

Umstellung eines ERP-Systems von Oracle Forms 6i auf .net/WPF

Stefan Basler/Tobias Lachmann
schrempp edv GmbH
Lahr

Schlüsselworte

Forms 6i, .net, WPF, Silverlight, Migration, Infrastruktur

Einleitung

Seit vielen Jahren herrscht in der Forms-Gemeinde große Unsicherheit über die Zukunft und Zukunftsfähigkeit von Forms. Zwar werden in regelmäßigen Abständen neue Versionen veröffentlicht. Aus unserer Sicht eignet sich Forms aber heute aus unterschiedlichen Gründen nicht mehr für die Entwicklung eines modernen, marktfähigen Produktes. Das bedeutet, dass wir unser Produkt, ein ERP-System mit weit über 1000 Forms-Masken, umstellen müssen.

Diese Umstellung muss unter zahlreichen Randbedingungen erfolgen: Anwendungslogik und Datenmodell müssen weitgehend erhalten bleiben, um einen Parallelbetrieb von Forms- und „neuen“ Anwendungen zu ermöglichen. Die neue Infrastruktur muss es unseren Anwendungsentwicklern ermöglichen, ähnlich produktiv zu sein wie mit Forms und muss die durch Forms bereitgestellten Funktionalitäten (Validierung, Locking, ...) umfassen. Es muss möglich sein, dem Anwender ein modernes und ergonomisches UI zur Verfügung zu stellen und schließlich sollte die gewählte Technik eine lange Lebensdauer haben.

Wir beschreiben in dem Vortrag unseren Entscheidungsprozess bis zur Auswahl von WPF. Wir stellen die verwendeten Komponenten vor und vergleichen ein komplexeres Forms-Modul mit der Entsprechung in WPF.

Geschichte und Hintergründe

Die Firma schrempp edv GmbH entwickelt bereits seit 1988 Software mit Oracle Forms (ab Version 2.3) und Oracle Reports. Zur Datenhaltung wurde eine Oracle-Datenbank der Version 5 verwendet. Im Laufe der Jahre wurden Anwendungen und Datenbanken auf die jeweils neuen Versionen der Entwicklungsumgebung (Forms 3.0, 4.0, 4.5, 6i) bzw. der Datenbank migriert. Aktuell wird Forms in der Version 6i und die Datenbankversion 11.2 verwendet.

Die Hauptprodukte sind das Warenwirtschaftspaket POS und das ERP System SIVAS mit zusammen mehr als 2000 Datenbankobjekten (Tabellen, Prozeduren, Packages), weit über 1000 Forms-Masken und vielen hundert größtenteils kundenspezifischen Reports. Zurzeit arbeiten etwa 20 Entwickler an diesen Paketen.

Auswahl einer neuen Software-Infrastruktur

Mit Beginn des Internetzeitalters hat Oracle die Entwicklungsumgebung Forms/Reports zu einer Browseranwendung weiterentwickelt. Seit Version 9 gab es keine Clientversionen mehr. Leider hat sich die Weiterentwicklung dabei darauf beschränkt, die „alte“ Funktionalität in den Browser zu migrieren. Forms wurde dabei nicht oder für uns nicht erkennbar inhaltlich weiterentwickelt und es entstanden sehr früh Gerüchte über eine Abkündigung von Forms. Diese Gerüchte haben sich zwar bis heute nicht bewahrheitet, es ist aber auch keine echte Entwicklungstätigkeit im Bereich Forms/Reports mehr erkennbar. Eine Migration auf Forms 11 wurde nur als „Zwischenschritt“ auf dem Weg zu ADF propagiert. Sie hätte einen erheblichen Aufwand ohne für uns erkennbaren Nutzen und für unsere Kunden eine erhebliche Investition in Hard- und Software bedeutet, so dass wir diese Alternative relativ früh ausschlossen.

Nach der Entscheidung gegen Forms waren nur noch zwei Alternativen vorhanden, ADF und Microsofts .net-Framework.

Für ADF sprach in erster Linie der Hersteller ORACLE, auf den wir uns bisher bei unseren Werkzeugen verlassen haben. Damit wären Entwicklungsumgebung und Datenhaltung nach wie vor aus einer Hand und die Kommunikation zwischen beiden sollte (analog Forms) reibungslos funktionieren. Allerdings hat ADF auf uns zum damaligen Zeitpunkt einen sehr komplexen und auch unfertigen Eindruck gemacht. Es war für uns nicht erkennbar wie wir unsere Software mit der gegebenen Anzahl an Entwicklern hätten umstellen können. Wir haben diesen Ansatz danach nicht mehr weiterverfolgt, so dass wir ADF zum aktuellen Zeitpunkt nicht bewerten können.

Die letzte Alternative blieb Microsofts .net. Im Unterschied zu ADF ist .net bereits seit längerem als Entwicklungsumgebung etabliert. Es existieren viele Komponentenhersteller, die sich auf .net spezialisiert haben und es gibt eine ausreichende Anzahl an Entwicklern mit entsprechendem Know-How. Weiterhin sprach für .net die sehr gute Windowsintegration, die speziell in unserem Umfeld (CAD-Anbindung, FiBu-Anbindung, Office-Anbindung, ...) enorm wichtig ist und von Kunden als selbstverständlich vorausgesetzt wird. Die Entscheidung für .net fiel schließlich Ende 2009. Innerhalb von .net hatten wir die Wahl zwischen WinForms und Windows Presentation Foundation (WPF). Da WPF die neuere, modernere und auch mächtigere Technologie ist, haben wir uns trotz steilerer Lernkurve gegenüber WinForms für WPF entschieden.

Auswahl des benötigten Toolsets

In einem ersten Schritt musste ein geeigneter Provider für den Zugriff auf die Oracle Datenbank gefunden werden. Hier standen der Provider von Microsoft, der Oracle Data Provider für .net und der Data Provider der Firma Devart zur Auswahl. Letzterer konnte uns letztlich vor allem aufgrund seines „Direct-Mode“ (keine Clientinstallation nötig) überzeugen.

Um den Zugriff auf die relationalen Datenbanktabellen in Oracle aus dem objektorientierten .net Framework zu ermöglichen wurde ein Objektrelationaler (O/R) Mapper gesucht. Wie im .net-Umfeld üblich gibt es hier sehr viele Möglichkeiten. Letztlich haben wir uns für den Mapper Entity Framework von Microsoft entschieden. Damit konnte verhindert werden, dass ein Produkt eines weiteren Herstellers verwendet wird. Zusätzlich wurde Entity Framework seit der ersten Version von Devart unterstützt.

Die UI-Komponenten, welche im Standard mit .net ausgeliefert werden sind in ihrer Funktionalität in der Regel eher einfach gehalten. Aus diesem Grund gibt es eine ganze Anzahl von

Komponentenherstellern, welche erweiterte und deutlich mächtigere Oberflächenelemente anbieten. Unter den großen Anbietern haben wir uns für Devexpress entschieden.

Als Entwicklungsumgebung wird Visual Studio 2010 verwendet und das .net Framework kommt in der aktuellen Version 4.0 zum Einsatz. Als Sprache wird C# verwendet.

Mit den eingesetzten Werkzeugen ist ein Ausführen der Anwendung ohne Installation auf dem Client möglich. Es muss lediglich das .net Framework 4.0 installiert sein.

Entwicklung einer Basisinfrastruktur

Oracle Forms ist ein Werkzeug, das dem Entwickler sehr viel Arbeit abnimmt und das schnelle Entwickeln von Programmen ermöglicht. Nachdem die Entscheidung der Basistechnologie gefallen war, galt es mit .net/WPF eine leicht verwendbare Infrastruktur zu entwickeln, welche es ermöglicht einen gleichen oder zumindest ähnlichen Grad an Produktivität wie mit Oracle Forms zu erreichen.

Wichtige Voraussetzung für die zu entwickelnde Umgebung war, dass das bisherige Datenmodell und auch die Anwendungslogik (in Form von PL/SQL-Packages auf der Datenbank) ohne Anpassungen verwendet werden kann.

Es wurde eine Vielzahl von Funktionalitäten in einer Basis-Bibliothek implementiert, welche zum Teil weit über die Möglichkeiten von Oracle Forms hinausgehen und Forms ähnliche Entwicklungsgeschwindigkeit ermöglichen. Stichwortartig sind diese:

- Frei konfigurierbare Verzeichnisstruktur für Laufzeitumgebung
- Clientlose, frei konfigurierbare Verbindung auf beliebig viele Oracle-Datenbanken
- Login-Handling
- Erstellung und Verwaltung der User-Connection auf Datenbank-Server
- Erstellung von EF-Modellen aus Oracle-Datenmodell in separate DLL
- Erstellung von Wrapper-Klassen für Datenbankfunktionen in separate DLL
- Hilfsfunktionen für lesenden und schreibenden Datenzugriff
- Datenvalidierung (Standard- und kundenspezifische Validierung auf Feld- und Satzebene)
- Konfigurierbare, standardisierte Suche
- Rechteverwaltung auf Funktions- und Item-Ebene
- Mehrsprachigkeit
- Freie Textformatierung
- Asynchrone Erstellung von Auswahllisten
- Asynchrone Ausführung langlaufender Prozesse
- Aufruf von bestehenden Oracle-Forms- und Oracle-Reports-Anwendungen mit Parameterübergabe
- Einheitliche Optik/Bedienung über Styles
- Anwenderbezogenes Speichern von Maskenlayouts
- Erweiterung der Layout- und Grid-Controls um datenbankspezifische Funktionalitäten
- Tree-Control mit integriertem Filterpanel
- Integration von Auswahllisten mit Autocomplete- und Grid-Funktionalität

Ergonomie und UI-Design

Durch Oracle Forms waren die Möglichkeiten der Benutzerführung und das UI-Designs weitgehend vorgegeben. In WPF gibt es praktisch keine systembedingten Einschränkungen im Bezug auf Benutzerführung und Design. Diese Tatsache ist zum einen eine große Chance, gleichzeitig aber auch ein entsprechend großen Risiko. Im Laufe des Projekts hat sich herausgestellt, dass wir ohne externe Hilfe nicht in der Lage waren, ein modernes Ergonomie-Konzept zu erstellen. Es wurde ein auf UI-Design und Ergonomie spezialisierter Dienstleister beauftragt dieses Konzept zu erstellen.

Projektverlauf

Nach der ersten Evaluierungsphase und der Entscheidung für ein bestimmtes Toolset war es uns wichtig, „echte“ Anwendungen zu erstellen. Aus diesem Grund haben wir uns entschieden, unsere internen Anwendungen (Supportabwicklung, EDM, ...) umzustellen. Wir konnten damit unsere Konzepte an ausreichend komplexen Programmen mit allen relevanten Fragestellungen bereits von Beginn an testen. Ein weiterer Vorteil war, dass bereits von Anfang an eine relativ große Anzahl von Anwendern mit unterschiedlichen HW-Konfigurationen bzw. unterschiedlichen Betriebssystemen (Windows XP, Vista, Windows 7, 32-Bit, 64-Bit) die neu erstellte Software aktiv verwendeten.

Wie erwartet, wurden mehrere Iterationen benötigt, um eine stabile und für Anwendungsentwickler akzeptable Infrastruktur zu erstellen.

Aktueller Stand

Seit etwa einem halben Jahr laufen alle internen Anwendungen ausschließlich als WPF-Anwendungen. Es wurden erste Standardmodule umgestellt, die sich zurzeit bei einigen Kunden in der Testphase befinden. Auch wurde auf Basis der WPF-Infrastruktur ein neues BI-Modul entwickelt, das sehr intensiv WPF-Funktionalitäten nutzt und das inzwischen bei mehreren Kunden produktiv eingesetzt wird.

Es wurden bereits Teile des Entwicklungsteams in WPF und der Basis-Infrastruktur geschult. Zum jetzigen Zeitpunkt programmieren sechs Entwickler ausschließlich mit WPF.

Beispiel umgestellter Masken

Um den Schulungsaufwand für unsere Anwender so gering wie möglich zu halten, sollte das neue Masken-Layout dem aktuellen weitgehend entsprechen. Von dieser Vorgabe sollte nur abgewichen werden, wenn offensichtliche Designfehler in der aktuellen Version bestehen, bzw. wenn die WPF-Architektur ein analoges Design nicht vorsieht.

Das folgende Beispiel zeigt eine Projektübersicht in Forms und die analoge Übersicht in WPF.

CRM-Portal - angemeldeter Benutzer: sivas@sivas-entwicklung

Aktion Editieren Feld Datensatz Abfrage Hilfe

CRM-Portal

SIVAS CRM-Portal

Vorgangsmanagement

- Projektmanagement
 - alle Projekte
 - Projekte im Anfragestadium
 - Projekte im Angebotsstadium
 - Projekte im Auftragsstadium
 - Projekte im Erledigungsstadium
 - Projekte nach Status
 - Projekte über Mitbewerberprodukte
 - eigene Projekte
 - alle Projektpositionen (Teilprojekte)
 - eigene Projektpositionen (Teilprojekte)
 - Projektsuche
 - Seriennummersuche
- Auftragsmanagement
 - alle Aufträge
 - Auftragstyp Aufträge
 - Auftragstyp Angebote
 - Auftragstyp Strichangebote
 - Auftragstyp Lieferscheine
 - eigene Aufträge
- Rechnungen
 - alle Rechnungen
 - Auftragstyp Rechnungen
 - Auftragstyp Sammelrechnungen
 - Auftragstyp Gutschriften
 - eigene Rechnungen
- Geschäftspartner und Ansprechpartner
 - Vorgänge zum Geschäftspartner
 - Vorgänge zum Ansprechpartner
 - Projekte
 - Projektpositionen (Teilprojekte)
 - kaufmännische Aufträge

Projekt	Prio	Stichwort	Projektpartner
MU 2011 1	0	A2 Montage Projekt	SIVAS Lag
P2 2011 161	0	A2 teste	Vertreter
P2 2011 162	0	A2 Test Projekt-Neuanlage aus NET	SIVAS Lag
P2 2011 163	0	A2 Test ohne Projektpartner	
P2 2011 164	0	A2 Neuanlage Projekt .Net	
P1 2011 1000	0	A2 Siloförderanlage in Werk Friesen	Schrempp
P3 2011 1015	0	A2 äpknm	SIEMENS A
P3 2011 1016	0	A2 tg	Kunde tg
P1 2011 1371	0	A2 48324 Sendenhorst Schleiten	
P1 2011 1372	0	A2 60323 Frankfurt Myliusstr.	
P1 2011 1373	0	A2 24582 Groá Buchwald	
P1 2011 1374	0	A2 33602 Bielefeld	
P1 2011 1375	0	A2 41836 H?ckelhoven Burgstr.	
P1 2011 1376	0	A2 63633 Birstein Sudetenstr.	
P1 2011 1377	0	A2 66538 Neunkirchen	
P1 2011 1378	0	A2 82467 Garmisch-Partenkirchen	
P1 2011 1379	0	A2 15890 Eisenh?ttenstadt Lindenalle	
P1 2011 1380	0	A2 46514 Schermbeck Pastoratsweg	
P1 2011 1381	0	A2 90478 N?rnberg Dutzendteichstr.	
P1 2011 1382	0	A2 14974 Ludwigsfelde	
P1 2011 1383	0	A2 53894 Mechernich An der Linde	
P1 2011 1384	0	A2 4626 Schm?lln	
P1 2011 1385	0	A2 8485 Lengenfeld	
P1 2011 1386	0	A2 89231 Neu-Ulm Augsburg Str.	
P1 2011 1387	0	A2 78056 Villingen-Schwenningen Walc	
P1 2011 1388	0	A2 25946 Nebel	

Abb. 1: Übersicht über eigene Projekte in Forms

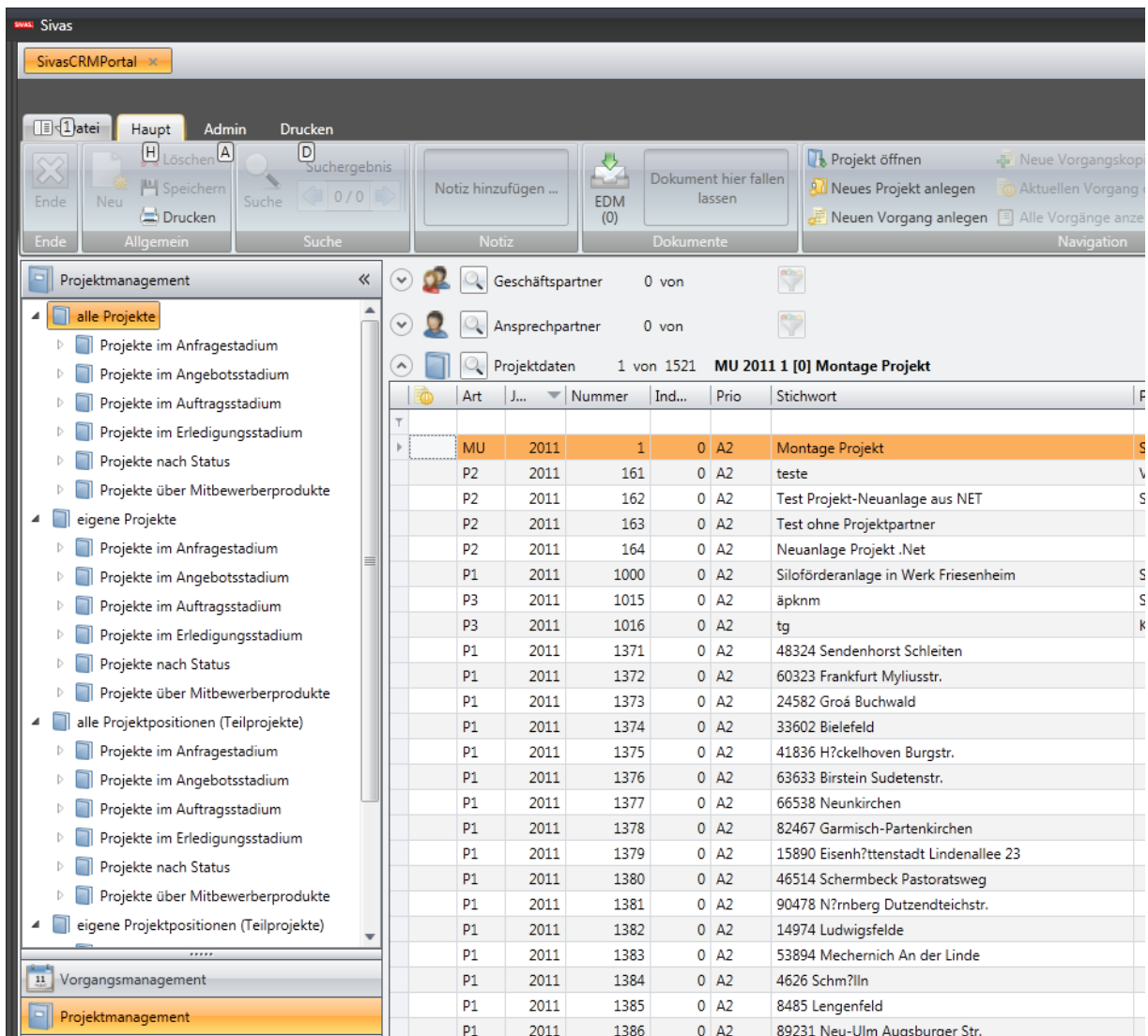


Abb. 2: Übersicht über eigene Projekte in WPF

Ausblick

Es ist geplant, alle Entwickler bis Ende des Jahres zu schulen. Das ERP-Vertriebsmodul soll ebenfalls bis zu diesem Zeitpunkt vollständig auf WPF umgestellt sein.

Als Ergänzung zu WPF wird die Eignung von Silverlight untersucht, um Module relativ unverändert auch im Browser zur Verfügung zu stellen.

Ab 01.01.2012 soll keine Neuentwicklung in Forms mehr stattfinden.

Fazit

Die bisherigen Ergebnisse bestätigen uns in unserer Entscheidung für .net. Es gibt aus unserer Sicht keine für uns wichtigen Punkte, die in der gewählten Umgebung nicht realisierbar sind. Die Mächtigkeit der neuen Entwicklungsumgebung und die damit einhergehende Komplexität führen

allerdings zu einer im Vergleich mit Forms deutlich steileren Lernkurve. Um eine zu Forms analog Entwicklungsgeschwindigkeit zu erreichen, ist vorab ein hoher Aufwand für die Infrastruktur notwendig. Die Umstellung wird erst möglich oder deutlich erleichtert, wenn die Anwendungslogik größtenteils in der Datenbank liegt.

Wie bei allen Softwareprojekten war auch in diesem Fall ein frühzeitiges Einbinden von Anwendern und Kunden sehr wichtig. Das ist umso mehr, als die aktuelle Version von mehr als 1000 Anwendern eingesetzt wird und ein „sanfter“ Übergang auf die neue Version erreicht werden muss.

Trotz der beschriebenen Eignung der neuen Infrastruktur gehen wir aber von einer längeren Co-Existenz von Forms und .net aus.

Kontaktadresse:

Stefan Basler
Schrempp edv GmbH
Rainer-Haungs-Str. 7
D-77933 Lahr

Telefon: +49 (0) 7821-95090
Fax: +49 (0) 7821-950999
E-Mail: s.basler@schrempp-edv.de
Internet: www.schrempp-edv.de