

ObjectVision: Ein Formular, mit dem man rechnen kann ...

Matthias Kraft

Programmieren ohne große EDV-Vorkenntnisse – das ist der hohe Anspruch, den ObjectVision, das neueste Produkt aus dem Hause Borland, an sich stellt. Ein weiterer wesentlicher Anspruch, der bereits dem Pakungsumschlag zu entnehmen ist, ist die Vielfalt der Kommunikationsmöglichkeiten, die das Programm verspricht. Hierdurch werden zwei Modeworte aufgegriffen, die derzeit einen hohen Stellenwert in der EDV-Landschaft haben: das Konzept der visuellen Programmierung und der Integration. Der folgende Artikel beschreibt das Konzept von ObjectVision und gibt einen Überblick über die einzelnen Funktionselemente. Eine detaillierte Beschreibung eines Projektes bleibt einem weiteren Beitrag vorbehalten.

Das ObjectVision-Konzept

Grundidee von ObjectVision ist das aktive Formular. Ausgehend von der Prämisse, daß sich ein großer Teil der Bürotätigkeit durch das Ausfüllen von Formularen erledigen läßt, und der Beobachtung, daß dies tatsächlich auch so praktiziert wird, soll ObjectVision durch die Belegung des Formulars zu einer spürbaren Arbeitserleichterung führen (Abb. 1).

Abb. 1:
Formular zur
vorläufigen Voll-
streckbarkeit

„Mitdenken“

Dabei ist der Computer nicht nur als ein alternatives Instrument zum Ausfüllen der einzelnen Felder gedacht, sondern das Formular „denkt“ in vielfältiger Weise mit:

- Formularfelder, deren Werte sich aus vorheriger Eingabe berechnen lassen, werden von ObjectVision berechnet.
- Daten, die bereits an anderer Stelle im System etwa in einer Datenbank oder einem Kalkulationsblatt vorliegen, können von dort übernommen werden.
- Bei der Dateneingabe zeigt ObjectVision automatisch an, welche Eingabe jeweils benötigt wird (automatisches „Prompt“). Insofern verhält es sich ähnlich wie ein einfaches Exper-

Matthias Kraft
ist Assistent am
Lehrstuhl für
Rechtswissenschaften
der Universität des Saarlandes
und Autor
des Programms
SOPHOS.

The screenshot shows a complex legal form with multiple overlapping windows. The main window is titled 'ObjectVision - ZPO-VV.OVD'. It has a menu bar with 'Datei', 'Bearbeiten', 'Formular', 'Feld', 'Ansicht', 'Arbeitsebene', and 'Hilfe'. The form is divided into several sections:

- Beteiligte (Hauptformular):** Contains checkboxes for 'Kläger anwaltlich vertreten' and 'Beklagter anwaltlich vertreten', and fields for 'Kosten des Klägers' and 'Kosten des Beklagten'.
- Verfahrensdaten (Hauptformular):** Contains a date field '202/91', a 'Streitwert' field with 'DM3.000,00', and 'Gemeinkosten' 'DM288,00'.
- Urteil gem § 708 ZPO:** Contains a 'Kostenanteil des Klägers' field with 'DM2.000,00', 'Kosten pro Anwalt' 'DM525,00', and 'Kostenanteil des Beklagten' '66,67%'. It also has a 'Tenor VV (Vollständig)' section with 'Kostenanteil Kl.' '33' and 'Kostenanteil Bkl.' '67'.
- Other sections:** 'Sache Mustermann v. UH', 'Obsiegen des Klägers', 'Sicherheitsleistung Kl.', 'Abwendungsbezugnis Bkl.', and 'Abwendungsbezugnis Kl.'.

tensystem, da es immer unter Berücksichtigung der vorhandenen Daten entscheidet, welche Information als nächste relevant ist.

Datenausgabe

Neben dieser sehr effizienten Art, Daten zu erfassen oder zu berechnen, stehen vielfältige Möglichkeiten der Datenausgabe zur Verfügung.

- Das Formular kann in hoher Qualität gedruckt werden.
- Daten können an andere Programme (Datenbank, Tabellenkalkulation oder Textverarbeitung) übermittelt werden.
- ObjectVision kann selbständig die Textverarbeitung ansteuern und durch Zusammenstellung von Textbausteinen einen Fließtext erzeugen.

Entwickler und Anwender

ObjectVision setzt sich im Prinzip aus zwei Modulen zusammen: In der ObjectVision-Terminologie wird von (Arbeits-)Ebenen gesprochen.

Das Entwicklermodul steht nur in der Vollversion zur Verfügung. Hier werden die sogenannten Anwendungen entworfen. Sie bestehen aus einem oder mehreren Formularen, die als eine Datei abgespeichert werden. Das Entwicklermodul splittet sich in drei Arbeitsebenen. Zwischen diesen und der Anwenderebene kann beliebig über ein Menü hin und her geschaltet werden. So können bei der Entwicklung immer sofort die Auswirkungen einer Änderung getestet werden. Im Anwendermodul wird das fertige Formular vom Endanwender eingesetzt. Es werden

also unter Ausnutzung der oben dargelegten Funktionalität Daten eingegeben und je nach Formular ausgewertet.

Dieses Modul ist als sogenannte Run-Time-Version auch einzeln verfügbar. Die Run-Time-Version dient der Weitergabe von Entwicklungen, ohne daß sich der Empfänger eine eigene Vollversion zulegen muß. So können also größere organisatorische Einheiten (etwa ein Gericht oder eine Kanzlei) mit einer einzigen Vollversion für alle Arbeitsstationen kostengünstig Anwendungen entwerfen.

Formulargestaltung

Wesentliches Element des Entwicklermoduls ist die Formular-Arbeitsebene. Hier wird die Oberfläche der Anwendung, d. h. ihrer einzelnen Formulare gestaltet.

Hierzu werden die einzelnen Eingabefelder der Formulare zunächst mit einem in der Anwendung (also für alle Formulare) singulären Namen benannt und dann mit Hilfe der Maus auf einem Formular plaziert. Für jedes Feld können folgende Attribute festgelegt werden:

- Die graphische Gestaltung, z. B. Umrandung, Hintergrundfarbe oder Schriftart.
- Datenformat (z. B. Text, Währung, Prozent, Festkomma oder wahr/falsch).
- Mit dem Datenformat (unglücklicherweise) verknüpft ist die Eingabetechnik, wie z. B. Texteingabe, Listen oder Boxen zum Ankreuzen.

Größe und Lage der Felder können mit der Maus frei variiert werden. Ein Feld kann in einer Anwendung mehrmals auf verschiedenen Formularen verwendet werden. Es wird dabei durch seinen Namen identifiziert. Die einzelnen Attribute können hierbei variieren. Neben den Datenfeldern sind auch erläuternde Text- oder Graphikfelder auf dem Formular möglich. Hierzu gehört die Möglichkeit, Bitmuster-Graphiken, z. B. aus Paint-

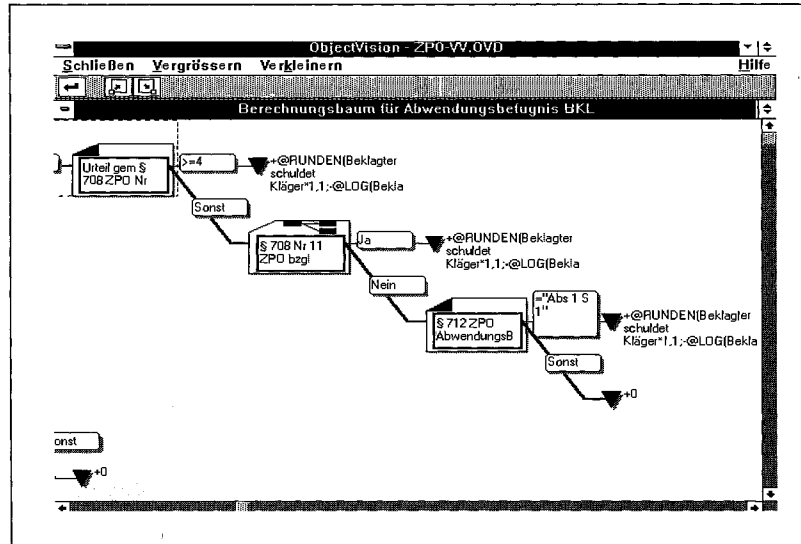


Abb. 2:
Berechnungsbaum
für die
Abwendungs-
befugnis des
Beklagten

Brush, über OLE in ein Formular einzubinden. Des weiteren können beliebige Hilfstexte hinter jedem Feld angelegt werden, die der Benutzer mit <F1> aufrufen kann.

Entwicklung der Funktionalität

Nach der graphischen Gestaltung der Formulare beginnt die eigentliche Arbeit:

Die Funktionalität des Formulars muß entwickelt werden, dem Formular wird also „Leben eingehaucht“ (Abb. 2). Dieser Arbeitsgang vollzieht sich im Prinzip wieder in mehreren Schritten, die jedoch nicht zwingend in der gleichen Reihenfolge ablaufen.

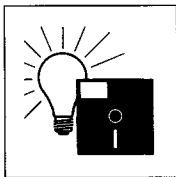
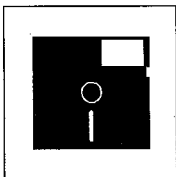
1. Eingabe der Formeln, mit deren Hilfe sich bestimmte Felder berechnen lassen.
2. Herstellung der Verbindung zu Programmen, aus denen Daten übernommen werden.
3. Entwicklung einer Routine zur Ausgabe der Ergebnisse des Formulars.

Zentrales Werkzeug für den ersten Schritt der Entwicklung von Algorithmen ist der sogenannte Berechnungsbaum. Jedes Feld kann einen derartigen Entscheidungsbaum im Hintergrund haben. Er führt über mehrere Äste je nach vorangegangenen Eingaben des Benutzers zu der Formel,

mit der der Wert des Feldes berechnet werden kann. Auf diese Weise können also unterschiedliche Formeln zur Berechnung eines Feldes führen. Der Baum wird im wesentlichen mit der Maus entwickelt. Über eine Schalterleiste werden die Knoten und Blätter in den Baum eingefügt. Lediglich die Formeln an den Blättern müssen noch eingetippt werden. Dabei steht in einer Liste eine Formelbibliothek zur Verfügung, die etwa der einer normalen Tabellenkalkulation entspricht. Über DLLs kann diese Formelsammlung nahezu beliebig erweitert werden. Diese Arbeitsweise ermöglicht es im Prinzip, frei von Fragen der Programmierung Entscheidungsabläufe und Algorithmen zu entwerfen. Das einzige Problem – und das ist kein geringes – ist die Analyse der zugrundeliegenden Normen.

„Links“ zu Datenquellen

Der Eingabe von Berechnungsformeln folgt die Anlage von Links zu anderen Datenquellen. Hierzu besitzt ObjectVision die Möglichkeit, dBase-, Paradox- und Btrieve-Daten direkt einzulesen, anzulegen und zu aktualisieren. Zudem ist der Rückgriff auf Daten aller DDE-fähigen Programme mittels Hotlink möglich. Auf diese Weise kann dem Anwender die Eingabe bereits



erfaßter Daten erspart werden, und es können Differenzen zwischen unterschiedlichen Anwendungen vermieden werden.

Ergebnis-Ausgabe

Letzter Schritt ist eine Implementierung der Ergebnis-Ausgabe. Hierzu stehen unter anderem folgende Wege zur Verfügung:

- Ausdruck des Formulars,
- Übertragen der Daten in andere Programme oder in eine Datenbank,
- Erstellung eines Fließtextes in der Textverarbeitung.

Ein Ausdruck des Formulars bedarf keiner weiteren Implementierung, da er via Menü „Datei | Drucken“ erfolgt. Wichtig ist hier eine graphisch auf den Drucker und das Papier abgestimmte Gestaltung des zu druckenden Formulars. Die Ausgabe der Daten in eines der von ObjectVision unterstützten Da-

tenbankformate ist ebenfalls recht einfach. Die Übertragung der berechneten Daten via DDE erfordert dagegen eine intensivere Beschäftigung mit den Möglichkeiten des dynamischen Datenaustauschs unter Windows. Hat man sich einmal der Mühe unterzogen, so kann man ObjectVision dazu einsetzen, Daten in nahezu jedes DDE-Programm zu exportieren und selbst dort Makros auszulösen. Dies ermöglicht es beispielsweise, eine Textverarbeitung dazu aufzufordern, bestimmte Textbausteine in ein Dokument einzufügen und bestimmte Daten dort einzutragen. Mit Hilfe sogenannter Ereignisbäume (das sind Spiegelbilder der Entscheidungsbäume, die jedoch auch nur auf Ereignisse wie Mausclicks, Datenänderung etc. reagieren) können nicht nur die gefundenen Ergebnisse, sondern auch die zugrundeliegenden Entscheidungsabläufe in ein Textverarbeitungsdokument ausgegeben werden. Hierdurch werden we-

sentlich präzisere Formulierungen von Begründungen ermöglicht, als sie bisher in der Praxis beispielsweise durch den Verweis auf eine Paragraphenkette üblich sind.

Zusammenfassung

Mit ObjectVision lassen sich verhältnismäßig leicht individuelle und Standardanwendungen für den juristischen Arbeitsplatz realisieren. Es kann dabei auch gut als Bindeglied zwischen mehreren Anwendungen dienen. Für die Praxis interessant ist sicherlich der verhältnismäßig moderate Preis von unter 600,- DM. Hierin ist eine Run-Time-Version enthalten, die an andere Benutzer verteilt werden darf. Auf diese Weise ist in einem überschaubaren Kreis von Benutzern (etwa einem Gericht oder einer Kanzlei) lediglich ein Exemplar der Vollversion nötig.

jurpc.zip - jurpc.zip - jurpc.zip - jurpc.zip - jurpc.zip - jurpc.zip - jurpc.zip - jurpc.zip

TeleTrusT: Digitale Risiken

Der „Verein zur Förderung der Vertrauenswürdigkeit von Informations- und Kommunikationstechniken“ (TeleTrusT Deutschland e. V; Geschäftsstelle: c/o eec, Frankfurter Str. 84, W-6368 Bad Vilbel 1) offeriert zum Preis von 20,- DM (zzgl. Porto und Verpackung) eine Sammelmappe mit Veröffentlichungen aus der Arbeit der TeleTrusT-Arbeitsgruppen.

Zum Jahresbericht kommen Blätter mit folgenden Themen hinzu:

- Arbeitsstand und Entwicklungsperspektiven in der Arbeitsgruppe Juristische Aspekte, *Ulrich Seidel*, GMD, 3 S.;
- Aufgabenstellung der Gruppe Sicherheitsarchitekturen, *Dietrich Kruse*, SNI, 2 S.;

- Digitale Unterschrift: Aspekte der Verbindlichkeit, *Siegfried Herda*, GMD, 13 S.;
- Digitale Unterschrift: Bedrohungsanalyse, *Dietrich Zeller*, SEL, 13 S.;
- Digitale Unterschrift: Technische Referenzszenarien/Risikoanalyse, *Klaus Müller*, SNI, 18 S.;
- Security Information Objects - The TeleTrusT/TeleSec Certificate, *Wolfgang Schneider*, GMD, 12 S.;
- Object Identifiers - Procedure of Registration, *Klaus Truöl*, GMD, 4 S.;
- Object Identifiers - Register, *Klaus Truöl*, GMD, 5 S.;
- Security Information Objects - The TeleTrust-Sedu (Security Enhanced Data Unit), *Wolfgang Schäfer*, DATEV e. G., 14 S.;

- Übersicht über nationale und internationale Sicherheitsprojekte, *Hartmut Schmidt*, IBM, 17 S.;
- Computer-Sicherheit vor dem Hintergrund von Export-Restriktionen und deren Liberalisierungsmöglichkeiten, *Roderich Gorny*, SNI, 3 S.

An EDV-Recht interessierte Juristen finden in dieser Sammlung eine Fülle gedanklicher Anregungen, so etwa gleich zu Beginn eine Definition des elektronischen Dokuments von *Ulrich Seidel*: „Ein EDV-Dokument ist eine mit einem technischen Mittel verkörperte, jedoch nicht schriftlich niedergelegte Erklärung, die allgemein oder für Eingeweihte verständlich ist, den Aussteller erkennen läßt und die zum Beweis einer rechtlich erheblichen Tatsache bestimmt ist.“