

MICROSOFT .NET

.NET-Smart Clients führen auf der Zielgerade

So sehen Sieger aus: Wer die Computer in der Zukunft vernetzt, war lange unentschieden. Mal hatten die Rich Clients die Nase vorn, mal die Thin Clients. Beide Lösungen haben Vor- und Nachteile. Nun ist es Microsoft gelungen, die Vorteile beider Technologien zu vereinen und deren Nachteile zu minimieren.

Es waren einfache Terminals, welche zu Beginn des Computerzeitalters und der vernetzten Rechnerwelten als Thin Client den Zugang zu den Ressourcen eines Großrechners ermöglichten. Doch dann kamen die PCs, und Rich Clients nutzten die nun auch lokal verfügbare Rechenleistung, um ihre Geschwindigkeit zu erhöhen. Erst mit dem Internet holten die flexiblen Thin Clients mit ihren offenen Kommunikationsplattformen wieder auf. Sie nutzten globale Standards für Transfer und Interpretation von Informationen und haben so neue Maßstäbe in der heterogenen Zusammenarbeit von IT-Systemen gesetzt.

Das Ende der DLL-Hell

Rich Clients sind in Bezug auf Bedienbarkeit, Anwenderkomfort und Funktionalität das Maß der Dinge. Dafür sind Installation und Wartung aufwändig, Unternehmen benötigen dafür oft sogar eine eigene Infrastruktur.

Anders stellen sich Browser dar. Als dynamische Thin Clients passen sie sich schnell an das Serverangebot an und können so mit wenig Aufwand immer aktuell gehalten werden. Das geht zwar auf Kosten des Komforts und der Handhabbarkeit, was ihre Beliebtheit beim Publikum jedoch nicht schmälert. Im Gegenteil, viele der IT-Verantwortlichen sehen heute in der Einführung von Browseranwendungen eine der effizientesten Methoden, um Betriebs- und Wartungskosten zu beherrschen.

Microsoft stellt mit dem .NET Framework eine neue Infrastruktur bereit, die exakt auf die Anforderungen der Anwender zugeschnitten ist, und hat damit im Endspurt um smarte Endgeräte das Tempo nochmals erhöht. Eine deutliche Verbesserung bietet die so genannte Side-by-Side Execution. Sie erlaubt, dass Visual Basic-Anwendungen gleiche Komponenten jetzt auch in unterschiedlichen Versionen problemlos ausführen können. Die bisherige Inkompatibilität der DLL-Dateien (Dynamic Link Library) – jede Version der verwendeten Komponenten nutzt eine zu den anderen nicht kompatible DLL – ist damit endgültig überwunden. Selbst lokalisierte Komponenten lassen sich dank der Side-by-Side Execution ohne Probleme parallel nutzen.

Microsoft .NET: Vereint das Beste aus allen Clients

.NET ist eine Infrastrukturplattform für Anwendungen; sie stellt eine Laufzeitumgebung und Bausteine zur Verfügung, um Lösungen schnell und effizient zu realisieren. Interoperabilität, auch mit Produkten von Drittherstellern, ist durch den Einsatz von Standardprotokollen wie TCP/IP, XML und SOAP gewährleistet. Und das nicht nur auf dem Server oder dem Desktop PC. Mobile Endgeräte wie Tablet PC, Pocket PC, Smartphone und auf Windows CE basierende industrielle Controller, so genannte eingebettete Geräte, können in den Genuss der Vorteile des .NET Framework kommen. Für die eingebetteten Geräte

hat Microsoft eine spezielle Version des .NET Compact Framework entwickelt. Sie berücksichtigt deren besondere Ressourcenanforderungen und bietet eine zum Gerät passende Untermenge der Funktionalität des „großen“ Framework.

Die Entwickler der .NET-Infrastruktur haben besonders daran gearbeitet, drängende IT-Probleme wie Sicherheit, Wartung, Flexibilität und Benutzerkomfort zu verbessern. Heute eröffnen sich mit ihr neue Wege, die auch etablierte Lösungen wie Thin und Rich Clients auf der Zielgeraden hinter sich lassen: Microsoft hat alle Vorteile der existierenden Clients zu

samtan Aufrufkette der Anwendung nur ein vom Administrator zugelassener Programmcode befindet.

Durch diese Art der Softwareverteilung sinken die normalerweise gerade bei Installation und Wartung anfallenden Kosten für eine hochfunktionale Client-Anwendung enorm, Software-Updates entfallen, weil die neuen Komponenten auf dem Server bereitgestellt werden.

Nicht allein der Einsatz von Anwendungen wird mit .NET erheblich vereinfacht. Mit Hilfe der neuen Technologien ist es möglich, sowohl Browser- als auch Windows-Formularanwendungen für den Benutzer transparent

komponenten liegen auch in diesem Fall auf einem Webserver im Intranet und werden mit denselben Mechanismen wie eine Smart-Client-Anwendung geladen. Und dies bietet die gleichen Vorteile bezüglich Installation, Wartung und Funktionalität.

Die für ein solches Dokument bereitstehenden Informationsquellen sind unbegrenzt und reichen vom dynamischen Zugriff auf andere Dokumente auf Dateifreigaben über Datenbanken bis hin zur Nutzung von Webservices, wenn diese im Netzwerk zur Verfügung stehen.

Dabei ist es möglich, hoch entwickelte On-/Offline-Funktionalität zu

Security überprüft zudem während der Laufzeit eines Programms, ob sich auch wirklich nur zugelassener Code, mit entsprechenden Zugriffsrechten, im Speicher des Systems befindet. Zusätzlich wird auch das Vertrauensverhältnis zum Speicherort jeder in einer Anwendung verwendeten Komponente validiert. Die Konfiguration der Rechte einer Anwendung erfolgt zum Beispiel über Active Directory und kann so von einem Administrator unternehmenszentral und kostengünstig vorgenommen werden.

Mit Hilfe der .NET Technologie wird also sichergestellt, dass weder schädliche Zugriffe auf die Hardware noch auf die Informationen eines Benutzers möglich sind.

.NET macht mobil

Da Smart Clients und Smart Documents sich für den Benutzer transparent und auf jeder .NET-fähigen Plattform verwenden lassen, sind sie nicht für ein bestimmtes Gerät limitiert. Die von ihnen angebotene Funktionalität steht dem Anwender auf seinem Arbeitsplatzrechner, seinem Laptop, Tablet PC, auf seinem Pocket PC oder, zukünftig, auch auf einem Smartphone zur Verfügung. Die aufwändige Installation früherer Tage vereinfacht sich dabei, ähnlich wie bei Desktop-Rechnern, zum Laden einer URL oder zum Öffnen eines Dokuments. Dabei spielt es keine Rolle, an welchem Ort der Benutzer sich befindet. Selbst wenn zum Verwendungszeitpunkt kein Netzwerk zur Verfügung steht, kümmern sich die .NET-Anwendungskomponenten darum, dass offline erfolgte Änderungen später in die im Netzwerk vorhandenen Dokumente oder Dienste einfließen.

Hohes Potenzial: .NET-Smart-Client-Technologien

Die Microsoft-Smart-Client-Technologie überwindet Barrieren, die Anwendungsentwickler seit mehr als zehn Jahren einreißen möchten. So ist es möglich, die Vorteile der Rich und Thin Clients zu einer leistungsfähigen Lösung zu kombinieren, ohne dabei die spezifischen Nachteile des jeweiligen Clients in Kauf nehmen zu müssen.



Office 2003 nutzt dies: Dokumente sind strukturierte und sich selbst beschreibende Datenquellen, die ohne großen Aufwand in eine Lösung integriert werden können. Da sie gleichzeitig alle Anlagen der Smart Clients besitzen, verwischt die Grenze zwischen Dokument und Applikation zunehmend. Das Ergebnis ist eine äußerst wandelbare Anwendungsplattform, die sich schnell und einfach den spezifischen Bedürfnissen des Benutzers anpasst.



Mit dem Smart-Client-Konzept ist es möglich, die Vorteile der Rich und Thin Clients zu einer leistungsfähigen Lösung zu kombinieren, ohne dabei die spezifischen Nachteile des jeweiligen Clients in Kauf nehmen zu müssen.

einem runden Gesamtkonzept als .NET-Smart Clients verbunden, ohne dabei die oben erwähnten Nachteile in Kauf nehmen zu müssen.

.NET-Smart Clients: schnell, sicher, komfortabel und preiswert

Smart Clients können, basierend auf der neuen Infrastruktur des .NET-Gesamtsystems, die Installationsfesseln ablegen, welche die Rich Clients immer noch bremsen. Via URL aus einer HTML-Seite oder aus einer .NET Anwendung geladen, laufen sie lokal im System des Anwenders. .NET übernimmt dabei die Versionskontrolle und lädt erforderliche Komponenten aus einem virtuellen Verzeichnis eines Webservers nach, ohne dass weitere Schritte von Seiten des Administrators nötig wären. Einträge in die Registry des Windows-Systems werden dabei ebenso wenig benötigt, wie Verzeichnisstrukturen angelegt werden müssen.

Für Sicherheit ist gesorgt, denn es greifen zwei Sicherheitsmechanismen: Assemblies (.NET-Programmkomponenten) werden nur geladen, wenn sie von einer vertrauenswürdigen Quelle stammen. Die Code-Access-Sicherheit der .NET-Laufzeitumgebung stellt zusätzlich sicher, dass sich in der ge-

zueiner Gesamtlösung zu kombinieren. Dadurch gewinnt der Anwender erheblich an Komfort, welcher nicht, wie bisher, durch Mehraufwand bei der IT-Infrastruktur und bei der Entwicklung teuer erkaufte werden muss. Smart Clients stellen darüber hinaus ihre Funktionalität auch offline zur Verfügung und bieten aus diesem Grund im mobilen Einsatz erhebliche Vorteile.

Office 2003 Smart Documents: immer aktuell

Office 2003 geht gleichermaßen neue Wege, um sich in die .NET Infrastruktur zu integrieren. Erstmals werden Layout-Informationen und Daten getrennt. Das zur Speicherung der Daten verwendete Format ist XML. Dieser Schritt hat weitreichende Folgen. So können nun auch einfache Dokumente ohne großen Aufwand als strukturierte und sich selbst beschreibende Datenquellen eingesetzt werden. Dies erleichtert die Integration in Informationsmanagement-Systeme enorm. Doch Microsoft geht weiter: Excel- oder Word-Dokumente sind selbst in der Lage, .NET-Anwendungskomponenten zu hosten, und werden so zu äußerst transportablen und in den Inhalten immer aktuellen Smart Documents. Die Anwendungs-

implementieren. Die Smart Documents sind so immer auf dem aktuellen Stand. Sie greifen auf Funktionalitäten im Netzwerk zu, die Office um auf den Anwendungsfall angepasste Funktionen erweitern.

Zur Umsetzung solcher Lösungen werden die Entwickler durch die Microsoft Visual Studio Tools für Microsoft Office 2003 unterstützt, welche Bestandteil der Office 2003 Developer Edition sind. Sie ermöglichen, integriert in die Softwareentwicklungsumgebung VS.NET, den Zugriff von .NET-Anwendungen auf das Office-Objektmodell, um dem Entwickler maximalen Komfort und Effizienz zu gewährleisten. Etablierte Office-Automationsmechanismen, wie VBA werden dabei nicht beeinträchtigt.

Smart auch gegen Viren

Durch die große Zahl von Viren für Office und die COM-Technologie ist Sicherheit zu einem zentralen Thema der neuen Entwicklungsinfrastruktur .NET geworden, die in vollem Umfang auch den Smart Clients und Documents zur Verfügung steht. So können Client-Komponenten mit einem Zertifikat signiert werden, welches den Hersteller des Codes genau identifiziert. Die .NET-Code Access