', 12.45, @neuenummer OUTPUT
select @neuenummer
go

/* Test der Stored Procedure: Fehlermeldung */
declare @neuenummer int
exec insertArtikel '', 12.45, @neuenummer OUTPUT
go
```

Funktionen

```
CREATE FUNCTION fn_HoleOrt (@p1z varchar(10))
    RETURNS varchar(50)
AS
BEGIN
    return (SELECT PoOrt FROM tPlzOrt WHERE PoPlz = @p1z)
END
GO
```

5 Trigger

Trigger sind mit gespeicherten Prozeduren vergleichbar, die auf Grund einer Datenbankaktion automatisch ausgeführt werden.

Man unterscheidet:

- **DML-Trigger** (seit SQL Server 7.0 möglich): werden durch eine INSERT-, UPDATE- oder DELETE-Aktion ausgelöst
- **DDL-Trigger** (ab SQL Server 2005): werden durch DDL-Statements wie CREATE, ALTER, DROP ausgelöst

Bei den DML-Trigger unterscheidet man weiter:

- **AFTER-Trigger**: werden **nach** einem INSERT, UPDATE oder DELETE ausgeführt
- **INSTEAD OF-Trigger**: werden **statt** eines INSERT, UPDATE oder DELETE ausgeführt

```
create trigger trNeuerKunde on tKunden
after insert as
begin
    set nocount on
    insert dbo.tProtokol1
        select getdate(), user_name(), 'insert', inserted.KdNr
    from inserted
    set nocount off
end
```

Die logische **inserted**-Tabelle enthält die einzufügenden bzw. eingefügten Datensätze. Analog dazu gibt es auch eine logische **deleted**-Tabelle, die gelöschte Datensätze enthält.

```
/* Testen des Triggers */
insert tAuftrag (KdNr, Datum, MitarbeiterNr)
values (109, '24.05.2007', 220);
```

Beispiele

```
alter trigger trAendernKunde on tKunden
after update as
begin
    set nocount on
    insert dbo.tProtokol1
        select getdate(), user_name(), 'update', inserted.KdNr
    from inserted
    set nocount off
end
```

```
create trigger trLoeschenKunde on tKunden
after delete as
begin
    set nocount on
    insert dbo.tProtokol1
        select getdate(), user_name(), 'delete', deleted.KdNr
    from deleted
    set nocount off
end
```

Windows Workflow Foundation

Thomas Reinwart

1. Workflow - Einführung

Warum einen Workflow verwenden

Ein Workflow ist ein Modell bestehend aus Aktivitäten das einen Prozess beschreibt. Diese Aktivitäten koordinieren Personen oder Maschinen.

Mit der Einbindung eines Workflows gibt es in der Architektur einer Applikation neue Möglichkeiten, die *Business Logic (BL)* zu definieren. Statt wie bisher unüberschaubaren Code mit vielen *if else*-Zweigen zu kodieren, gibt es nun die Möglichkeit, dies in einen Workflow zu verpacken. So lässt sich nun die BL mittels eines graphischen Designers erstellen, damit ist dies wesentlich flexibler bei Veränderungen. Zudem ist es übersichtlich dokumentiert. Das betrifft auch weit komplexere Prozesse eines Unternehmens. Mit einem Workflow lassen sich Abläufe besser koordinieren.

Windows Workflow Foundation (WF)

Der Vorteil, einen Workflow zu verwenden ist, dass damit der Prozess recherchiert, beschrieben und daraus ein Modell erstellt wird. Meist werden die Prozesse als *UML, use cases* oder *flow chart* dokumentiert. Beim Verwenden von WF sind die Beschreibung und der Workflow ein und das selbe System, d.h. es gibt eine visualisierte Darstellung des Workflow beim Design und zur Laufzeit, der Code der Activities ist mit dem Workflow verbunden. Somit gibt es keine getrennte Dokumentation die auseinander laufen kann.

Welcher meiner laufenden Workflows derzeit in welchem Status sich in welchem Step befindet, zeigt mir Tracking im selben Design visuell an.

Ein Workflow kann im Designer verändert werden, ohne dass dabei Code geändert werden muss. Ich kann einen Workflow in ein Produkt einbauen, dabei wird jeder Workflow für den Kunden angepasst, der Rest bleibt gleich. Bzw. der Kunde passt sich seinen Workflow selber an.

Meist läuft am Rechner eine Vielzahl von Programmen, die nur von Zeit zu Zeit echte Dienste leisten. Die meiste Zeit jedoch verbringen diese Anwendungen damit, auf Usereingaben oder auf ein anderes programatisches Event zu warten. Die Anwendungen laufen also und tun nichts anderes als Ressourcen zu verbrauchen.

In einem Workflow können die Objekte serialisiert und deserialisiert werden, die Persistierung in die Datenbank. Somit ist der Status, in dem sich der Workflow der Anwendung befindet eingefroren. Wenn nun ein bestimmtes Event eintritt, beginnt der Workflow wieder zu arbeiten. Durch diese persistieren des Zustandes ist es möglich, das Laden auf einen anderen Rechner und Zeitpunkt durchzuführen. Man hat damit selber keinen Implementierungsaufwand, sich um die Persistierung eines Workflows Status zu kümmern.

Ein Workflow kann in jedem Projekttyp eingebunden werden, vom simplen Comand Line Tool bis Windows Forms, XAML, Asp.net. Innerhalb des Workflows kann über Code Activities

wiederrum eigene Assemblies aufgerufen werden, oder auch Webservice eingebunden werden.

Komponenten

- *Base Activity Library*: mit der WF mitgelieferte Activities
- *Runtime engine*
- *Runtime services*: Hosting und Kommunikation
- *WF Visual Designer*: Control kann auch in der eigener Applikation gehostet werden

1.1 Sequential oder State Workflows verwenden

Die Windows Workflow Foundation bietet zwei Workflow Arten an, den *Sequential* und den *State Workflow*. Beim Projektbeginn muss ich mir im Klaren sein, welcher Workflow besser zu meinem Projekt passt.

1.1.1 Der State Workflow

Der Workflow befindet sich in einem bestimmten Status und wartet auf Events um in einen anderen Status zu gelangen. Der neue Status kann irgendein anderer Status sein. Status können verschachtelt sein. Gut geeignet für menschliche Interaktion.

Dieser Workflow führt die Aktivitäten aufgrund des Status aus. Der Workflow hängt stark von den externen Modulen ab. Userinteraktion bzw. Ergebnis der externen Module beeinflussen den Status. Zustandsübergänge werden durch externe Events ausgelöst.

Er eignet sich dann, wenn mehr Useraktion gewünscht wird. Wenn dieser Workflow eine Benachrichtigung an einen anderen Benutzer oder ein anderes System sendet, gibt es Antworten in Form von Events. Dadurch wird ein Prozess Status durch einen anderen Prozess Status getauscht.

Ein Beispiel für einen State Workflow ist das Zusammenspiel von verschiedenen Interaktionen einer Produktion eine Maschine mit einem Bediener.

Start – Material entnehmen – Material reinigen – Material bearbeiten – Material Kontrolle – Material bearbeiten – Material Ausschuss oder *OK*

Dazwischen kann jeder einzelne Step auch wieder zum Vorgänger springen. Der Workflow wird hier lange Zeit in einem der Steps verbringen.

Der Unterschied zum *Sequential Workflow* ist, dass beim *State Workflow* auch ein vorheriger Status erreicht werden kann.

1.1.2 Der Sequential Workflow

Der Plan eines sequentiellen Workflows ist vorgegeben. Er kann Bedingungen, Schleifen usw. beinhalten. Dies passt mehr zu automatisierten Prozessen oder starren Protokollen. Er hat einen definierten Start und ein definiertes Ende, kann aber auch ewig laufen.

Dieser Workflow führt die Activities hintereinander aus. Der Workflow hat dabei ausführbaren Tasks unter Kontrolle. Es kann Userinteraktion geben, der Workflow kann aber auch vollautomatisch laufen.

Er ist ideal für die Umsetzung von Business Prozessen, die komplex sind.

Dieser Workflow arbeitet eine Activity nach der anderen ab, es sei denn eine Exception wird während aber Workflow Abarbeitung gefeuert oder wenn eine *TerminateActivity* ausgeführt wird.

Beispiel: Daten von einer Quelle lesen, Daten verarbeiten, eine Benachrichtigung schicken, Ergebnisdaten schreiben. Die Userinteraktion ist also nicht unbedingt notwendig.

Zur Verfügung stehende Activities im Workflow

Windows Workflow v3.0

- Pointer
- CallExternalMethod
- Code
- Compensate
- CompensatableSequence
- ConditionedActivityGroup
- Delay
- EventDriven
- EventHandlingScope
- FaultHandler
- HandleExternalEvent
- IfElse
- InvokeWebService
- InvokeWorkflow
- Parallel
- Policy
- Replicator
- SetState
- Sequence
- State
- StateInitialization
- StateFinalization
- Suspend
- SynchronizationScope
- Terminate
- Throw
- TransactionScope
- CompensatableTransactionScope
- WebServiceInput
- WebServiceOutput
- WebServiceFault
- While

Windows Workflow v3.5

- Pointer
- ReceiveActivity
- SendActivity

Activity: alles in einem Workflow ist eine Activity, auch der Workflow selber, der einen bestimmten Typ einer Activity darstellt.

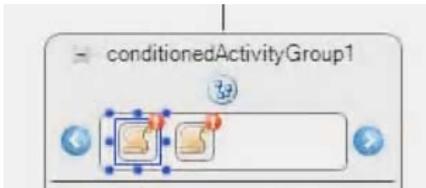
CallExternalMethod: Damit lassen sich Methoden außerhalb eines Workflows aufrufen. InterfaceType und MethodName müssen dabei definiert werden. InterfaceType gibt an, welches Runtime Service genutzt werden soll, wenn einen Activity ausgeführt wird. MethodName gibt an, welche Methode des angegebenen Interface aufgerufen werden soll.

Code: Hiermit lässt sich eigener .net Code im Workflow ausführen. Der Sourcecode ist getrennt vom Workflow.

Compensate: Ist eine Error Activity und kann zu einer Exception Handler Activity hinzugefügt werden. Kann genutzt werden um ein Rollback der Änderungen im Falle eines Fehlers durchzuführen.

Compensatable Sequence: Kann für einen bestimmten Typ eines Fehlers verwendet werden. Funktioniert ähnlich wie mit Datenbank Transaktionen. Der Unterschied ist, dass die Datenbank solange sperrt bis alle Updates fertig sind, also für eine kurze Zeit. Eine Compensate Action kann bei Transaktionen verwendet werden, die lange andauern, um die original Transaktion rückgängig zu machen.

Conditioned Activity Group: Dies ist eine bedingte Aktivität, die andere Aktivitäten, basierend auf Bedingungen, die für die Gruppe gelten, oder die Aktivitäten die der Gruppe zugeordnet, ausführt.



Delay: Bei der Verwendung einer DelayActivity wird der Wartezeitraum und die Methode angegeben, die im Fall der Zeitüberschreitung ausgeführt wird. Im Beispiel eine Genehmigungsworkflows kann man ein Delay verwenden. Wenn nun eine Person innerhalb des angegebenen Zeitraums nicht reagiert, wird der Alternativzweig im Workflow (z.B. andere Person kontaktieren) verwendet.

Event Driven: Diese Activity kann andere Activities beinhalten, die ausgeführt werden wenn dieses Event eintritt.

Event Handling Scope: kann eine Anzahl von Activities beinhalten. Während diese Activities ausgeführt werden, können die Events empfangen werden.

Fault Handler: Eine Error handling Activity, die wie ein catch block funktioniert.

Handle External Event: Die HandleExternalEventActivity wird in Verbindung mit der CallExternalMethodActivity für die In und Out Kommunikation mit einem lokalen Dienst verwendet. Sie können diese Aktivitäten direkt für die allgemeine Kommunikation verwenden. Oder Sie können Unterklassen von HandleExternalEventActivity und CallExternalMethodActivity erstellen um Aktivitäten, die streng an bestimmte Ereignisse und Methoden an ein Interface gebunden ist das ein ExternalDataExchangeAttribute besitzt, zu erzeugen.

If Else: Kann man sich als if else so wie in der Programmiersprache vorstellen. Über eine Condition wird die Bedingung festgelegt.

Invoke Web Service: kann ein Methode eines Webservice mittels Proxyklasse ausführen. Parameter können übergeben und empfangen werden.

Invoke Workflow: Ruft den zugeordneten Workflow auf Parallel: Im Workflow werden Activities als single thread ausgeführt. Es ist also keine wirkliche parallele Ausführung. Die Workflow runtime verarbeitet die Activity Queue pro Workflow Instanz im Prinzip von FIFO (first in, first out) ab.

Policy: Dies stellt eine Auflistung von Regeln dar. Eine Regel hat eine Bedingung und Action(en), die durchgeführt werden, wenn die Bedingung erfüllt ist. Das erlaubt einen regelbasierten Workflow, anstatt einem IfElse basierendem Workflow.

Replicator: Ist auch eine Bedingungsaktivität. Wie wenn man eine "For Each" Anweisung verwendet. Zur Laufzeit werden eine Anzahl von Instanzen einer Aktivität erstellt, die abgeschlossen werden müssen, bevor die Replikator Aktivität beendet wird.

Set State Activity: Dies ist ein flow activity, die für den Statuswechsel bei einem State Machine Workflow verwendet wird.

State Activity: Ist eine flow activity und stellt einen Status in einem State Machine Workflow dar.

State Initialization Activity: Diese Aktivität ist Bestandteil einer State Aktivität, die aus anderen Aktivitäten besteht, die ausgeführt werden, wenn die State Aktivität initialisiert wird.

Sequence: Ist eine zusammengesetzte Aktivität, die mehrere sequenzielle Aktivitäten beinhaltet. Es bietet eine einfache Möglichkeit, diese Tätigkeiten zu verknüpfen, die damit in Folge ausgeführt werden.

Suspend: Ist eine flow-Aktivität und kann den Ablauf eines Workflows für ein Einschreiten pausieren, wenn eine Fehlerbedingung (error condition) eintritt.

Synchronization Scope: Synchronisiert Workflow Aktivitäten

Terminate: Diese Aktivität beendet die Ausführung des Workflows sofort, wenn ein Fehler auftritt. Sie protokolliert auch den Fehler.

Throw: Mit dieser Aktivität können Sie eine Exception auslösen.

Transaction Scope: Diese Aktivität bietet Transaktionsunterstützung. In dieser Aktivität können sich Aktivitäten befinden, die eine Transaktion bilden.

Compensatable Transaction Scope: Es gibt zwei Aktivitäten die dieses Interface implementieren: CompensatableTransactionScopeActivity und CompensatableSequenceActivity.

WebService Input, Output, Fault: WF Webservice Integration

While: Dies ist eine bedingte Aktivität und ist wie eine "while" Bedingung in der Anwendung. Es wird eine andere Aktivität ausgeführt, bis eine Bedingung erfüllt ist. Die Bedingung kann ein Code oder einfach eine Regel basierende Bedingung sein.

Condition: es gibt die CodeCondition und die RuleConditionReference

CodeCondition: diese führt eine Methode der eigene code behind Klasse aus. Diese gibt true oder false zurück. Mit Code Condition sind Entwickler vertraut, hiermit lassen sich komplexe Bedingungen einfacher als in einer RuleConditionReference abbilden. Das Ergebnis einer CodeCondition wird als Result Property der ConditionalEventArgs zurückgeliefert.

RuleConditionReference: diese kann extern als XML gespeichert werden, die Extension ist .rules. Das hat den Vorteil, dass dies extern durch andere Tools geändert werden kann, auch zur Laufzeit.

Parameterübergabe in Workflows: Bei einer Activity gibt es normalerweise einen Zusammenhang zu den Bewegungsdaten, also zum Beispiel zur Customer Id. Zum Unterschied zum Aufruf von Klassen in .net, wo Parameter bei Methoden übergeben werden, gibt es bei der Parameterübergabe in Workflows ein Dictionary von Name/Values.

1.1.3 Für welchen Workflow soll ich mich entscheiden?

Ein State Workflow wartet auf externe Events bevor er in einen anderen Step springt.

Ein Sequential Workflow ist ein zusammenhängender Fluss von Activities. Activities in einem Sequential Workflow warten nicht auf externe Instanzen um dann in den nächsten Step zu springen.

Man kann jeden State Workflow auch als Sequential Workflow abbilden. Früher oder später wird dieser aber unüberschaubar werden.

2. Workflow Speicherformat

XOML (Extensible Object Markup Language) ist eine auf XML basierende Syntax. Ein Workflow kann als XOML gespeichert werden.

2.1 SqlPersistenceService

Per default laufen alle Workflows im Memory, solange auch die Applikation läuft, die den Workflow hostet. Zum Persistieren jeder einzelner Instanz des laufenden Workflows kann das mitgelieferte SqlPersistenceService verwendet werden. Damit wird der Zwischenstand der Workflows gespeichert, die Host Applikation kann jederzeit beendet und neu gestartet werden, außerdem können Workflows aus dem Speicher entladen werden wenn sie idle gehen. (Spart Memory, Performance)

Der Workflow kann damit am Filesystem oder in der Datenbank (SQL Server, SQL Express, ...) gespeichert werden.

```
// Add the SqlWorkflowPersistenceService
WorkflowPersistenceService persistenceService =
new SqlWorkflowPersistenceService (
    "Initial Catalog=SqlPersistenceService;" +
    "Data Source=localhost;" +
    "Integrated Security=SSPI;",
    true,
    TimeSpan.FromHours(1.0),
    TimeSpan.FromSeconds(5.0));
workflowRuntime.AddService(persistenceService);
```

Parameter

ConnectionString: Verbindung zur bestehenden Persist DB

UnloadOnIdle: muss auf true stehen, sonst wird nichts persistiert. Gibt an, dass der Workflow entladen wird, wenn er idle ist. Default Wert false.

LoadingInterval: Zeitintervall, in der die Persist DB gepollt wird, um die persist records zu schreiben.

DB Installationsscript:

c:\WINDOWS\Microsoft.NET\Framework\v3.0\Windows Workflow Foundation\SQL\EN\

Typ: SqlPersistanzService DB und Tracking DB können einzeln und getrennt verwendet werden. Wenn beides verwendet wird, empfiehlt sich aber beides in eine DB (SQL Server 2000, SQL Server 2005, oder SQL Server 2005 Express Edition) zu geben.

Microsoft Distributed Transaction Coordinator Service muss gestartet werden. Bei Verwendung eines Remote Servers muss Port 135 für Microsoft Distributed Transaction Center (MSDTC) offen sein.

2.2 Tracking

Da Workflows im Hintergrund laufen und die Workflow an sich auch komplexe Ausmaße annehmen werden, können diese mittels Tracking visualisiert werden. Bei der Instanziierung des Workflows wird das SqlTracking Service angegeben. Somit wird jede Instanz der Workflows getrackt.

Das Tracking ist für laufende als auch für beendete Workflows möglich, d.h. ich kann mir während des Ausführens als auch im Nachhinein den durchlaufenen Workflow ansehen. Die Darstellung erfolgt graphisch, der durchlaufene Workflow wird abgehackt dargestellt, der aktuellen Step mit einem grünen Dreieck.

DB Installationsscript liegt im Installationsverzeichnis:

c:\WINDOWS\Microsoft.NET\Framework\v3.0\Windows Workflow Foundation\SQL\EN\

3. Praxis

3.1 Workflow Hosting

Beispielanwendungen

- Automatisierungen von Abläufen
- Dokumenten Management
- Ticket Systeme
- Seitennavigation
- Komplexe Business Logic
- Häufige Änderungen an der Business Logic

Das Hosting kann in einem Service, über Webservice, asp.net, ... erfolgen.

In diesem Beispiel in einem Windows Service:

3.2 Workflow Tracking

Das Tool wird in den Samples mitgeliefert.

Workflow Monitor Sample

<http://msdn.microsoft.com/en-us/library/ms741723.aspx>

Um dies nutzen zu können, sind folgende Schritte notwendig:

- SQL Tracking DB muss angelegt worden sein.
- Eigener Workflow ist design und compiliert, Workflow wurde gestartet
- Erzeugt Workflow Assembly entweder ins Bin Verzeichnis vom Workflow Monitor kopieren oder die Assembly in den GAC geben (gacutil -i demo.dll)
- Workflow Tracking starten, Tracking DB angeben

3.3 Praxis Beispiel

In jedem organisieren Unternehmen gibt es wiederkehrende Abläufe, die man in einem Workflow Prozess abbilden kann. Dies betrifft einfache und komplizierte Vorgänge. Alleine durch die Analyse innerhalb einer Firma erge-

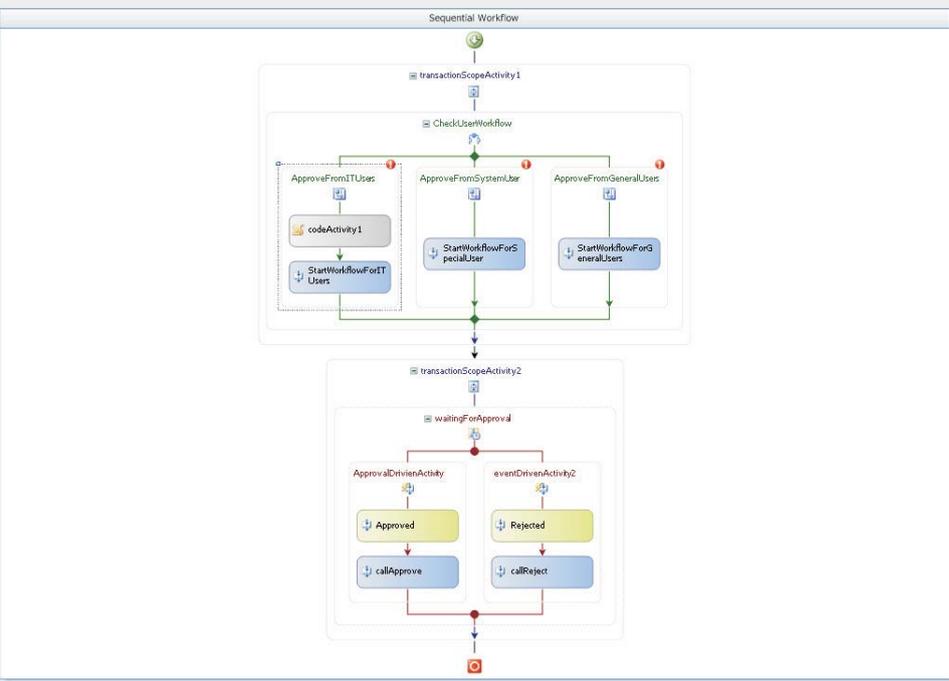
```
#region fields
WorkflowRuntime wfruntime = new WorkflowRuntime();
static AutoResetEvent _waitHandle = new AutoResetEvent(false);
private System.Collections.ObjectModel.ReadOnlyCollection<WorkflowInstance>
_loadedWorkflows;
#endregion
protected override void OnStart(string[] args)
{ ThreadPool.QueueUserWorkItem(new WaitCallback(RunWorkflow), args);
_wfruntime.WorkflowCompleted +=
new EventHandler<WorkflowCompletedEventArgs>(OnWorkflowCompleted);
}
private void RunWorkflow(object obj) {
try
{
// Load WF from SqlWorkflowPersistenceService // Create the WorkflowRuntime
using (WorkflowRuntime workflowRuntime = new WorkflowRuntime())
{
// Add the SqlWorkflowPersistenceService service
WorkflowPersistenceService persistenceService =
new SqlWorkflowPersistenceService (
"Initial Catalog=SqlPersistenceService;" +
"Data Source=localhost;Integrated Security=SSPI;",
true, TimeSpan.FromHours(1.0), TimeSpan.FromSeconds(5.0));
workflowRuntime.AddService(persistenceService);
// Tracking
workflowRuntime.AddService (
new SqlTrackingService("Initial Catalog=Tracking;Data Source=localhost;" +
Integrated Security=SSPI;"));
// Set up the WorkflowRuntime event handlers
workflowRuntime.WorkflowCompleted += OnWorkflowCompleted;
workflowRuntime.WorkflowIdled += OnWorkflowIdled;
workflowRuntime.WorkflowPersisted += OnWorkflowPersisted;
workflowRuntime.WorkflowUnloaded += OnWorkflowUnloaded;
workflowRuntime.WorkflowLoaded += OnWorkflowLoaded;
workflowRuntime.WorkflowTerminated += OnWorkflowTerminated;
workflowRuntime.WorkflowAborted += OnWorkflowAborted;
_loadedWorkflows = workflowRuntime.GetLoadedWorkflows();
Console.WriteLine("Workflows loaded: ", _loadedWorkflows.Count);
}
}
}
catch (Exception ex)
{
EventLog.WriteEntry("Hosting WF in an Windows Service", e.ToString(),
EventLogEntryType.Information, 1000);
}
}
```

ben sich oft Schwachstellen, Lücken oder Missverständnisse die Arbeitsvorgänge undurchsichtig, unkontrollierbar und zeitaufwendig gestalten.

Es bietet sich an, dies als elektronischen Workflow abzubilden.

Anhand eines einfachen Beispiels möchte ich dies veranschaulichen.

Es handelt sich dabei um einen Genehmigungsprozess, bei dem ein bereitgestelltes Dokument durch einen Gruppe oder einen Benutzer akzeptiert werden muss.

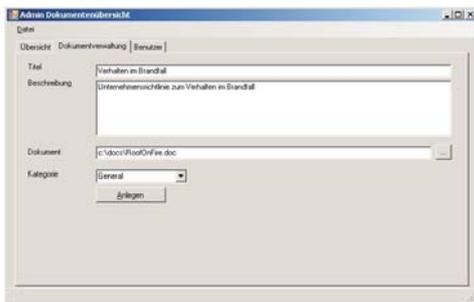


Ablauf

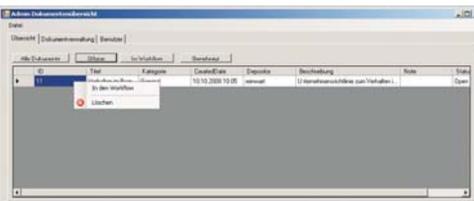
Eine leitende Person erstellt Dokumente und stellt diese in den Workflow. In dem Workflow sind drei Abteilungen definiert: IT User, System User, und allgemeine Benutzer.

Im Workflow sind diese Gruppen durch eine IfElseActivity aufgeteilt, daher kann jede Gruppe eine besondere Business Logic (BL) aufweisen. Eine solche BL lässt sich in einer CodeActivity unterbringen. In einer solchen CodeActivity lässt sich eigener Code ausführen.

Die Nutzung des zuvor erstellten Workflows ist in diesem Fall eine simple .net Windows Forms Anwendung, ist aber vom Workflow völlig unabhängig.



Ein Dokument mit den Attributen Titel, Beschreibung und Kategorie wird angelegt.



In der Übersichtsliste erscheint das Dokument und kann in den Workflow übernommen werden.

Technischer Teil

In der Applikation wurde das Persistence Service und das TrackingService der Windows Workflow Foundation eingebunden.

Persistence Service

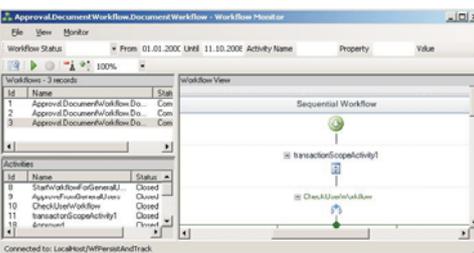
Jede Workflow Änderung eines Status wird in die Datenbank geschrieben.

TrackingService

Bietet die Möglichkeit, bei Instanz eines Workflow den aktuellen Status zu erfassen

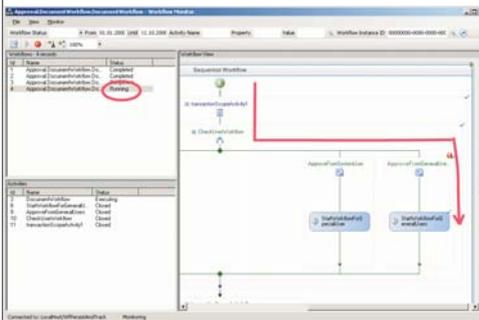
Tracking Monitor

Zeigt den aktuellen Workflow an

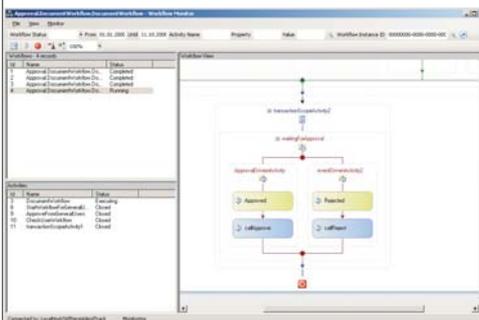


Damit lässt sich jeder aktuelle Step jedes einzelnen Workflow Überwachen und verständlich darstellen.

Das Dokument wurde in der Workflow gestellt, der Workflow läuft. (workflow running)

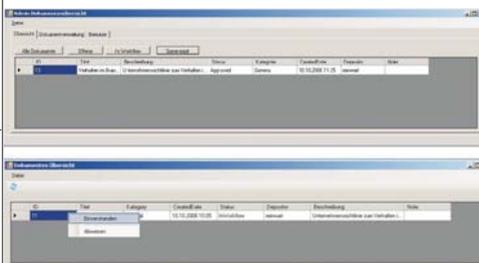


Der Workflow hat nun den Weg für die Kategorie General User genommen und wartet nun auf die Bestätigung durch den zugewiesenen User.



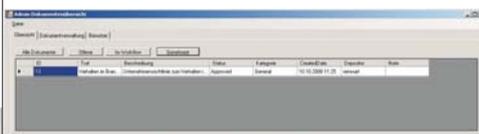
Der User hat nun die Möglichkeit, das Dokument, das nun diesen Workflow durchläuft zu bestätigen oder abzuweisen.

Für den User gibt es eine andere Applikation. Zu Beginn muss er sich authentifizieren, nach der Anmeldung sieht er die für ihn bestimmten Dokumente, für die eine Zustimmung notwendig ist.



Er liest sich den Inhalt des Dokuments durch und ist damit einverstanden.

Die Anwendung aus der Sicht des Administrators: Das Dokument wurde genehmigt und ist als „Approved“ in der Liste geführt.



Der Administrator hat laufend die Übersicht, welche User die Dokumente bereits akzeptiert haben. Bei einer Überschreitung eines Zeitraumes könnte man in diesem Demo Workflow ohne viel Aufwand auch um eine Eskalationsstufe (z.B. automatische generiertes Email) erweitern.

4. Einschränkungen mit WF 3.0

Performance: Microsoft selbst dokumentiert dies in dem Dokument "Performance Characteristics of Windows Workflow Foundation".

(<http://msdn.microsoft.com/en-us/library/aa973808.aspx>)

Versionierung von Workflows

Ein Workflow kann jederzeit geändert werden, aber die bestehenden Workflows kann man nicht mit der neuen Definition weiterlaufen lassen. Wenn Sie bei einem Prozess plötzlich am Ende noch einen Schritt brauchen, können Sie diesen also nicht bei Workflows anwenden, die bereits gestartet sind (auch wenn diese langlebig sind).

Keine Kompatibilität zu SSIS-Workflows

Microsoft SQL Server Integration Services (SSIS) bietet eine ähnliche Umgebung wie WF, ist jedoch in keiner Weise kompatibel zu WF. Das Entwicklungsteam von SSIS hat sich aktiv gegen die Verwendung von WF als Basis ausgesprochen (einerseits war SSIS vor WF auf dem Markt, andererseits ist SSIS mehr auf Massendatenverarbeitung fokussiert, dafür wäre WF zu langsam).

Designer Control Hosting

Eigener Code zusätzlich zum WF Control notwendig, für einen WF Beginner nicht einfach. Mit dem Designer Control der Version 4.0 wird es einfacher sein.

5. Zukünftige Version WF 4.0

Auf der PDC 2008 wurde von Kenny Wolf, einem Microsoft Architect von WF und WCF, die Zukunft von WF in der Version 4.0 präsentiert. Workflow 4.0 soll zusammen mit dem .net Framework 4.0 und Visual Studio 2010 auf den Markt kommen.

In der Vorführung erkennt man die neue einfachere Oberfläche für das Design eines Workflow Prozesses. Aufgrund der Erkenntnis, das die WF Versionen 3.0 und 3.5 Schwächen im Bereich Funktionsumfang, Bedienung, Komplexität und Leistung aufweisen und die Kundenwünsche damit nicht immer befriedigt werden können, wurde WF von Grund auf neu geschrieben. Das Konzept der Aktivitäten im Workflow und die Persistierung und die Möglichkeit von Monitoring bleibt. Es soll eine 10 bis 100-fache Leistungssteigerung geben, ebenso eine volle Kontrolle über die Persistierung. WF 4.0 wird rein deklarativ sein.

WF 4.0 vs. WF 3.0

- Activity
 - Authoring is simpler and takes much less code
 - Fully declarative workflows and activities
 - Alignment across Expressions, Rules, and Activities
 - Seamless Composition Across Flow Styles
- Runtime
 - 10-100X Performance Improvements
 - Full control over persistence
 - Flow-in Transactions
 - Partial Trust Support
 - Integrates with WCF, WPF, ASP.NET
- Tools
 - Designer Performance and Usability
 - Rehosting Improvements
 - Unified Debugging Experience

Preparing for WF 4.0

- 3.0/3.5 workflows continue to work (on the 3.0 WF runtime)
- Can use 3.0 Activities within a 4.0 Workflow
- Guidance on how to prepare 3.0/3.5 code

Windows Workflow Foundation

Thomas Reinwart

1. Workflow - Einführung

Warum einen Workflow verwenden

Ein Workflow ist ein Modell bestehend aus Aktivitäten das einen Prozess beschreibt. Diese Aktivitäten koordinieren Personen oder Maschinen.

Mit der Einbindung eines Workflows gibt es in der Architektur einer Applikation neue Möglichkeiten, die *Business Logic (BL)* zu definieren. Statt wie bisher unüberschaubaren Code mit vielen *if else*-Zweigen zu kodieren, gibt es nun die Möglichkeit, dies in einen Workflow zu verpacken. So lässt sich nun die BL mittels eines graphischen Designers erstellen, damit ist dies wesentlich flexibler bei Veränderungen. Zudem ist es übersichtlich dokumentiert. Das betrifft auch weit komplexere Prozesse eines Unternehmens. Mit einem Workflow lassen sich Abläufe besser koordinieren.

Windows Workflow Foundation (WF)

Der Vorteil, einen Workflow zu verwenden ist, dass damit der Prozess recherchiert, beschrieben und daraus ein Modell erstellt wird. Meist werden die Prozesse als *UML, use cases* oder *flow chart* dokumentiert. Beim Verwenden von WF sind die Beschreibung und der Workflow ein und das selbe System, d.h. es gibt eine visualisierte Darstellung des Workflow beim Design und zur Laufzeit, der Code der Activities ist mit dem Workflow verbunden. Somit gibt es keine getrennte Dokumentation die auseinander laufen kann.

Welcher meiner laufenden Workflows derzeit in welchem Status sich in welchem Step befindet, zeigt mir Tracking im selben Design visuell an.

Ein Workflow kann im Designer verändert werden, ohne dass dabei Code geändert werden muss. Ich kann einen Workflow in ein Produkt einbauen, dabei wird jeder Workflow für den Kunden angepasst, der Rest bleibt gleich. Bzw. der Kunde passt sich seinen Workflow selber an.

Meist läuft am Rechner eine Vielzahl von Programmen, die nur von Zeit zu Zeit echte Dienste leisten. Die meiste Zeit jedoch verbringen diese Anwendungen damit, auf Usereingaben oder auf ein anderes programatisches Event zu warten. Die Anwendungen laufen also und tun nichts anderes als Ressourcen zu verbrauchen.

In einem Workflow können die Objekte serialisiert und deserialisiert werden, die Persistierung in die Datenbank. Somit ist der Status, in dem sich der Workflow der Anwendung befindet eingefroren. Wenn nun ein bestimmtes Event eintritt, beginnt der Workflow wieder zu arbeiten. Durch diese persistieren des Zustandes ist es möglich, das Laden auf einen anderen Rechner und Zeitpunkt durchzuführen. Man hat damit selber keinen Implementierungsaufwand, sich um die Persistierung eines Workflows Status zu kümmern.

Ein Workflow kann in jedem Projekttyp eingebunden werden, vom simplen Comand Line Tool bis Windows Forms, XAML, Asp.net. Innerhalb des Workflows kann über Code Activities

wiederrum eigene Assemblies aufgerufen werden, oder auch Webservice eingebunden werden.

Komponenten

- *Base Activity Library*: mit der WF mitgelieferte Activities
- *Runtime engine*
- *Runtime services*: Hosting und Kommunikation
- *WF Visual Designer*: Control kann auch in der eigener Applikation gehostet werden

1.1 Sequential oder State Workflows verwenden

Die Windows Workflow Foundation bietet zwei Workflow Arten an, den *Sequential* und den *State Workflow*. Beim Projektbeginn muss ich mir im Klaren sein, welcher Workflow besser zu meinem Projekt passt.

1.1.1 Der State Workflow

Der Workflow befindet sich in einem bestimmten Status und wartet auf Events um in einen anderen Status zu gelangen. Der neue Status kann irgendein anderer Status sein. Status können verschachtelt sein. Gut geeignet für menschliche Interaktion.

Dieser Workflow führt die Aktivitäten aufgrund des Status aus. Der Workflow hängt stark von den externen Modulen ab. Userinteraktion bzw. Ergebnis der externen Module beeinflussen den Status. Zustandsübergänge werden durch externe Events ausgelöst.

Er eignet sich dann, wenn mehr Useraktion gewünscht wird. Wenn dieser Workflow eine Benachrichtigung an einen anderen Benutzer oder ein anderes System sendet, gibt es Antworten in Form von Events. Dadurch wird ein Prozess Status durch einen anderen Prozess Status getauscht.

Ein Beispiel für einen State Workflow ist das Zusammenspiel von verschiedenen Interaktionen einer Produktion eine Maschine mit einem Bediener.

Start – Material entnehmen – Material reinigen – Material bearbeiten – Material Kontrolle – Material bearbeiten – Material Ausschuss oder *OK*

Dazwischen kann jeder einzelne Step auch wieder zum Vorgänger springen. Der Workflow wird hier lange Zeit in einem der Steps verbringen.

Der Unterschied zum *Sequential Workflow* ist, dass beim *State Workflow* auch ein vorheriger Status erreicht werden kann.

1.1.2 Der Sequential Workflow

Der Plan eines sequentiellen Workflows ist vorgegeben. Er kann Bedingungen, Schleifen usw. beinhalten. Dies passt mehr zu automatisierten Prozessen oder starren Protokollen. Er hat einen definierten Start und ein definiertes Ende, kann aber auch ewig laufen.

Dieser Workflow führt die Activities hintereinander aus. Der Workflow hat dabei ausführbaren Tasks unter Kontrolle. Es kann Userinteraktion geben, der Workflow kann aber auch vollautomatisch laufen.

Er ist ideal für die Umsetzung von Business Prozessen, die komplex sind.

Dieser Workflow arbeitet eine Activity nach der anderen ab, es sei denn eine Exception wird während aber Workflow Abarbeitung gefeuert oder wenn eine *TerminateActivity* ausgeführt wird.

Beispiel: Daten von einer Quelle lesen, Daten verarbeiten, eine Benachrichtigung schicken, Ergebnisdaten schreiben. Die Userinteraktion ist also nicht unbedingt notwendig.

Zur Verfügung stehende Activities im Workflow

Windows Workflow v3.0

-  Pointer
-  CallExternalMethod
-  Code
-  Compensate
-  CompensatableSequence
-  ConditionedActivityGroup
-  Delay
-  EventDriven
-  EventHandlingScope
-  FaultHandler
-  HandleExternalEvent
-  IfElse
-  InvokeWebService
-  InvokeWorkflow
-  Parallel
-  Policy
-  Replicator
-  SetState
-  Sequence
-  State
-  StateInitialization
-  StateFinalization
-  Suspend
-  SynchronizationScope
-  Terminate
-  Throw
-  TransactionScope
-  CompensatableTransactionScope
-  WebServiceInput
-  WebServiceOutput
-  WebServiceFault
-  While

Windows Workflow v3.5

-  Pointer
-  ReceiveActivity
-  SendActivity

Activity: alles in einem Workflow ist eine Activity, auch der Workflow selber, der einen bestimmten Typ einer Activity darstellt.

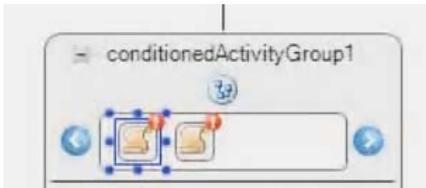
CallExternalMethod: Damit lassen sich Methoden außerhalb eines Workflows aufrufen. InterfaceType und MethodName müssen dabei definiert werden. InterfaceType gibt an, welches Runtime Service genutzt werden soll, wenn einen Activity ausgeführt wird. MethodName gibt an, welche Methode des angegebenen Interface aufgerufen werden soll.

Code: Hiermit lässt sich eigener .net Code im Workflow ausführen. Der Sourcecode ist getrennt vom Workflow.

Compensate: Ist eine Error Activity und kann zu einer Exception Handler Activity hinzugefügt werden. Kann genutzt werden um ein Rollback der Änderungen im Falle eines Fehlers durchzuführen.

Compensatable Sequence: Kann für einen bestimmten Typ eines Fehlers verwendet werden. Funktioniert ähnlich wie mit Datenbank Transaktionen. Der Unterschied ist, dass die Datenbank solange sperrt bis alle Updates fertig sind, also für eine kurze Zeit. Eine Compensate Action kann bei Transaktionen verwendet werden, die lange andauern, um die original Transaktion rückgängig zu machen.

Conditioned Activity Group: Dies ist eine bedingte Aktivität, die andere Aktivitäten, basierend auf Bedingungen, die für die Gruppe gelten, oder die Aktivitäten die der Gruppe zugeordnet, ausführt.



Delay: Bei der Verwendung einer DelayActivity wird der Wartezeitraum und die Methode angegeben, die im Fall der Zeitüberschreitung ausgeführt wird. Im Beispiel eine Genehmigungsworkflows kann man ein Delay verwenden. Wenn nun eine Person innerhalb des angegebenen Zeitraums nicht reagiert, wird der Alternativzweig im Workflow (z.B. andere Person kontaktieren) verwendet.

Event Driven: Diese Activity kann andere Activities beinhalten, die ausgeführt werden wenn dieses Event eintritt.

Event Handling Scope: kann eine Anzahl von Activities beinhalten. Während diese Activities ausgeführt werden, können die Events empfangen werden.

Fault Handler: Eine Error handling Activity, die wie ein catch block funktioniert.

Handle External Event: Die HandleExternalEventActivity wird in Verbindung mit der CallExternalMethodActivity für die In und Out Kommunikation mit einem lokalen Dienst verwendet. Sie können diese Aktivitäten direkt für die allgemeine Kommunikation verwenden. Oder Sie können Unterklassen von HandleExternalEventActivity und CallExternalMethodActivity erstellen um Aktivitäten, die streng an bestimmte Ereignisse und Methoden an ein Interface gebunden ist das ein ExternalDataExchangeAttribute besitzt, zu erzeugen.

If Else: Kann man sich als if else so wie in der Programmiersprache vorstellen. Über eine Condition wird die Bedingung festgelegt.

Invoke Web Service: kann ein Methode eines Webservice mittels Proxyklasse ausführen. Parameter können übergeben und empfangen werden.

Invoke Workflow: Ruft den zugeordneten Workflow auf Parallel: Im Workflow werden Activities als single thread ausgeführt. Es ist also keine wirkliche parallele Ausführung. Die Workflow runtime verarbeitet die Activity Queue pro Workflow Instanz im Prinzip von FIFO (first in, first out) ab.

Policy: Dies stellt eine Auflistung von Regeln dar. Eine Regel hat eine Bedingung und Action(en), die durchgeführt werden, wenn die Bedingung erfüllt ist. Das erlaubt einen regelbasierten Workflow, anstatt einem IfElse basierendem Workflow.

Replicator: Ist auch eine Bedingungsaktivität. Wie wenn man eine "For Each" Anweisung verwendet. Zur Laufzeit werden eine Anzahl von Instanzen einer Aktivität erstellt, die abgeschlossen werden müssen, bevor die Replikator Aktivität beendet wird.

Set State Activity: Dies ist ein flow activity, die für den Statuswechsel bei einem State Machine Workflow verwendet wird.

State Activity: Ist eine flow activity und stellt einen Status in einem State Machine Workflow dar.

State Initialization Activity: Diese Aktivität ist Bestandteil einer State Aktivität, die aus anderen Aktivitäten besteht, die ausgeführt werden, wenn die State Aktivität initialisiert wird.

Sequence: Ist eine zusammengesetzte Aktivität, die mehrere sequenzielle Aktivitäten beinhaltet. Es bietet eine einfache Möglichkeit, diese Tätigkeiten zu verknüpfen, die damit in Folge ausgeführt werden.

Suspend: Ist eine flow-Aktivität und kann den Ablauf eines Workflows für ein Einschreiten pausieren, wenn eine Fehlerbedingung (error condition) eintritt.

Synchronization Scope: Synchronisiert Workflow Aktivitäten

Terminate: Diese Aktivität beendet die Ausführung des Workflows sofort, wenn ein Fehler auftritt. Sie protokolliert auch den Fehler.

Throw: Mit dieser Aktivität können Sie eine Exception auslösen.

Transaction Scope: Diese Aktivität bietet Transaktionsunterstützung. In dieser Aktivität können sich Aktivitäten befinden, die eine Transaktion bilden.

Compensatable Transaction Scope: Es gibt zwei Aktivitäten die dieses Interface implementieren: CompensatableTransactionScopeActivity und CompensatableSequenceActivity.

WebService Input, Output, Fault: WF Webservice Integration

While: Dies ist eine bedingte Aktivität und ist wie eine "while" Bedingung in der Anwendung. Es wird eine andere Aktivität ausgeführt, bis eine Bedingung erfüllt ist. Die Bedingung kann ein Code oder einfach eine Regel basierende Bedingung sein.

Condition: es gibt die CodeCondition und die RuleConditionReference

CodeCondition: diese führt eine Methode der eigene code behind Klasse aus. Diese gibt true oder false zurück. Mit Code Condition sind Entwickler vertraut, hiermit lassen sich komplexe Bedingungen einfacher als in einer RuleConditionReference abbilden. Das Ergebnis einer CodeCondition wird als Result Property der ConditionalEventArgs zurückgeliefert.

RuleConditionReference: diese kann extern als XML gespeichert werden, die Extension ist .rules. Das hat den Vorteil, dass dies extern durch andere Tools geändert werden kann, auch zur Laufzeit.

Parameterübergabe in Workflows: Bei einer Activity gibt es normalerweise einen Zusammenhang zu den Bewegungsdaten, also zum Beispiel zur Customer Id. Zum Unterschied zum Aufruf von Klassen in .net, wo Parameter bei Methoden übergeben werden, gibt es bei der Parameterübergabe in Workflows ein Dictionary von Name/Values.

1.1.3 Für welchen Workflow soll ich mich entscheiden?

Ein State Workflow wartet auf externe Events bevor er in einen anderen Step springt.

Ein Sequential Workflow ist ein zusammenhängender Fluss von Activities. Activities in einem Sequential Workflow warten nicht auf externe Instanzen um dann in den nächsten Step zu springen.

Man kann jeden State Workflow auch als Sequential Workflow abbilden. Früher oder später wird dieser aber unüberschaubar werden.

2. Workflow Speicherformat

XOML (Extensible Object Markup Language) ist eine auf XML basierende Syntax. Ein Workflow kann als XOML gespeichert werden.

2.1 SqlPersistenceService

Per default laufen alle Workflows im Memory, solange auch die Applikation läuft, die den Workflow hostet. Zum Persistieren jeder einzelner Instanz des laufenden Workflows kann das mitgelieferte SqlPersistenceService verwendet werden. Damit wird der Zwischenstand der Workflows gespeichert, die Host Applikation kann jederzeit beendet und neu gestartet werden, außerdem können Workflows aus dem Speicher entladen werden wenn sie idle gehen. (Spart Memory, Performance)

Der Workflow kann damit am Filesystem oder in der Datenbank (SQL Server, SQL Express, ...) gespeichert werden.

```
// Add the SqlWorkflowPersistenceService
WorkflowPersistenceService persistenceService =
new SqlWorkflowPersistenceService (
    "Initial Catalog=SqlPersistenceService;" +
    "Data Source=localhost;" +
    "Integrated Security=SSPI;",
    true,
    TimeSpan.FromHours(1.0),
    TimeSpan.FromSeconds(5.0));
workflowRuntime.AddService(persistenceService);
```

Parameter

ConnectionString: Verbindung zur bestehenden Persist DB

UnloadOnIdle: muss auf true stehen, sonst wird nichts persistiert. Gibt an, dass der Workflow entladen wird, wenn er idle ist. Default Wert false.

LoadingInterval: Zeitintervall, in der die Persist DB gepollt wird, um die persist records zu schreiben.

DB Installationsscript:

c:\WINDOWS\Microsoft.NET\Framework\v3.0\Windows Workflow Foundation\SQL\EN\

Typ: SqlPersistanzService DB und Tracking DB können einzeln und getrennt verwendet werden. Wenn beides verwendet wird, empfiehlt sich aber beides in eine DB (SQL Server 2000, SQL Server 2005, oder SQL Server 2005 Express Edition) zu geben.

Microsoft Distributed Transaction Coordinator Service muss gestartet werden. Bei Verwendung eines Remote Servers muss Port 135 für Microsoft Distributed Transaction Center (MSDTC) offen sein.

2.2 Tracking

Da Workflows im Hintergrund laufen und die Workflow an sich auch komplexe Ausmaße annehmen werden, können diese mittels Tracking visualisiert werden. Bei der Instanziierung des Workflows wird das SqlTracking Service angegeben. Somit wird jede Instanz der Workflows getrackt.

Das Tracking ist für laufende als auch für beendete Workflows möglich, d.h. ich kann mir während des Ausführens als auch im Nachhinein den durchlaufenen Workflow ansehen. Die Darstellung erfolgt graphisch, der durchlaufene Workflow wird abgehackt dargestellt, der aktuellen Step mit einem grünen Dreieck.

DB Installationsscript liegt im Installationsverzeichnis:

c:\WINDOWS\Microsoft.NET\Framework\v3.0\Windows Workflow Foundation\SQL\EN\

3. Praxis

3.1 Workflow Hosting

Beispielanwendungen

- Automatisierungen von Abläufen
- Dokumenten Management
- Ticket Systeme
- Seitennavigation
- Komplexe Business Logic
- Häufige Änderungen an der Business Logic

Das Hosting kann in einem Service, über Webservice, asp.net, ... erfolgen.

In diesem Beispiel in einem Windows Service:

3.2 Workflow Tracking

Das Tool wird in den Samples mitgeliefert.

Workflow Monitor Sample

<http://msdn.microsoft.com/en-us/library/ms741723.aspx>

Um dies nutzen zu können, sind folgende Schritte notwendig:

- SQL Tracking DB muss angelegt worden sein.
- Eigener Workflow ist design und compiliert, Workflow wurde gestartet
- Erzeugt Workflow Assembly entweder ins Bin Verzeichnis vom Workflow Monitor kopieren oder die Assembly in den GAC geben (gacutil -i demo.dll)
- Workflow Tracking starten, Tracking DB angeben

3.3 Praxis Beispiel

In jedem organisieren Unternehmen gibt es wiederkehrende Abläufe, die man in einem Workflow Prozess abbilden kann. Dies betrifft einfache und komplizierte Vorgänge. Alleine durch die Analyse innerhalb einer Firma erge-

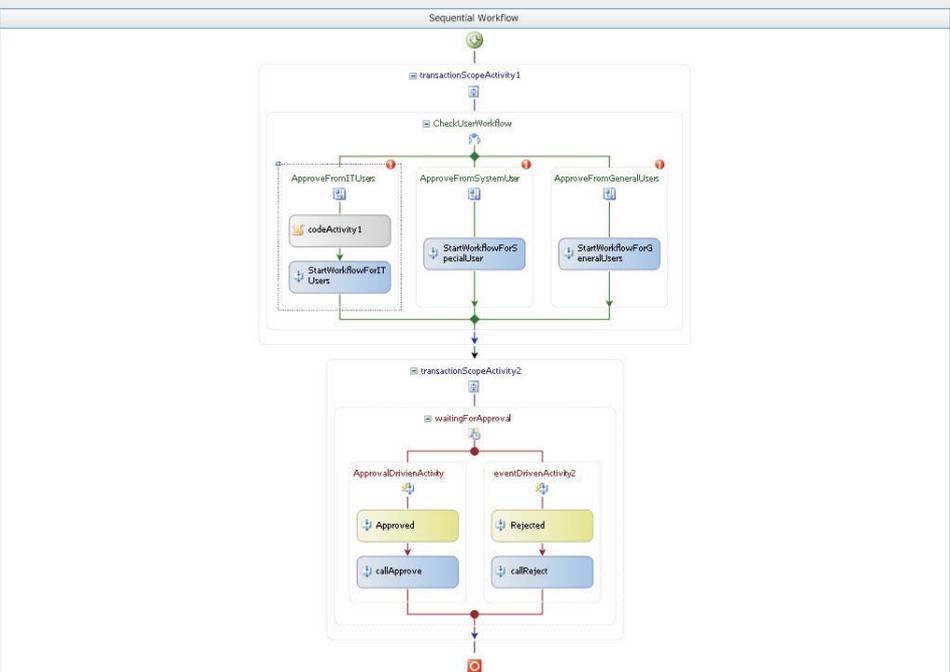
```
#region fields
WorkflowRuntime wfruntime = new WorkflowRuntime();
static AutoResetEvent _waitHandle = new AutoResetEvent(false);
private System.Collections.ObjectModel.ReadOnlyCollection<WorkflowInstance>
_loadedWorkflows;
#endregion
protected override void OnStart(string[] args)
{ ThreadPool.QueueUserWorkItem(new WaitCallback(RunWorkflow), args);
_wfruntime.WorkflowCompleted +=
new EventHandler<WorkflowCompletedEventArgs>(OnWorkflowCompleted);
}
private void RunWorkflow(object obj) {
try
{
// Load WF from SqlWorkflowPersistenceService // Create the WorkflowRuntime
using (WorkflowRuntime workflowRuntime = new WorkflowRuntime())
{
// Add the SqlWorkflowPersistenceService service
WorkflowPersistenceService persistenceService =
new SqlWorkflowPersistenceService (
"Initial Catalog=SqlPersistenceService;" +
"Data Source=localhost;Integrated Security=SSPI;",
true, TimeSpan.FromHours(1.0), TimeSpan.FromSeconds(5.0));
workflowRuntime.AddService(persistenceService);
// Tracking
workflowRuntime.AddService (
new SqlTrackingService("Initial Catalog=Tracking;Data Source=localhost;" +
"Integrated Security=SSPI;"));
// Set up the WorkflowRuntime event handlers
workflowRuntime.WorkflowCompleted += OnWorkflowCompleted;
workflowRuntime.WorkflowIdled += OnWorkflowIdled;
workflowRuntime.WorkflowPersisted += OnWorkflowPersisted;
workflowRuntime.WorkflowUnloaded += OnWorkflowUnloaded;
workflowRuntime.WorkflowLoaded += OnWorkflowLoaded;
workflowRuntime.WorkflowTerminated += OnWorkflowTerminated;
workflowRuntime.WorkflowAborted += OnWorkflowAborted;
_loadedWorkflows = workflowRuntime.GetLoadedWorkflows();
Console.WriteLine("Workflows loaded: ", _loadedWorkflows.Count);
}
}
}
catch (Exception ex)
{
EventLog.WriteEntry("Hosting WF in an Windows Service", e.ToString(),
EventLogEntryType.Information, 1000);
}
}
```

ben sich oft Schwachstellen, Lücken oder Missverständnisse die Arbeitsvorgänge undurchsichtig, unkontrollierbar und zeitaufwendig gestalten.

Es bietet sich an, dies als elektronischen Workflow abzubilden.

Anhand eines einfachen Beispiels möchte ich dies veranschaulichen.

Es handelt sich dabei um einen Genehmigungsprozess, bei dem ein bereitgestelltes Dokument durch einen Gruppe oder einen Benutzer akzeptiert werden muss.

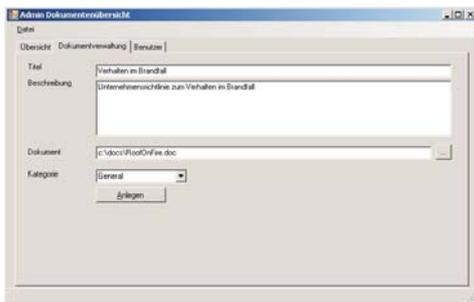


Ablauf

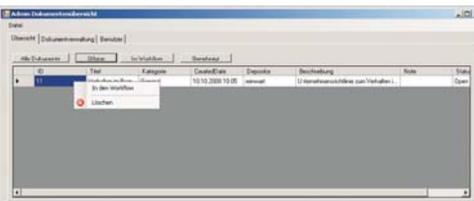
Eine leitende Person erstellt Dokumente und stellt diese in den Workflow. In dem Workflow sind drei Abteilungen definiert: IT User, System User, und allgemeine Benutzer.

Im Workflow sind diese Gruppen durch eine IfElseActivity aufgeteilt, daher kann jede Gruppe eine besondere Business Logic (BL) aufweisen. Eine solche BL lässt sich in einer CodeActivity unterbringen. In einer solchen CodeActivity lässt sich eigener Code ausführen.

Die Nutzung des zuvor erstellten Workflows ist in diesem Fall eine simple .net Windows Forms Anwendung, ist aber vom Workflow völlig unabhängig.



Ein Dokument mit den Attributen Titel, Beschreibung und Kategorie wird angelegt.



In der Übersichtsliste erscheint das Dokument und kann in den Workflow übernommen werden.

Technischer Teil

In der Applikation wurde das Persistence Service und das TrackingService der Windows Workflow Foundation eingebunden.

Persistence Service

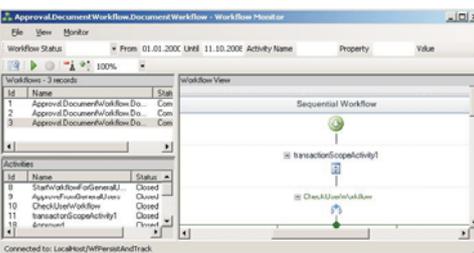
Jede Workflow Änderung eines Status wird in die Datenbank geschrieben.

TrackingService

Bietet die Möglichkeit, bei Instanz eines Workflow den aktuellen Status zu erfassen

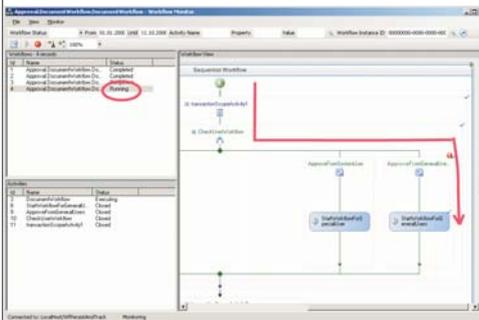
Tracking Monitor

Zeigt den aktuellen Workflow an

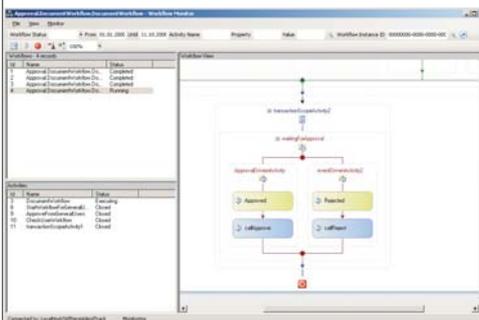


Damit lässt sich jeder aktuelle Step jedes einzelnen Workflow Überwachen und verständlich darstellen.

Das Dokument wurde in der Workflow gestellt, der Workflow läuft. (workflow running)

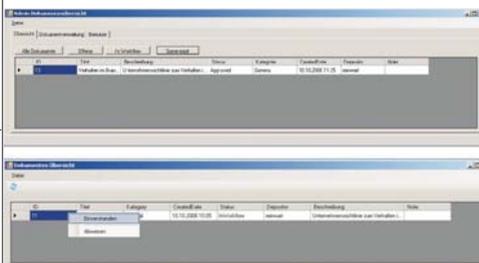


Der Workflow hat nun den Weg für die Kategorie General User genommen und wartet nun auf die Bestätigung durch den zugewiesenen User.



Der User hat nun die Möglichkeit, das Dokument, das nun diesen Workflow durchläuft zu bestätigen oder abzuweisen.

Für den User gibt es eine andere Applikation. Zu Beginn muss er sich authentifizieren, nach der Anmeldung sieht er die für ihn bestimmten Dokumente, für die eine Zustimmung notwendig ist.



Er liest sich den Inhalt des Dokuments durch und ist damit einverstanden.

Die Anwendung aus der Sicht des Administrators: Das Dokument wurde genehmigt und ist als „Approved“ in der Liste geführt.



Der Administrator hat laufend die Übersicht, welche User die Dokumente bereits akzeptiert haben. Bei einer Überschreitung eines Zeitraumes könnte man in diesem Demo Workflow ohne viel Aufwand auch um eine Eskalationsstufe (z.B. automatische generiertes Email) erweitern.

4. Einschränkungen mit WF 3.0

Performance: Microsoft selbst dokumentiert dies in dem Dokument "Performance Characteristics of Windows Workflow Foundation".

(<http://msdn.microsoft.com/en-us/library/aa973808.aspx>)

Versionierung von Workflows

Ein Workflow kann jederzeit geändert werden, aber die bestehenden Workflows kann man nicht mit der neuen Definition weiterlaufen lassen. Wenn Sie bei einem Prozess plötzlich am Ende noch einen Schritt brauchen, können Sie diesen also nicht bei Workflows anwenden, die bereits gestartet sind (auch wenn diese langlebig sind).

Keine Kompatibilität zu SSIS-Workflows

Microsoft SQL Server Integration Services (SSIS) bietet eine ähnliche Umgebung wie WF, ist jedoch in keiner Weise kompatibel zu WF. Das Entwicklungsteam von SSIS hat sich aktiv gegen die Verwendung von WF als Basis ausgesprochen (einerseits war SSIS vor WF auf dem Markt, andererseits ist SSIS mehr auf Massendatenverarbeitung fokussiert, dafür wäre WF zu langsam).

Designer Control Hosting

Eigener Code zusätzlich zum WF Control notwendig, für einen WF Beginner nicht einfach. Mit dem Designer Control der Version 4.0 wird es einfacher sein.

5. Zukünftige Version WF 4.0

Auf der PDC 2008 wurde von Kenny Wolf, einem Microsoft Architect von WF und WCF, die Zukunft von WF in der Version 4.0 präsentiert. Workflow 4.0 soll zusammen mit dem .net Framework 4.0 und Visual Studio 2010 auf den Markt kommen.

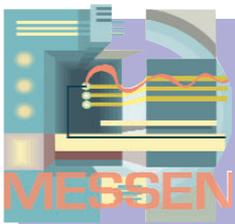
In der Vorführung erkennt man die neue einfachere Oberfläche für das Design eines Workflow Prozesses. Aufgrund der Erkenntnis, das die WF Versionen 3.0 und 3.5 Schwächen im Bereich Funktionsumfang, Bedienung, Komplexität und Leistung aufweisen und die Kundenwünsche damit nicht immer befriedigt werden können, wurde WF von Grund auf neu geschrieben. Das Konzept der Aktivitäten im Workflow und die Persistierung und die Möglichkeit von Monitoring bleibt. Es soll eine 10 bis 100-fache Leistungssteigerung geben, ebenso eine volle Kontrolle über die Persistierung. WF 4.0 wird rein deklarativ sein.

WF 4.0 vs. WF 3.0

- Activity
 - Authoring is simpler and takes much less code
 - Fully declarative workflows and activities
 - Alignment across Expressions, Rules, and Activities
 - Seamless Composition Across Flow Styles
- Runtime
 - 10-100X Performance Improvements
 - Full control over persistence
 - Flow-in Transactions
 - Partial Trust Support
 - Integrates with WCF, WPF, ASP.NET
- Tools
 - Designer Performance and Usability
 - Rehosting Improvements
 - Unified Debugging Experience

Preparing for WF 4.0

- 3.0/3.5 workflows continue to work (on the 3.0 WF runtime)
- Can use 3.0 Activities within a 4.0 Workflow
- Guidance on how to prepare 3.0/3.5 code



Messtechnik für den Profi:

- ▶ Netzqualitätsanalyser
- ▶ Transientenrekorder
- ▶ Energieanalyser
- ▶ Schutzmaßnahmenprüfgeräte
- ▶ Schreiber
- ▶ Einbauanalyser
- ▶ Stromzangen



Mobile Computer für alle Fälle:

- ▶ Notebooks
- ▶ Industrie-Notebooks
- ▶ Sonderlösungen



Panasonic Toughbook CP-28



Mikrocontroller, Entwicklungstools und Baugruppen:

- ▶ Compiler
- ▶ Debugger
- ▶ Betriebssysteme
- ▶ Starterkits
- ▶ Minimodule



Familien:

C166 & ST10
8051, C500, C800
M16C, 77k, TLCS900
TriCore, Carmel
MIPS, DSP56xxx,
68xxx, PowerPC



Wir entlasten Sie mit folgenden Dienstleistungen:

- ▶ Messen und Protokollieren der Netzqualität
- ▶ Auffinden von Netzstörungen
- ▶ Schulungen zum Thema Netzqualität
- ▶ Produktschulungen



MTM-Systeme
Ing. Gerhard Muttenthaler
Hadrawagasse 36
1220 Wien

fon +43 1 2032814
fax +43 1 2021303
mail office@mtm.at
web www.mtm.at

Produktinformationen und
Nützliches unter:
www.mtm.at

NEUE INFINEON STARTERKITFAMILIE

Zur neuen XC16xFamilie sind nun auch die Starterkit's erhältlich.

Die Starterkits für XC161CI, XC164CS und XC167CI werden "ready to use" incl. Software geliefert

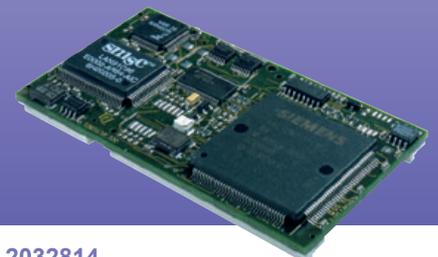
Der optimale Start für Ihre 16 Bit Mikrocontrolleranwendung!



16-Bit Applikationen mit Ethernet-Anbindung

Mit dem TQM167UE bietet TQ-Components ein voll lauffähiges embedded 16-Bit Microcontroller System mit Ethernet-Schnittstelle auf einer Fläche von nur 80 x 44 mm² an. Als Systemkern dient der Infineon SAB-C167CR Microcontroller. Der Speicherausbau von 1 MB SRAM sowie 1 MB FLASH erlaubt auch größere Programme laufen zu lassen. Die RS232 Schnittstelle und 4-fach UART stellen die komplette Verbindung zur Außenwelt des Moduls her. Das TQM167UE bietet zusätzlich den Super I/O-Contoller FDC37C669, z.B. zur Anbindung eines Floppy-Laufwerks.

Besonders einfach ist das Programmhandling. Über die mitgelieferte Download-SW können eigene Programme einfach und komfortabel auf das Modul geladen werden. Um den Einstieg in die Modulwelt zu vereinfachen, liefert TQ-Components das Starterkit zu oben beschriebenen Modul, die komplette "Plug and Play" Lösung unter der Bezeichnung STK167UE.



Info bei MTM-Systeme unter www.mtm.at oder +43 1 2032814