

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

World Wide Wings (WWW) ist eine fiktive Fluggesellschaft, die als Fallbeispiel für eine mehrschichtige, komponentenbasierte .NET 3.5-Anwendung dient. Herr Dr. Schwichtenberg verwendet dieses Fallbeispiel in einigen Büchern, Artikeln und Vorträgen. Dies ist eine Kurzdokumentation. Mehr Erläuterungen zu finden Sie in den Büchern.

WICHTIGE HINWEISE

Dieses Fallbeispiel wird als eine Beispielanwendung ehrenamtlich entwickelt. In Bezug auf professionelle Softwareentwicklung hat die Anwendung den Status eines frühen Prototypen. Bitte erwarten Sie daher keine vollständige, fehlerlose Anwendung.

Außerdem beachten Sie bitte: Da es sehr aufwendig ist, eine Anwendung parallel in zwei Sprachen (C# und VB) zu pflegen, sind einige Lösungen nur in jeweils einer Sprache realisiert. C# ist die bevorzugte Sprache (entsprechend dem Trend im .NET-Markt).

1 Versionsgeschichte

Die aktuellste Version basiert auf .NET 3.5 Service Pack 1. Frühere Versionen des Beispiels basierten auf .NET 2.0 und .NET 3.0. Sie können frühere Versionen herunterladen unter www.world-wide-wings.de.

Version	Erscheinungs-termin	Bemerkung
0.1	Mai 2005	Verwendet im Buch ".NET 2.0 Crashkurs – Beta-Edition" (Microsoft Press, ISBN 3-86063-987-0)
0.2	Januar 2006	Verwendet im Buch ".NET 2.0 Crashkurs, 2. Auflage" (Microsoft Press, ISBN 3-86063-531-X)
0.3	März 2006	Verwendet im Buch "ASP.NET 2.0 mit Visual Basic 2005 – Das Entwicklerbuch" (Microsoft Press, ISBN 3-86063-544-1)
0.3.1	29.4.2006	Erweiterte und neu strukturierte Fassung, die auf der CD-ROM zu dem Buch "ASP.NET 2.0 mit Visual Basic 2005 – Das Entwicklerbuch" (Microsoft Press, ISBN 3-86063-544-1) ausgeliefert wurde.
0.3.5	31.5.2006	.NET Remoting-Fassade ergänzt: <ul style="list-style-type: none"> • Remoting Server in Projekt WWWings_RemotingServer • Remoting Client in Console_VB Console_VB greift nun alternativ auch auf Webservice zu. Kleinere Verbesserungen in der Webanwendung.
0.3.6	28.7.2006	<ul style="list-style-type: none"> • Buchungscodergänzt im Datenmodell • Mehr Funktionen (Buchung, Passagierverwaltung, Webservices, Einsatz von ASP.NET 3.0 "Atlas" am Beispiel der Flugverwaltung) in Webanwendung. • Mehr Dokumentation im Code. Diese Fassung in auf der CD-ROM zu dem Buch "ASP.NET 2.0 mit C# 2005 – Das Entwicklerbuch" (Microsoft Press, ISBN 3-86063-546-8) veröffentlicht worden.
0.3.7	25.11.2006	<ul style="list-style-type: none"> • Anpassung auf ASP.NET 1.0 Version Beta 2 • Mehr AJAX-Funktionen in der Webanwendungen
0.4	18.12.2006	Erste Version, die .NET 3.0-Bibliotheken verwendet. Die Version umfasst Beispiele für WF und WCF. WPF ist noch nicht enthalten.
0.4.1	05.01.2007	Die Windows Forms-Anwendung wurde erheblich umgestaltet. Die einzelnen Fenster des Teilanwendung "Call Center" bilden eine zusammenhängende MDI-Anwendung. Die Kommunikation erfolgt mit den WCF-Diensten.
0.4.2	17.01.2007	Vorabversion der im Buch ".NET 3.0 Crashkurs" (Microsoft Press, ISBN 3866455011) verwendeten Version. Inklusive WPF-Beispiel.
0.4.3	24.02.2007	Version zur DOTNET-Konferenz 1/2007. Neu sind PowerShell-Commandlets sowie Beispiele zur Nut-

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

		zung des AJAX Control Toolkits.
0.4.3.5	15.03.2007	Version zum Buch ".NET 3.0 Crashkurs" (Microsoft Press, ISBN 3866455011) verwendeten Version.
0.4.5.6	1.8.2007	iX-CD 10/2007 mit kleineren Verbesserungen
0.5.0.2	10.4.2008	Erste Version für .NET 3.5 und Visual Studio 2008. Enthalten im Buch Version zum Buch "ASP.NET 3.5 Crashkurs" (ausgeliefert auf der Buch-DVD und im Web)
0.5.1.0	4.5.2008	Unit Tests und Geschäftslogik ergänzt
0.5.2.0	8.9.2008	Version zum Buch ".NET 3.5 Crashkurs" (ausgeliefert im Web, da es leider keine Buch-DVD kam). Basiert auf .NET 3.5 Service Pack 1
0.5.3.0	17.11.2008	Version zu den Büchern "ASP.NET 3.5 mit C# 2008" und "ASP.NET 3.5 mit Visual Basic 2008" (beide erscheinen im Dezember 2008 bei Microsoft Press).

2 Szenario

Die Fluggesellschaft World Wide Wings (kurz WWWings oder WWW) ist eine Charter-Fluggesellschaft. Sie bietet nationale, europäische und interkontinentale Flüge an. Die verschiedenen WWWings-Anwendungen verwalten Flüge, Passagiere, Flugbuchungen und Mitarbeiter der Fluggesellschaft.



Abbildung: Logo der Fluggesellschaft World Wide Wings

Um das Beispiel nicht zu komplex und unübersichtlich zu machen, wurden folgende Annahmen getroffen:

- Es gibt keinen Flugplan, sondern nur individuelle Flüge, die einmalig zu einem bestimmten Termin stattfinden.
- Bei den Buchungen einer Route aus mehreren Flügen werden die Datumsangaben nicht berücksichtigt.
- Es gibt nur eine Buchungsklasse.
- Alle Plätze auf einem Flug kosten den gleichen Einheitspreis.
- Jedes Flugzeug wird nur von einem Piloten geflogen.
- Es gibt keine Flugbegleiter.

3 Wichtige Hinweise

Bitte beachten Sie zu dem World Wide Wings-Fallbeispiel die folgenden Punkte:

- Die Implementierung des Fallsbeispiels ist nur ein Prototyp mit starken Vereinfachungen an verschiedenen

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

Stellen. Erwarten Sie bitte keine vollständige Fluggesellschaftsverwaltungssoftware. Dies wäre wirtschaftlich als Nebenprodukt eines solchen Buches nicht zu leisten. Das Fallbeispiel lässt Ihnen noch genug Raum zum Üben.

- Einige Funktionen (z.B. Pflege der Flugtabelle und Buchen eines Flugs und Buchen eines Flugs) sind redundant, d.h. mehrfach auf verschiedene Weise (z.B. in einer Seite oder Assistentenbasiert implementiert), um die verschiedenen Vorgehensweisen aufzeigen und vergleichen zu können. Bitte wundern Sie sich also nicht, wenn Sie in unteren Schichten des Projekts funktionsidentische Implementierungen von Aufgaben finden.
- Die Webanwendung vermischt verschiedene Architekturmodelle in verschiedenen Bereichen in einer Anwendung. Alle Versuche, eine Vielzahl von verschiedenen Ansätzen in verschiedenen Projekten zu pflegen haben zu einer unhandlich großen Anzahl von Projekten geführt. Daher wurden die Projekte so weit wie möglich konsolidiert.
- Die Implementierung des Fallbeispiels kann in den Büchern aufgrund der Seitenrestriktionen nur in kleinen Ausschnitten gezeigt werden. Im Buch sind daher nur kurze Ausschnitte des Programmcodes abgedruckt. In den Listing-Unterschriften ist jedoch in eckigen Klammern ein Verweis auf den Fundort der Implementierung enthalten. Im Download-Paket zu diesem Buch finden Sie eine vollständigere Implementierung.
- Nicht alle Mechanismen von .NET können sinnvoll an diesem Fallbeispiel gezeigt werden, ohne das Szenario unüberschaubar aufzublähen. Soweit wie sinnvoll, stützen sich alle Beispiele irgendwie auf das Szenario.
- An einigen Stellen sind aus Vereinfachungsgründen ganz bewusst Servernamen und Pfad in den Programmcodes hineincodiert wurden. Dies ist kein Anschauungsbeispiel für Softwarearchitektur, sondern die didaktische Notwendigkeit, um in einem Buch die Beispiele kurz und überschaubar zu halten.
- Einige Teile des Fallbeispiels liegen sowohl in C# als auch Visual Basic vor. Die Übersetzung ist zum Teil durch Sprachkonverter erfolgt ist bietet daher nicht immer optimalen Quellcode.
- Die gesamte Anwendung ist im Auslieferungszustand darauf ausgelegt, auf einem System zu arbeiten. Wenn Sie die Anwendung verteilen wollen, müssen Sie Konfigurationen (Verbindungszeichenfolgen, Webservice- und Remoting-URLs) anpassen. Im Standard steht der Rechnername auf "." bzw. "localhost", was jeweils den lokalen Rechner bezeichnet.

4 Datenmodell

Die nachstehende Abbildung zeigt das Datenmodell der WWW-Datenbank, die sowohl als Microsoft Access 2003-Datenbank (*WorldWideWings.mdb*) als auch als Microsoft SQL Server 2005-Datenbank (*WorldWideWings.mdf*) enthalten ist.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

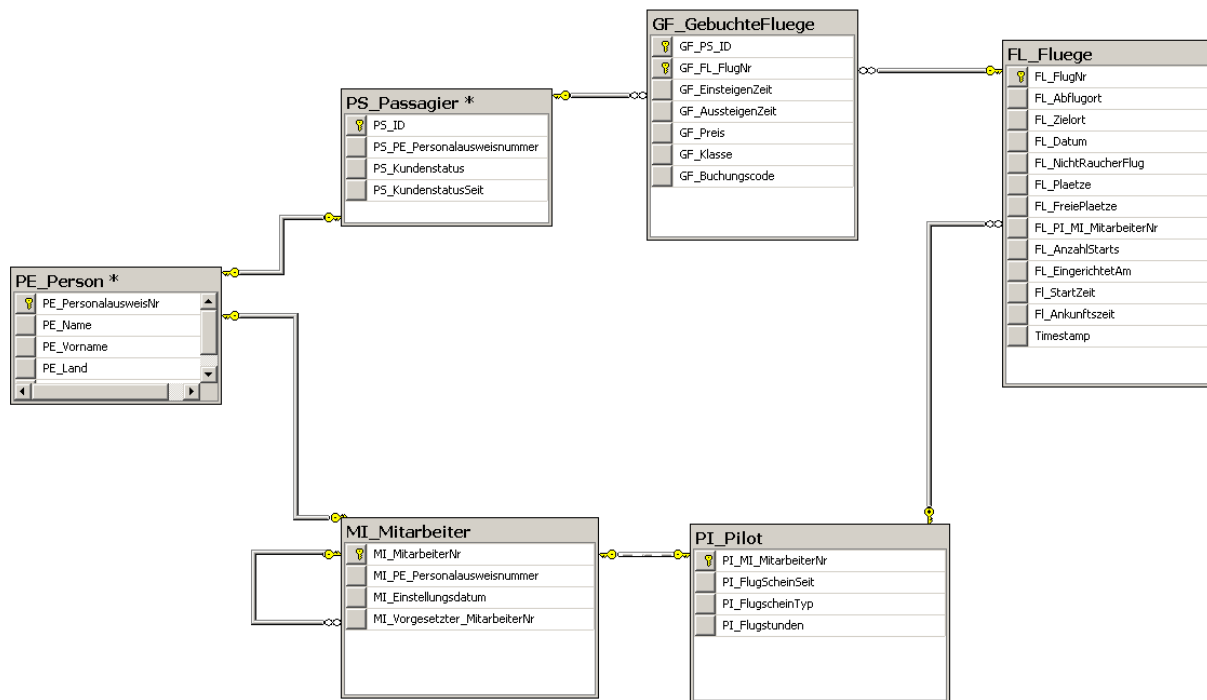


Abbildung: Datenmodell des Fallbeispiels

5 Mehrschichtarchitektur

Das Fallbeispiel ist mehrschichtig aufgebaut (siehe nächste Abbildung): Web- und Windows-Benutzerschnittstellen sind von Geschäftslogik und Datenzugriff getrennt. Die Benutzerschnittstellensteuerung greift wahlweise direkt auf die Geschäftslogik zu (d.h. die Geschäftslogik wird im Prozess des Clients ausgeführt) oder aber die Benutzerschnittstellensteuerung greift auf einen dedizierten Anwendungsserver zu. Als Kommunikationstechnik stehen drei alternative Lösungen zur Verfügung:

- WCF (gehostet im IIS oder einem eigenen Anwendungsserver, der als Systemdienst oder Konsolenanwendung läuft)
- ASMX (gehostet im IIS)
- .NET Remoting (gehostet in einem eigenen Anwendungsserver, der als Systemdienst oder Konsolenanwendung läuft)

Im Fall von WCF und ASMX gibt es eine serviceorientierte Fassade. .NET Remoting hingegen setzt direkt auf dem domänenspezifischen Objektmodell auf. Die Datenzugriffsschicht läuft immer im gleichen Prozess wie die Geschäftslogik. Eine Verteilung ist hier noch nicht vorgesehen.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

Mehrschicht-Modell

WorldWideWings-Architektur

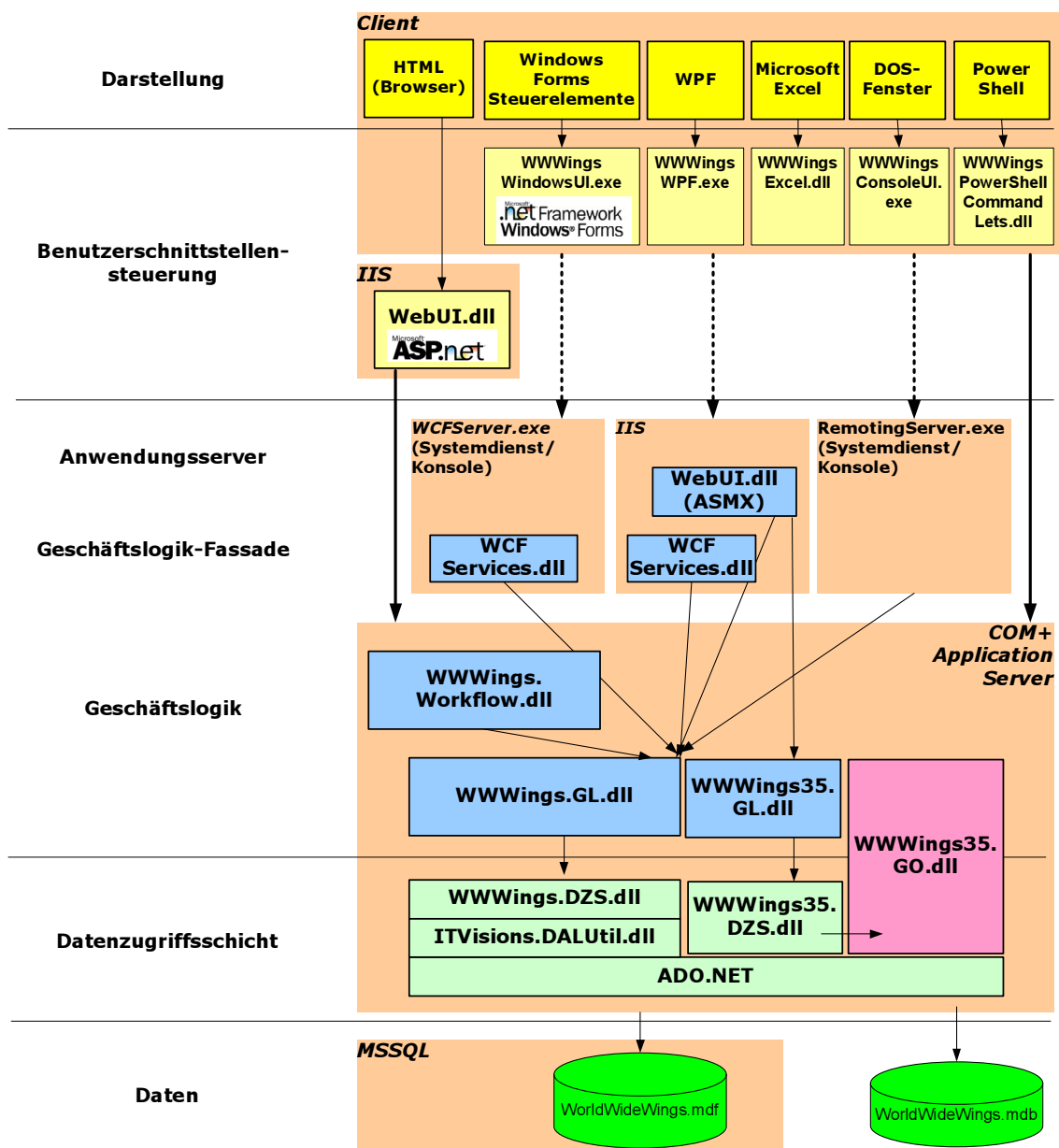


Abbildung: Gesamtarchitektur des Fallbeispiels

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

WICHTIG: An einigen Stellen wird im Fallbeispiel bewusst ganz auf Mehrschichtigkeit verzichtet. Mit den Designern und Assistenten von Visual Studio können sehr schnell Anwendungen "zusammengeklickt" werden, die den Datenzugriffscode mit der Benutzerschnittstelle vermengen. Da am Markt weiterhin eine Nachfrage nach solchen Rapid Application Development (RAD)-Werkzeugen besteht und Visual Studio noch stärker als die Vorgängerversionen diese Szenarien unterstützt, sind diese Werkzeuge auch im vorliegenden Buch beschrieben – und sei es nur, um Ihnen die Gefahren ihrer Anwendung zu zeigen.

6 Objektmodell

Die nachfolgende Grafik zeigt das von der *wwwings.GL.dll* angebotene domänenspezifische Objektmodell. Das Objektdiagramm wurde mit Visual Studio erstellt. Visual Studio unterstützt die Erstellung von Klassendiagrammen (Visual Studio-Elementtyp *Klassendiagramm*), die automatisch mit dem Quellcode synchronisiert werden.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

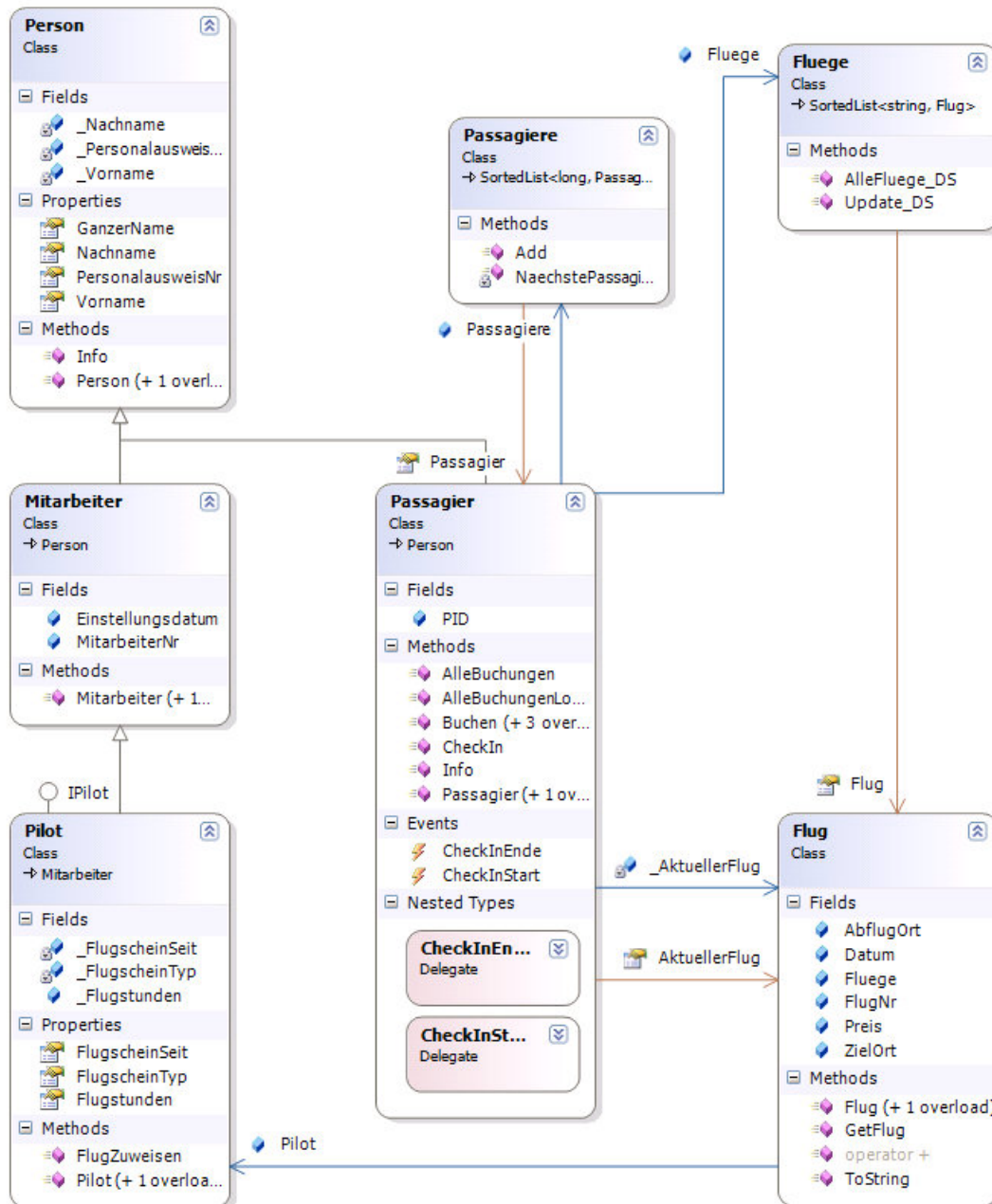


Abbildung: Fluggesellschaft_Klassendiagramm.cd

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

7 Projekte und Projektmappen

Die Anwendung ist in zahlreiche Projekte zerlegt, von denen einige sowohl in Visual Basic als auch in C# existieren. Durchweg ist die C#-Implementierung aber weiter fortgeschritten.

Baustein	Anwendungsart	Visual Basic-Projekt	C#-Projekt	Erläuterung
DALUtil	DLL	ITVisions_DALUtil.vbproj	ITVisions_DALUtil.csproj	Hilfsbibliothek zur vereinfachten Nutzung von ADO.NET (inkl. Provider-Fabriken)
Datenzugriffsschicht (.NET 2.0/3.0-Stil)	DLL	WWWings_DZS_VB.vbproj	WWWings_DZS_CS.csproj	Demonstriert den Datenzugriff mit ADO.NET und der DALUtil-Bibliothek
Datenzugriffsschicht (.NET 3.5-Stil)	DLL	<i>Nicht verfügbar</i>	WWWings35_DAL.csproj	Demonstriert den Datenzugriff mit LINQ-to-SQL
Geschäftslogik (.NET 2.0/3.0-Stil)	DLL	WWWings_GL_VB.vbproj	WWWings_GL_CS.csproj	Demonstriert die Implementierung eines domänenspezifischen Objektmodells (inkl. Einsatz von Generics) sowie Serviceorientierung auf Basis der ADO.NET-basierten Datenzugriffsschicht
Geschäftslogik (.NET 3.5-Stil)	DLL	<i>Nicht verfügbar</i>	WWWings35_GL.csproj	Demonstriert die Implementierung eines domänenspezifischen Objektmodells (inkl. Einsatz von Generics) sowie Serviceorientierung auf Basis der LINQ-to-SQL-basierten Datenzugriffsschicht
Geschäftslogikobjekt	DLL	<i>Nicht verfügbar</i>	WWWings35_GO..csproj	Realisiert den Datenkontext und die Geschäftsobjekte für WWWings35_DAL.csproj und WWWings35_BL.csproj
Anwendungsserver (.NET Remoting)	EXE	WWWings_RemotingServer	<i>Nicht verfügbar</i>	.NET Remoting Server für Geschäftslogik, programmiert als Windows-Systemdienst, der aber auch von der Konsole aus gestartet werden kann
WCF-Fassade	DLL	WWWings_WCFServices	<i>Nicht verfügbar</i>	WCF-Dienste für Zugang zu der Geschäftslogik, zum Hosting in IIS und eigenem WCF-Anwendungsserver.
Anwendungsserver (WCF)	EXE	<i>Nicht verfügbar</i>	WWWings_WCFServer	WCF-Server für Geschäftslogik, programmiert als Windows-Systemdienst, der aber auch von der Konsole aus gestartet werden kann
Konsolen-Client	EXE	WWWings_ConsoleUI_VB.vbproj	WWWings_ConsoleUI_CS.csproj	Einfache Testroutinen für Geschäftslogik und Datenzugriff (lokal und z.T. via .NET Remoting)

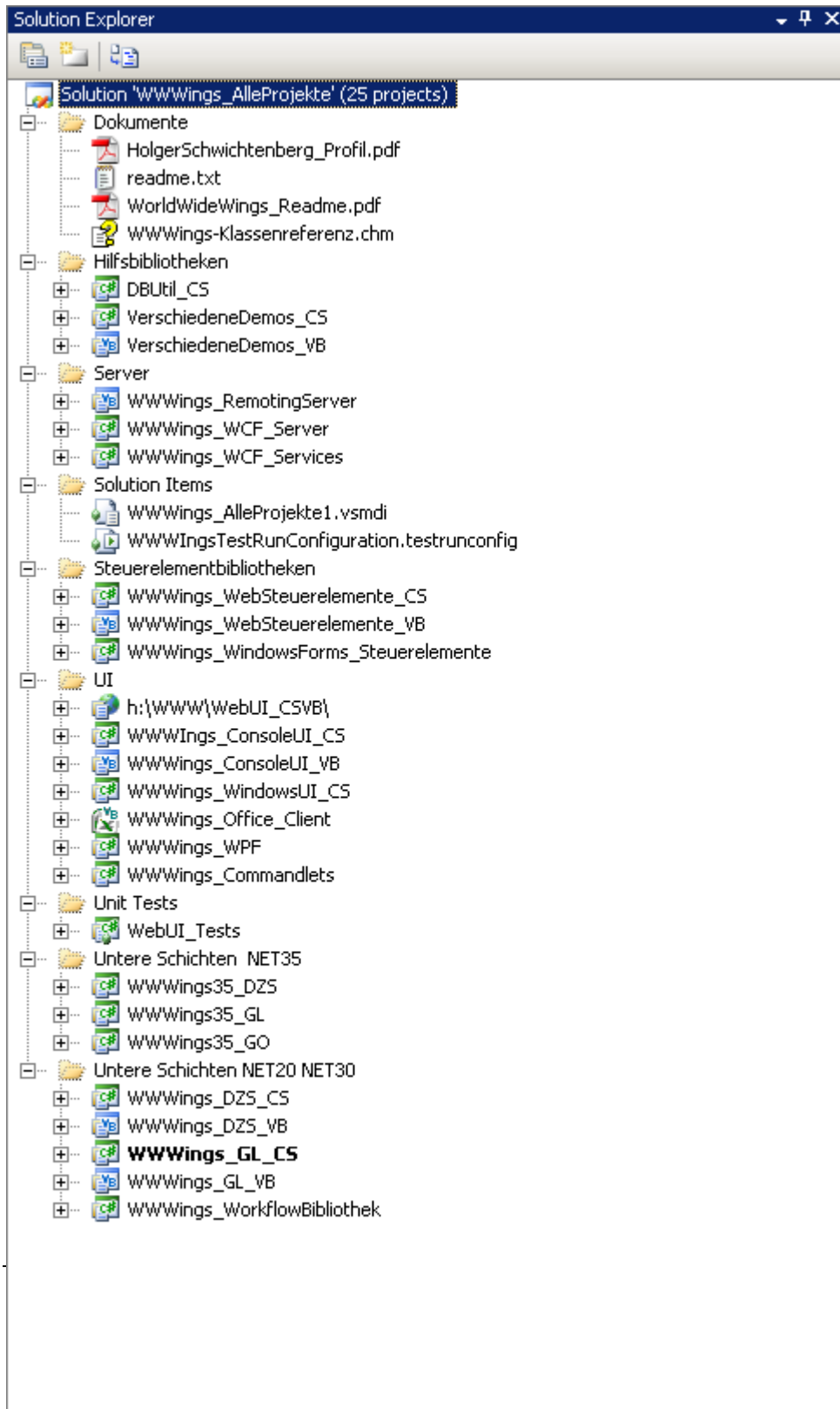
Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

Windows-Client mit Windows Forms	EXE	WWWings_WindowsUI_VB.vbproj	WWWings_WindowsUI_CS.csproj	Windows-Anwendung, erstellt mit Windows Forms
Windows-Steuererelemente für Windows Forms	DLL	WWWings_Steuererelemente.vbproj	WWWings_Steuererelemente.csproj	Steuererelementbibliothek für den Windows-Client
Windows-Client mit WPF	EXE	<i>Nicht verfügbar</i>	WWWings_WPF.cs	Windows-Anwendung, erstellt mit WPF
Sammlung von kleineren Beispielen ohne Zusammenhang zur Gesamtanwendung	DLL	VerschiedeneDemos_VB.vbproj	VerschiedeneDemos_CS.csproj	Von der Gesamtanwendung losgelöste Beispiele für die Verwendung der FCL (z.B. ADO.NET, System.Net) und die Syntax der jeweiligen Programmiersprache (z.B. Datentypen). Aufruf durch den Konsolen-Client oder den Web-Client.
Web-Client	Mehrere DLL	WebUI_CSVB		Web-Client, in dem gemischt die Sprachen VB und C# verwendet werden. Gleichzeitig Fassade für ASMX- und WCF-Dienste.
Webserversteuererelemente	DLL	WWWings_WebSteuererelemente.vbproj	WWWings_WebSteuererelemente.csproj	Steuererelementbibliothek für den Web-Client
Unit Tests	DLL	<i>Nicht verfügbar</i>	WebUI_Tests.csproj	VSTS-Testprojekt für Web-Client
Client für Microsoft Office	XSL/DLL	WWWings_Office_Client.vbproj	<i>Nicht verfügbar</i>	VSTO-Anwendung für Microsoft Office, die Daten aus dem World Wide Wings-Beispiel visualisiert
PowerShell Commandlets	DLL	WWWings_PowerShell_Commandlets	<i>Nicht verfügbar</i>	Commandlets für die Windows-PowerShell für die WWWings-Geschäftslogik: Get-Flug, Get-Passagier, New-Buchung, etc.

Tabelle: Auf dem World Wide Wings-Beispiel basierende Visual Studio 2005-Projekte

Die Projekte sind in mehreren Projektmappen unterschiedlich zusammengestellt für verschiedene Anwendungsfälle, zum Beispiel enthält "WWWings_Web.sln" nur die Projekte, die für den Web-Client benötigt werden. Die Projektmappe "WWWings_AlleProjekte.sln" enthält alle Projekte.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"



Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

Abbildung: Projektmappe mit allen Projekten

8 Funktionen der World Wide Wings-Webanwendung

Das World Wide Wings-Fallbeispiel enthält eine Webanwendung, die zahlreiche ASP.NET-Funktionen demonstriert.

8.1 Anwendungsfälle

Diese Beispielanwendung implementiert eine grafische Oberfläche sowohl für Endkunden als auch Verwaltungsmitarbeiter und Partner der World Wide Wings-Fluggesellschaft. Funktionen der Anwendung sind insbesondere:

- Navigation mit Baumdarstellung und Menü
- Anzeige des aktuellen Standorts in der Website durch *Breadcrumbs* (wörtlich: »Brotkrumen«)
- Darstellung und Änderungen von Datenbankinhalten und XML-Dateien
- Einheitliche Gestaltung für alle Seiten, definiert in einer Vorlagenseite (Masterpage)
- Kundenbereich, der nur für registrierte Nutzer zugänglich ist
- Registrieren eines Nutzers
- Anmelden/Abmelden für registrierte Nutzer
- Zusendung eines vergessenen Kennworts
- Änderung des Kennworts
- Registrierte Benutzer können die Gestaltung der Website wählen (Personalisierung).
- Portalseite mit durch den Endbenutzer konfigurierbaren Webparts
- Administrationsseiten mit der Möglichkeit, Benutzer anzulegen, zu ändern und zu löschen.
- Verwaltung der Flugdaten in Tabellen- und Einzeldatendarstellung
- Flugbuchung durch Endkunden mit einem Assistenten oder im Profimodus
- Bereitstellen von Webservices zum Abruf des Flugplans

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

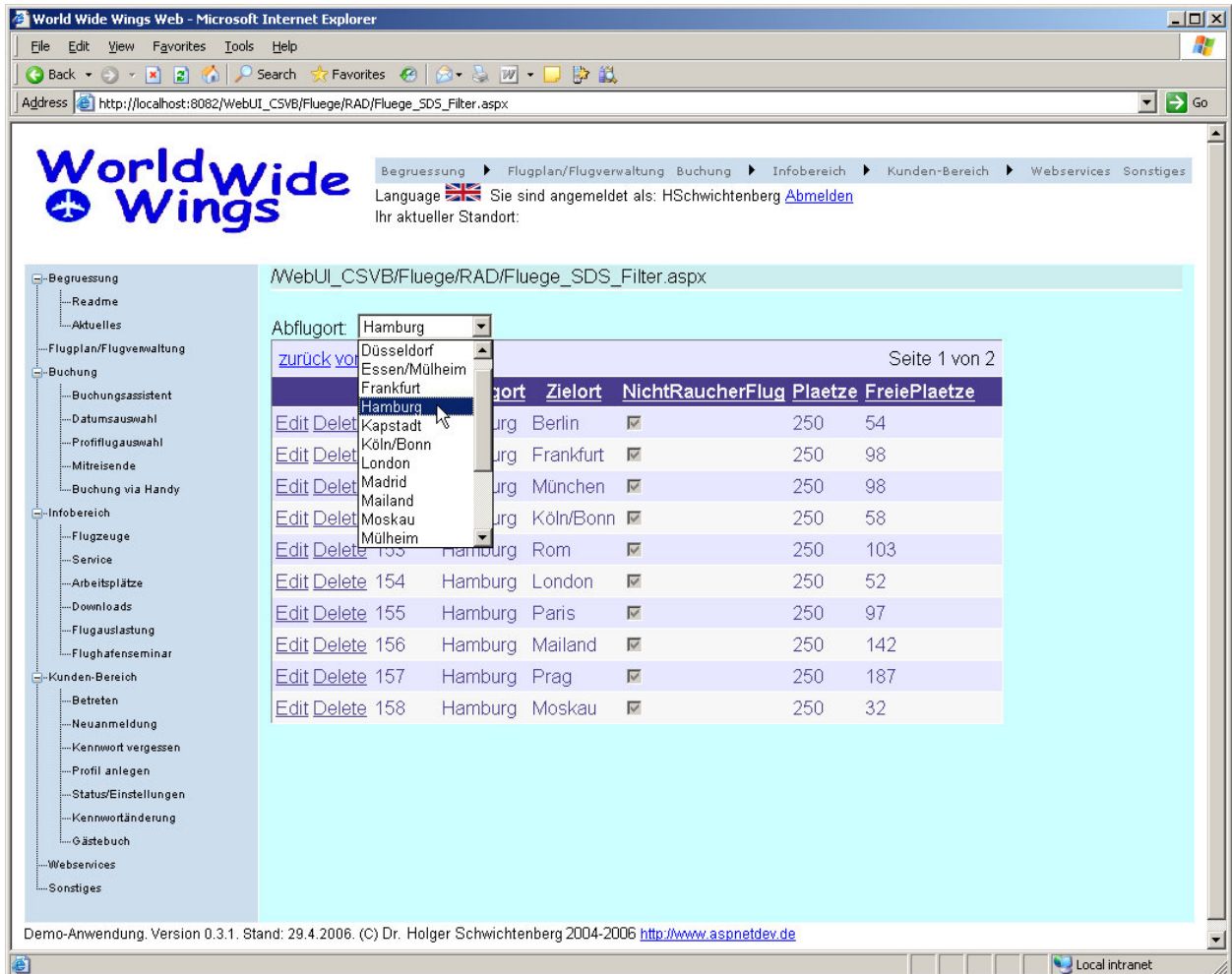


Abbildung: Beispielanwendung (WorldWideWings-Web)

8.2 Gezeigte ASP.NET-Funktionen

Insbesondere folgende Möglichkeiten von ASP.NET werden in der Anwendung verwendet:

- Einsatz der Grundsteuerelemente wie Label, Button, DropDownList, etc.
- Validierungssteuerelemente
- Container-Steuerelemente wie Panel, MultiView und Wizard.
- Navigationssteuerelemente wie Menu, TreeView und SiteMapPath
- Einsatz selbstdefinierter Steuerelemente (User Controls und Custom Controls)

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

- Masterpages, Themes, Skins
- Sitemaps
- Seitenübergänge (Redirect, Transfer, Cross Page Postbacks)
- Zustandsverwaltung (Session, Viewstate, Application)
- Formularbasierte Authentifizierung
- Sicherheitssteuerelemente (Login, LoginView, ChangePassword, CreateUserWizard, etc.)
- Datengebundene Steuerelemente (GridView, DetailsView, FormView, DataList, Repeater, etc.)
- Rapid Application Development (RAD)-Datenbindung (SqlDataSource)
- Datenbindung in mehrschichtigen Anwendungen (ObjectDataSource)
- Darstellung von relationalen Datenbanken und XML-Dokumenten
- ASP.NET 2.0 Client Callbacks
- ASP.NET 2.0 AJAX Extensions
- Konfiguration durch web.config
- Einsatz der mobilen Steuerelementen (Mobile Controls)
- Mehrsprachigkeit
- Speicherung von Profildaten
- URL-Remapping / URL-Rewriting
- Zwischenspeicherung (Caching auf Seiten-, Steuerelement- und Datenebene)
- Erzeugung dynamischer Grafiken mit System.Drawing
- Multi-Threading in Webanwendung mit Fortschrittsanzeige auf dem Client
- Einbettung von Windows Forms-Steuerelementen in Webseiten
- Dynamische Erzeugung von Steuerelementen zur Laufzeit
- Webparts
- ASP.NET Webservices (ASMX)
- Nutzung der eingebauten Objekte (Request, Response, Server, etc.)
- Zahlreiche Beispiele zur Nutzung der .NET-Klassenbibliothek (ADO.NET, System.IO, System.Diagnostics, etc.)
- Erweitern der ASP.NET-Infrastruktur durch Provider

Diese Liste erhebt keinen Anspruch auf Vollständigkeit.

8.3 Architektur

Die Webanwendung verwendet in der aktuellen Version direkt die Geschäftslogik und die Datenzugriffsschicht im Prozess des Webservers. Ein dedizierter Anwendungsserver wird nicht eingesetzt. Dies kann aber leicht geändert werden.

Hinweise zu den .NET 3.5-Code-Beispielen

"World Wide Wings"

Die Webanwendung kann wahlweise über den in Visual Studio 2005 integrierten Webserver als auch den IIS gestartet werden. Sie müssen die Konfiguration für den IIS allerdings manuell vornehmen (vgl. Bücher "ASP.NET 2.0", Kapitel 11).

9 Funktionen der World Wide Wings-Desktopanwendung

Das World Wide Wings-Fallbeispiel enthält eine Windows-Desktop-Anwendung, die zahlreiche Windows Forms-Funktionen demonstriert und in Ausschnitten hier im Kapitel besprochen wird.

9.1 Anwendungsfälle

Diese Beispielanwendung implementiert eine grafische Oberfläche für die Verwaltungsmitarbeiter der World Wide Wings-Fluggesellschaft. Funktionen der Anwendung sind insbesondere:

- Stammdatenverwaltung für Flüge (Flugrouten, Daten, Platzanzahl)
- Verwaltung der Passagierdaten
- Buchung von Flügen (auch mit mehreren Teilstrecken)
- Stornierung von Flugbuchungen
- Ausdruck von Flugtickets

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

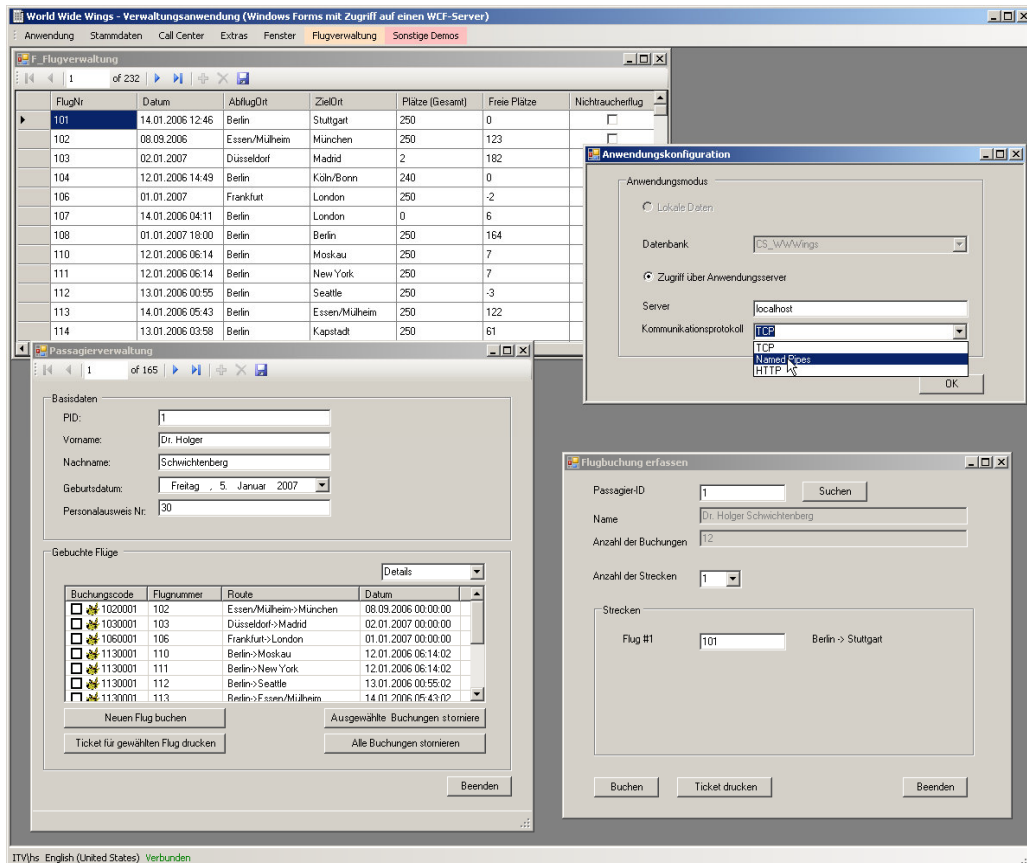


Abbildung: Die Desktop-Anwendung mit Multi-Document-Interface (MDI)

9.2 Gezeigte Windows Forms-Funktionen

Insbesondere folgende Möglichkeiten von Windows Forms werden in der Anwendung verwendet:

- Single- und Multi-Document-Interface-Fenster
- Menüs
- Symbolleisten
- Statuszeilen
- Verwendung der Grundsteuerelemente wie Button, TextBox, Label, ComboBox, etc.
- Selbsterstellte Steuerelemente (User Controls)
- Datenbindung mit dem DataGridView
- Datenbindung an einfache Steuerelemente

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

- Rapid Application Development (RAD)-Datenbindung
- Datenbindung in mehrschichtigen Anwendungen (Anbindung an WCF-Dienste)
- Individuelles Zeichnen
- Drucken
- Nutzung der Zwischenablage
- Drag&Drop
- Lokalisierung
- Einstellungen speichern auf Anwendungs- und Benutzerebene
- Click-Once-Deployment

Diese Liste erhebt keinen Anspruch auf Vollständigkeit.

9.3 Architektur

Teile der Anwendung (Startbildschirm sowie die Fenster der Menüs *Stammdaten* und *Call Center*) verwenden die Windows Communication Foundation (WCF) zum Zugriff auf einen selbstentwickelten Anwendungsserver. Wenn dieser WCF-Anwendungsserver nicht läuft, kann dieser Teil der Anwendung nicht funktionieren.

Der Buchungsassistent verwendet nicht WCF, sondern die Windows Workflow Foundation (WF). Hierzu ist Voraussetzung, dass sowohl die Datenbank *WorldWideWings.mdf* als auch die Datenbank *WWWings_Workflow.mdf* von dem lokalen System aus verfügbar sind. Die Konfiguration dieser Datenbankstandorte erfolgt direkt über die Anwendungskonfigurationsdatei.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

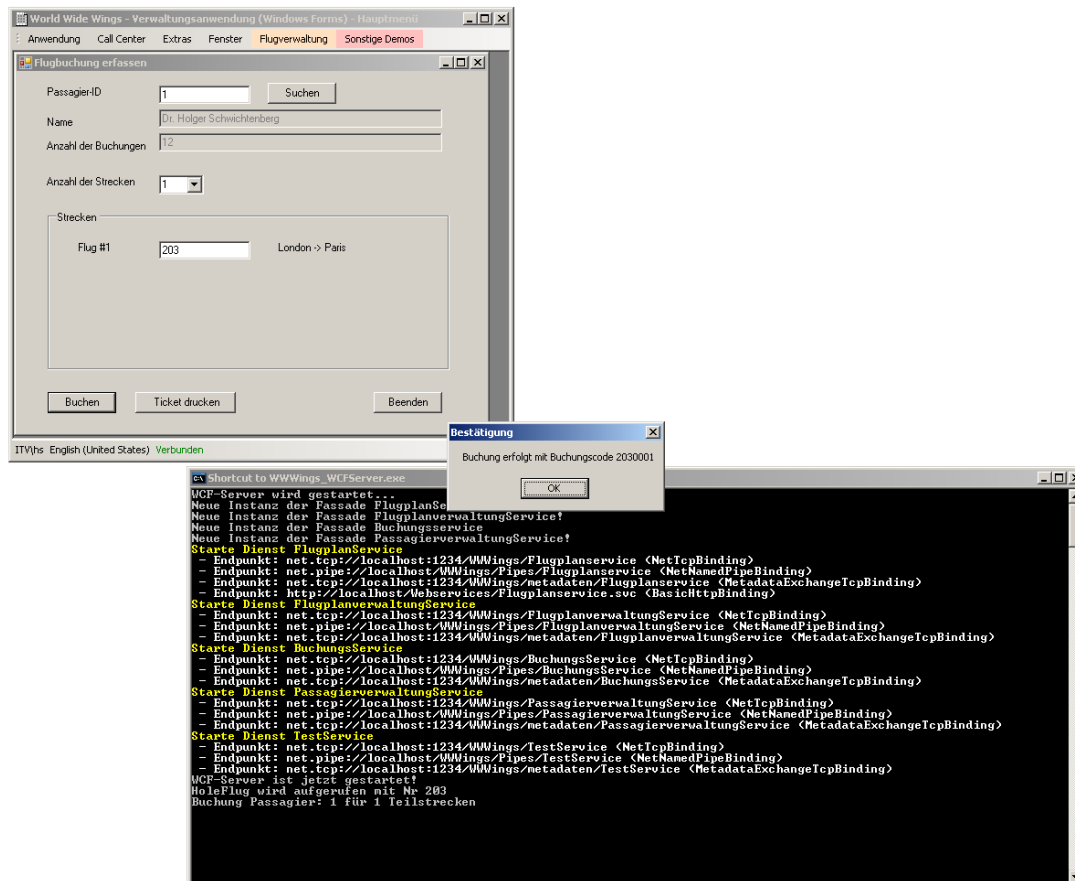


Abbildung: Die Windows Forms-Anwendung zusammen mit dem Protokollfenster des Anwendungsservers, der hier als Konsolenanwendung läuft

9.4 Hauptmenü

Das Hauptmenü zeigt einerseits Informationen über den Client an und andererseits über den Anwendungsserver. Das Hauptmenü enthält ein Timer-Steuererelement, das alle zwei Sekunden den Anwendungsserver mit einer Art "Ping" aufruft und mit den von dem Server übermittelten Informationen die Anzeige aktualisiert.

Die am weitesten fortgeschrittenen und am besten integrierten Anwendungsteile findet man in den Menüs *Stammdaten* und *Call Center*. *Extras* enthält neben den Anwendungseinstellungen auch die Möglichkeit, die Datenbank auf ihren Ursprungszustand zurückzusetzen. *Fenster* erlaubt die Verwaltung der geöffneten untergeordneten Fenster. Die farblich hervorgehobenen beiden letzten Menüs enthalten weitere isolierte Beispiele, die sich nicht in die Call Center-Anwendung einfügen (Menüpunkt *Sonstige Demos*) bzw. alternative Implementierungen der Flugverwaltung (Menüpunkt *Flugverwaltung*) zeigen.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

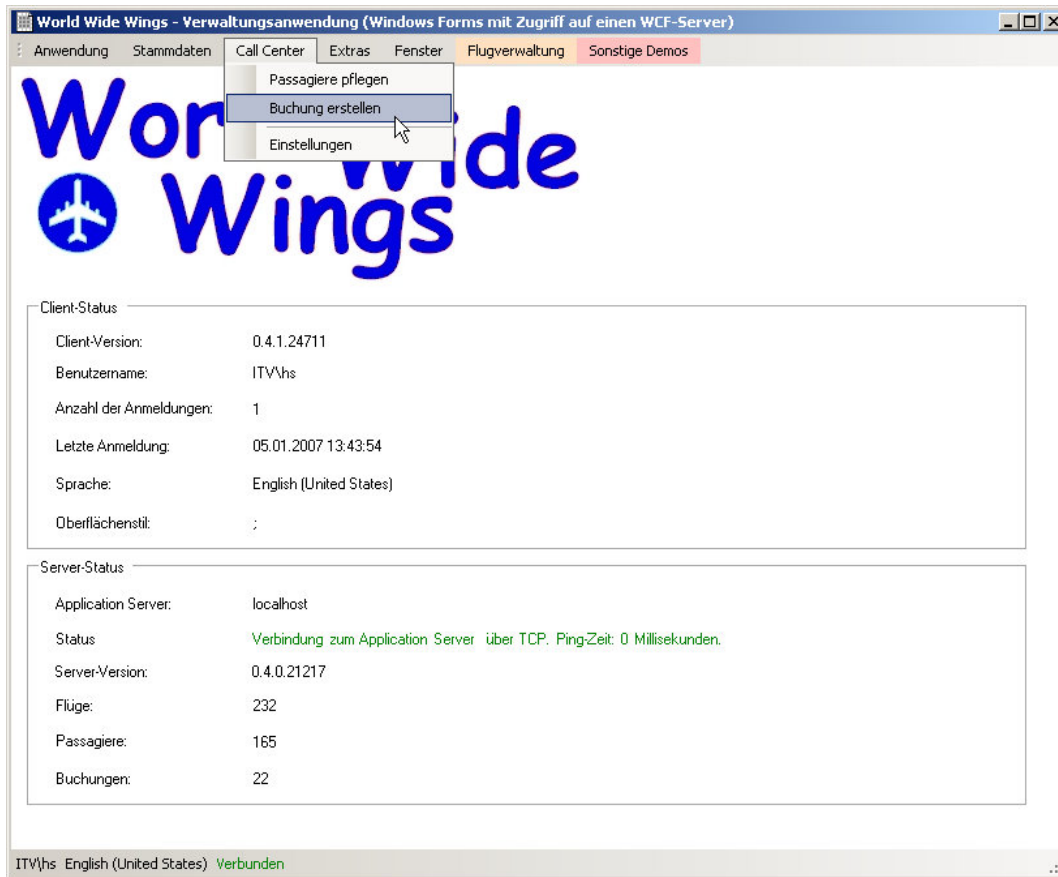


Abbildung: Hauptmenü der Windows Forms-Anwendung für World Wide Wings

9.5 Konfiguration

Über die Menüpunkte *Stammdaten/Einstellungen*, *Call Center/Einstellungen* bzw. *Extras/Einstellungen* kann man den Rechnernamen für den Anwendungsserver sowie das zu verwendende Kommunikationsprotokoll (Named Pipes, TCP oder HTTP) festlegen. Die Einstellungen werden benutzerspezifisch abgelegt.

[Hinweis Beginn](#)

Wie man in dem Konfigurationsfenster erkennt ist vorgesehen, dass die Anwendung anstelle des Zugriffs auf den Anwendungsserver auch direkt auf die Datenbank zugreifen kann. Dann würden die Geschäftslogik- und die Datenzugriffsassembly in den Prozess der Windows Forms-Anwendung geladen und dort ausgeführt. Tatsächlich bindet die Windows Forms-Anwendung diese Assemblies bereits, jedoch ist die Umschaltung in der aktuellen Version der Anwendung noch nicht implementiert. Die direkt gebundenen Assemblies werden aber außerhalb der Hauptanwendung (Menüpunkte *Flugverwaltung* und *Sonstige Demos*) verwendet.

[Hinweis Ende](#)

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

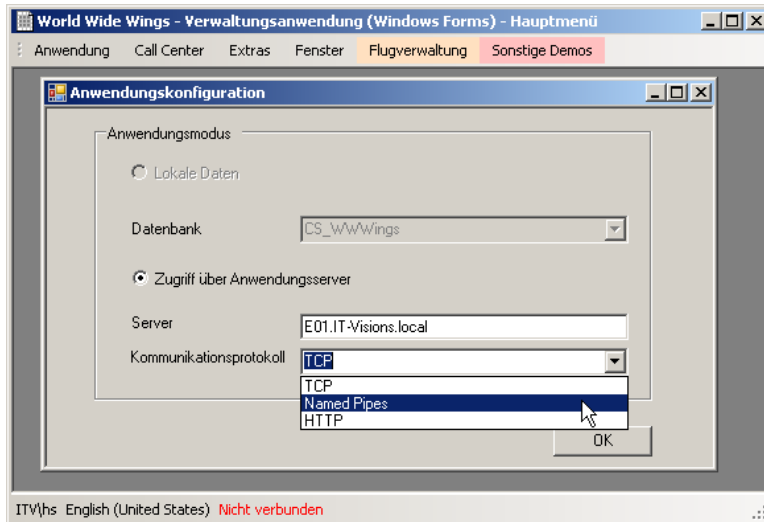
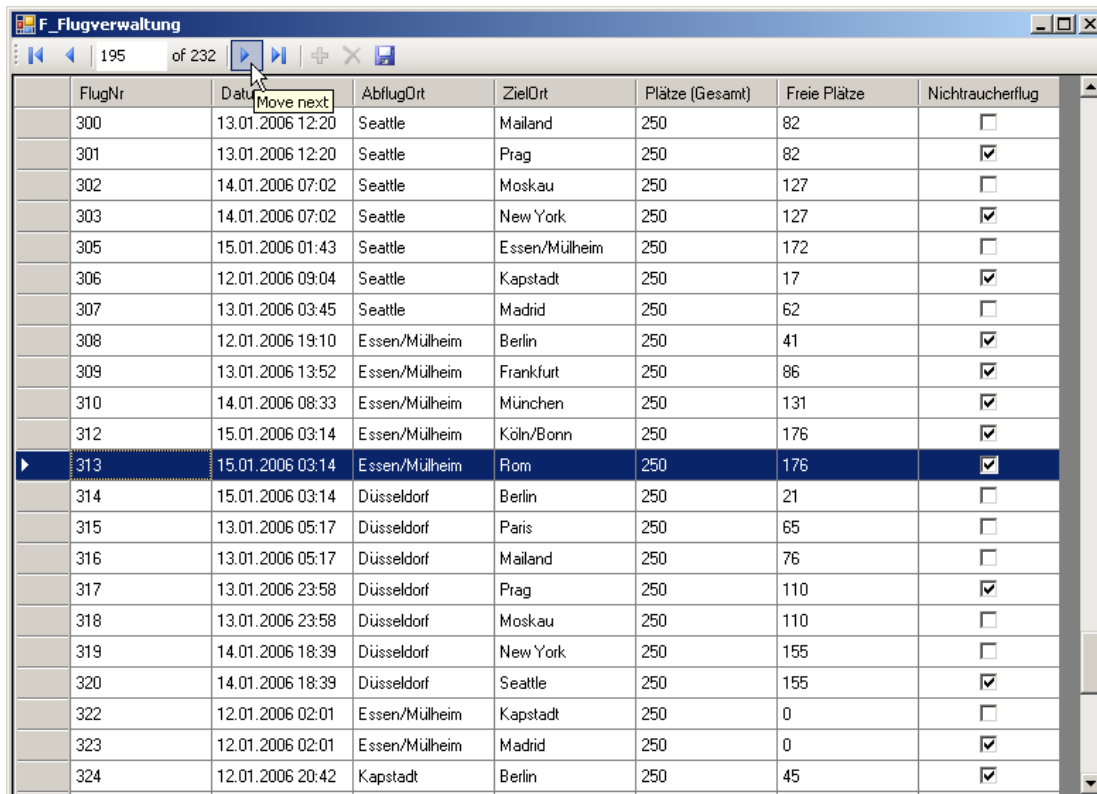


Abbildung: Konfiguration des Anwendungsservers

9.6 Stammdatenverwaltung

Die Stammdatenverwaltung umfasst eine tabellarische Ansicht aller Flüge, die dazu dient, die Stammdaten für die Flüge (Routen, Platzanzahl) zu ändern.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"



FlugNr	Datum	AbflugOrt	ZielOrt	Plätze (Gesamt)	Freie Plätze	Nichtraucherflug
300	13.01.2006 12:20	Seattle	Mailand	250	82	<input type="checkbox"/>
301	13.01.2006 12:20	Seattle	Prag	250	82	<input checked="" type="checkbox"/>
302	14.01.2006 07:02	Seattle	Moskau	250	127	<input type="checkbox"/>
303	14.01.2006 07:02	Seattle	New York	250	127	<input checked="" type="checkbox"/>
305	15.01.2006 01:43	Seattle	Essen/Mülheim	250	172	<input type="checkbox"/>
306	12.01.2006 09:04	Seattle	Kapstadt	250	17	<input checked="" type="checkbox"/>
307	13.01.2006 03:45	Seattle	Madrid	250	62	<input type="checkbox"/>
308	12.01.2006 19:10	Essen/Mülheim	Berlin	250	41	<input checked="" type="checkbox"/>
309	13.01.2006 13:52	Essen/Mülheim	Frankfurt	250	86	<input checked="" type="checkbox"/>
310	14.01.2006 08:33	Essen/Mülheim	München	250	131	<input checked="" type="checkbox"/>
312	15.01.2006 03:14	Essen/Mülheim	Köln/Bonn	250	176	<input checked="" type="checkbox"/>
313	15.01.2006 03:14	Essen/Mülheim	Rom	250	176	<input checked="" type="checkbox"/>
314	15.01.2006 03:14	Düsseldorf	Berlin	250	21	<input type="checkbox"/>
315	13.01.2006 05:17	Düsseldorf	Paris	250	65	<input type="checkbox"/>
316	13.01.2006 05:17	Düsseldorf	Mailand	250	76	<input type="checkbox"/>
317	13.01.2006 23:58	Düsseldorf	Prag	250	110	<input checked="" type="checkbox"/>
318	13.01.2006 23:58	Düsseldorf	Moskau	250	110	<input type="checkbox"/>
319	14.01.2006 18:39	Düsseldorf	New York	250	155	<input type="checkbox"/>
320	14.01.2006 18:39	Düsseldorf	Seattle	250	155	<input checked="" type="checkbox"/>
322	12.01.2006 02:01	Essen/Mülheim	Kapstadt	250	0	<input type="checkbox"/>
323	12.01.2006 02:01	Essen/Mülheim	Madrid	250	0	<input checked="" type="checkbox"/>
324	12.01.2006 20:42	Kapstadt	Berlin	250	45	<input checked="" type="checkbox"/>

Abbildung: Verwaltung der Flüge

9.7 Call Center-Anwendung

Die Menü *Call Center* hinterlegten Fenster bieten eine durchgehende Teilanwendung mit den Bereichen Passagierverwaltung, Flugbuchung und Ticketausdruck.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

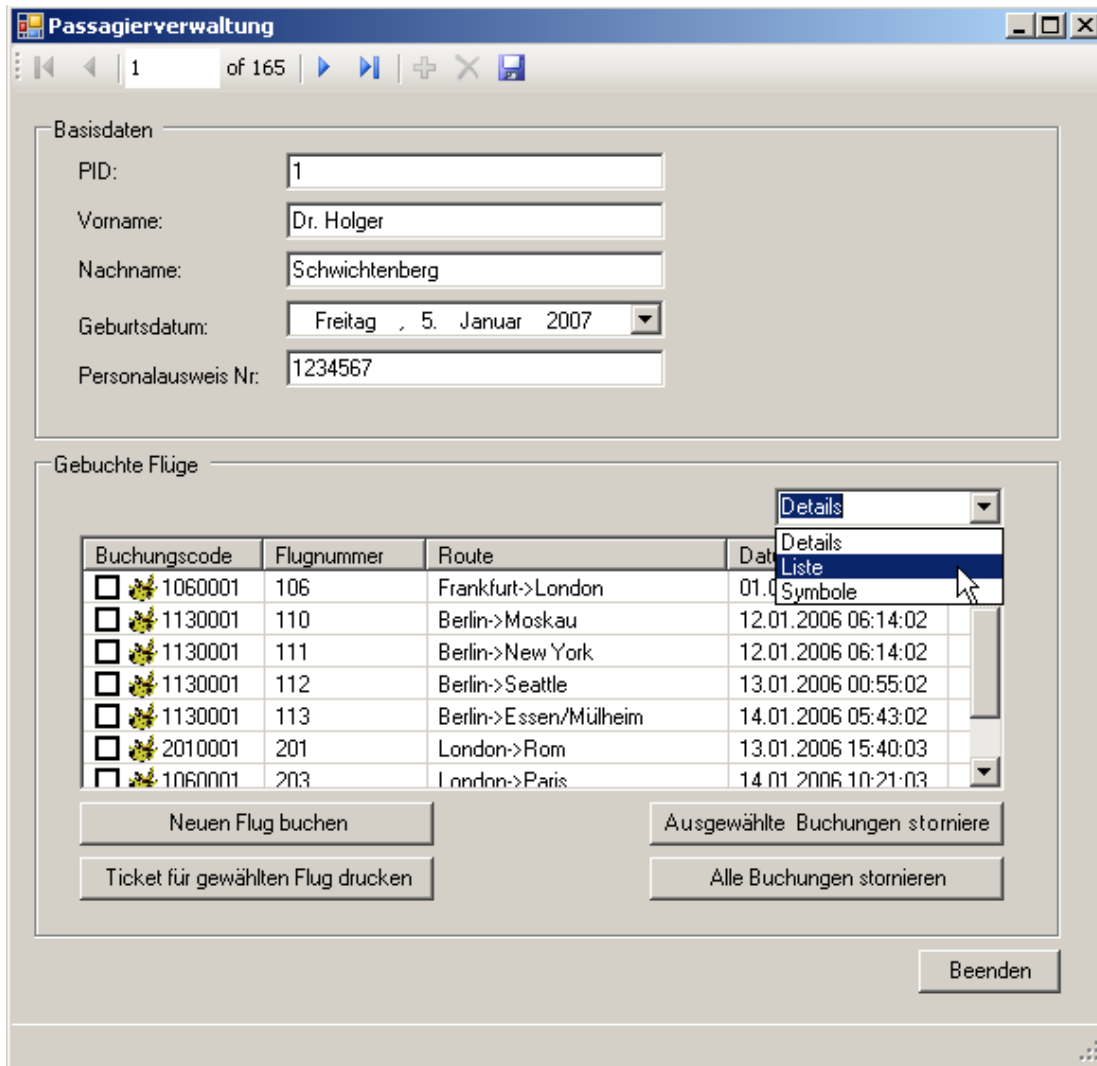


Abbildung: Verwaltung der Passagiere

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

Flugbuchung erfassen

Passagier-ID:

Name:

Anzahl der Buchungen:

Anzahl der Strecken:

Strecken:

Flug #1	<input type="text" value="313"/>	Essen/Mülheim -> Rom
Flug #2	<input type="text" value="191"/>	Rom -> New York
Flug #3	<input type="text" value="290"/>	New York -> Kapstadt

Abbildung: Erstellung einer neuen Buchung mit einer einfachen Maske

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

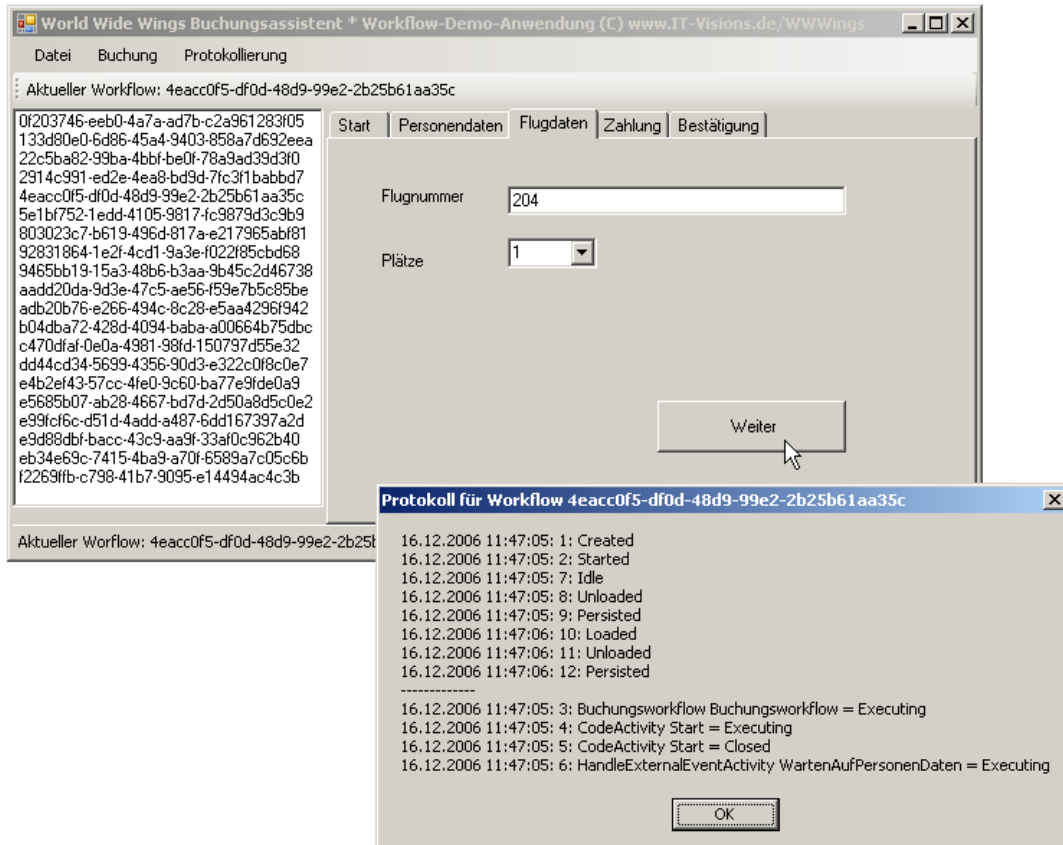


Abbildung: Der Buchungsassistent verwendet Windows Workflow Foundation

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"



Abbildung: Flugticket (Bildschirmanzeige)

In der Call Center-Teilanwendung findet u.a. folgende Anwendungsbeispiele für Windows Forms-Steuer-elemente und -Datenbindung.

Fenster	Verwendete Steuer-elemente	Weitere verwen-dete Techniken	Verwendete WCF-Dienste/WCF-Operationen
F_Hauptmen u.cs	MenuStrip, StatusS- trip, PictureBox, GroupBox, Label, Timer, NotifyIcon	MDI-Fenster, Benutzerspezifi- sche Anwen- dungseinstellun- gen	TestService.Ping()
F_Flugverwalt ung.cs	BindingSource, Bin- dingNavigator, GridView.	Datenbindung (Tabelle)	Dienste.FlugplanverwaltungService.Hol Alle() FlugplanverwaltungServi- ce.SpeichereFlug()
F_Passagierv erwaltung.cs	BindingSource, Bin- dingNavigator, GroupBox, Label, TextBox, DateTime- Picker, ListView, Imagelist, Combo-	Datenbindung (Einzeldatenan- sicht)	PassagierverwaltungServi- ce.HoleAllePassagiere() BuchungsServi- ce.HoleBuchungenFuerPassagier() BuchungsService.BuchungenLoeschen()

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

	Box, Button, StatusStrip		AlleBuchungenLoeschenFuerPassagier() BuchungsService.HoleBuchungen()
F_Buchungs.cs	TextBox, ComboBox, Button, GroupBox, Label	dynamisch erzeugte Steuerelemente	PassagierverwaltungService.HolePassagier() FlugplanverwaltungService.HoleFlug() BuchungsService.HoleBuchungenFuerPassagier() BuchungsService.MehrfachBuchungErstellenMitIDs()
F_Buchungsassistent.cs	MenuStrip, ToolStip, StatusStrip, TabControl, ListBox, TextBox, Label, Button, ComboBox, PictureBox	Zugriff auf WF-Workflow	Keine
F_Konfiguration.cs	GroupBox, RadioButton, Label, ComboBox	Benutzerspezifische Anwendungseinstellungen	Keine
F_Ticketdruck	PrintDocument, Button	Zeichnen mit System.Drawing	Keine

Tabella: Übersicht über die in der Call Center-Anwendung gezeigten Funktionen

10 Softwarevoraussetzungen für die aktuelle Version

Absolute notwendige Installationsvoraussetzungen für alle Teile der Anwendung sind:

- .NET Framework 3.5.21022.8
- .NET Framework 3.5 Service Pack 1 (3.5.30729.1) (ab World Wide Wings 0.5.2.0)
- Visual Studio 2008 (9.0.21022.8) Professional Edition
- Visual Studio 2008 Service Pack 1 (9.0.30729.1) (ab World Wide Wings 0.5.2.0)

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

- SQL Server 2005 Express Edition

Einige Teile der Anwendung erfordern außerdem folgende Installationspakete:

- Visual Studio 2008 Team Suite Edition (mit Service Pack 1!)
- SQL Server 2005 Standard oder Enterprise Edition (nur für ADO.NET Query Notifications)
- Microsoft Excel 2003 (nur für Office-Client)
- Visual Studio Tools for Microsoft Office 2008 (nur für Office-Client)
- Windows PowerShell 1.0
- MSMQ-Dienst (installiert von der Windows-CD) – für WCF-Server

Einige Projekte im Ordner `_ZUSATZ` erfordern zudem:

- ASP.NET 3.5 Extensions (für einige Zusatzprojekte)
- ADO.NET Entity Framework Beta 3
- Vanatec Open Access 4.0
- nNibernate 1.2

Bitte beachten Sie, dass mit anderen Versionen Probleme bei der Übersetzung und beim Betrieb auftreten können.

11 Installation und Konfiguration

Das Beispiel wird im Quellcode zusammen mit Visual Studio-Projektdateien ausgeliefert und muss von dem Empfänger mit einer Visual Studio-Edition oder dem Werkzeug `msbuild.exe` übersetzt werden.

11.1 Die World Wide Wings-Datenbank

Da die World Wide Wings-Datenbank in verschiedenen Projekten verwendet wird, muss Sie an einer zentralen Stelle liegen. Ausgeliefert wird sie im Unterverzeichnis `"/Datenbanken"` in einer SQL Server 2005-Version. Alternativ können Sie die Datenbank mit Hilfe der Skripte in `/Start/Setup/WWWings_Datenbank` erzeugen.

Ab Version 0.5.2.0 gibt es keine Microsoft Access-Version der Datenbank mehr!

Sie müssen die Verbindungszeichenfolgen in den Anwendungskonfigurationsdateien anpassen, damit der Pfad zu dem Datenbanken-Verzeichnis Ihrer Systemkonfiguration entspricht. Die ausgelieferte Konfiguration sieht die Verwendung der SQL Server-Datenbank vor, wobei die Datenbank unter dem Namen "WWWings" einem lokalen Microsoft SQL Server hinzugefügt ("attached") wurde.

Alternativ können Sie die SQL Server 2005-Datenbank statisch an einen entfernten SQL Server 2005/2008 anfügen oder dynamisch an einen SQL Server Express binden über `AttachDbFileName`. Dann müssen Sie in den Verbindungszeichenfolgen den Servername und andere Einstellungen gemäß Ihrer Systemkonfiguration anpassen.

Die Konfigurationsdateien, in denen Sie die Pfadangaben anpassen müssen, liegen bei .NET-Anwendungen im Standard im Wurzelordner des jeweiligen Projekts und heißen `app.config` oder `web.config`. Bitte schauen Sie hier in die Sektion `<connectionStrings>` -

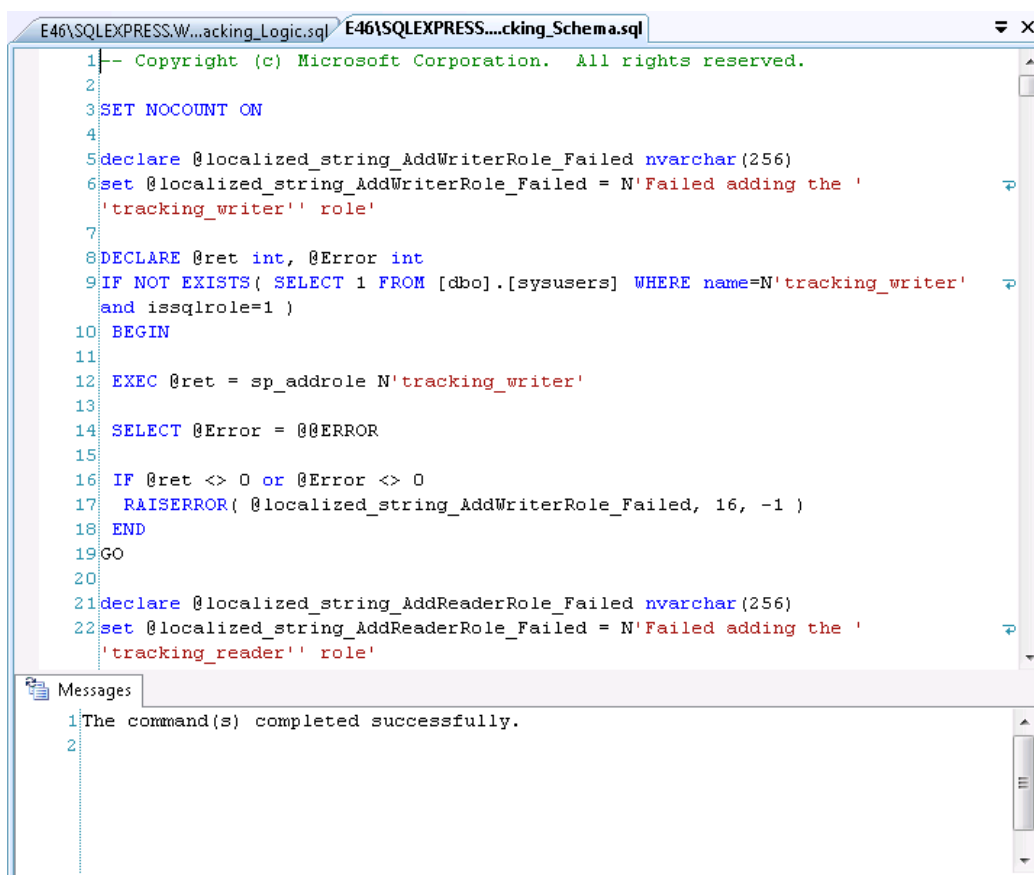
Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

Bitte beachten Sie, dass sich leider nicht alle Pfade als relative Pfade gestalten lassen. Die Standardeinstellung für alle Pfade ist `h:\www`. Wenn Sie die Anwendung dort ablegen, brauchen Sie keine Pfade anzupassen. Wenn Sie kein `h`-Laufwerk besitzen, können Sie mit dem DOS-Befehl `subst` einen entsprechenden Alias definieren. Wenn auch das nicht geht, müssen Sie die Pfade anpassen.

11.2 Weitere Datenbanken

Für die WCF- und WF-Beispiele müssen Sie auch deren Standard-Datenbanken zusätzlich installieren.

- Für WCF legen Sie eine Datenbank mit Namen "WWWings_WCF" und führen Sie dort dann die im .NET Framework mitgelieferten Standard-SQL-Skripte von Microsoft aus (SqlPersistenceProviderSchema.sql und SqlPersistenceProviderLogic.sql)
- Für WF legen Sie eine Datenbank mit Namen "WWWings_Workflow" und führen Sie dort dann die im .NET Framework mitgelieferten Standard-SQL-Skripte von Microsoft aus (SqlPersistenceService_Schema.sql, SqlPersistenceService_Logic.sql, Tracking_Schema.sql und Tracking_Logic.sql)



```
1-- Copyright (c) Microsoft Corporation. All rights reserved.
2
3SET NOCOUNT ON
4
5declare @localized_string_AddWriterRole_Failed nvarchar(256)
6set @localized_string_AddWriterRole_Failed = N'Failed adding the '
   'tracking_writer' role'
7
8DECLARE @ret int, @Error int
9IF NOT EXISTS( SELECT 1 FROM [dbo].[sysusers] WHERE name=N'tracking_writer'
   and issqlrole=1 )
10 BEGIN
11
12 EXEC @ret = sp_addrole N'tracking_writer'
13
14 SELECT @Error = @@ERROR
15
16 IF @ret <> 0 or @Error <> 0
17 RAISERROR( @localized_string_AddWriterRole_Failed, 16, -1 )
18 END
19 GO
20
21declare @localized_string_AddReaderRole_Failed nvarchar(256)
22set @localized_string_AddReaderRole_Failed = N'Failed adding the '
   'tracking_reader' role'
```

Messages

```
1 The command(s) completed successfully.
2
```

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

Abbildung: Ausführen der SQL-Skripte in Visual Studio

11.3 Komponentenpfad

Das World Wide Wings-Projekt verwendet auch zahlreiche Komponenten von Microsoft, www.IT-Visions.de und anderen Anbietern, die nicht im Quellcode mitgeliefert werden. Diese befinden sich im Verzeichnis /_Komponenten.

Name	Size	Type	Date Modified
AjaxControlToolkit.dll	1.228 KB	Application Extension	11.11.2008 22:29
AJAXExtensionsToolbox.dll	16 KB	Application Extension	13.12.2006 12:17
Castle.DynamicProxy.dll	76 KB	Application Extension	06.02.2007 13:12
Iesi.Collections.dll	32 KB	Application Extension	03.05.2007 09:42
Interop.ActiveDs.dll	104 KB	Application Extension	04.03.2008 21:20
ITV_DemoViewer.dll	28 KB	Application Extension	17.11.2008 20:51
log4net.dll	264 KB	Application Extension	30.10.2006 07:47
NHibernate.dll	1.060 KB	Application Extension	03.05.2007 09:42
nStuff.WebDevHelper.Server.dll	20 KB	Application Extension	30.08.2007 14:38
Readme.txt	1 KB	Text Document	17.11.2008 21:15
RssToolkit.dll	44 KB	Application Extension	21.02.2006 14:24
stdole.dll	16 KB	Application Extension	02.01.2006 11:10
WorldWideWings.snk	1 KB	Visual Studio Strong...	01.01.2008 17:08
WWWingsKomponentenOrdnerAlsAssemblyFolderRegistrieren.reg	1 KB	Registration Entries	17.11.2008 21:13

Damit alle Visual Studio-Projekte diese DLL finden können, sollten Sie einen Eintrag in der Registry vornehmen, wobei Sie den Pfad (im Standard h:\WWW_Komponenten) durch den Standort des Verzeichnisses in Ihrem Dateisystem ersetzen müssen.

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\.NETFramework\AssemblyFolders\WWWings]
```

```
@="H:\\WWW\\Komponenten"
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\.NETFramework\v2.0.50727\AssemblyFoldersEx\WWWings]
```

```
@="H:\\WWW\\_Komponenten"
```

Diese Registrierungseinträge werden vorgenommen durch die Datei WWWingsKomponentenOrdnerAlsAssemblyFolderRegistrieren.reg, die auch in dem Komponentenverzeichnis liegt.

12 Mögliche Fehler

Das Fallbeispiel wird vor jeder Auslieferung fehlerfrei übersetzt (siehe Bildschirmabbildung) und in den Kernelementen funktionell getestet. Laufzeitfehler können nicht ausgeschlossen werden, die Übersetzbarkeit ist jedoch getestet worden.

Warnungen sind keine Fehler: Die Projekte erzeugen zum Teil bewusst Warnungen, weil z.B. im Rahmen eines Geschwindigkeitsvergleiches zwischen frühem und spätem Binden einige Variablen nicht typisiert wurden. Andere Variablen sind noch nicht verwendet.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

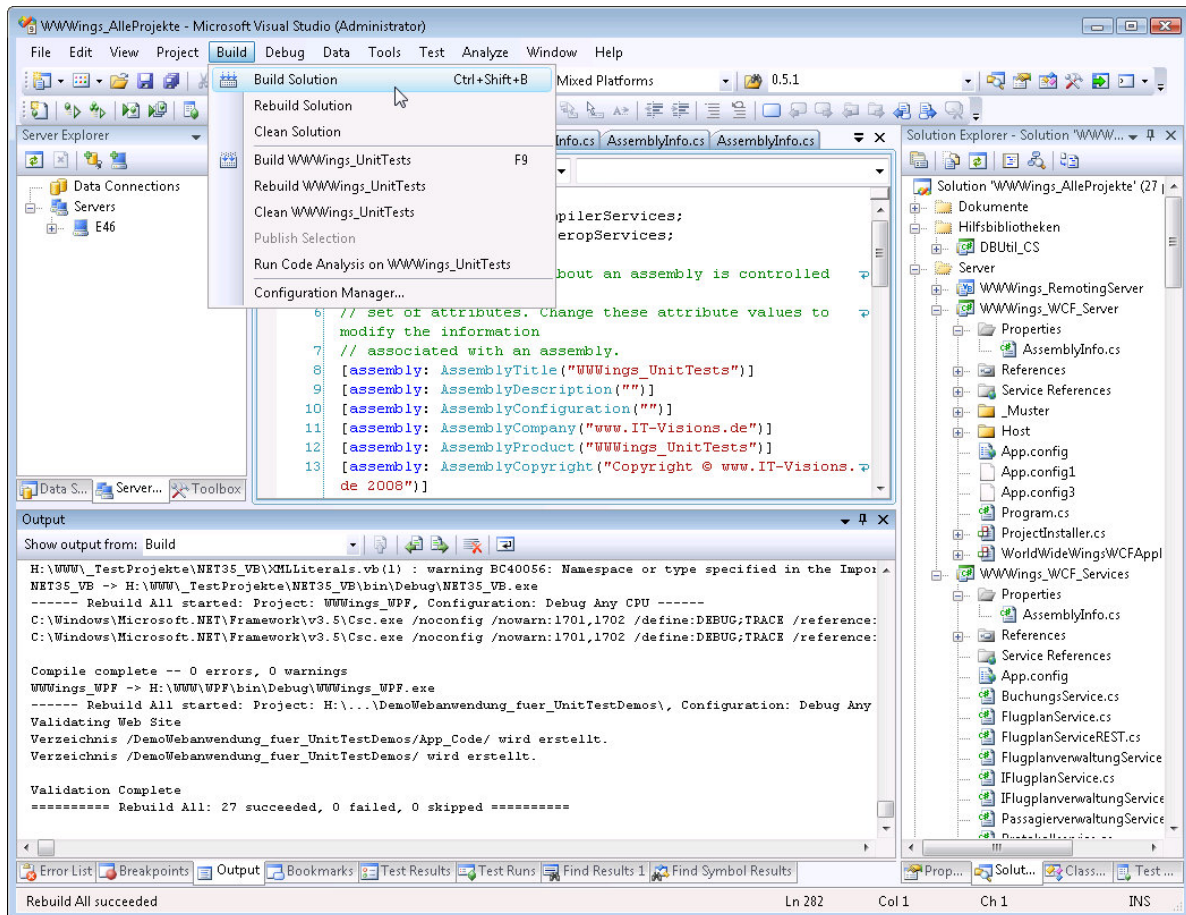


Abbildung: Beleg der fehlerfreien Übersetzung der Projektmappe mit allen Projekten (WWWings_AlleProjekte)

13 Dokumentation

Zu dem World Wings Wings-Projekt gehört eine HTML-Hilfe-Datei (.chm) enthalten, die die wesentlichen Klassen der Datenzugriffsschicht und der Geschäftslogik dokumentiert. Die Dokumentation wurde mit dem Werkzeug nDoc erstellt.

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

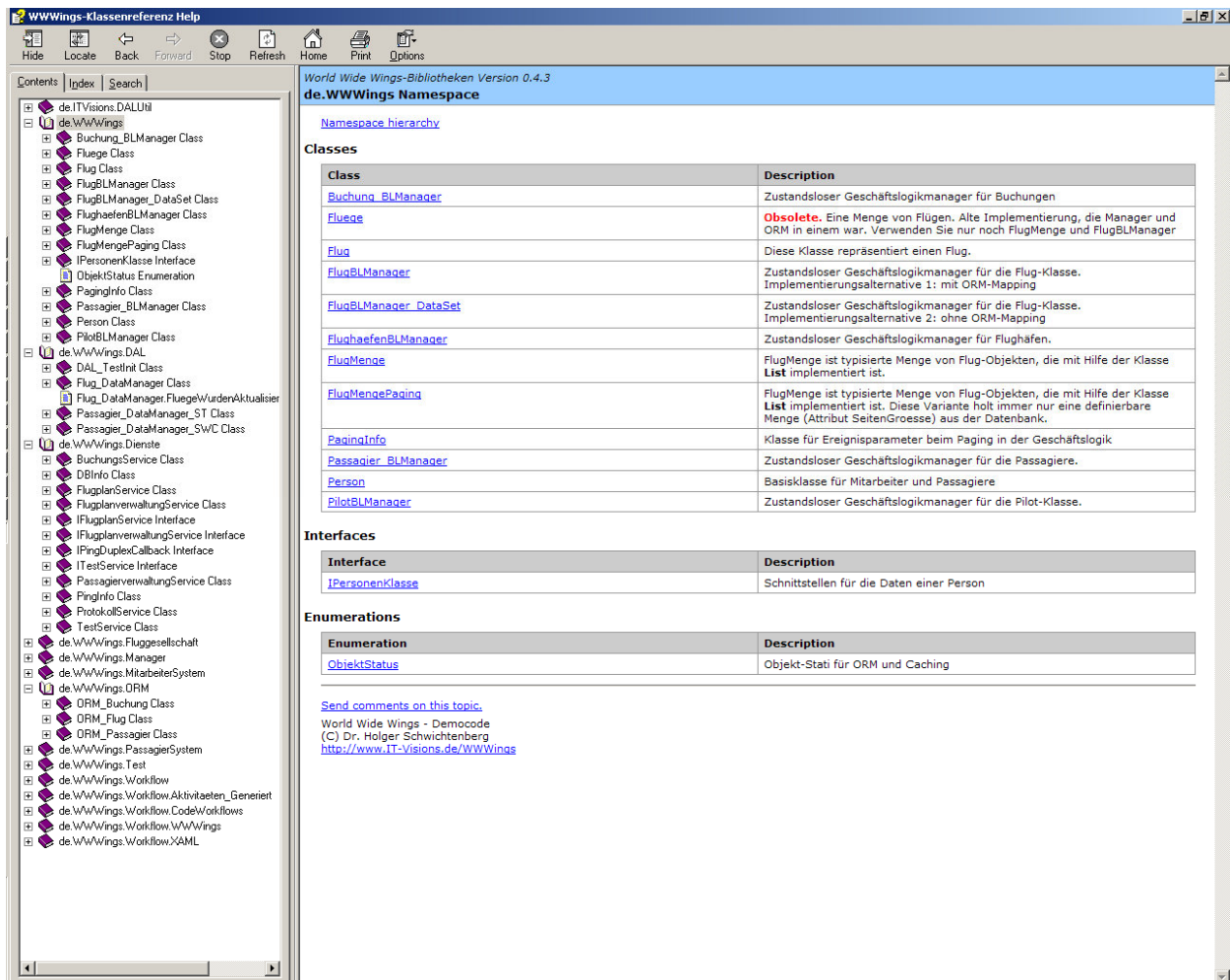


Abbildung: Dokumentation der Klassenbibliotheken

14 Aktualisierungen

Das Beispiel wird ständig aktualisiert und erweitert. Sie können die jeweils aktuelle Fassung auf folgender Webseite kostenlos beziehen:

<http://www.world-wide-wings.de>

Hinweise zu den .NET 3.5-Code-Beispielen "World Wide Wings"

15 Fragen zu diesem Beispiel

Wenn Sie Fragen zu diesem Beispiel haben, nutzen Sie bitte die webbasierten Foren auf der Homepage des Autors:

<http://www.IT-Visions.de/Foren>

Der Autor versichert Ihnen, dass die Beispiele bei ihm korrekt kompiliert wurden. Wenn die Beispiele bei Ihnen nicht kompilieren, prüfen Sie bitte die Softwarekonfiguration. Wenn die Beispiele nicht korrekt ausgeführt werden, prüfen Sie bitte, um in dem betreffenden Code alle Parameter auf ihre Umgebung angepasst sind.

16 Über den Autor Dr. Holger Schwichtenberg

Dr. Holger Schwichtenberg ist Diplom-Wirtschaftsinformatiker und hat im Bereich komponentenbasierter Softwareentwicklung promoviert. Seit 1995 ist er in der Softwareentwicklung tätig als Softwareentwickler, Softwarearchitekt und Entwicklungsleiter. Seit der ersten Stunde von .NET beschäftigt er sich intensiv mit dem .NET Framework. Durch seine Auftritte auf nationalen und internationalen Fachkonferenzen sowie zahlreiche Fachbücher für Addison-Wesley und Microsoft Press gehört er zu den bekanntesten .NET-Experten in Deutschland. Zu seinen Bestseller-Büchern gehören "Windows Scripting" und "Windows PowerShell" (beide Addison-Wesley) sowie ".NET Crashkurs" und "ASP.NET – Das Entwicklerbuch" (beide Microsoft Press). Außerdem ist Holger Schwichtenberg ständiger Mitarbeiter der Fachzeitschriften iX, dotnetpro und Windows ITPro.



Neben der Entwicklung von .NET-Anwendungen im Kundenauftrag unterstützt er mit seiner Firma www.IT-Visions.de kleine, mittlere und große Unternehmen durch Beratung und Schulung bei der Erstellung von Windows- und Web-basierten Anwendungen. In dem wettbewerbsrechtlichen Verfahren der EU gegen Microsoft war er in den Jahren 2006 und 2007 als technischer Gutachter tätig.

Von Microsoft ist er für sein Fachwissen ausgezeichnet als Microsoft Certified Solution Developer (MCSD), .NET Code Wise Member und Microsoft Most Valuable Professional (MVP) für Visual Developer/ASP.NET. Die von ihm betriebene Community-Website www.dotnetframework.de wurde von Microsoft als eine von 38 Websites in das Codezone Premier Website-Programm aufgenommen.

Weitere Informationen unter: <http://www.IT-Visions.de>

Weblog: <http://www.heise.de/ix/blog/1>