

Microsoft .NET in der Versicherungsbranche als Motor für konzernweite Systemintegration

Autor : Ingo Paszkowsky, Adrian Schuster

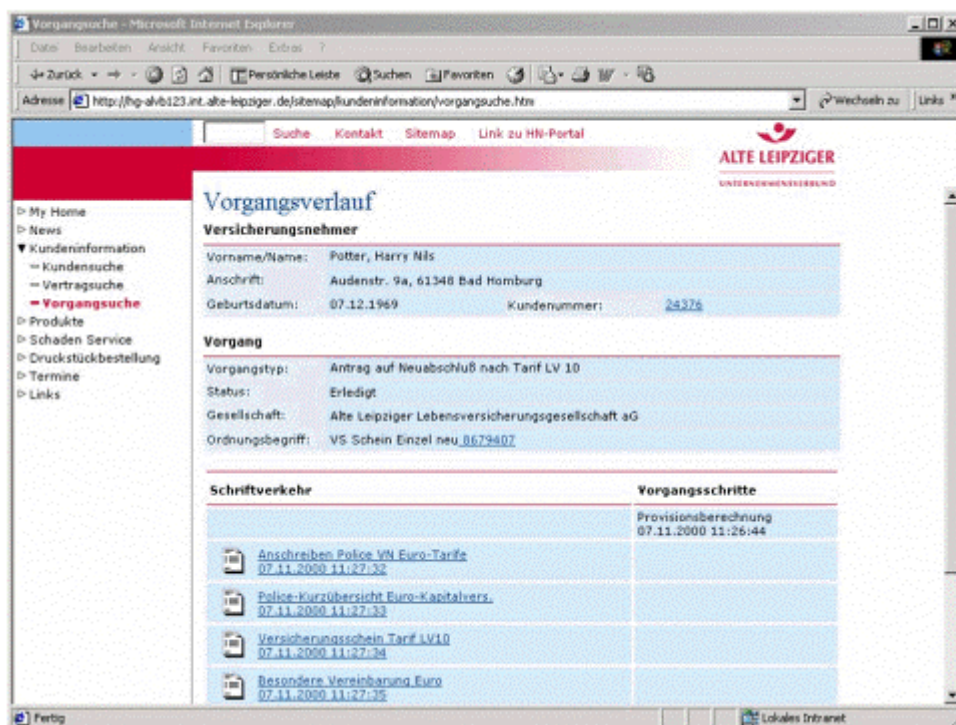
'Alte Leipziger' Unternehmensverbund, Oberursel

Mit einem Umsatz von 2,3 Milliarden Euro, Kapitalanlagen von ca.15 Milliarden Euro und 3.400 Mitarbeitern zählt die Alte Leipziger zu den mittelgroßen Versicherungsunternehmen und Finanzdienstleistern in Deutschland. Die Versicherungsgruppe bietet bedarfsgerechte, individuelle Problemlösungen für Privat- und Firmenkunden. Das Produkt- und Service-Angebot umfasst Lebens-, Kranken- und Sachversicherungen, Bauspar- und Investmentprodukte sowie spezielle Dienstleistungen wie Risikobegutachtung und Schadensprognosen.

Als serviceorientierter Dienstleister setzt der Alte Leipziger Unternehmensverbund auf eine vernetzte Multi-Kanal-Strategie zur Kommunikation mit Kunden, Geschäftspartnern und Mitarbeitern. Mit dem neuen Vermittlerportal steht für Geschäftspartner nun ein leistungsfähiges Angebot an Information, Service und Kommunikation online zur Verfügung.

Vermittlerportal sichert Wettbewerbsvorteil

Wichtigstes Ziel der neu entwickelten Portal-Anwendung ‚Kundeninformation‘ ist es, den Vermittlern einen Echtzeit-Zugang zu den operativen Systemen der Versicherung zur Verfügung zu stellen. Beispielsweise ermöglicht das Portal Recherchen nach Verträgen oder Kunden und liefert zugehörige Bearbeitungsstände zurück. Außerdem lassen sich Vorgangsschritte inkl. korrespondierender Geschäftsdokumente innerhalb selektierter Zeiträume suchen.



Das Kundeninformationsportal bietet Vermittlern einen Echtzeitzugriff auf operative Systeme der Alte Leipziger Versicherungen.

Der Browser-Zugriff auf aktuelle Vorgangs- und Kundendaten ist für die Zielgruppe ein einfacher und kostengünstiger Weg, Geschäftsabläufe zu optimieren und die Aussagefähigkeit ihren Kunden gegenüber zu erhöhen. Die Vorteile für die Alte Leipziger liegen in einer engeren Bindung der Makler, Mehrfach- und Generalagenten an das Unternehmen sowie in verbesserten Kundenbeziehungen. Und nicht zuletzt stellt diese Funktionalität des Portals ein Novum in der Branche dar. Sie steigert die Attraktivität des Versicherers für andere Vermittler - was wiederum positive Effekte bei der Neukundengewinnung erwarten lässt.

Integrationskraft entscheidet

Auch technologisch betritt die Alte Leipziger mit der Portal-Anwendung Neuland: Sie ist das erste Beispiel für den erfolgreichen Einsatz des Microsoft .NET Frameworks beim Alte Leipziger-Unternehmensverbund. Die Umsetzung des Projekts startete im Frühsommer 2001 und nutzte Microsoft Visual Studio .NET bereits in der Version Beta 2. Im Februar 2002 folgte dann die Pilotphase mit zunächst 20 ausgewählten Vermittlern. Aktuell sind es bereits 250 Vermittler, die das Portal produktiv nutzen, und der Rollout-Prozess ist wie geplant in vollem Gange.

Der Entscheidung für die .NET-Plattform von Microsoft lagen im Wesentlichen drei Hauptkriterien zugrunde:

- Webbasierte Interoperabilität zwischen beliebigen Anwendungen und Diensten.
- Konsequente Beschränkung auf offene Industriestandards wie XML und SOAP bei der technischen Kopplung von Komponenten und Systemen.
- Die Möglichkeit zu prozessgesteuerter Anwendungsentwicklung.

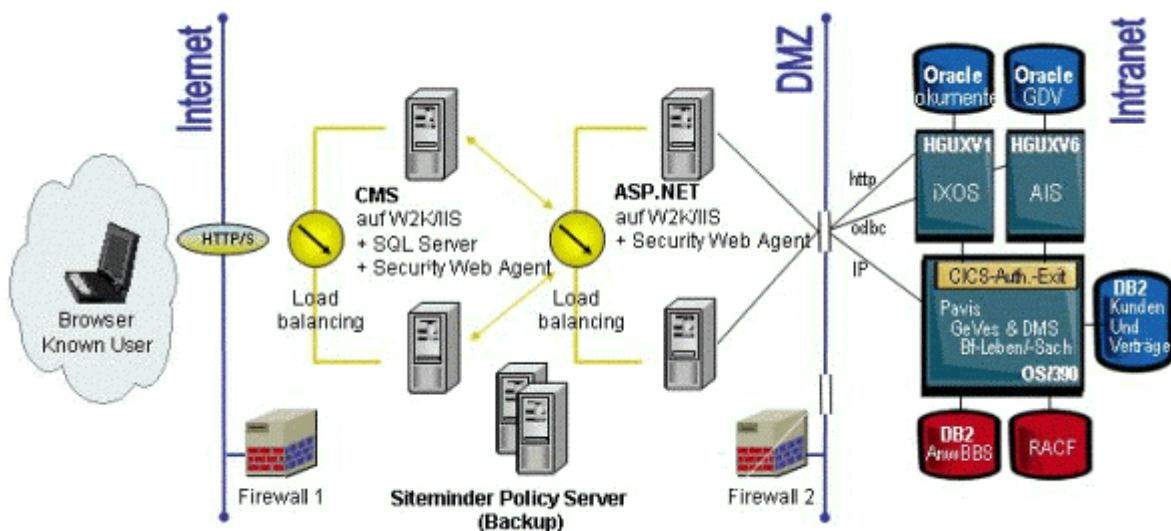
Michael Kuhn, verantwortlicher Softwarearchitekt für die Entwicklung der neuen Portal-Anwendung, begründet das Votum zugunsten von .NET folgendermaßen: "Gegenüber anderen Internet-Techniken wie JavaScript, ASP oder DHTML verspricht .NET die höchste Flexibilität bei der Integration unserer diversen Systeme und ihrer Zusammenführung unter die Oberfläche einer dynamischen Web-Anwendung." Damit ist auch gesagt, dass das Portal-Projekt ‚Kundeninformation‘ zu einem Testfall für das Potenzial der .NET-Strategie hinsichtlich der weiteren Anwendungsintegration bei den Gesellschaften der Alte Leipziger werden sollte. Als konkreten Vorteil von .NET führt Kuhn unter anderem die professionelle Entwicklungsumgebung Visual Studio .NET an, für die es bei anderen Alternativen keine vergleichbare Entsprechung gibt.

Die Architektur der Lösung

Das interne Backend der Anwendung ‚Kundeninformation‘ setzt sich aus verschiedenen operativen Kernsystemen zusammen:

- Statusinformationen zum Stand eines Vorgangs bezieht das Portal aus einem Windows-basierten Workflow-Managementsystem: der in Zusammenarbeit mit dem Systemhaus TeamWiSE, Mönchengladbach, entwickelten Versicherungslösung ‚Technische Anwendungsarchitektur‘, kurz TAA.
- Kunden- und Vertragsdaten entstammen einem zentralen GDV-Datenpool, wobei GDV hier als Name für das standardisierte Datenaustauschformat in der deutschen Versicherungswirtschaft steht.
- Elektronische Dokumente, wie gescannter Schriftverkehr, Anträge oder Schadensmeldungen, aber auch Versicherungspolizen oder Provisionsabrechnungen werden als tiff-Dateien in einem optischen Archiv von iXOS gehalten.

In der derzeitigen Ausbaustufe besteht die Portalarchitektur aus zwei mal zwei identischen Servern unter Microsoft Windows 2000 Server: je ein Serversystem für das Content Management System und für die Applikation ‚Kundeninformation‘. Beide Systeme werden über ein zweistufiges Load Balancing angesprochen. Die Nutzer-Autorisierung beziehungsweise Authentifizierung erfolgt über Web Security Agents, die auf einen zentralen Siteminder Policy Server zugreifen.



Architektur der ersten Ausbaustufe des Alte Leipziger-Vermittlerportals

Mehr Flexibilität bei der Kopplung von Systemen

Für die Versorgung des Portals mit Kunden- und Vertragsdaten aus dem GDV-Datenpool ist ein Java-basierter Web Service verantwortlich, auf den über die .NET-Klasse XMLDataDocument (mit der Methode Load) zugegriffen wird. Sie gehört zu jenen neuen Klassen, die unter .NET die Aufgaben des MSXML-Objekts XMLDOMDocument übernehmen und von den Vorteilen von .NET DOM profitieren.

.NET DOM in Kürze

- Stellt ein sprach- und plattformunabhängiges Programming Interface für HTML und XML-Dokumente bereit.
- Ermöglicht es Skripten und Programmen, auf Inhalt, Struktur und Stil eines Dokuments lesend und schreibend zuzugreifen.
- Definiert die Dokument-Struktur durch ein flexibles Objektmodell (Tree-View, Nodes, Elements, Attributes etc.).
- Liefert eine In-Memory-Repräsentation des Dokuments.
- W3C verabschiedete DOM Level 1 und DOM Level 2 Core.
- <http://www.w3.org/tr/rec-dom-level-1>
- <http://www.w3.org/tr/dom-Level-2-core>

Ebenfalls als Web Service ist der Transfer gescannter Dokumente aus dem IXOS-Archiv zum Portal realisiert. Demgegenüber wird die Verbindung zum Workflow-Managementsystem, und damit zu den aktuellen Vorgangsdaten, von einem C# Assembly sichergestellt, das sich eines der Kernkonzepte von ADO.NET zunutze macht: den .NET Data Provider.

Konkret handelt es sich hier um eine transaktionsorientierte Verbindung, wobei dem Portal ein DataSet und der TAA-Datenquelle ein OleDb-Provider zugeordnet ist. Diese Lösung ist prinzipiell aber nicht auf den heute praktizierten unidirektionalen Datenaustausch beschränkt. Im Gegenteil: Das mächtige DataSet auf der Portalseite soll künftig auch Informationen an das Workflow-Managementsystem der TAA liefern, spätestens dann nämlich, wenn die Vermittler Neukundendaten oder aktuelle Schadensmeldungen direkt in die Portal-Oberfläche ihres Browsers eingeben werden. Nach den Planungen von Michael Kuhn soll die Portal-Applikation diese interaktive Qualitätsstufe bereits Ende 2002 erreicht haben.

Spotlights ADO.NET

Interoperabilität dank XML

XML ist ein offener Web-Standard, er ist für Menschen lesbar. Zudem enthalten XML-Daten ihre eigene Beschreibung (Metadaten) und sind daher leicht zu verarbeiten. Aus diesem Grund nutzt ADO.NET durchgängig XML für die Repräsentation und den Transfer sämtlicher Daten.

Skalierbarkeit dank verbindungslosem DataSet

Physische Verbindungen werden nur für die extrem kurze Dauer einer Transaktion etabliert. Entsprechend gering ist die Gefahr von Data-Locking. ADO.NET verbindet Applikationen mit der vernetzten Welt des Internet - durch ein verbindungsloses Modell. Anwendungsentwickler müssen demnach weniger Rücksicht auf die Ressourcen der Datenbank-Schicht nehmen. Das einfache Ablaufschema von ADO.NET: Connect - Execute (ein oder mehrere Queries) - Disconnect.

Effizientere Entwicklung und Pflege dank strikt getrennter Lösungsschichten

Wie .NET allgemein, so liegt auch ADO.NET die strikte Trennung von Datenhaltung, Businesslogik und User Interface zugrunde. Anwendungen können daher aus voneinander unabhängigen Schichten aufgebaut werden. Zum Beispiel: je eine separate Schicht für den Datenzugriff, die Business- und die Nutzungslogik sowie eine unabhängige Oberfläche als Präsentationsschicht. Der Vorteil: Entwickler und Oberflächen-Designer können parallel arbeiten. Auch später wirken sich Updates oder Änderungen immer nur in einer Schicht aus.

Das User Interface des Portals ist mit ASP.NET, dem Nachfolger von ASP, realisiert. Es erfüllt zugleich auch die Aufgabe, tiff-Daten, die vom Web Service angeliefert werden, in Echtzeit in das pdf-Format zu transformieren.

Produktivitätsschub für Entwickler

"Mit .NET konnten wir unterschiedlichste Systeme und Plattformen sehr flexibel mit dem Vermittler-Portal koppeln. Der Entwicklungsaufwand war überraschend gering. Änderungen, die früher Tage beanspruchten, dauern heute nur wenige Stunden", fasst Michael Kuhn seine bisherigen Erfahrungen mit Visual Studio .NET zusammen. Bis jetzt haben sich fünf seiner Kollegen intensiv mit der neuen Plattform auseinandergesetzt. Trotz eines tiefgehenden Paradigmenwechsels habe es auch bei C# keine ernsthaften Einstiegsschwierigkeiten gegeben. Michael Kuhn erklärt dies unter anderem mit der relativ einfachen Syntax, die gleichwohl eine hundertprozentige Objektorientierung, ähnlich wie in C oder C++, bietet.

Dynamische Web-Anwendungen lassen sich seinen Worten zufolge per Drag und Drop "komponieren". Aus Server-Steuerelementen - zu denen in Visual Studio .NET nicht nur die herkömmlichen Buttons und Textfelder gehören, sondern zum Beispiel auch komplexe Tabellen - können dank der Vererbung von Eigenschaften neue Controls auf einfache Weise abgeleitet werden. Zudem lassen sich die neuen Steuerelemente in eigenen Klassen ablegen und können somit unternehmensweit wieder verwertet werden. Das wirkt der Mehrfachentwicklung gleicher Funktionalität entgegen und fördert die Einheitlichkeit von Quellcodes innerhalb des Unternehmens.

Parallel zur weiteren Öffnung des Vermittler-Portals werden bei der Alte Leipziger derzeit webbasierte Intranetanwendungen, die analoge Funktionen wie die Portal-Anwendung 'Kundeninformation' abbilden, von JavaScript, ASP beziehungsweise DHTML in die .NET Welt portiert. Ähnliches ist auch für das Visual Basic Framework und seine Integration mit dem Workflow-Managementsystem der TAA geplant. In diesem Zusammenhang weist Michael Kuhn auf einen weiteren Vorzug von .NET hin: *"Die Common Language Runtime (CLR) ermöglicht ein problemloses Miteinander von Managed und Unmanaged Code. Portierungen können demnach schrittweise, gemäß den vorhandenen Kapazitäten vollzogen werden."*

Managed versus Unmanaged

Unmanaged Compiler erzeugen direkt ausführbaren, zum Beispiel x86-spezifischen Code. Der Code ist also an eine bestimmte Hardware und ein bestimmtes Betriebssystem gebunden. Die Kompilation von Managed Code verläuft dagegen in zwei Schritten: Aus dem Quellcode einer .NET-Sprache wird zunächst ein Zwischencode in IL (Intermediate Language) generiert. IL ähnelt auf den ersten Blick Assembler, ist aber hardwareunabhängig. Aus diesem Zwischencode erzeugt der .NET JIT Compiler nativen Maschinencode, und zwar beim ersten Client-Aufruf einer Seite. Weil der Maschinencode von der gleichen CPU erzeugt wird, die ihn auch ausführen soll, lässt er sich besser speziell für die Zielplattform optimieren.

Die Common Language Runtime (CLR) enthält für jede unterstützte Plattform einen spezialisierten JIT Compiler. Voraussetzung für die Nutzung von Managed Code: Der Quellcode muss in einer der .NET-Sprachen wie Visual Basic .NET oder C++ .NET geschrieben sein.

Fazit

Bei der Alte Leipziger Versicherung hat die .NET-Strategie ihren Praxis-Test bestanden: Gestiegene Produktivität bei der Entwicklung sowie weniger Aufwand für Softwaretests und Wartung sparen Zeit und Kosten. Zudem können neue Lösungen ihren Nutzen deutlich früher als bisher generieren. Nicht zuletzt beschleunigt .NET die Integration von Anwendungen und Verfahren innerhalb und zwischen den Versicherungsgesellschaften sowie ihren externen Partnern.

Positiv ist auch die Resonanz der Vermittler, die von der stabilen Performance der gleichwohl funktionsreichen Anwendung profitieren. Künftig wird sich ihr Arbeitsalltag weiter vereinfachen. Denn das Portal wird sich vom webbasierten Auskunftsmedium zu einer interaktiven Plattform für eine verbesserte Zusammenarbeit der Alte Leipziger-Unternehmen mit ihren Partnern entwickeln.

Lösungskomponenten

Server:

Microsoft Windows 2000 Server,
Microsoft Internet Information Services 5.0,
Microsoft .NET Framework v1.0.3705
Microsoft SQL Server 2000

Frontend:

Browser (SSL 128)
Microsoft Internet Explorer ab Version 5.0
oder Netscape Navigator ab Version 4.7

Entwicklung:

Microsoft Visual Studio .NET (ASP.NET; C#)
Microsoft SQL Server 2000

Anwender:

Herr Michael Kuhn
Alte Leipziger Unternehmensverbund
Alte-Leipziger-Platz 1
61440 Oberursel
Telefon: (06171) 66-0
Fax: (06171) 66 7500-0
E-Mail: kuhn@alte-leipziger.de
Internet: www.alte-leipziger.de

„Die Inhalte der hier eingestellten Artikel stammen möglicherweise nicht von Microsoft, sondern von Dritten und werden Ihnen kostenlos zur Verfügung gestellt. Microsoft kann daher für die Richtigkeit und Vollständigkeit der Inhalte keine Haftung übernehmen.“