

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

EINGANGSBEMERKUNG

Diese Codebeispielsammlung wird ehrenamtlich entwickelt und sollte daher nicht mit den Maßstäben professioneller Softwareentwicklung gemessen werden. In Bezug auf professionelle Softwareentwicklung hat die Codebeispielsammlung allenfalls den Status eines sehr frühen Prototypen. Bitte erwarten Sie daher keine vollständige und fehlerlose Anwendung. Damit die Beispiele bei Ihnen kompilieren und laufen, müssen Sie Anpassungen an Ihre Umgebung vornehmen. Bitte lesen Sie dazu zuerst dieses Dokument.

1 Versionsgeschichte

Die aktuellste Version basiert auf .NET 4.0. Frühere Versionen des Beispiels basierten auf .NET 2.0 und .NET 3.0 sowie .NET 3.5 SP1. Sie können frühere Versionen unter www.world-wide-wings.de herunterladen.

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">• Buchungscode ergänzt 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 Nutzung des AJAX Control Toolkits.
0.4.3.5	15.03.2007	Verwendet im Buch ".NET 3.0 Crashkurs" (Microsoft Press, ISBN 3866455011).
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 gab). 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 erschienen im Dezember 2008 bei Microsoft Press).
0.5.3.0b	21.8.2009	Update des Readme-Dokuments

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

0.5.5.0	1.9.2009	Silverlight 3.0-Beispiel ergänzt Word 2007-Add-In und Word 2007-Vorlage ergänzt Dynamic Data mit ADO.NET Entity Framework ergänzt Weitere Beispiele zur Nutzung von ADO.NET Data Services in AJAX Beispiele für ASP.NET AJAX 4.0 (Templates, Data View, Data Provider) Änderung der Verzeichnisstruktur: Projekte in Unterverzeichnisse sortiert Erhebliche Erweiterung des Readme-Dokuments
0.6.0.0	30.12.2010	Erste Version für .NET 4.0 und Visual Studio 2010. Version zum Buch ".NET 4.0 Crashkurs" (ausgeliefert im Web, da es leider keine Buch-DVD gab). Highlights: <ul style="list-style-type: none">• C# 4.0 und VB 10.0• Entity Framework 4.0• Silverlight 4.0-Beispiele inklusive RIA Services• WPF 4.0• ASP.NET 4.0 (unter "Sonstige Beispiele")• Beispiele zur FCL 4.0 (unter "Sonstige Beispiele")

2 Szenario

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



Abbildung: Logo der Fluggesellschaft World Wide Wings

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 Stellen. Erwarten Sie bitte keine vollständige Fluggesellschaftsverwaltungssoftware. Das Fallbeispiel lässt Ihnen noch genug Raum zum Üben.
- Einige Funktionen (z.B. Pflege der Flugtabelle 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.
- Das Fallbeispiel vermischt verschiedene Architekturmodelle in verschiedenen Bereichen in einer Anwendung. Alle Versuche, eine Vielzahl von verschiedenen Ansätzen in verschiedenen Projekten zu pflegen haben zu einem unhandlich großen Anzahl von Projekten geführt. Daher wurden die Projekte so weit wie möglich konso-

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

lidiert.

- 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 wenigen Stellen sind aus Vereinfachungsgründen ganz bewusst Servernamen und Pfade in den Programmcode hineincodiert worden. Dies ist kein Anschauungsbeispiel für Softwarearchitektur, sondern die didaktische Notwendigkeit, um in einem Buch die Beispiele kurz und überschaubar zu halten.
- 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# oder Visual Basic). Wenige Teile des Fallbeispiels liegen sowohl in C# als auch Visual Basic vor. Die Übersetzung ist zum Teil durch Sprachkonverter erfolgt und bietet daher nicht immer optimalen Quellcode. C# ist die bevorzugte Sprache (entsprechend dem Trend im .NET-Markt).
- Dieses "Lies Mich"-Dokument beschreibt den Einstieg in die Beispielanwendung, will und kann aber nicht die Implementierung oder die zugrundeliegenden Technologien erläutern.

4 Datenmodell

4.1 Allgemeines zum Datenmodell

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

- Es gibt keinen Flugplan, sondern nur individuelle Flüge, die einmalig zu einem bestimmten Termin stattfinden.
- Die Abflug- und Zielorte werden zu jedem Flug als Zeichenkette abgelegt. Auf Normalisierung wird hier bewusst verzichtet.
- Es gibt nur eine Buchungsklasse.
- Alle Plätze auf einem Flug kosten den gleichen Einheitspreis.
- Bei den Buchungen einer Route aus mehreren Flügen werden die Datumsangaben nicht berücksichtigt.
- Jedes Flugzeug wird nur von einem Piloten geflogen.
- Es gibt keine Flugbegleiter.

4.2 Versionsunterschiede

Es gibt zwei Datenmodelle, beide für Microsoft SQL Server 2008.

4.3 Das ältere Datenmodell (Datenbankschema-Version 4)

Die nachstehende Abbildung zeigt das ältere Datenmodell der WWW-Datenbank, die als Microsoft SQL Server 2008-Datenbank (WWWWings4.mdf) enthalten ist. Die meisten Teile der Beispielanwendung verwenden dieses Datenmodell.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

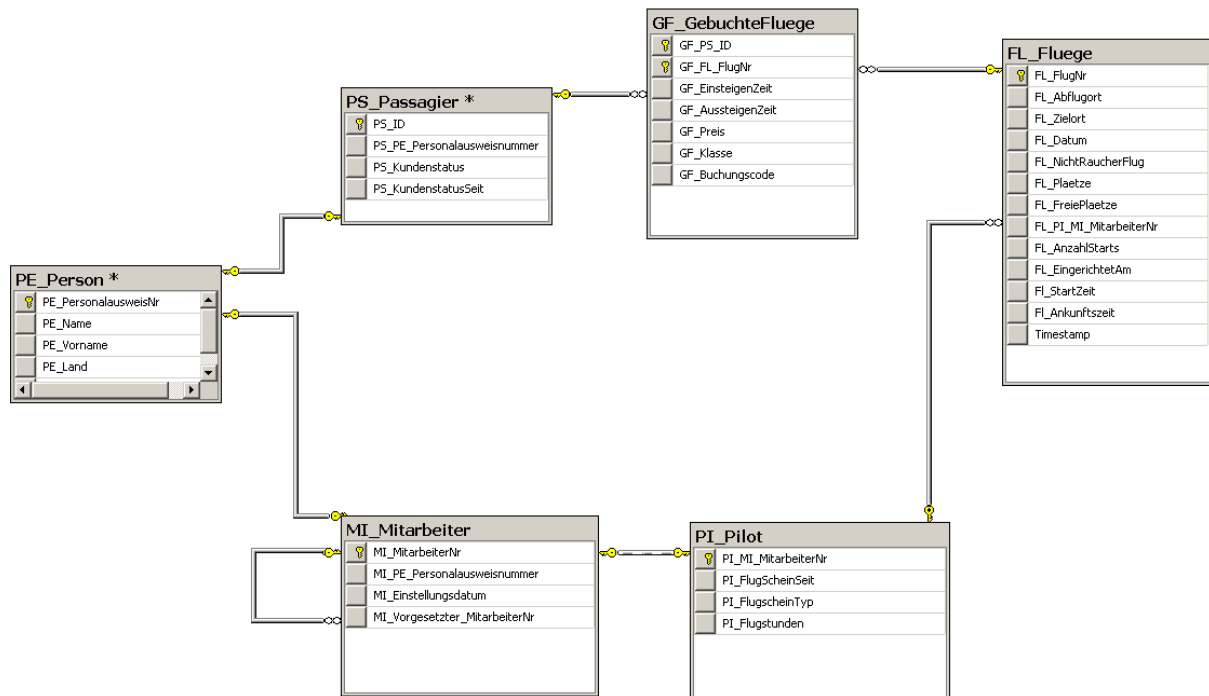


Abbildung: Älteres Datenmodell des Fallbeispiels (Version 4)

4.4 Das neuere Datenmodell (Datenbankschema-Version 6)

Die nachstehende Abbildung zeigt das neuere Datenmodell der WWW-Datenbank, die als Microsoft SQL Server 2008-Datenbank (WWWings6.mdf) enthalten ist. Einige neuere Teile der Beispielanwendung verwenden schon dieses Datenmodell.

Folgende Änderungen wurden gegenüber Schema-Version 4 vorgenommen (insbesondere für das ADO.NET Entity Framework):

- Es wird nun für die Buchungen eine reine M:N-Zwischentabelle (hier: Flug_Passagier) verwendet.
- Es wurde ein einheitlicher Primärschlüssel für die Tabellen Person, Mitarbeiter und Pilot gewählt.
- Auf die Tabellenkürzel wurde verzichtet.
- Die Tabellennamen wurden alle in den Singular gesetzt.
- Es wurden weiteren Spalten mit bisher nicht verwendeten Datentypen ergänzt (z.B. Foto, Memo).
- Bisher ungenutzte Spalten (z.B. EingerichtetAm, AnzahlStarts) wurden entfernt.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

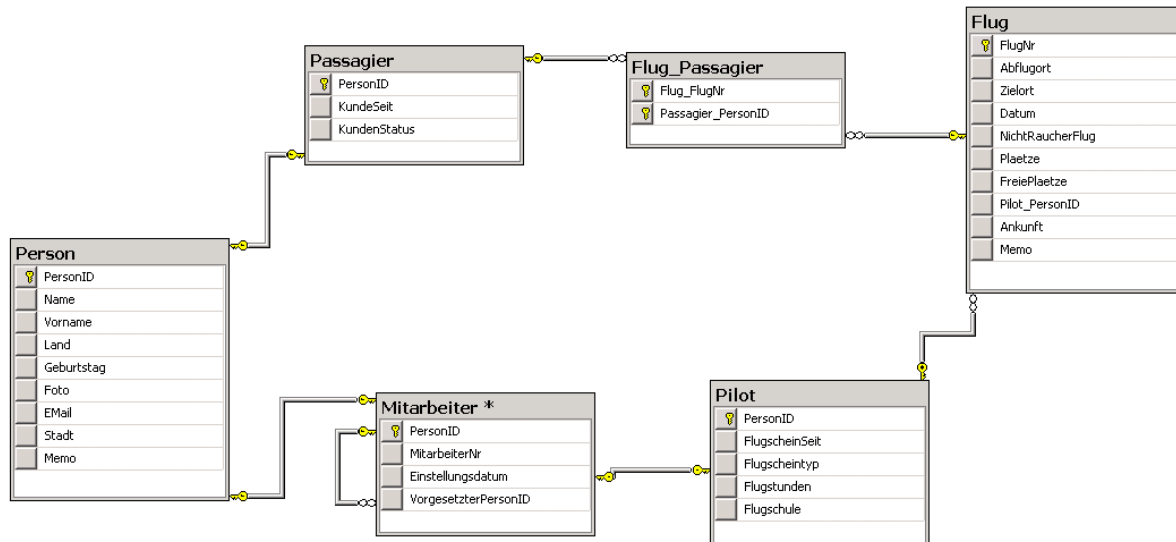


Abbildung: Neues Datenmodell des Fallbeispiels (Version 6)

HINWEIS: Die mitgelieferte Microsoft Access-Datenbank ist ein Export der Daten. Sie wird in den Beispielen aber nicht verwendet. Sie können die Beispiele aber entsprechend umbauen.

Eine Weitergabe von Löschoperationen (kaskadierendes Löschen/löschweitergabe) ist im Datenmodell nur für die 1:N-Beziehung zwischen Flug und Flug_Passagier (den Buchungen) vorgesehen. Eine Weitergabe bei Änderungen an Primärschlüsseln ist nicht vorgesehen, da die Primärschlüssel in diesem Fallbeispiel als unveränderlich gelten.

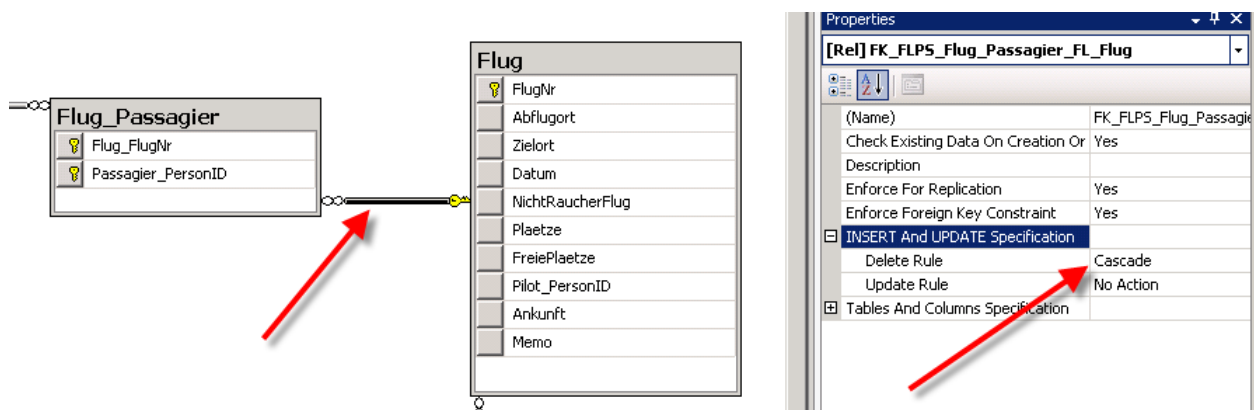


Abbildung: Löschweitergabe

5 Mehrschichtarchitektur

Das Fallbeispiel ist in vielen Teilen mehrschichtig aufgebaut: 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

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

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.

WICHTIG: An einigen Stellen wird im Fallbeispiel bewusst auf Mehrschichtigkeit ganz 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 eine Nachfrage nach solchen Rapid Application Development (RAD)-Werkzeugen besteht und Visual Studio diese Szenarien ja auch unterstützt, sind diese Werkzeuge auch in diesem Fallbeispiel enthalten – und sei es nur, um Ihnen die "Gefahren" ihrer Anwendung zu zeigen.

Das erste Bild zeigt die "alte" Architektur, die in allen Beispielen zum Einsatz kommt, die vor .NET 3.5 entstanden sind. Durchgezogene Pfeile sind prozessinterne Assembly-Referenzen. Gestrichelte Pfeile sind prozessübergreifende Kommunikationsbeziehungen. Die orangen Kästen stellen die Prozessgrenzen dar.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

Mehrschicht-Modell

WorldWideWings-Architektur

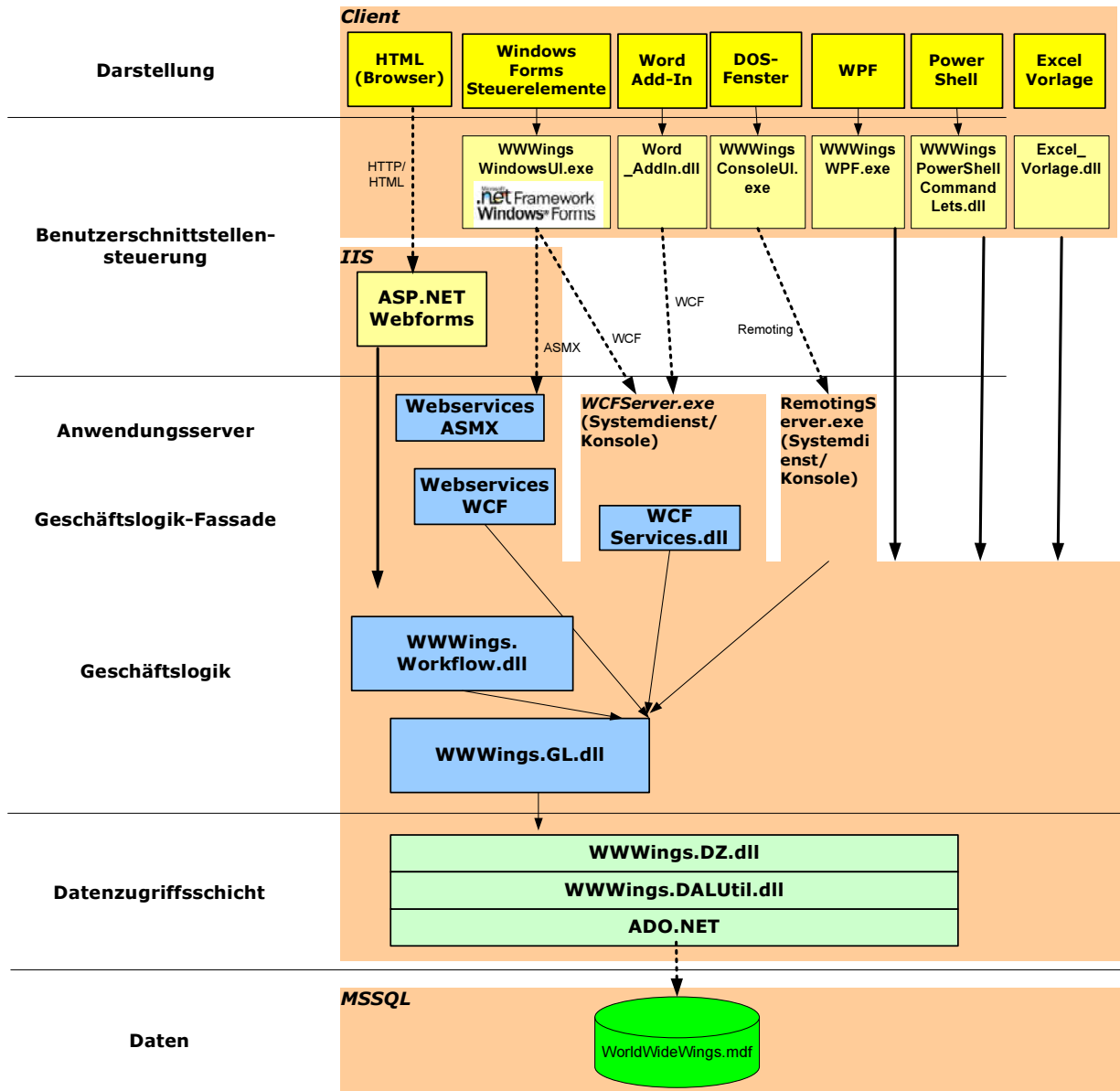


Abbildung: Gesamtarchitektur des Fallbeispiels (ältere Beispiele mit DB-Schemaversion 4)

Hinweise zu der .NET 4.0-Codebeispielsammlung "World Wide Wings"

Mehrschicht-Modell

WorldWideWings-Architektur

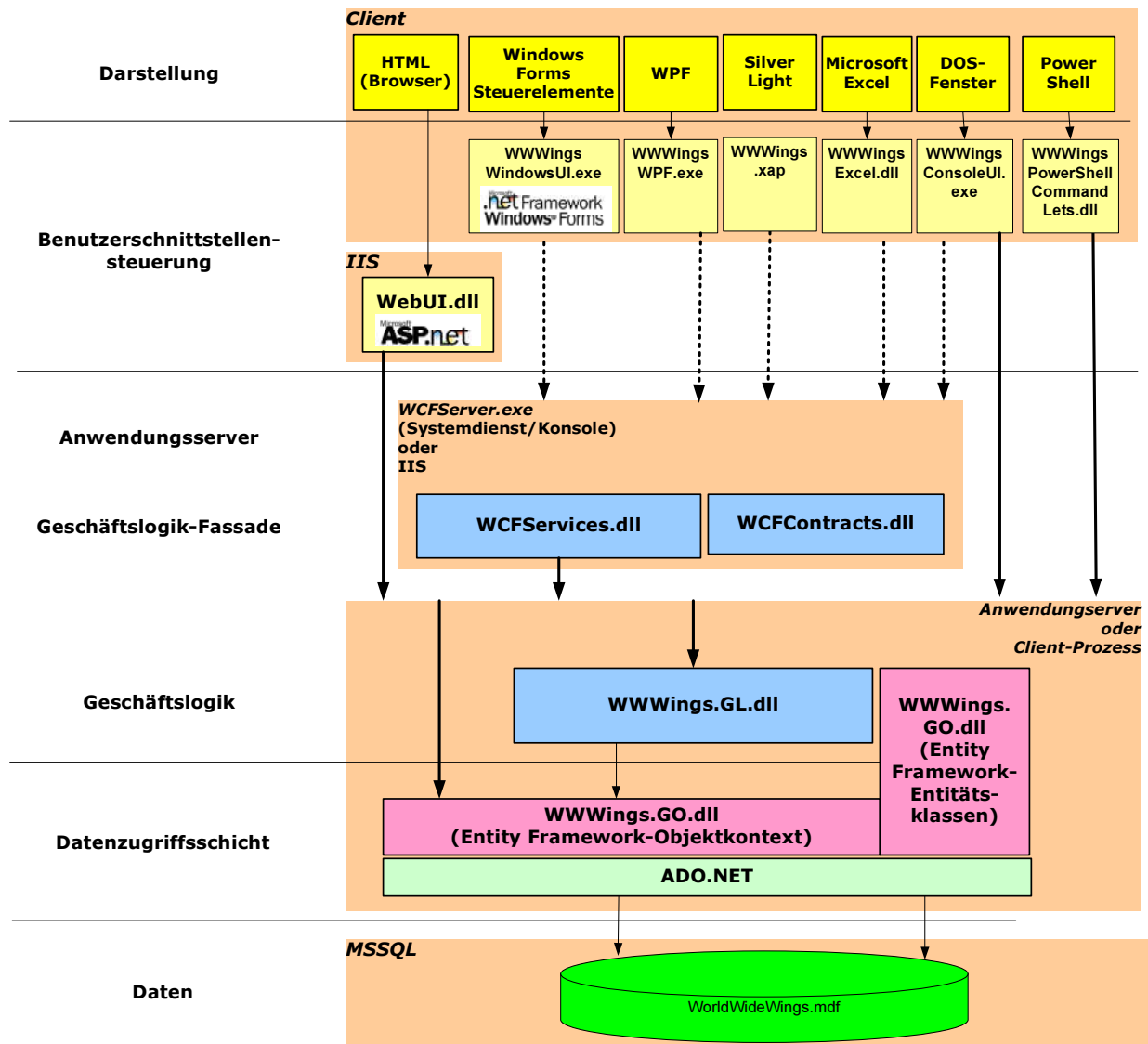


Abbildung: Gesamtarchitektur des Fallbeispiels (neuere Beispiele mit DB-Schemaversion 6)

6 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	DB-Schema	Visual Basic-Projekt	C#-Projekt	Erläuterung
DALUtil	DLL	4	ITVisions_DALUtil.vbproj	ITVisions_DALUtil.csproj	Hilfsbibliothek zur vereinfach-

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

					ten Nutzung von ADO.NET (inkl. Provider-Fabriken)
Datenzugriffsschicht (.NET 2.0/3.0-Stil ohne Entity Framework)	DLL	4	WWWings_DZS_ohneEF_VB.vbproj	WWWings_DZS_ohneEF_CS.csproj	Demonstriert den Datenzugriff mit ADO.NET und der DALUtil-Bibliothek
Geschäftslogik (.NET 2.0/3.0-Stil ohne Entity Framework)	DLL	4	WWWings_GL_ohneEF_VB.vbproj	WWWings_GL_ohneEF_CS.csproj	Demonstriert die Implementierung eines domänenspezifischen Objektmodells (inkl. Einsatz von Generics) sowie Serviceorientierung auf Basis der ADO.NET-basierten Datenzugriffsschicht
Geschäftslogik (mit ADO.NET Entity Framework)	DLL	6	Nicht verfügbar	WWWings_GL.csproj	Demonstriert die Implementierung eines domänenspezifischen Objektmodells (inkl. Einsatz von Generics) sowie Serviceorientierung auf Basis der Entity Framework-Modelle
Geschäftslogikobjekte (mit ADO.NET Entity Framework-Standardvorlage)	DLL	6	Nicht verfügbar	WWWings_GO_Standard.csproj	Entity Framework-Modell mit Standardvorlage inkl. Modell-erweiterungen
Geschäftslogikobjekte (mit ADO.NET Entity Framework-SelfTrackingEntities-Vorlage)	DLL	6	Nicht verfügbar	WWWings_GO_STE.csproj	Entity Framework-Modell mit STE-Vorlage inkl. Modellerweiterungen

Tabelle: Visual Studio-Projekte für untere Schichten

Baustein	Anwendungstyp	DB-Schema	Visual Basic-Projekt	C#-Projekt	Erläuterung
Anwendungsserver (.NET Remoting)	EXE	4	WWWings_RemotingServer	Nicht verfügbar	.NET Remoting Server für Geschäftslogik, programmiert als Windows-Systemdienst, der aber auch von der Konsole aus gestartet werden kann
Dienst-Verträge	DLL	4 + 6	Nicht verfügbar	WWWings_WCF_Contracts	Schnittstellendefinition für WCF-Dienste
WCF-Fassade	DLL	4 + 6	Nicht verfügbar	WWWings_WCFServices	WCF-Dienste für Zugang zu der Geschäftslogik, zum Hosting in IIS und eigenem WCF-Anwendungsserver.
Anwendungsserver (WCF)	EXE	4 + 6	Nicht verfügbar	WWWings_WCFServer	WCF-Server für Geschäftslogik, programmiert als Windows-Systemdienst, der aber auch von der Konsole aus gestartet werden kann
Client Proxies	DLL	4 + 6	Nicht verfügbar	WWWings_WCF_ClientProxies	Client Proxies für WCF-Dienste. Werden eingebunden in Konsolen- und Desktop-

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

					Projekte.
--	--	--	--	--	-----------

Tabelle: Visual Studio-Projekte für Anwendungsserver

Baustein	Anwen- dungsart	DB- Schema	Visual Basic-Projekt	C#-Projekt	Erläuterung
Konsolen-Client	EXE	4	WWWings_ ConsoleUI_VB.vbproj	WWWings_ ConsoleUI_CS.csproj	Einfache Testroutinen für Geschäftslogik und Datenzugriff (lokal und z.T. via .NET Remoting)
Windows-Client mit Windows Forms	EXE	4	Nicht verfügbar	WWWings_ WindowsUI_CS.csproj	Windows-Anwendung, erstellt mit Windows Forms
Windows-Steuererelemente für Windows Forms	DLL	4	WWWings_ Steuererelemente.vbproj	WWWings_ Steuererelemente.csproj	Steuerelementbibliothek für den Windows-Client
Windows-Client mit WPF	EXE	4	Nicht verfügbar	WWWings_ WPF.cs	Windows-Anwendung, erstellt mit WPF
Sammlung von kleineren Beispielen ohne Zusammenhang zur Gesamtanwendung	DLL	4	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 mit ASP.NET Webforms und ASP.NET AJAX	Mehrere DLL	4 und 6	/Web/Webforms		Web-Client, in dem gemischt die Sprachen VB und C# verwendet werden unter Einsatz von ASP.NET Webforms und ASP.NET AJAX. Gleichzeitig Host für einige ASMX- und WCF-Dienste.
Webserversteuer-elemente	DLL	4	WWWings_Web- Steuererelemente.vbproj	WWWings_ Web- Steuererelemente.csproj	Steuerelementbibliothek für den Web-Client
Web-Client mit ASP.NET Dynamic Data	Mehrere DLL	6	Nicht verfügbar	Web_DynData_EF und Web_DynData_LTS	Stammdatenpflege-Website auf Basis von ASP.NET Dynamic Data mit LINQ-to-SQL (LTS) oder ADO.NET Entity Framework (WF) im Untergrund.
SilverlightSharedLibrary	DLL	6	Nicht verfügbar	SilverlightSharedLibrary..csproj	Web- & Desktop-Client mit Microsoft Silverlight 4.0: Von Client und Server gemeinsam genutzt Bibliothek
SilverlightServices	DLL	6	Nicht verfügbar	SilverlightServices.csproj	Web- & Desktop-Client mit Microsoft Silverlight 4.0: Dienste für Silverlight
SilverlightServer	DLL	6	Nicht verfügbar	SilverlightServer.csproj	Web- & Desktop-Client mit Microsoft Silverlight 4.0: Webserver für Silverlight
SilverlightClientProxies	DLL	6	Nicht verfügbar	SilverlightClientProxies.csproj	Web- & Desktop-Client mit Microsoft Silverlight 4.0: Client Proxies für Silverlight
SilverlightClient	DLL	6	Nicht verfügbar	SilverlightClient.csproj	Web- & Desktop-Client mit Microsoft Silverlight 4.0: Silverlight-Client

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

Client für Microsoft Excel	XSL/DLL	4	WWWings_Office_Client.vbproj	Nicht verfügbar	VSTO-Dokument für Microsoft Excel, die Flugdaten aus dem World Wide Wings-Beispiel ausliest
Clients für Microsoft Word	DOC/DLL	4	Nicht verfügbar	OfficeWord_AddIn und OfficeWord_Vorlage	VSTO-Add-In bzw. Vorlage für Microsoft Word, das Flugdaten aus dem World Wide Wings-Beispiel ausliest und in das aktuelle Dokument einfügt
PowerShell Commandlets	DLL	4	WWWings_PowerShell_Commandlets	Nicht verfügbar	Commandlets für die Windows-PowerShell für die WWWings-Geschäftslogik: Get-Flug, Get-Passagier, New-Buchung, etc.

Tabelle: Visual Studio-Projekte für Client/Oberflächen

Baustein	Anwendungsart	DB-Schema	Visual Basic-Projekt	C#-Projekt	Erläuterung
Unit Tests	DLL	4 und 6	Nicht verfügbar	WebUI_Tests.csproj	VSTS-Testprojekt
MVC Unit Tests	DLL	4 und 6	Nicht verfügbar	WWWings_UnitTests.csproj	VSTS-Testprojekt für MVC-Client

Tabelle: Test-Projekte

Baustein	Anwendungsart	DB-Schema	Visual Basic-Projekt	C#-Projekt	Erläuterung
VisualStudioInstallerSetup	DLL	n/a	Nicht verfügbar	VisualStudioInstallerSetup.vdproj	Setup-Projekt für WPF-Anwendung
SilverlightClickOnce	DLL	n/a	Nicht verfügbar	WWWings_UnitTests.csproj	Click Once-Installation-Routine für Silverlight-Anwendung

Tabelle: Setup-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 – die Übersetzung dauert daher hier am längsten. Es wird empfohlen die spezifischen Projektmappen und nicht "WWWings_AlleProjekte.sln" zu verwenden.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

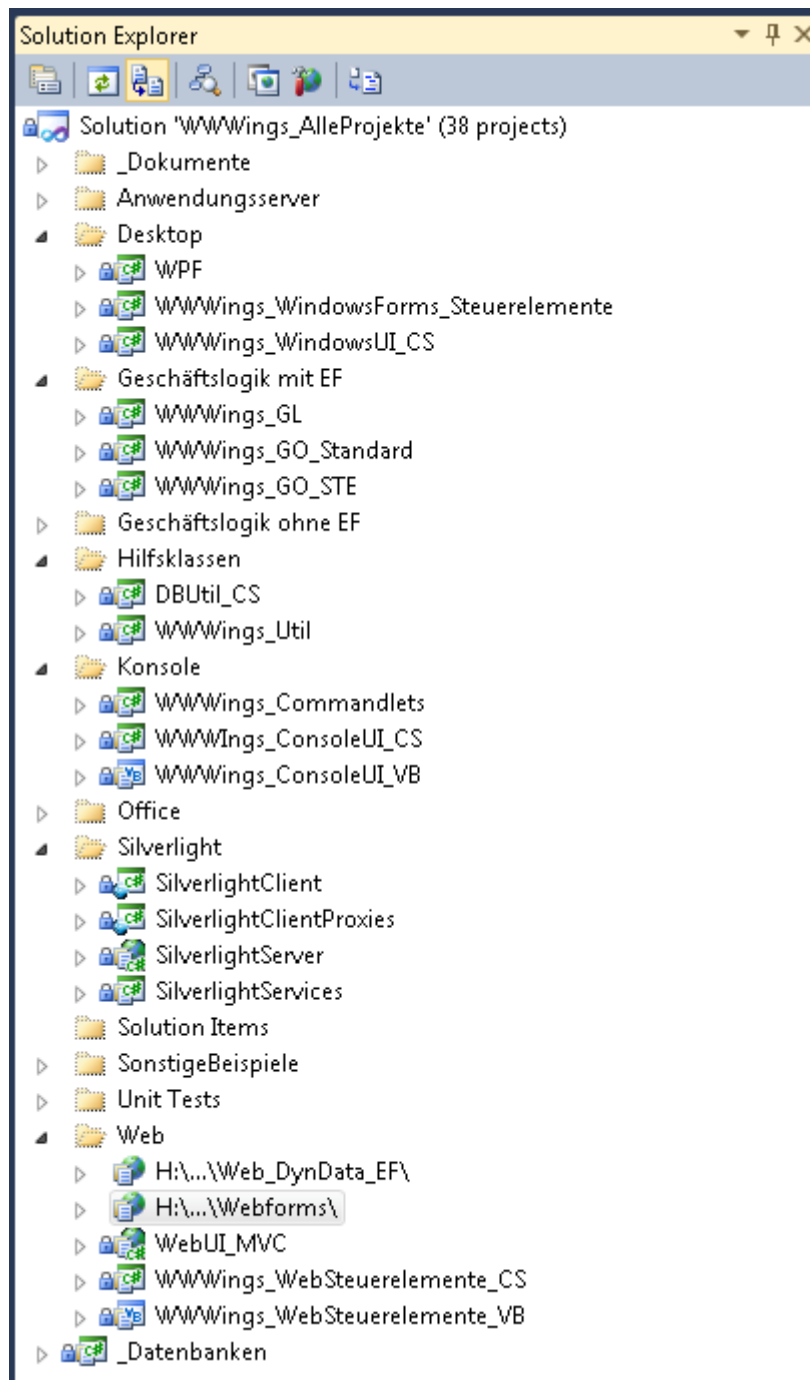


Abbildung: Projektmappe mit allen Projekten

7 Funktionsüberblick

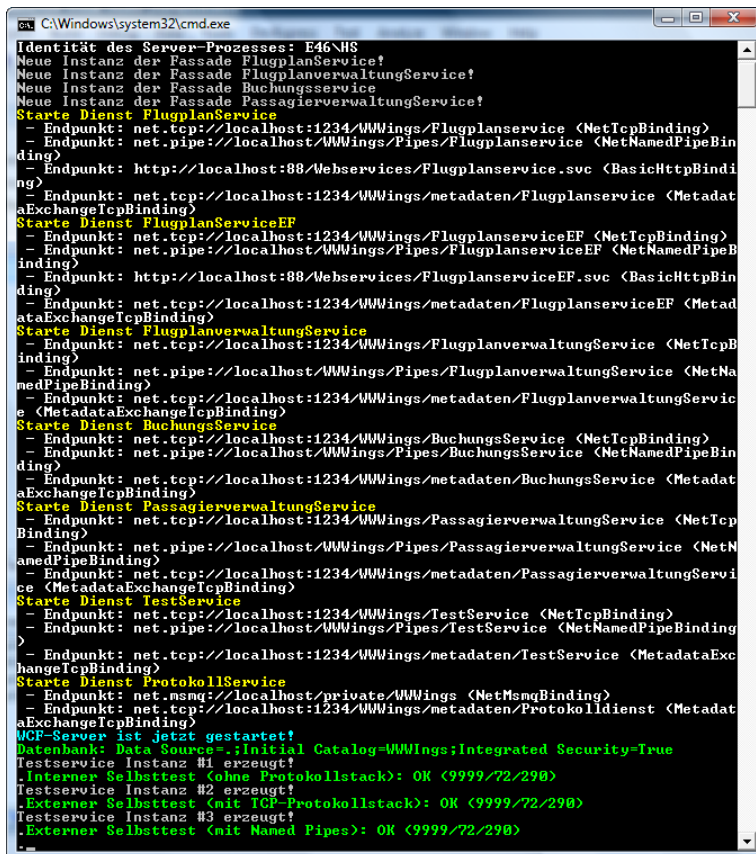
Dieses Kapitel gibt einen Überblick über die Funktionen, die das World Wide Wings-Fallbeispiel anbietet.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

7.1 World Wide Wings-Anwendungsserver

Es gibt mehrere Anwendungsserver in World Wide Wings:

- WCF-Anwendungsserver (siehe Bildschirmabbildung): Stellt Dienste per TCP, Named Pipes und HTTP bereit. Ein eigenständiges Projekt. Läuft als Konsolenanwendung oder Windows-Systemdienst.
- IIS-Anwendungsserver: Bereitstellen von einigen HTTP-basierten Webservices über ASP.NET Webservices (ASMX) und WCF. Teil des ASP.NET Webforms-Projekts. Gehostet im Visual Studio-Entwicklungswebserver oder den Internet Information Services (IIS).
- .NET Remoting-Anwendungsserver. Ein eigenständiges Projekt. Läuft als Konsolenanwendung oder Windows-Systemdienst.



```
CA\Windows\system32\cmd.exe
Identität des Server-Prozesses: E46\HS
Neue Instanz der Fassade FlugplanService?
Neue Instanz der Fassade FlugplanverwaltungService?
Neue Instanz der Fassade BuchungsService
Neue Instanz der Fassade PassagierverwaltungService?
Starte Dienst FlugplanService
- Endpunkt: net.tcp://localhost:1234/WWings/FlugplanService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/FlugplanService <NetNamedPipeBinding>
- Endpunkt: http://localhost:88/Webservices/FlugplanService.svc <BasicHttpBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/netdaten/FlugplanService <MetadataExchangeIcpBinding>
Starte Dienst FlugplanServiceEF
- Endpunkt: net.tcp://localhost:1234/WWings/FlugplanServiceEF <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/FlugplanServiceEF <NetNamedPipeBinding>
- Endpunkt: http://localhost:88/Webservices/FlugplanServiceEF.svc <BasicHttpBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/netdaten/FlugplanServiceEF <MetadataExchangeIcpBinding>
Starte Dienst FlugplanverwaltungService
- Endpunkt: net.tcp://localhost:1234/WWings/FlugplanverwaltungService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/FlugplanverwaltungService <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/netdaten/FlugplanverwaltungService <MetadataExchangeIcpBinding>
Starte Dienst BuchungsService
- Endpunkt: net.tcp://localhost:1234/WWings/BuchungsService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/BuchungsService <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/netdaten/BuchungsService <MetadataExchangeIcpBinding>
Starte Dienst PassagierverwaltungService
- Endpunkt: net.tcp://localhost:1234/WWings/PassagierverwaltungService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/PassagierverwaltungService <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/netdaten/PassagierverwaltungService <MetadataExchangeIcpBinding>
Starte Dienst TestService
- Endpunkt: net.tcp://localhost:1234/WWings/TestService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/TestService <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/netdaten/TestService <MetadataExchangeIcpBinding>
Starte Dienst ProtokollService
- Endpunkt: net.msmq://localhost/private/WWings <NetMsmqBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/netdaten/Protokolldienst <MetadataExchangeIcpBinding>
WCF-Server ist jetzt gestartet!
Datenbank: Data Source=.;Initial Catalog=WWings;Integrated Security=True
Testservice Instanz #1 erzeugt!
Interne Selbsttest (ohne Protokollstack): OK (9999/72/290)
Testservice Instanz #2 erzeugt!
Externe Selbsttest (mit TCP-Protokollstack): OK (9999/72/290)
Testservice Instanz #3 erzeugt!
Externe Selbsttest (mit Named Pipes): OK (9999/72/290)
```

Abbildung: Start des Anwendungsservers an der Konsole

7.2 Funktionen der World Wide Wings-Hauptwebanwendung

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

7.2.1 Anwendungsfälle

Diese Beispielanwendung implementiert eine grafische Oberfläche sowohl für Endkunden als auch für Verwal-

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

tungsmitarbeiter 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)
- Registrieren eines Benutzers
- Kundenbereich, der nur für registrierte Nutzer zugänglich ist
- 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 (über ASMX und WCF)

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

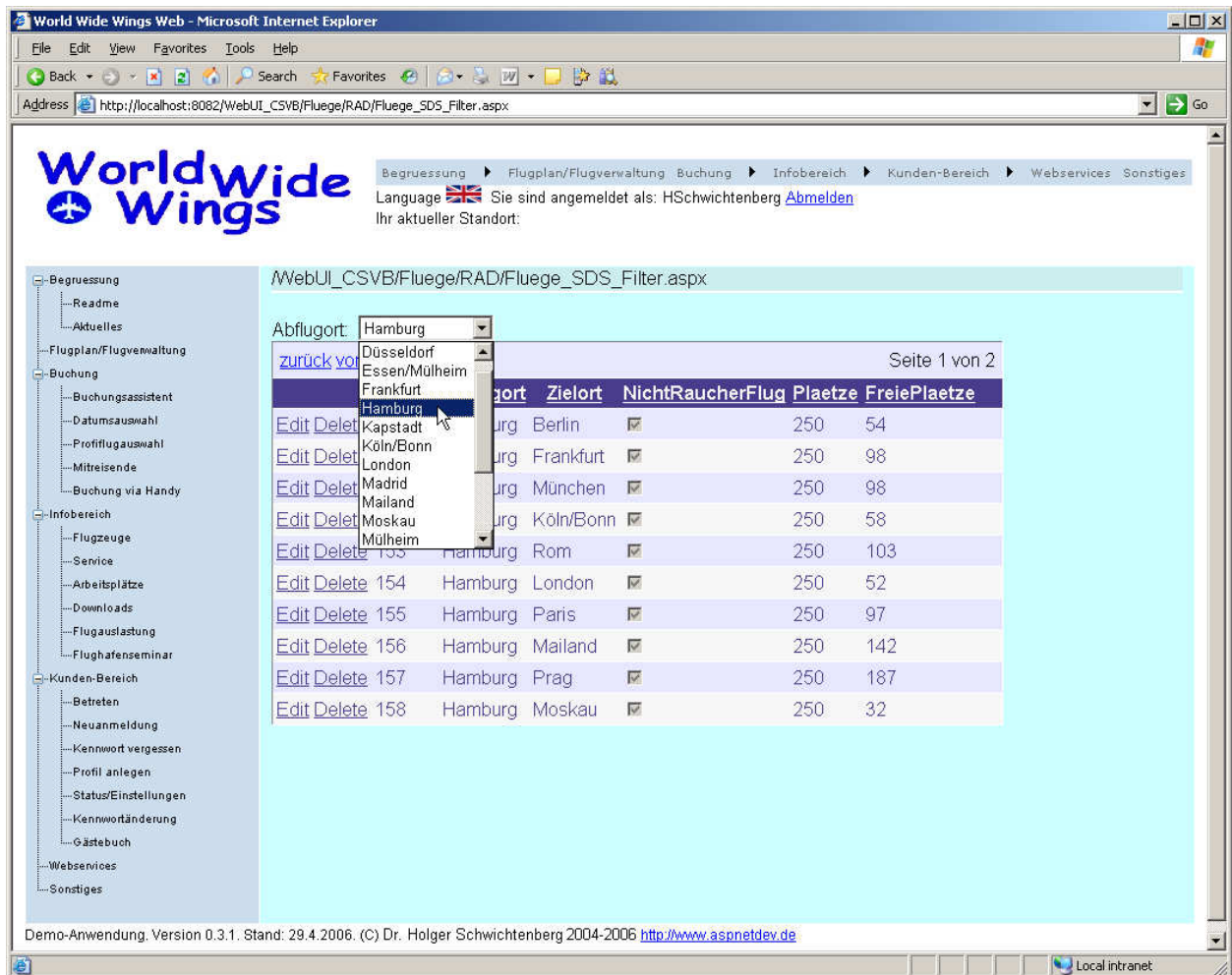


Abbildung: Beispielanwendung (WorldWideWings-Web)

7.2.2 Gezeigte ASP.NET-Funktionen

Insbesondere folgende Möglichkeiten von ASP.NET Webforms 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)
- Masterpages, Themes, Skins
- Sitemaps
- Seitenübergänge (Redirect, Transfer, Cross Page Postbacks)
- Zustandsverwaltung (Session, Viewstate, Application)
- Formularbasierte Authentifizierung

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

- 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 Client Callbacks
- ASP.NET AJAX Extensions (Partielle Seitenerzeugung und AJAX Services)
- Konfiguration durch web.config
- Einsatz der mobilen Steuerelemente (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) und WCF-Webservices
- 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.

7.2.3 Architektur

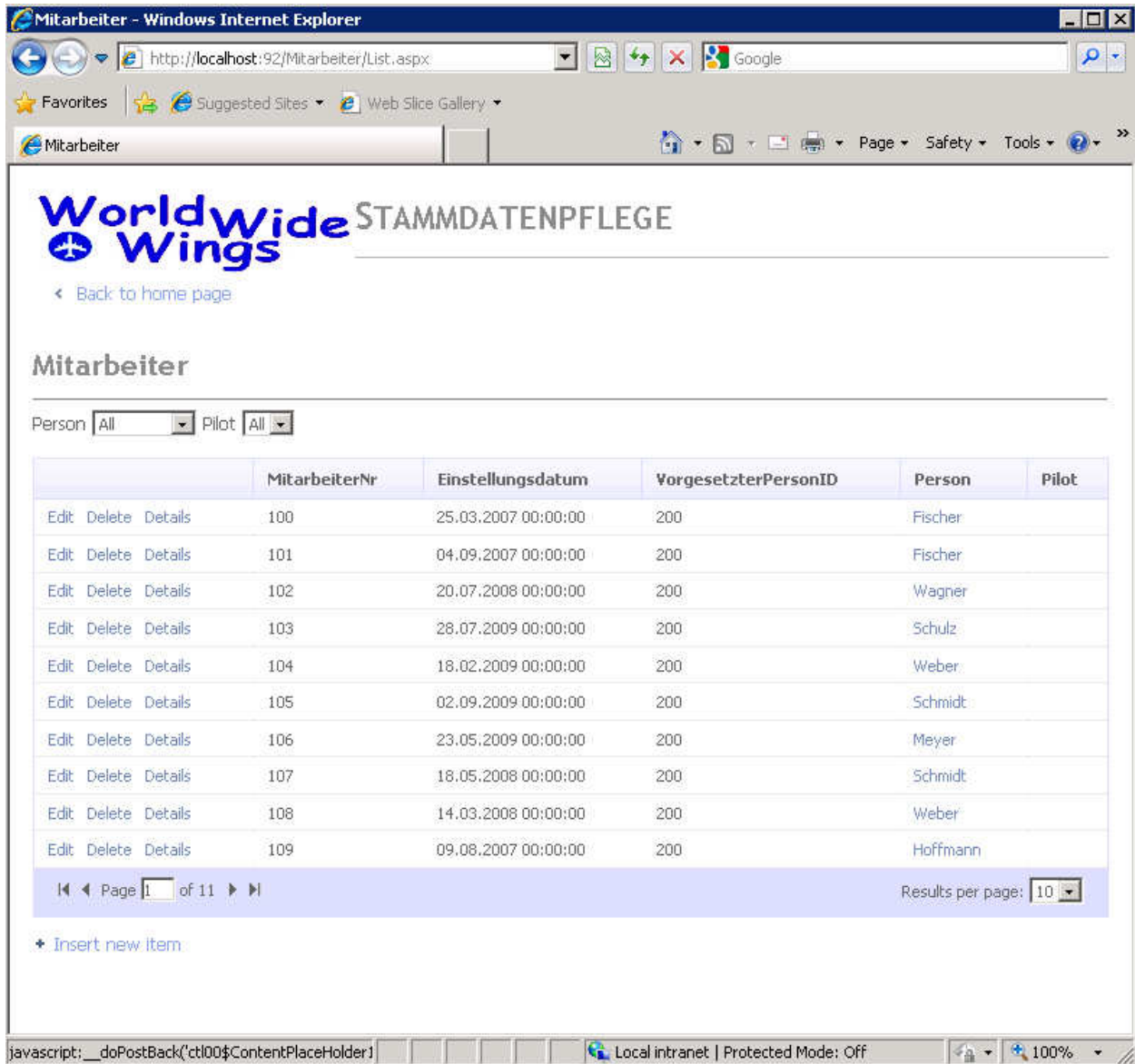
Die Webanwendung verwendet in der aktuellen Version direkt die Geschäftslogik und die Datenzugriffsschicht im Prozess des Webserver. Ein Anwendungsserver wird nicht eingesetzt. Dies könnte aber leicht geändert werden.

Die Webanwendung kann wahlweise über den in Visual Studio integrierten Webserver als auch den IIS gestartet werden. Sie müssen die Konfiguration für den IIS allerdings manuell vornehmen (vgl. Bücher des Autors zu ASP.NET).

7.3 Funktionen der World Wide Wings-Stammdaten-Webanwendung

Die Stammdatenwebanwendung ist mit Hilfe von ASP.NET Dynamic Data realisiert und dient der einfachen Pflege der Stammdaten.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



World Wide Wings STAMMDATENPFLEGE

[Back to home page](#)

Mitarbeiter

Person Pilot

	MitarbeiterNr	Einstellungsdatum	VorgesetzterPersonID	Person	Pilot
Edit Delete Details	100	25.03.2007 00:00:00	200	Fischer	
Edit Delete Details	101	04.09.2007 00:00:00	200	Fischer	
Edit Delete Details	102	20.07.2008 00:00:00	200	Wagner	
Edit Delete Details	103	28.07.2009 00:00:00	200	Schulz	
Edit Delete Details	104	18.02.2009 00:00:00	200	Weber	
Edit Delete Details	105	02.09.2009 00:00:00	200	Schmidt	
Edit Delete Details	106	23.05.2009 00:00:00	200	Meyer	
Edit Delete Details	107	18.05.2008 00:00:00	200	Schmidt	
Edit Delete Details	108	14.03.2008 00:00:00	200	Weber	
Edit Delete Details	109	09.08.2007 00:00:00	200	Hoffmann	

Page 1 of 11 Results per page: 10

[+ Insert new item](#)

7.4 Funktionen der World Wide Wings-Silverlight-Anwendung

Die Silverlight-Anwendung erlaubt bisher den Abruf und die Änderung von Flugdaten. Die Bearbeitung der Daten ist im Out-Of-Browser-Betrieb auch mit Microsoft Excel möglich. Außerdem gibt es Druckfunktionen und die Möglichkeit zum lokalen Zwischenspeichern.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

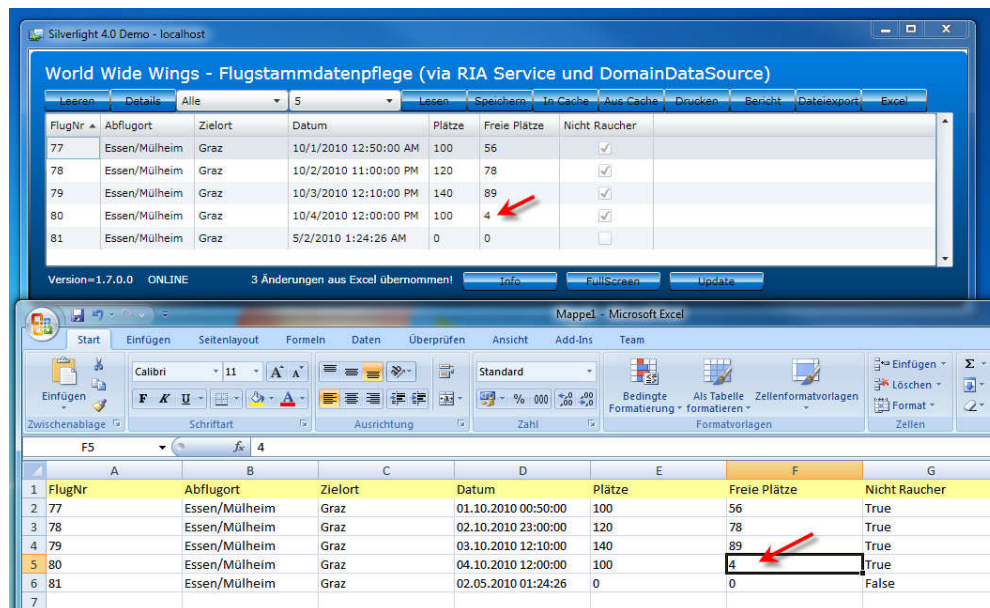


Abbildung: Beispiel aus der Silverlight-Anwendung mit Kopplung an Microsoft Excel (wahlweise im Browser oder außerhalb des Browsers)

7.5 Funktionen der World Wide Wings-Desktopanwendung (Windows Forms)

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

7.5.1 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 der .NET 4.0- Codebeispielsammlung "World Wide Wings"

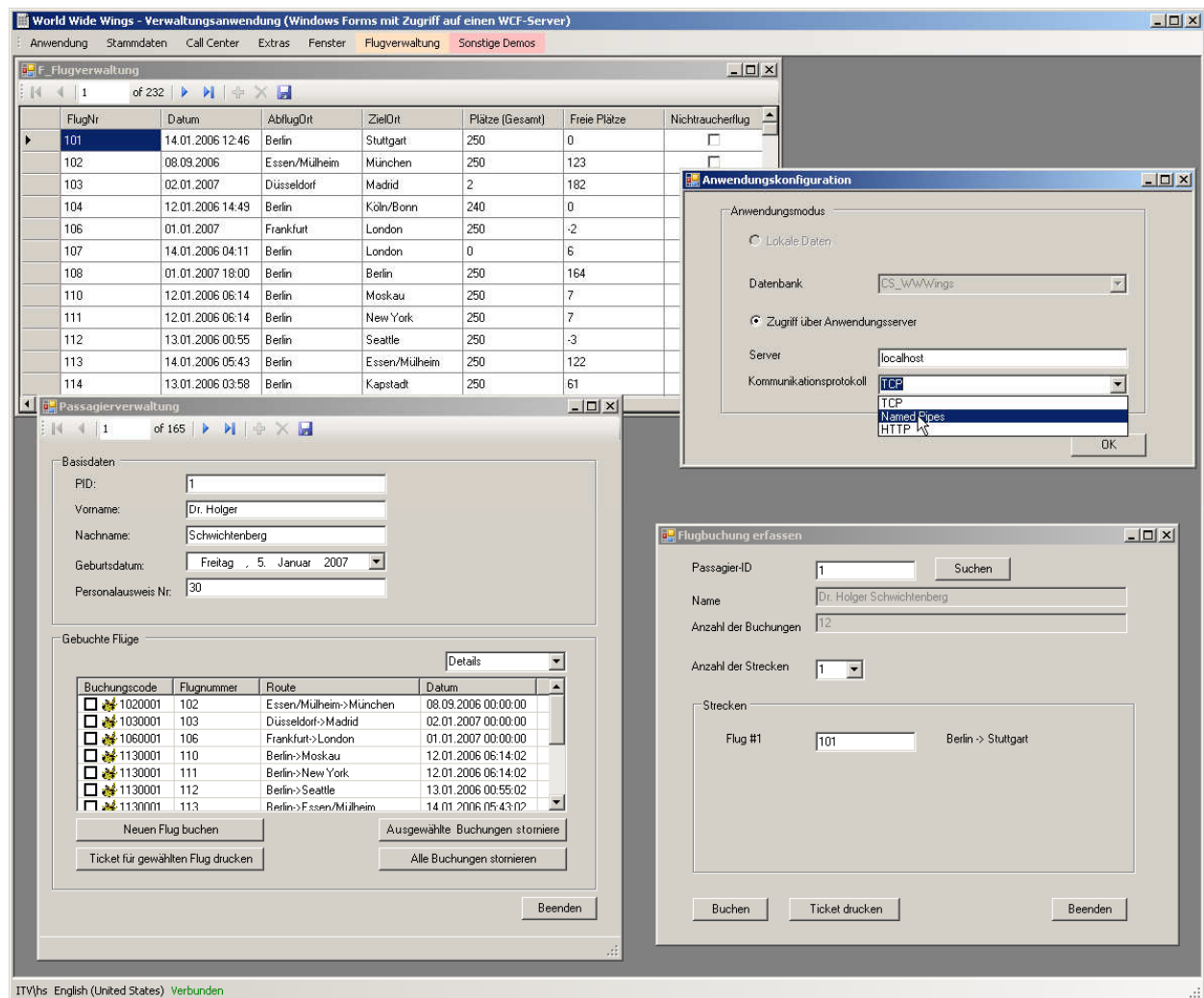


Abbildung: Die Desktop-Anwendung mit Multi-Document-Interface (MDI)

7.5.2 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 der .NET 4.0- Codebeispielsammlung "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.

7.5.3 Architektur

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

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

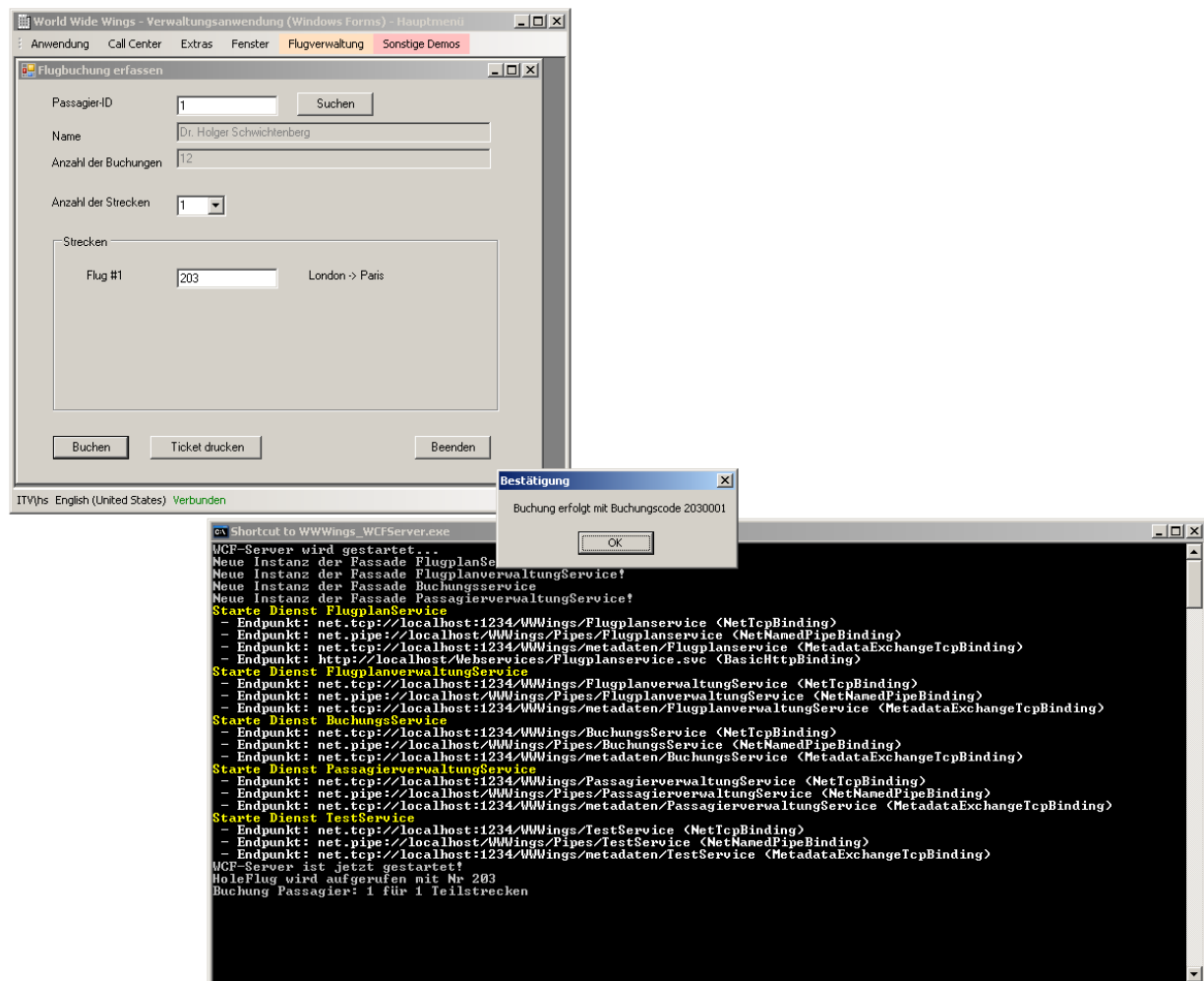


Abbildung: Die Windows Forms-Anwendung zusammen mit dem Protokollfenster des Anwendungsservers, der hier als Konsolenanwendung läuft

HINWEIS: Der Buchungsassistent basiert auf der Windows Workflow Foundation (WF) - allerdings ohne Einsatz von WCF. Für den Buchungsassistent ist daher 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 Anwendungsconfigurationsdatei.

7.5.4 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 der .NET 4.0- Codebeispielsammlung "World Wide Wings"

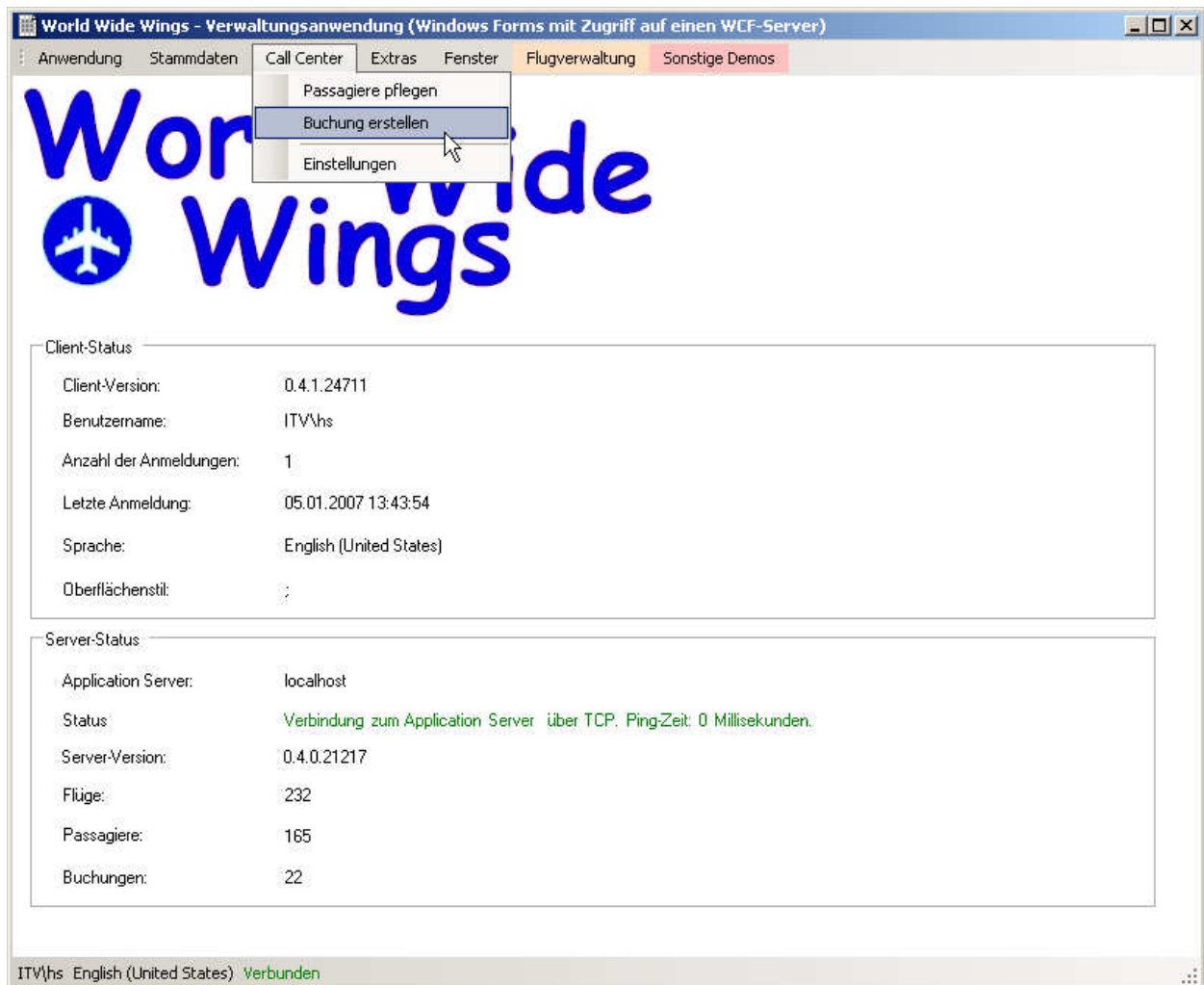


Abbildung: Hauptmenü der Windows Forms-Anwendung für World Wide Wings

7.5.5 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 der .NET 4.0- Codebeispielsammlung "World Wide Wings"

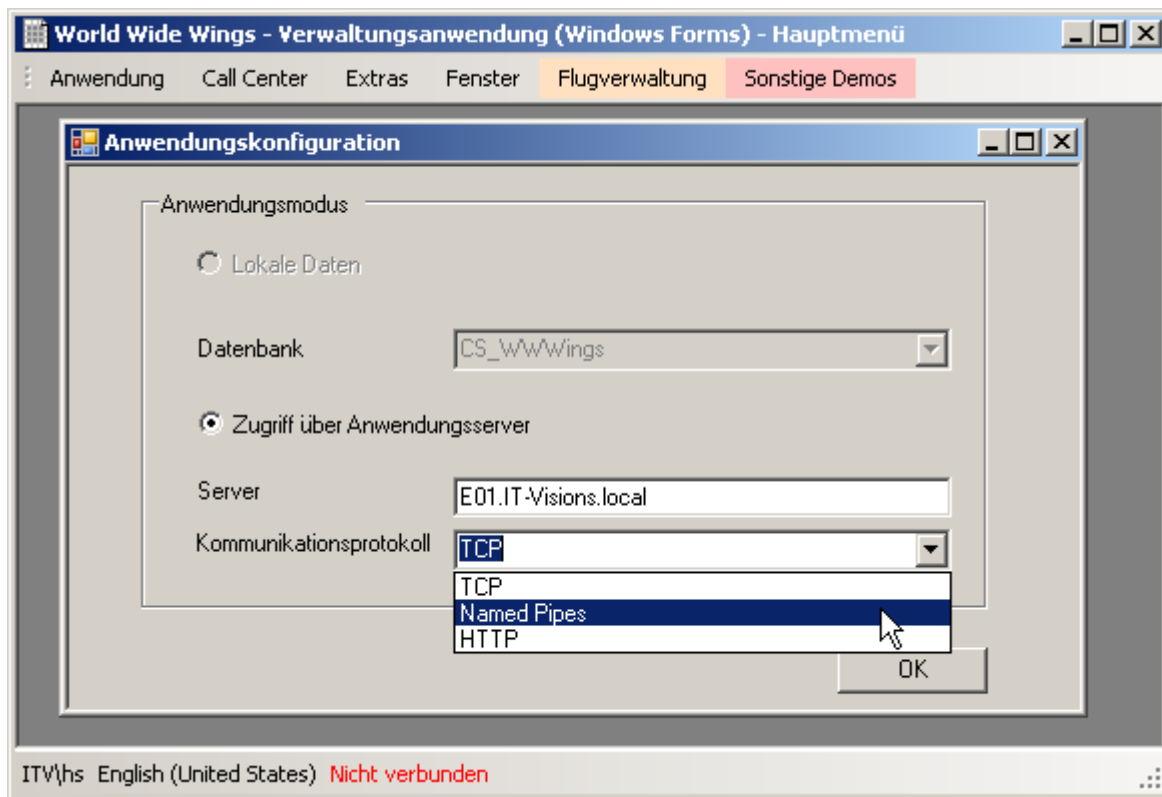
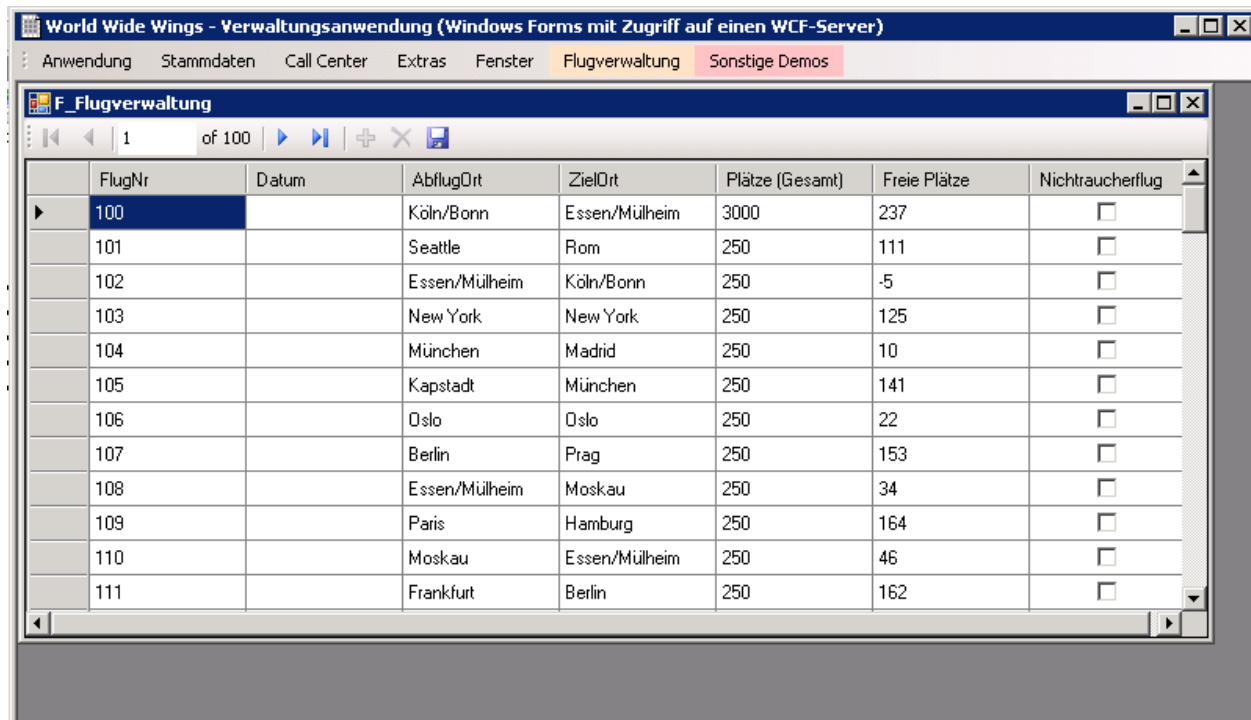


Abbildung: Konfiguration des Anwendungsservers

7.5.6 Stammdatenverwaltung

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

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



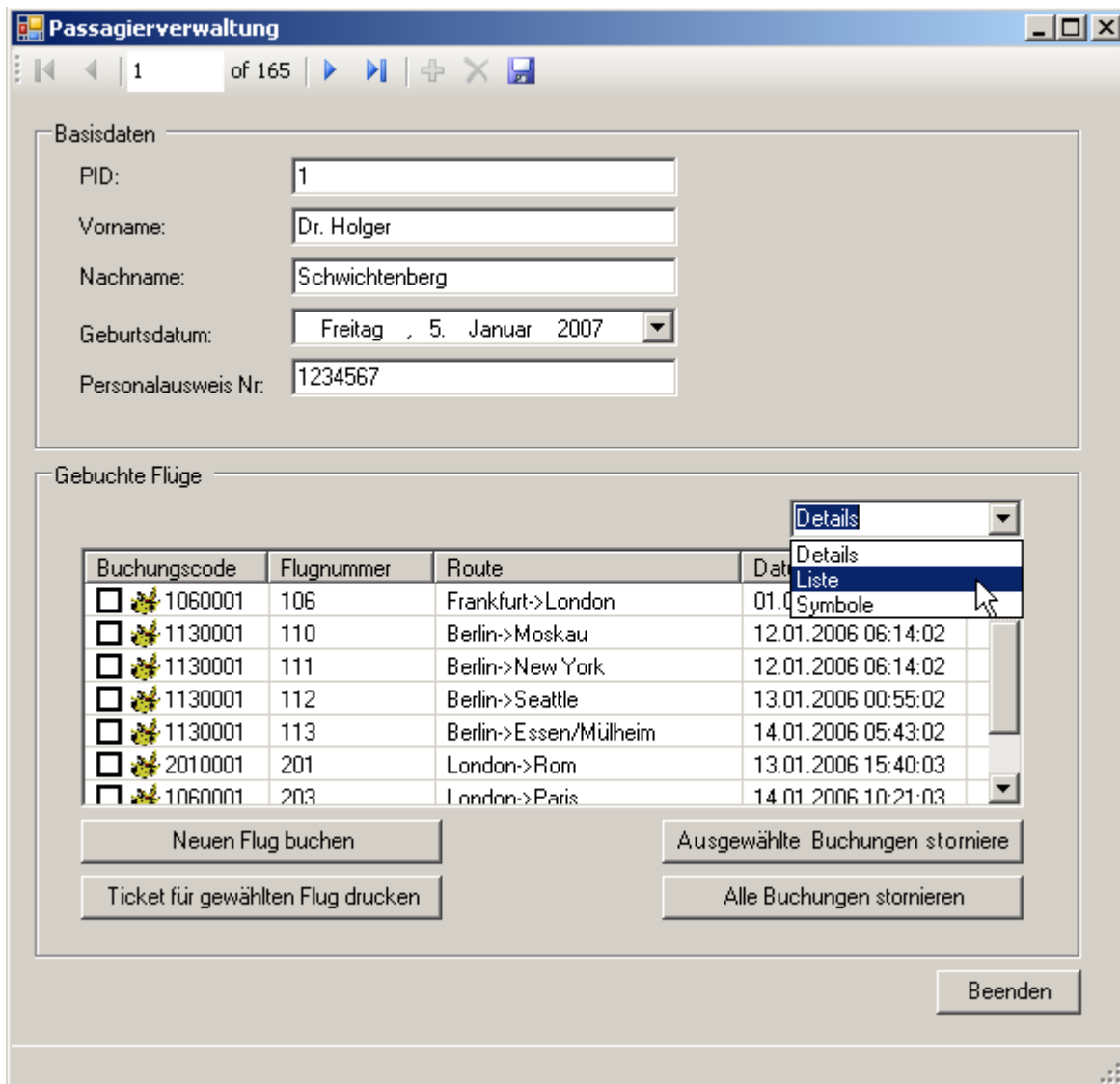
FlugNr	Datum	AbflugOrt	ZielOrt	Plätze (Gesamt)	Freie Plätze	Nichtraucherflug
100		Köln/Bonn	Essen/Mülheim	3000	237	<input type="checkbox"/>
101		Seattle	Rom	250	111	<input type="checkbox"/>
102		Essen/Mülheim	Köln/Bonn	250	-5	<input type="checkbox"/>
103		New York	New York	250	125	<input type="checkbox"/>
104		München	Madrid	250	10	<input type="checkbox"/>
105		Kapstadt	München	250	141	<input type="checkbox"/>
106		Oslo	Oslo	250	22	<input type="checkbox"/>
107		Berlin	Prag	250	153	<input type="checkbox"/>
108		Essen/Mülheim	Moskau	250	34	<input type="checkbox"/>
109		Paris	Hamburg	250	164	<input type="checkbox"/>
110		Moskau	Essen/Mülheim	250	46	<input type="checkbox"/>
111		Frankfurt	Berlin	250	162	<input type="checkbox"/>

Abbildung: Verwaltung der Flüge

7.5.7 Call Center-Anwendung

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

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



The screenshot shows a Windows application titled "Passagierverwaltung". It has a standard Windows interface with a title bar, menu bar, and toolbar. The main window is divided into two sections: "Basisdaten" (Basic Data) and "Gebuchte Flüge" (Booked Flights).

Basisdaten:

- PID: 1
- Vorname: Dr. Holger
- Nachname: Schwichtenberg
- Geburtsdatum: Freitag, 5. Januar 2007
- Personalausweis Nr.: 1234567

Gebuchte Flüge:

A table lists booked flights with columns: Buchungscode, Flugnummer, Route, and Date. A context menu is open over the table, showing options: Details, Liste, and Symbole. Below the table are four buttons: "Neuen Flug buchen", "Ausgewählte Buchungen storniere", "Ticket für gewählten Flug drucken", and "Alle Buchungen stornieren". A "Beenden" button is at the bottom right.

Buchungscode	Flugnummer	Route	Date
<input type="checkbox"/> 1060001	106	Frankfurt->London	01.01.2006 06:14:02
<input type="checkbox"/> 1130001	110	Berlin->Moskau	12.01.2006 06:14:02
<input type="checkbox"/> 1130001	111	Berlin->New York	12.01.2006 06:14:02
<input type="checkbox"/> 1130001	112	Berlin->Seattle	13.01.2006 00:55:02
<input type="checkbox"/> 1130001	113	Berlin->Essen/Mülheim	14.01.2006 05:43:02
<input type="checkbox"/> 2010001	201	London->Rom	13.01.2006 15:40:03
<input type="checkbox"/> 1060001	203	London->Paris	14.01.2006 10:21:03

Abbildung: Verwaltung der Passagiere

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

The screenshot shows a Windows application window titled "Flugbuchung erfassen". It contains several input fields and buttons. At the top, there is a "Passagier-ID" field with the value "1" and a "Suchen" button. Below this is a "Name" field with the text "Dr. Holger Schwichtenberg". The "Anzahl der Buchungen" field contains the value "10". The "Anzahl der Strecken" field is a dropdown menu currently showing "3". Below these fields is a section titled "Strecken" which contains three rows of flight details. Each row has a "Flug #" field and a corresponding route description. At the bottom of the window, there are three buttons: "Buchen" (highlighted with a mouse cursor), "Ticket drucken", and "Beenden".

Strecken	
Flug #1	313 Essen/Mülheim -> Rom
Flug #2	191 Rom -> New York
Flug #3	290 New York -> Kapstadt

Abbildung: Erstellung einer neuen Buchung mit einer einfachen Maske

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

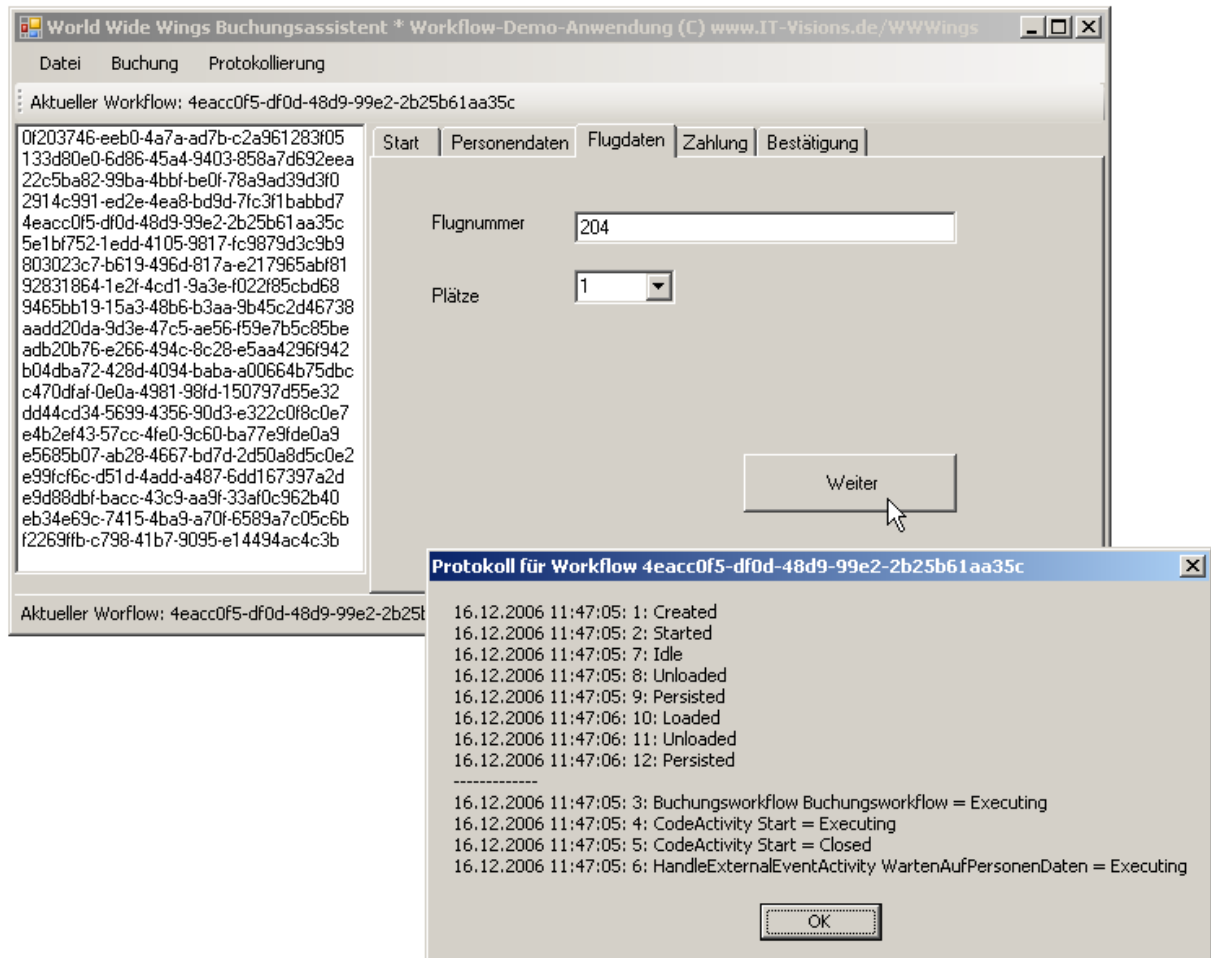


Abbildung: Der Buchungsassistent verwendet Windows Workflow Foundation

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

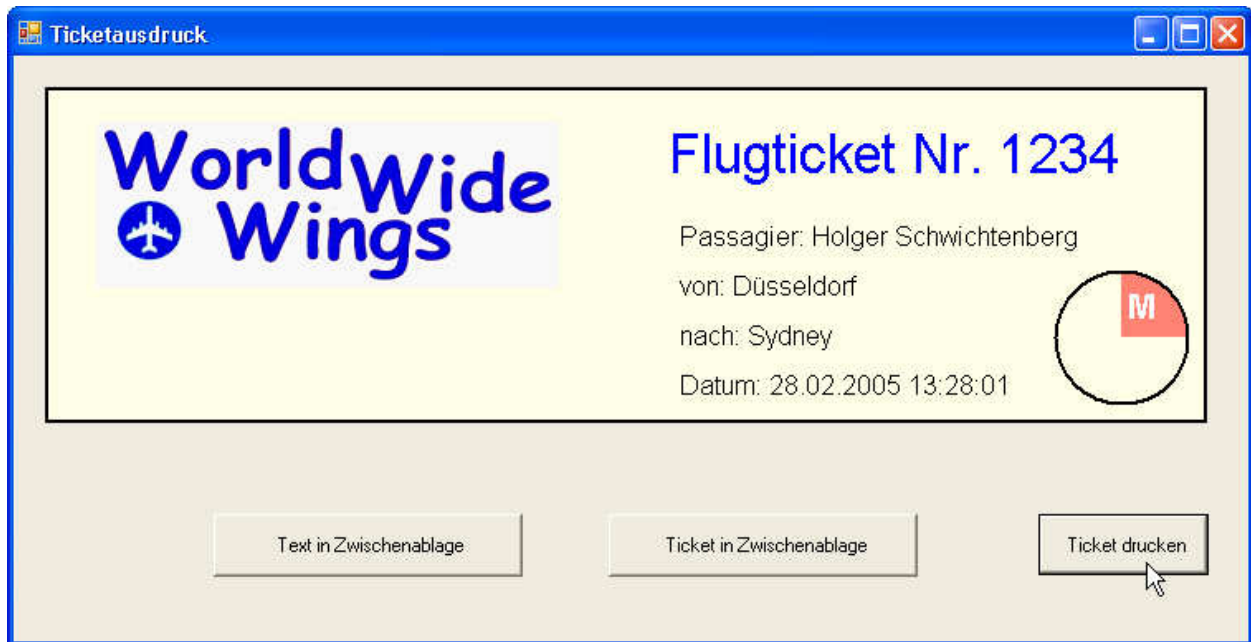


Abbildung: Flugticket (Bildschirmanzeige)

In der Call Center-Teilapplenung findet man u.a. folgende Anwendungsbeispiele für Windows Forms-
Steuerelemente 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 der .NET 4.0- Codebeispielsammlung "World Wide Wings"

	Box, Button, StatusStrip		AlleBuchungenLoeschenFuerPassagier() BuchungsService.HoleBuchungen()
F_Buchung.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

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

7.6 Funktionen der World Wide Wings-Desktopanwendung (WPF)

Auch in diesem Kapitel dient eine Anwendung aus World Wide Wings als Beispiel, das eine Vielzahl von Funktionen im Kontext aufzeigt. Die WPF-Anwendung für die World Wide Wings-Fluggesellschaft besteht derzeit aus einem Rahmenfenster [MainWindows.xaml] mit zwei Masken:

- Eine Bildschirmmaske zur Flugbuchung [F_Buchung.xaml]
- Eine Bildschirmmaske zur Flugverwaltung [F_Flugverwaltung.xaml]

Das Rahmenfenster demonstriert folgende Funktionen:

- Fenster fester Größen mit MinHeight/MaxHeight/MinWidth/MaxWidth oder ResizeMode="NoResize"
- WPF-Ribbon-Steuerelement mit Tabs und Anwendungsmenü
- Klassisches Menü
- Umschalten zwischen Ribbon und klassischem Menü

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

- Befehlsbindung mit Deaktivieren von Funktionen
- Umschalten zwischen selbstdefinierten Steuerelementen im Inhaltsbereich (Die beiden Masken sind als User Controls realisiert!)
- Integration mit der Windows 7-Taskbar

Die Bildschirmmaske zur Flugbuchung demonstriert folgende Funktionen:

- WPF-Anwendungsobjekt und WPF-Fenster
- Layout-Elemente DockPanel, StackPanel und Grid
- Kontextmenüs
- Grundsteuerelemente Label, Button, TextBox, ComboBox, ListBox, CheckBox
- Datenbindung an Geschäftsobjekte
- Zeichnen (Trennlinien)
- Formatvorlagen
- Eigene Darstellung für Steuerelemente (runde Schaltflächen)
- Transformation (Querstellung des Auswahlfeldes)
- Animation (Bewegung der *Buchen*-Schaltfläche beim Klicken)
- Ereignisbindung
- Befehlsbindung

Die Bildschirmmaske zur Flugverwaltung, die man aus dem Menü "Anwendung" der Flugbuchung heraus aufrufen kann, demonstriert folgende Funktionen:

- WPF-Anwendungsobjekt und WPF-Fenster
- Layout-Elemente DockPanel, StackPanel und Grid
- DataGrid-Steuerelement
- Grundsteuerelemente ComboBox, Button, StatusBar und DatePicker
- Datenbindung mit der CollectionViewSource

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

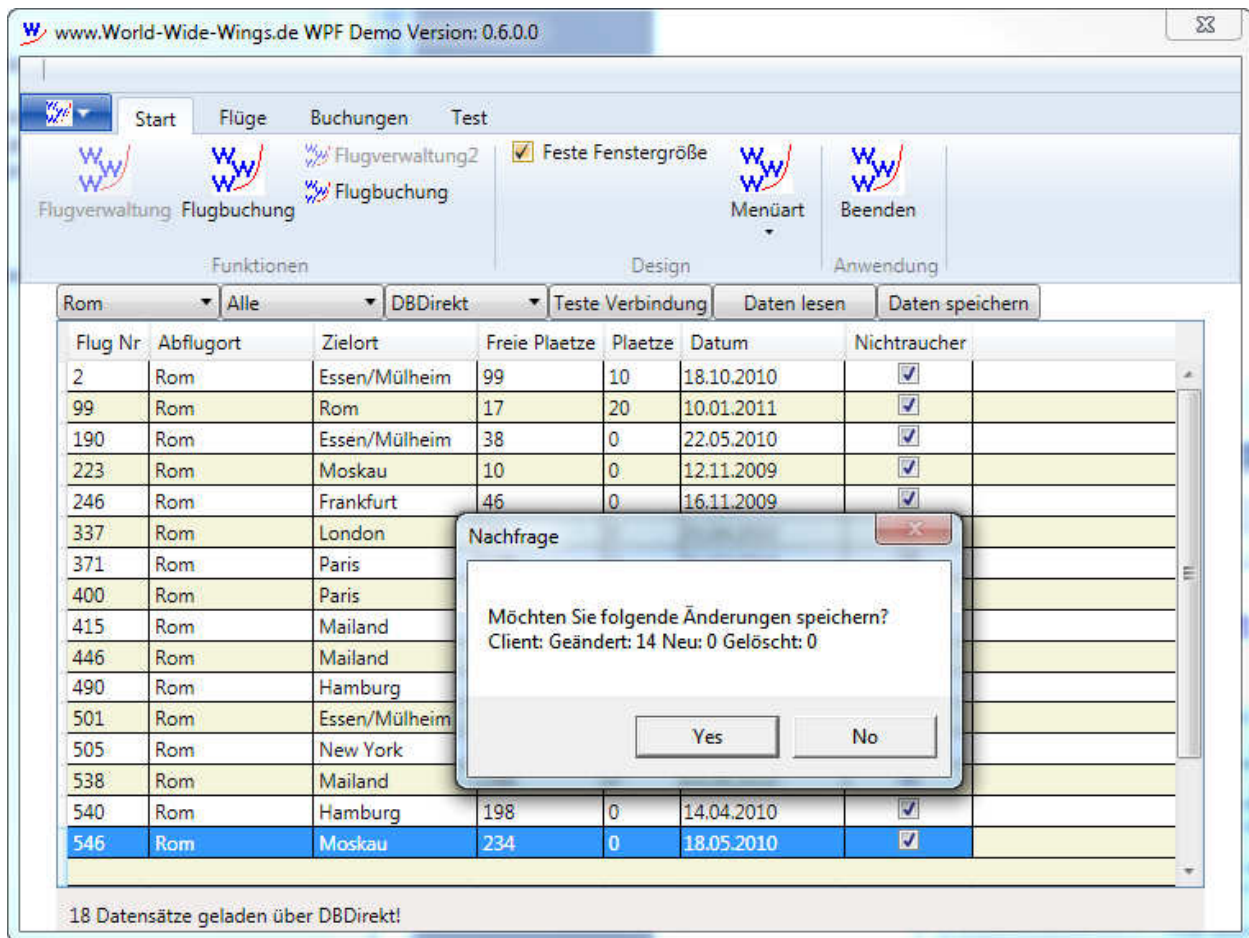
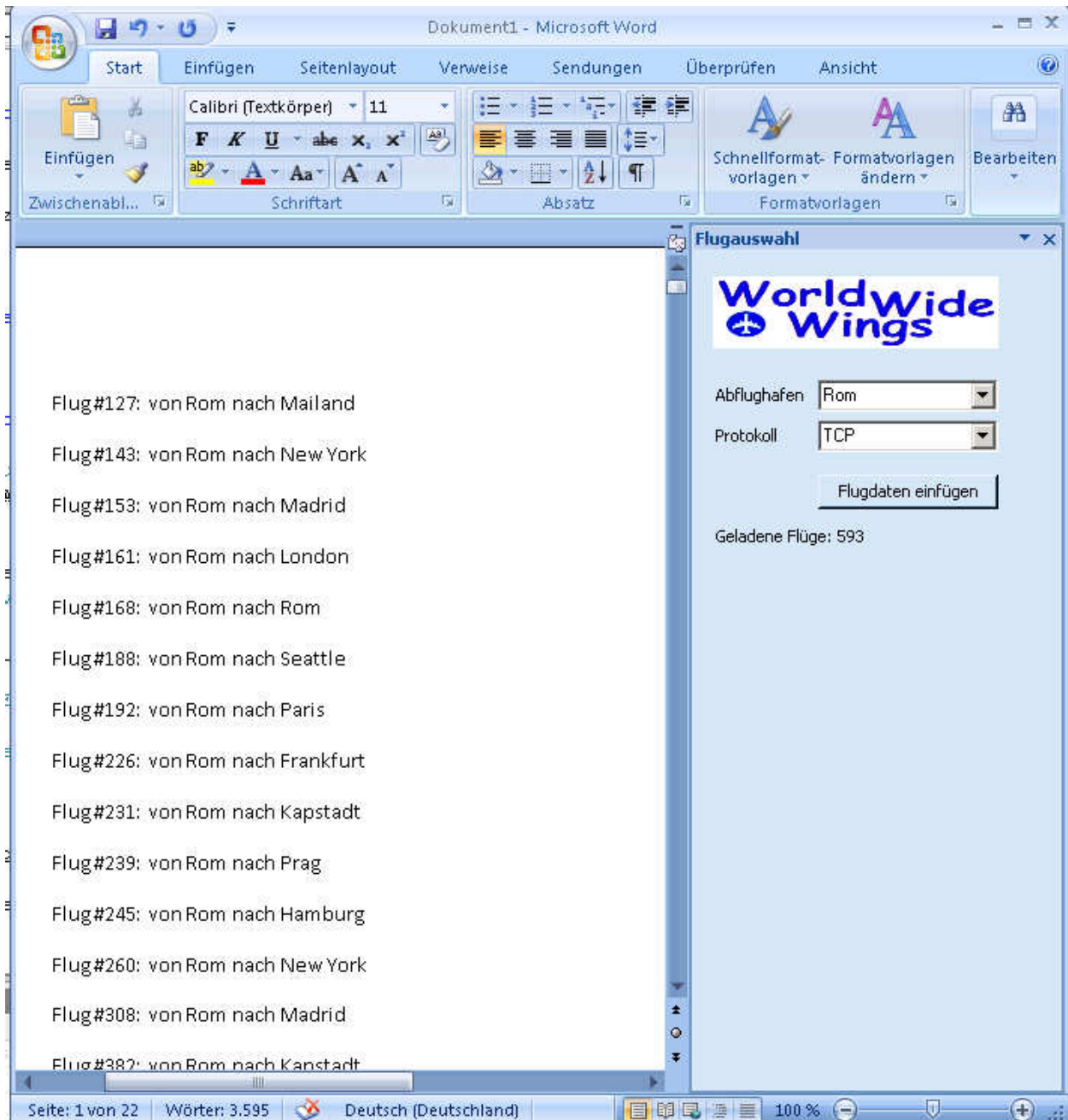


Abbildung Fehler! Formatvorlage nicht definiert..1: Flugverwaltung mit WPF

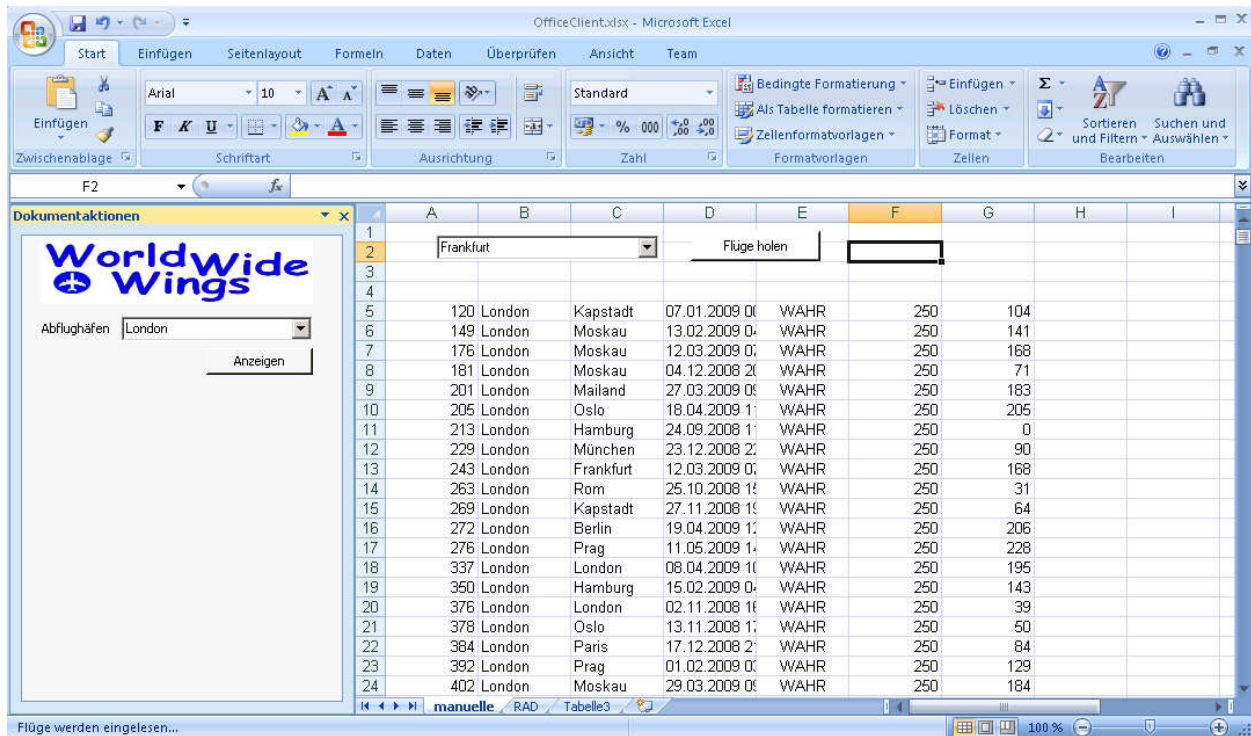
7.7 Funktionen der Office-Anwendungen

Die Office-Erweiterungen für Word und Excel zeigen eine Seitenleiste, über die man Flugdaten abrufen kann. Excel greift direkt auf die Datenbank zu, während Word über den WCF Application Server geht. Der Application Server muss also gestartet sein.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



7.8 Funktionen der Konsolenanwendung

Die Konsolenanwendung zeigt ein über Tasten steuerbares Menü, mit dem man einige Beispiele aufrufen kann.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



```
file:///H:/WWW/Konsole/ConsoleUI_CS/bin/Debug/ConsoleUI_CS.EXE
World Wide Wings Konsolen-Anwendung
1. Aufruf dieses Benutzers.

World Wide Wings - Konsole - HAUPTMENÜ

D ... Demo-Viewer
A ... Alle Flüge ausgeben
B ... Alle Flüge ausgeben <Seitenweises Abholen der Daten>
C ... Alle Flüge auf der Strecke Berlin - München ausgeben
F ... Daten zu einem Flug
O ... Alle Flüge von einem Abflugort
P ... Anzahl der freien Plätze ändern
N ... Neuer Flug
L ... Flug löschen
U ... Prohebuchung
S ... Sonstige Demos <Demoviewer>
I ... Flüge neu initialisieren
J ... Passagiere neu initialisieren
Z ... Plätze neu initialisieren
W ... Test-Workflow <Konsole>
U ... Workflow-Starter <GUI>
1 ... Demo 1: Lokale Datenbank
2 ... Demo 2: Lokale Datenbank
3 ... Demo 3: WCF Ein-Weg/Zwei-Weg/Sessions
4 ... Demo 4: WCF Duplex
5 ... Demo 5: WCF MSMQ
6 ... Demo 6: WCF Sitzung Operation
7 ... Demo 7: WCF Sitzung Count
8 ... Demo 8: WCF Sitzung beenden
9 ... Demo 9: WCF Sitzung erneuern
0 ... Demo 10: WCF Durable Services
E ... Programm beenden
```

8 Softwarevoraussetzungen für die aktuelle Version

Absolut notwendige Installationsvoraussetzungen für alle Teile der Anwendung sind:

- .NET Framework 4.0.30319.1
- Visual Studio 2010 (4.0.30319.1) Professional Edition
- SQL Server 2008 R2 Express Edition

Einige Teile der Anwendung erfordern außerdem folgende Installationspakete:

- MSMQ-Dienst in Windows (für WCF-Anwendungsserver)
- Visual Studio 2010 Ultimate
- SQL Server 2008 R2 Standard oder Enterprise Edition (nur für ADO.NET Query Notifications)
- Microsoft Word 2007/2010 und Excel 2007/2010 (nur für Office-Clients)
- Visual Studio Tools for Microsoft Office 2010 (nur für Office-Clients)
- Microsoft Silverlight SDK 4.0 (nur für Silverlight-Client)
- Windows PowerShell 2.0
- MSMQ-Dienst (installiert von der Windows-CD) – für WCF-Server

Einige Projekte im Ordner _SONSTIGEBEISPIELE erfordern zudem:

- Telerik/Vanatec Open Access 4.0

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

- nHibernate 2.1

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

WICHTIG: Wenn Sie die optionalen Softwarevoraussetzungen nicht erfüllen können, werden einige Projekte nicht laden, nicht kompilieren oder nicht korrekt arbeiten. Schließen Sie diese Projekte aus der jeweiligen Projektmappe aus.

9 Konfiguration der Beispielanwendung

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. Vor dem ersten Übersetzen sollten Sie die nachstehenden Punkte ausführen.

WICHTIG: Die gesamte Anwendung ist im Auslieferungszustand darauf ausgelegt, auf einem einzigen 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.

9.1 Rechte/Benutzerkontensteuerung

Diese Anleitung geht davon aus, dass Sie auf dem System als Administrator ohne Benutzerkontensteuerung (UAC ab Vista) arbeiten. Ohne Administratorrechte bzw. mit Benutzerkontensteuerung könnte es Probleme mit der Allokierung von Ports geben.

9.2 Softwarevoraussetzungen

Bitte stellen Sie vor dem Einrichten sicher, dass Sie die Softwarevoraussetzungen erfüllen.

9.3 TCP-Ports

Die Beispielanwendung verwendet folgende TCP-Ports, die frei sein müssen (sonst müssen Sie die Ports in den Konfigurationsdateien ändern!).

Anwendungsteil	Dienst	Port
Web/Webform	Weboberfläche	90
Web/DynamicData_LTS	Weboberfläche	91
Web/DynamicData_EF	Weboberfläche	92
Web/Silverlight.Web	Weboberfläche	93
Web/MVC	Weboberfläche	94
WCFServert / IIS	Diverse WCF-Dienste	87
WCFServert – HTTP-Bindungen	Diverse WCF-Dienste	88
WCFServert – TCP-Bindungen	Diverse WCF-Dienste	1234
WCFServert – Router	Router für WCF-Dienste	1111

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

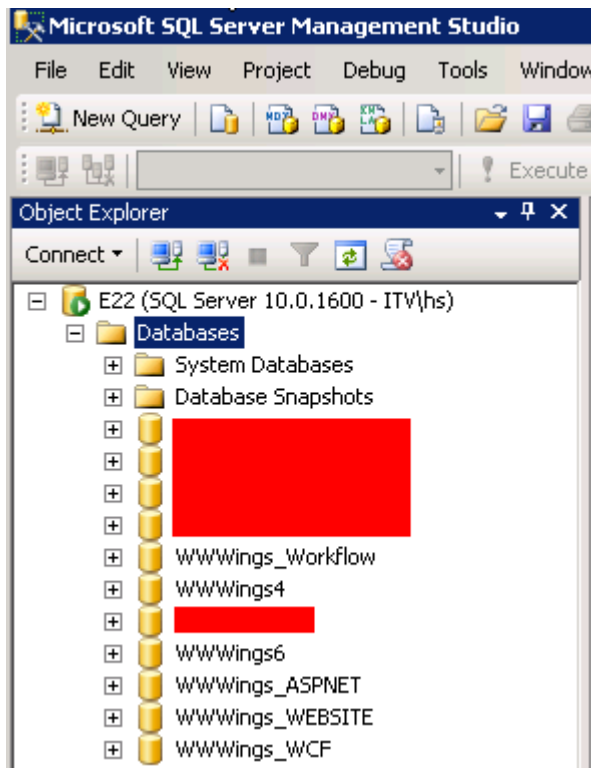


Abbildung: Zustand nach Einrichten aller für World Wide Wings notwendigen Datenbanken

9.4.1 Die World Wide Wings-Hauptdatenbanken

Es gibt zwei Versionen der Hauptdatenbank und eine Version der Website-Datenbank. Sie müssen alle dem lokalen SQL Server hinzufügen.

Datenbankdatei	Datenbankname	Verbindungszeichenfolge/Connection String
_Datenbanken/Version4/WWWings4.mdf	WWWings4	Data Source=.; Initial Catalog=WWWings4; Integrated Security=True
_Datenbanken/Version6/WWWings6.mdf	WWWings6	Data Source=.; Initial Catalog=WWWings6; Integrated Security=True
_Datenbanken/SonstigeDatenbanken /WWWings_Wesbite.mdf	WWWings_Website	Data Source=.; Initial Catalog=WWWings_Website; Integrated Security=True

HINWEIS: Für das ADO.NET Entity Framework (EF) gibt es spezielle Verbindungszeichenfolgen, die die o.g. Verbindungszeichenfolge beinhalten. Bitte achten Sie beim Anpassen darauf, dass Sie auch die EF-Verbindungszeichenfolgen anpassen.

Beispiel für eine EF-Verbindungszeichenfolge:

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

```
metadata=res://*/EF.EFModell.csdl|res://*/EF.EFModell.ssdl|res://*/EF.EFModell.msl;  
provider=System.Data.SqlClient;provider connection string=&quot;Data Source=.;Initial Cata-  
log=WWWings6;Integrated Security=True;MultipleActiveResultSets=True&quot;;
```

9.4.2 Einrichten der .NET-Standarddatenbanken

Für die ASP.NET-, WCF- und WF-Beispiele müssen Sie auch die drei Microsoft Standard-Datenbanken zusätzlich auf dem lokalen Microsoft SQL Server installieren:

- WWWings_WCF
- WWWings_WF
- WWWings_ASPNET

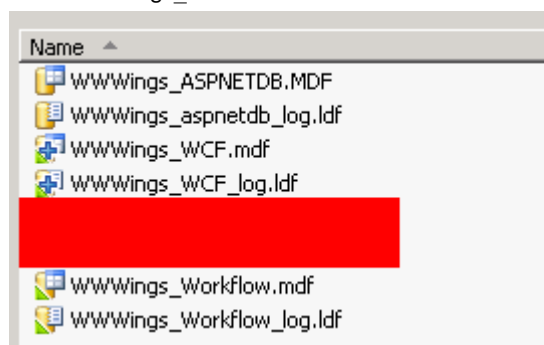


Abbildung: Datenbankdaten für die drei o.g. Datenbanken

Sie haben folgende alternative Möglichkeiten:

- Für WCF legen Sie eine Datenbank mit Namen "WWWings_WCF" an und führen Sie dort dann die im .NET Framework mitgelieferten Standard-SQL-Skripte von Microsoft aus (SqlPersistenceProviderSchema.sql und SqlPersistenceProviderLogic.sql – meist installiert im Verzeichnis WCF: C:\Windows\Microsoft.NET\Framework\v3.5\SQL\EN).
- Für WF legen Sie eine Datenbank mit Namen "WWWings_Workflow" an und führen Sie dort dann die im .NET Framework mitgelieferten Standard-SQL-Skripte von Microsoft aus (Tracking_Schema.sql und Tracking_Logic.sql, SqlPersistenceService_Schema.sql und SqlPersistenceService_Logic.sql meist installiert im Verzeichnis C:\Windows\Microsoft.NET\Framework\v3.0\Windows Workflow Foundation\SQL\EN)
- Für ASP.NET legen Sie eine Datenbank mit Namen "ASPNETDB" an und führen Sie dort dann alle im .NET Framework mitgelieferten Standard-SQL-Skripte für ASP.NET aus:

Name	Date modified	Type	Folder
InstallCommon.sql	05.01.2008 13:26	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallMembership.sql	20.07.2009 12:33	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallPersistSqlState.sql	20.07.2009 12:33	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallPersonalization.sql	18.09.2006 23:32	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallProfile.SQL	18.09.2006 23:32	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallRoles.sql	18.09.2006 23:32	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallSqlState.sql	20.07.2009 12:33	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallSqlStateTemplate.sql	20.07.2009 12:33	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)
InstallWebEventSqlProvider.sql	18.09.2006 23:32	Microsoft SQL Se...	v2.0.50727 (C:\windows\Microsoft.NET\Framework)

9.5 Option "AttachDBFileName"

Es gibt natürlich andere Möglichkeiten mit einem Microsoft SQL Server zu arbeiten (z.B. entfernter SQL Server oder AttachDbFileName). Dann müssten Sie in den Verbindungszeichenfolgen den Servername und andere

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

Einstellungen gemäß Ihrer Systemkonfiguration anpassen. Bitte beachten Sie, dass der Connection String in allen Projekten vorkommt. Leider stellt Microsoft noch keinen Mechanismus zur projektübergreifenden Ablage von Connection Strings bereit.

Beispiel: Verbindungszeichenfolge für AttachDbFileName:

```
Data Source=.;Initial
Catalog=H:\WWW\_Datenbanken\Version4\WWWings5.MDF;Integrated Security=True;
user instance=true;
```















Tipp: Verbindungszeichenfolgen finden Sie unter: <http://www.connectionstrings.com/>

9.6 Projektmappenwahl

Wählen Sie die Projektmappe, die Ihrer Zielsetzung am nächsten kommt (z.B. WWWings_Web.sln, WWWings_Window.sln, WWWings_Office.sln). Sie können auch WWWings_Alle.sln öffnen, doch dauert die Übersetzung hier am längsten!

9.7 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]
```

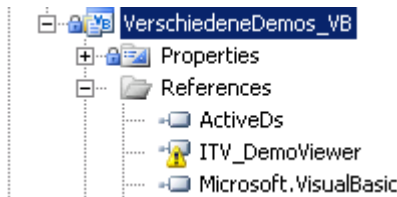
```
@="H:\\WWW\\_Komponenten"
```

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

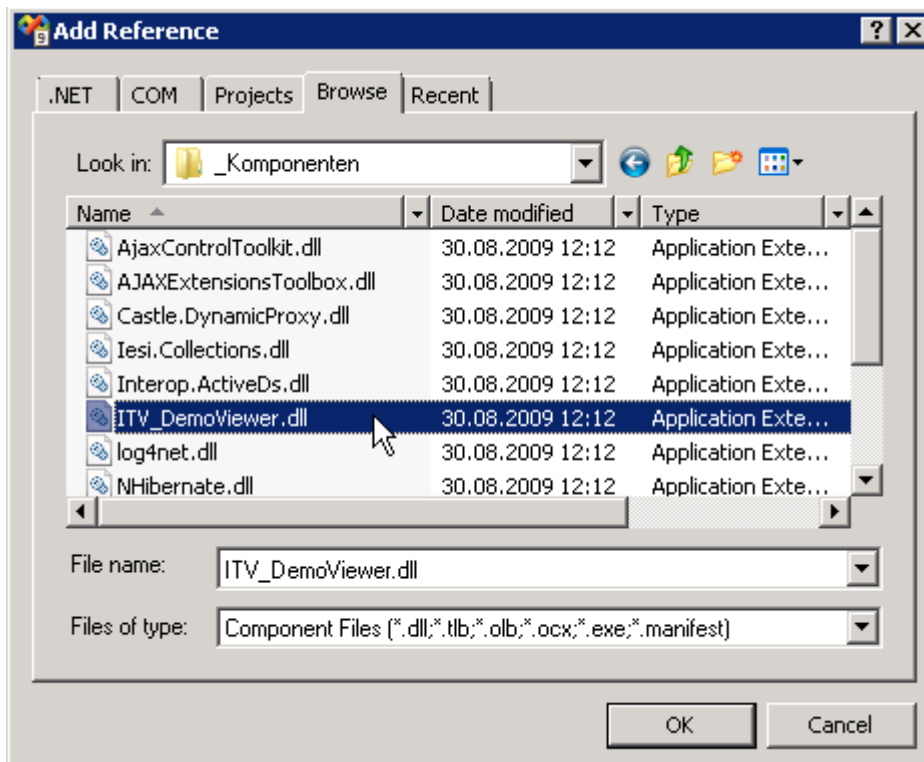
Sollte Visual Studio dennoch die DLLs nicht finden können (was auf einigen Systemen beobachtet wurde), so erneuern Sie bitte die Referenzen.

So erkennen Sie fehlende Referenzen, bei denen die DLL nicht gefunden wurde:

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



Wählen Sie "Add Reference" (deutsch: "Verweis hinzufügen") und wählen Sie die entsprechende DLL aus dem Verzeichnis _Komponenten:



Danach sollte das gelbe Ausrufezeichen in der Liste der Referenzen verschwunden sein!

9.8 Dateisystempfade

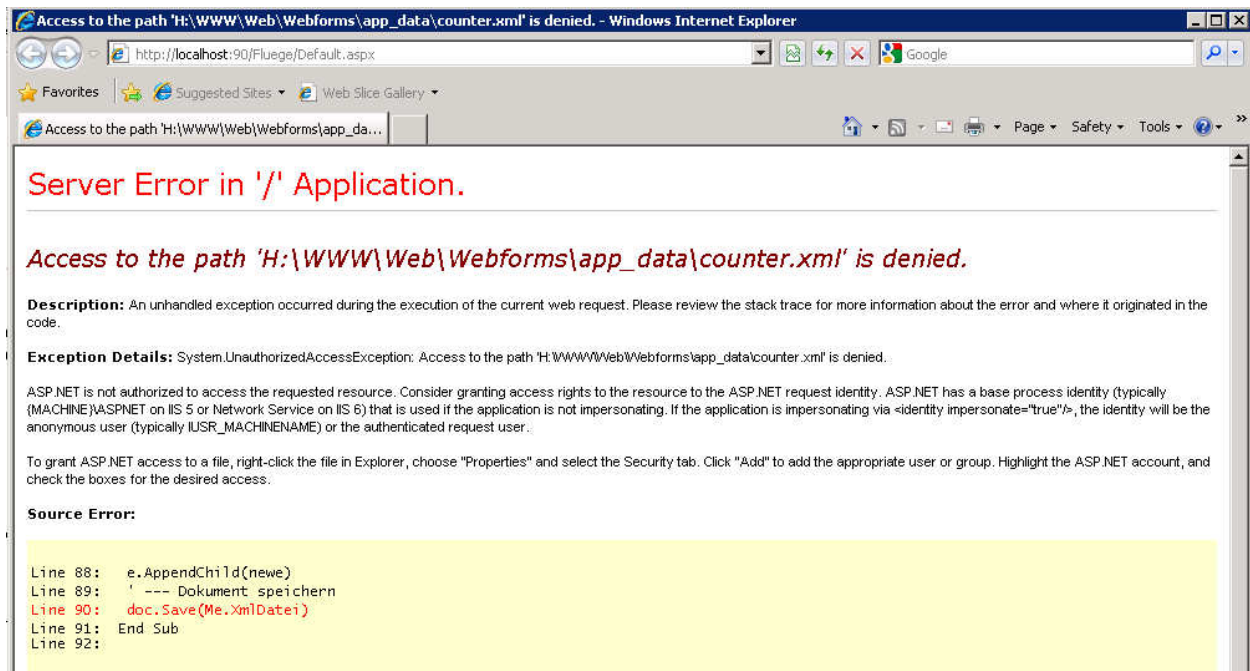
Bitte beachten Sie, dass sich leider nicht alle Pfade als relative Pfade gestalten lassen. Die Standardeinstellung für alle Pfade ist h:\www. Wenn Sie die Anwendung dort ablegen, brauchen Sie keine Pfade anzupassen.

Die Konfigurationsdateien, in denen Sie die Pfadangaben anpassen müssen, liegen bei .NET-Anwendungen im Standard im Wurzelordner des jeweiligen Startprojekts und heißen app.config oder web.config.

9.9 Dateisystemrechte

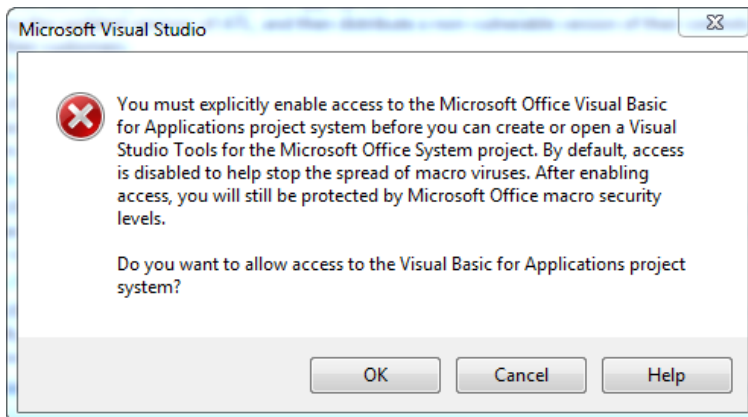
Der Webserver, auf dem Sie die Webforms-Anwendung betreiben, muss Schreibrechte auf dem Verzeichnis App_Data haben, sonst sehen Sie Fehler wie den folgenden.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



9.10 Sonstige Punkte

Wenn Sie beim Kompilieren oder Starten gefragt werden, ob Sie "Debuggen" oder "VBA" erlauben wollen, wählen Sie bitte "Ja".



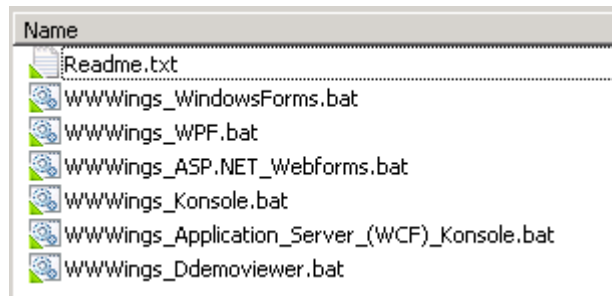
10 Start der Anwendung

Dieses Kapitel dokumentiert, wie die wichtigsten Anwendungsteile gestartet werden, nachdem Sie die Anwendung auf Ihrem System konfiguriert und kompiliert haben.

10.1 Batch-Dateien

Im Hauptordner _START finden Sie mehrere Batch-Dateien zum Start der zentralen Anwendungsteile.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

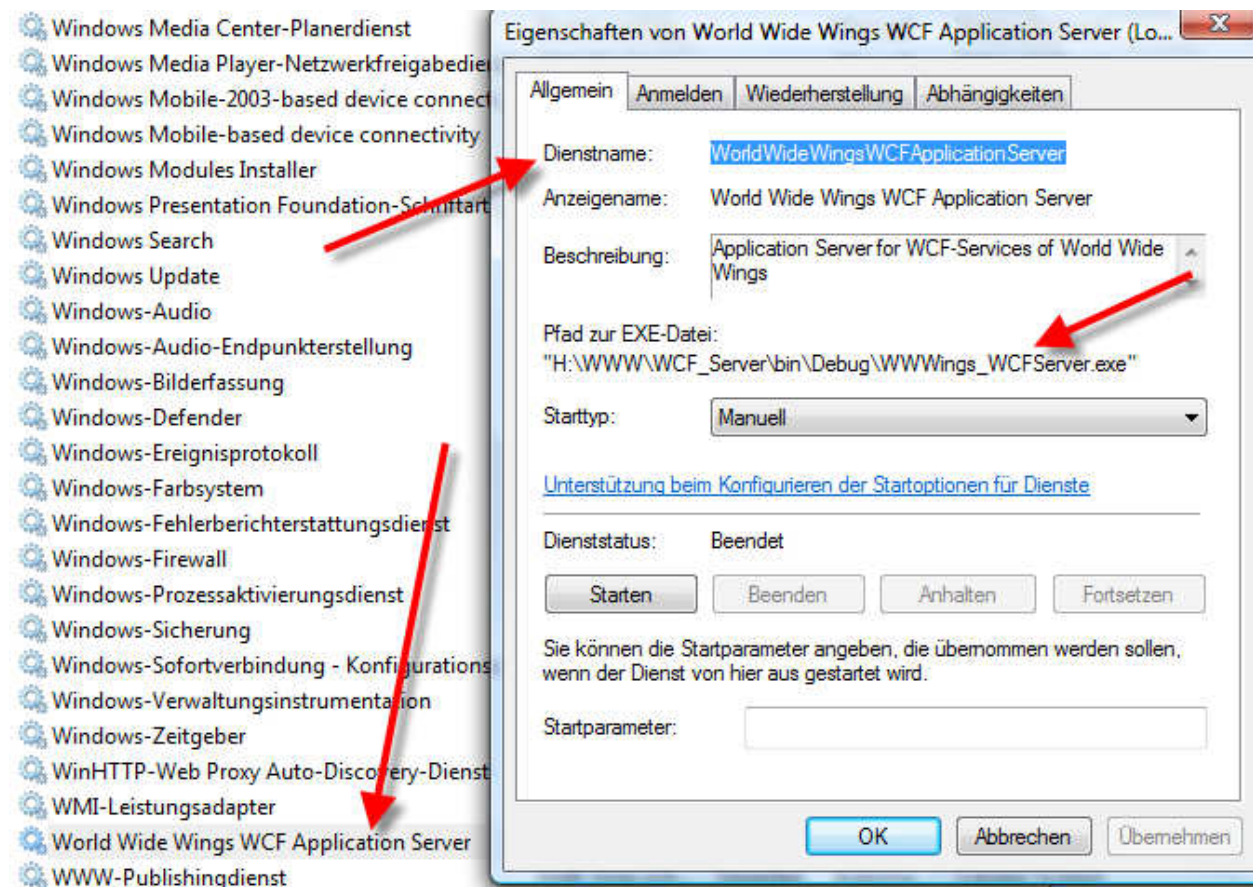


10.2 Start des WCF-Servers (Verzeichnis WCF_Server)

Den WCF-Server können Sie entweder als ausgabelosen Windows-Dienst oder als Konsolenanwendung mit hilfreichen Ausgaben starten.

10.2.1 Start als Windows-Dienst

1. Installation mit `installutil.exe WWWings_WCFServer.exe`
2. Start mit `net start WorldWideWingsWCFApplicationServer`



Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

```
C:\Program Files\VS08\VC>net start worldwidewingswcfapplicationserver
World Wide Wings WCF Application Server wird gestartet..
World Wide Wings WCF Application Server konnte nicht gestartet werden.

Der Dienst hat keinen Fehler gemeldet.

Sie erhalten weitere Hilfe, wenn Sie NET HELPMSG 3534 eingeben.
```

10.2.2 Start als Konsolenanwendung

mit Parameter "Debug":

WWWings_WCFServer.exe debug

Der folgende Screenshot zeigt den Start des WCF-Servers.

Tipp: Alternativ können Sie starten:

/_Start/WWWings_Application_Server_(WCF)_Konsole.bat

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

```

C:\Windows\system32\cmd.exe
Identität des Server-Prozesses: E46\HS
Neue Instanz der Fassade FlugplanService!
Neue Instanz der Fassade FlugplanverwaltungService!
Neue Instanz der Fassade Buchungsservice
Neue Instanz der Fassade PassagierverwaltungService!
Starte Dienst FlugplanService
- Endpunkt: net.tcp://localhost:1234/WWings/FlugplanService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/FlugplanService <NetNamedPipeBinding>
- Endpunkt: http://localhost:88/Webservices/FlugplanService.svc <BasicHttpBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/metadaten/FlugplanService <MetadataExchangeTcpBinding>
Starte Dienst FlugplanServiceEF
- Endpunkt: net.tcp://localhost:1234/WWings/FlugplanServiceEF <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/FlugplanServiceEF <NetNamedPipeBinding>
- Endpunkt: http://localhost:88/Webservices/FlugplanServiceEF.svc <BasicHttpBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/metadaten/FlugplanServiceEF <MetadataExchangeTcpBinding>
Starte Dienst FlugplanverwaltungService
- Endpunkt: net.tcp://localhost:1234/WWings/FlugplanverwaltungService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/FlugplanverwaltungService <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/metadaten/FlugplanverwaltungService <MetadataExchangeTcpBinding>
Starte Dienst Buchungsservice
- Endpunkt: net.tcp://localhost:1234/WWings/Buchungsservice <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/Buchungsservice <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/metadaten/Buchungsservice <MetadataExchangeTcpBinding>
Starte Dienst PassagierverwaltungService
- Endpunkt: net.tcp://localhost:1234/WWings/PassagierverwaltungService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/PassagierverwaltungService <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/metadaten/PassagierverwaltungService <MetadataExchangeTcpBinding>
Starte Dienst TestService
- Endpunkt: net.tcp://localhost:1234/WWings/TestService <NetTcpBinding>
- Endpunkt: net.pipe://localhost/WWings/Pipes/TestService <NetNamedPipeBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/metadaten/TestService <MetadataExchangeTcpBinding>
Starte Dienst Protokollservice
- Endpunkt: net.msmq://localhost/private/WWings <NetMsmqBinding>
- Endpunkt: net.tcp://localhost:1234/WWings/metadaten/Protokolldienst <MetadataExchangeTcpBinding>
WCF-Server ist jetzt gestartet!
Datenbank: Data Source=.;Initial Catalog=WWings;Integrated Security=True
Testservice Instanz #1 erzeugt!
.Interner Selbsttest (ohne Protokollstack): OK <9999/72/290>
Testservice Instanz #2 erzeugt!
.Externer Selbsttest (mit ICF-Protokollstack): OK <9999/72/290>
Testservice Instanz #3 erzeugt!
.Externer Selbsttest (mit Named Pipes): OK <9999/72/290>

```

10.3 Start der Clients

Anwendungsart	Standort ("Release" statt "Debug", falls Sie in Visual Studio die Konfiguration auf "Release" gestellt haben)	Batch-Datei
Konsolen-Client	\Konsole\ConsoleUI_CS\bin\Debug\ConsoleUI_CS.exe	_Start/WWWings_Konsole.bat
Windows Forms-Client	\Desktop\WindowsForms_CS\bin\Debug\WorldWideWings_WindowsUI.exe	_Start/WWWings_WindowsForms.bat

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

WPF-Client	\Desktop\WPF\bin\Debug\ WWWings_WPF.exe	WWWings_WPF.bat
Word_AddIn / Word_Vorlage / Ex- cel_Vorlage	Bitte starten Sie das jeweilige Visual Studio-Projekt im Debugger. Es gibt derzeit kein Setup-Projekt dafür.	
ASP.NET Web-forms	http://localhost:90 (wenn der Visual Studio Webserver läuft oder Sie die Anwendung in den IIS eingebunden haben)	WWWings_ASPNET_WebForms.bat
ASP.NET Dynamic Data LTS	http://localhost:91 (wenn der Visual Studio Webserver läuft oder Sie die Anwendung in den IIS eingebunden haben)	
ASP.NET Dynamic Data EF	http://localhost:92 (wenn der Visual Studio Webserver läuft oder Sie die Anwendung in den IIS eingebunden haben)	
ASP.NET MVC	http://localhost:94 (wenn der Visual Studio Webserver läuft oder Sie die Anwendung in den IIS eingebunden haben)	
Silverlight-Anwendung	http://localhost:93 (wenn der Visual Studio Webserver läuft oder Sie die Anwendung in den IIS eingebunden haben)	

11 Mögliche Fehler

Das Fallbeispiel wird vor jeder Auslieferung fehlerfrei übersetzt (zum Beweis siehe Bildschirmabbildung) und in den Kernelementen funktionell von mindestens drei Personen auf drei verschiedenen Systemen getestet. Fehler können natürlich dennoch nicht ausgeschlossen werden, da es sich hier nur um ein ehrenamtliches Beispielpjekt handelt und nicht um ein professionelles Produkt (und auch da gibt es wohl manchmal Fehler ☺)

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

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

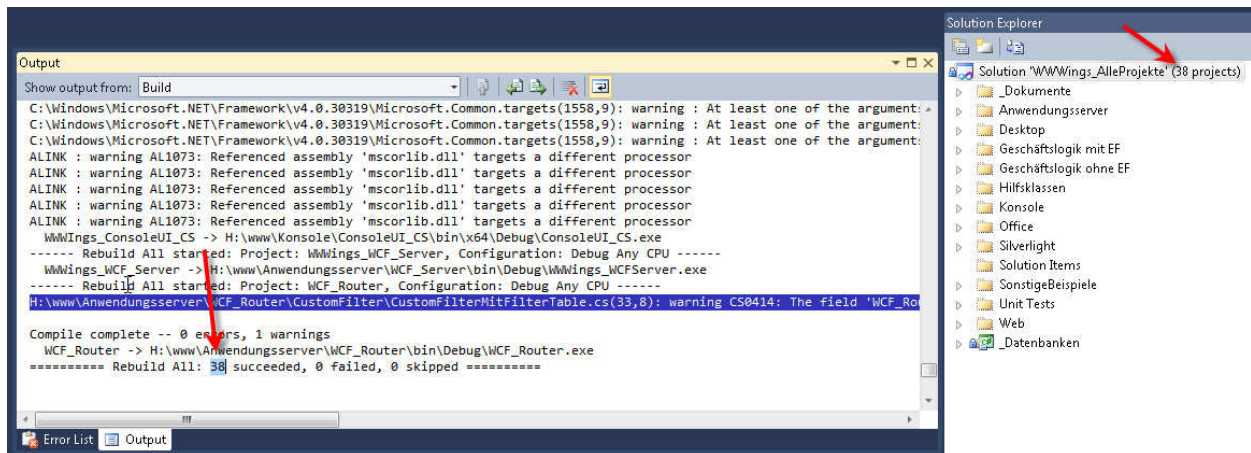


Abbildung: Beleg der fehlerfreien Übersetzung der Projektmappe mit allen 38 Projekten (WWings_AlleProjekte)

12 Wie finde ich bestimmte Codebeispiele?

Diese Tipps können Ihnen helfen, bestimmte Beispiele im Fallbeispiel zu finden:

Wenn Sie in der Webanwendung die Menüpunkte oben am Bildschirm und die Äste im Baum links am Bildschirm mit der Maus überfahren, sehen Sie in den Tooltips, welche Beispiele sich hinter dem jeweiligen Punkt verbergen.

In verschiedenen Ordnern der Webanwendung (z.B. /Fluege/) gibt es eine default.aspx-Seite, die selbst kein Beispiel ist, sondern die in diesem Ordner enthaltenen Beispiele verlinkt und kurz erläutert.

Solche Navigationshilfen kann es naturgemäß in anderen Projektarten nicht geben. Im Zweifel ist die Volltextsuche von Visual Studio die schnellste Möglichkeit, eine bestimmte Codestelle aus einem meiner Bücher oder den Einsatz einer bestimmten .NET-Klasse zu finden.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"



Abbildung: Hilfeseiten in der Webanwendung

13 Dokumentation der Klassen

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

Hinweise zu der .NET 4.0-Codebeispielsammlung "World Wide Wings"

World Wide Wings-Klassenreferenz Help

Hide Locate Back Forward Stop Refresh Home Print Options

Contents Index Search

- + de.ITVisions.DALUnit
 - + de.WW/Wings:
 - Buchung_BLMManager Class
 - Fluege Class
 - Flug Class
 - FlugBLMManager Class
 - FlugBLMManager_DataSet Class
 - FlughafenBLMManager Class
 - FlugMenge Class
 - FlugMengePaging Class
 - IPersonenKlasse Interface
 - ObjektStatus Enumeration
 - PagingInfo Class
 - Passagier_BLMManager Class
 - Person Class
 - PilotBLMManager Class
 - + de.WW/Wings:DAL
 - DAL_Testintf Class
 - Flug_DataManager Class
 - Flug_DataManager.FluggeWurdenAktualisier
 - Passagier_DataManager_ST Class
 - Passagier_DataManager_SWC Class
 - + de.WW/Wings: Dienste
 - BuchungsService Class
 - DBInfo Class
 - FlugplanService Class
 - FlugplanverwaltungService Class
 - IFlugplanService Interface
 - IFlugplanverwaltungService Interface
 - IPingDuplexCallback Interface
 - ITestService Interface
 - PassagierverwaltungService Class
 - PingInfo Class
 - ProtokollService Class
 - TestService Class
 - + de.WW/Wings: Fluggesellschaft
 - de.WW/Wings: Manager
 - de.WW/Wings: MitarbeiterSystem
 - + de.WW/Wings: ORM
 - ORM_Buchung Class
 - ORM_Flug Class
 - ORM_Passagier Class
 - + de.WW/Wings: PassagierSystem
 - de.WW/Wings: Test
 - de.WW/Wings: Workflow
 - de.WW/Wings: Workflow.Aktivitaeten_Generiert
 - de.WW/Wings: Workflow.CodeWorkflows
 - de.WW/Wings: Workflow.WW/Wings
 - de.WW/Wings: Workflow.XAML

World Wide Wings-Bibliotheken Version 0.4.3

de.WWWings Namespace

[Namespace hierarchy](#)

Classes

Class	Description
Buchung_BLMManager	Zustandsloser Geschäftslogikmanager für Buchungen
Fluege	Obsolete. Eine Menge von Flügen. Alte Implementierung, die Manager und ORM in einem war. Verwenden Sie nur noch FlugMenge und FlugBLMManager
Flug	Diese Klasse repräsentiert einen Flug.
FlugBLMManager	Zustandsloser Geschäftslogikmanager für die Flug-Klasse. Implementierungsalternative 1: mit ORM-Mapping
FlugBLMManager_DataSet	Zustandsloser Geschäftslogikmanager für die Flug-Klasse. Implementierungsalternative 2: ohne ORM-Mapping
FlughafenBLMManager	Zustandsloser Geschäftslogikmanager für Flughäfen.
FlugMenge	FlugMenge ist typisierte Menge von Flug-Objekten, die mit Hilfe der Klasse List implementiert ist.
FlugMengePaging	FlugMenge ist typisierte Menge von Flug-Objekten, die mit Hilfe der Klasse List implementiert ist. Diese Variante holt immer nur eine definierbare Menge (Attribut SeitenGrosse) aus der Datenbank.
PagingInfo	Klasse für Ereignisparameter beim Paging in der Geschäftslogik
Passagier_BLMManager	Zustandsloser Geschäftslogikmanager für die Passagiere.
Person	Basisklasse für Mitarbeiter und Passagiere
PilotBLMManager	Zustandsloser Geschäftslogikmanager für die Pilot-Klasse.

Interfaces

Interface	Description
IPersonenKlasse	Schnittstellen für die Daten einer Person

Enumerations

Enumeration	Description
ObjektStatus	Objekt-Stati für ORM und Caching

[Send comments on this topic.](#)

World Wide Wings - Democode
(C) Dr. Holger Schwichtenberg
<http://www.IT-Visions.de/WWWings>

Abbildung: Dokumentation der Klassenbibliotheken

14 Aktualisierungen

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

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

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, ob in dem betreffenden Code alle Parameter auf ihren Umgebung angepasst sind.

Hinweise zu der .NET 4.0- Codebeispielsammlung "World Wide Wings"

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 und dotnetpro.



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 bis 2009 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>