

Visualisierungssysteme auf Basis von .Net 3.0

Mit Windows Vista hat Microsoft [1] das in die Jahre gekommene XP-Design grundlegend modernisiert – hardwareunterstützte Transparenzeffekte und Animationen auf dem Desktop sind nur die augenfälligsten Neuerungen. Die eigentlichen, für den industriellen Anwender bedeutsamen Veränderungen liegen jedoch unter der Oberfläche: Mit dem kürzlich vorgestellten .Net-Framework in der Version 3.0, der Ablösung der alten Windows-Programmierschnittstelle durch die Windows Presentation Foundation (WPF) und der Umstellung auf XAML als Beschreibungssprache für grafische Benutzeroberflächen hat Redmond die Windows-Programmierung auf eine neue Grundlage gestellt. Auch Anwender von Visualisierungssystemen können profitieren.

Peter Tanneberg

musste, kann das Aussehen eines Elements und die Verknüpfung mit den jeweiligen Prozessvariablen nun interaktiv verändert werden. Ob eine Prozessgröße nun als Balken oder Zeigerinstrument dargestellt werden soll, macht für die Visualisierung keinen Unterschied. Auch dynamische Anzeigen, die zuvor aufwendig programmiert werden mussten, sind mit der neuen Technologie mit wenigen Zeilen Text generiert (Bild 1).

Die entscheidende Neuerung, die die Windows Presentation Foundation mit sich bringt, ist XAML, ein erweiterter XML-Dialekt. XAML ist eine deskriptive Sprache, in der alle Bildschirmausgaben, egal ob 2D- oder 3D-Objekte, Bewegungen, Animationen und sämtliche anderen Effekte definiert werden. Aus diesen Klartextbeschreibungen bzw. einem binären Zwischencode wird dann zur Laufzeit beispielsweise ein Button, eine Dialogbox oder jedes andere GUI-Objekt erzeugt. Die Darstellung beruht durchgängig auf Vektorgrafiken, so dass auch unterschiedliche Bildschirmauflösungen und -größen kein Problem mehr darstellen. Auch beim Rotieren oder Zoomen sind daher keine unschönen Treppen oder Pixeleffekte zu sehen.

XAML und die Folgen

Die strikte Trennung von Programmlogik und Bedienoberfläche vereinfacht auch das Leben des Automatisierers, denn nun können sich Spezialisten wie Grafiker und Designer um das Aussehen und die Ergonomie der Visualisierungsober-

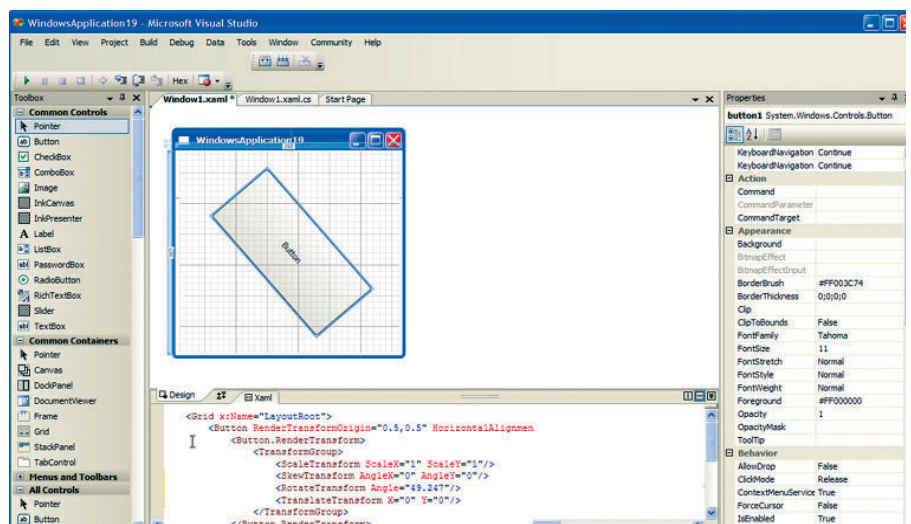


Bild 1. XAML-Beispiel: Änderungen – hier wird gerade ein Button definiert – erfolgen interaktiv bzw. direkt im Klartext

fläche kümmern und ihr Know-how in ein Projekt einbringen. Dazu werden Grafiken und Animationen, die zum Beispiel mit Hilfe von Microsofts Expression Interactive Designer erstellt wurden, als XAML-Dateien ausgegeben, die wiederum ohne weitere Verarbeitung oder Konvertierung direkt in das Visualisierungsprojekt importiert werden können. Änderungen und Varianten lassen sich durchspielen, ohne dass die darunterliegende Programmlogik angepasst werden müsste. Nebenbei erleichtert dies auch die Umsetzung von Vorgaben des Corporate Designs. Während man zuvor mit den zwar zahlreichen, aber eben doch „vorgefertigten“ Standardtemplates leben, oder sich seine eigenen programmieren

Visualisierungsapplikationen von Smart bis Professional

Seit über zwei Jahrzehnten entwickelt Inosoft [2] Visualisierungsapplikationen, zunächst für DOS, dann für Windows-Systeme, die auf bewährte Industriestandards für Software, Programmierung und Kommunikation setzen. Die aktuelle Version, VisiWinNet, basiert auf dem .net-Framework und ist in zwei Grundvarianten verfügbar. Die kleinere Version Smart ist für Projekte von geringer bis mittlerer Komplexität geeignet. Hervorzuheben ist die Möglichkeit, das System in OEM-Tools und -Umgebungen einzubetten, da sämtliche Teile als eigenständige .net-Komponenten realisiert sind, die sich leicht in andere .net-Softwareumgebun-

Dipl.-Ing. Peter Tanneberg ist Geschäftsführer der Inosoft GmbH in Hiddenhausen.

E-Mail:
peter.tanneberg@inosoft.com



gen integrieren lassen. Am anderen Ende des Leistungsspektrums steht die Professional-version, die z. B. für High-End-Maschinenvisualisierungen oder Scada-Applikationen eingesetzt werden kann. Die gesamte Palette der .net-Programmiersprachen kann genutzt werden, um den Funktionsumfang an die eigenen Bedürfnisse anzupassen. So ist es z. B. möglich, das System zur automatisierten Generierung von Visualisierungsoberflächen für Serienmaschinen einzusetzen. Man muss aber keinesfalls programmieren, sondern kann auch die mitgelieferten Templates nutzen, die die typische Automatisierungsaufgaben wie Nutzerverwaltung, Einheiten- und Sprachumschaltung, Rezeptverwaltung, Datenbankzugriffe usw. bereits abdecken. Für den Einsatz in FDA-regulierten Anwendungen bietet das System alle erforderlichen Funktionen zur Benutzer- und Passwortverwaltung und zur Implementierung eines Audit-Trails. Zur Produktpalette gehört außerdem mit VisiWinnet Web ein Webvisualisierungssystem für Intra- und Internetaapplikationen, das mit Hilfe der Ajax-Technologie im Browser das Look & Feel einer Desktopapplikation nachbildet. Das System ist eine reine Port-80-Anwendung und arbeitet daher problemlos mit Firewalls zusammen.

Visualisierung der dritten Art

Einen Vorgeschmack auf die neuen Möglichkeiten gab das von dem Textilmaschinenhersteller Monforts [3] und dem Bremer Institut für Integriertes Design [4] entwickelte Konzept zur Bedienung von Anlagen zur Textilveredlung. Die mit dem iF communication design award 2006 ausgezeichnete Applikation nutzt Vektorgrafiken zur Darstellung der Maschinen; über eine Lupe können einzelne Funktionsbereiche angesteuert und unterschiedlichen Detailabstufungen angewählt werden. Derartige Konzepte las-

sen sich mit Hilfe der XAML-Technik mit geringerem Aufwand als vorher realisieren (Bild 2).

Konsequenterweise ist die Umstellung der Visualisierungssoftware auf .net in der Version 3.0 in vollem Gange. Dank der universellen Verbreitung von Windows und der Vorteile wird sich das XAML-Format in absehbarer Zeit auf breiter Front durchsetzen und von allen gängigen Grafik- und Designtools unterstützt werden.

Fazit

Mit XAML und WPF steht eine einheitliche Technologie für alle Anforderungen, von der dezentralen Visualisierung bis zur Fernwartung, von der Desktop- bis zur Browserapplikation zur Verfügung. Die Umstellung auf XAML sorgt im Bereich der Visualisierung für mehr Flexibilität und Freiheiten bei der Gestaltung der Bedienoberflächen.

Literatur

- [1] Microsoft Deutschland GmbH:
www.microsoft.de
- [2] Inosoft GmbH: www.inosoft.com
- [3] A. Monforts Textilmaschinen GmbH & Co. KG:
www.monforts.de
- [4] Institut für integriertes Design:
www.iidbremen.de



Bild 2. Vektorbasiertes Interface für Monforts-Maschinen – Detailansicht nach Bedarf (rechts)