

SharePoint- Server 2010

Das Entwicklerbuch

Marc-André Zhou

Marc André Zhou

SharePoint Server 2010

Das Entwicklerbuch

Marc André Zhou
SharePoint Server 2010
ISBN: 978-3-86802-261-2

© 2012 entwickler.press
Ein Imprint der Software & Support Media GmbH

Bibliografische Information Der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen
Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über
<http://dnb.ddb.de> abrufbar.

Ihr Kontakt zum Verlag und Lektorat:
Software & Support Media GmbH
entwickler.press
Darmstädter Landstr. 108
60598 Frankfurt am Main
Tel.: +49 (0)69 630089-0
Fax: +49 (0)69 930089-89
lektorat@entwickler-press.de
<http://www.entwickler-press.de>

Lektorat: Sebastian Burkart
Korrektur: Frauke Pesch
Satz: Dominique Kalbassi
Belichtung, Druck & Bindung: M.P. Media-Print Informationstechnologie GmbH, Paderborn

Alle Rechte, auch für Übersetzungen, sind vorbehalten. Reproduktion jeglicher Art (Fotokopie, Nachdruck, Mikrofilm, Erfassung auf elektronischen Datenträgern oder anderen Verfahren) nur mit schriftlicher Genehmigung des Verlags. Jegliche Haftung für die Richtigkeit des gesamten Werks kann, trotz sorgfältiger Prüfung durch Autor und Verlag, nicht übernommen werden. Die im Buch genannten Produkte, Warenzeichen und Firmennamen sind in der Regel durch deren Inhaber geschützt.

Inhaltsverzeichnis

Vorwort	11
1 Entwicklungsumgebung	13
1.1 Allgemeine Anforderungen	14
1.1.1 Hardware	14
1.1.2 Software	14
1.2 SharePoint 2010 unter Windows 7	15
1.2.1 Installation vorbereiten	16
1.2.2 Installation durchführen	18
1.2.3 Verwendung einer vorhandenen MS-SQL-Server-Instanz	20
1.3 SharePoint Server 2010 unter Windows Server 2008 R2	22
1.3.1 Installation und Konfiguration des Serverbetriebssystems	23
1.3.2 Windows-Server-2008-Rollen	23
1.3.3 SharePoint-2010-Installation	28
1.4 SharePoint 2010 Easy Setup Script	35
1.5 Contoso-Umgebung	36
1.5.1 Hyper-V-Server installieren und konfigurieren	37
1.5.2 Contoso Image verwenden	38
1.6 Entwicklerwerkzeuge	39
1.6.1 Visual Studio 2010	39
1.6.2 Möglichkeiten in Visual Studio 2010	40
1.6.3 Community Kit for SharePoint	44
1.6.4 SPDisposeCheck	45
1.6.5 SharePoint Manager	45
1.6.6 E-Mail-Funktionen	46
1.7 Lösungstypen	48
1.7.1 Ausführungsmodell von Sandkastenlösungen	49
1.7.2 Sandkastenlösungen vs. Farm-Level-Lösungen	52
1.7.3 Festlegung des Lösungstyps	53
2 Das SharePoint-Objektmodell	57
2.1 SharePoint-Struktur	57
2.1.1 Serverarchitektur	58
2.1.2 Seitenarchitektur	60

2.2	Server Object Model (SOM)	64
2.2.1	Verbindung und Inhalte auslesen	64
2.2.2	Inhalte rekursiv auslesen	66
2.2.3	Teamsite programmatisch anlegen	68
2.2.4	Benachrichtigungen setzen (Alerts)	72
2.2.5	Arbeiten mit Listen	75
2.2.6	Abfrage umfangreicher Listen	77
2.2.7	Cache für Abfrageergebnisse	84
2.2.8	Abfragen mit CAML	86
2.2.9	LINQ to SharePoint	90
2.2.10	Speicherverwaltung	97
2.3	Client Object Model (COM)	99
2.3.1	Funktionsweise	100
2.3.2	Verfügbare Bibliotheken	101
2.3.3	Verbindung und Inhalte auslesen	103
2.3.4	Load und ExecuteQuery	104
2.3.5	Listeninhalte auslesen	106
2.3.6	Teamsite programmatisch anlegen	109
2.3.7	Load vs. LoadQuery	113
2.3.8	Daten filtern mit LINQ	114
2.4	SharePoint Online (Office 365)	116
2.4.1	Funktionen von SharePoint Online	117
2.4.2	Lösungstypen	118
2.4.3	Ein Web Part für SharePoint Online umsetzen	119
3	Entwurf von SharePoint-Lösungen	125
3.1	SharePoint als Lösungsplattform?	126
3.2	Konzeptionelle Planung einer SharePoint-Lösung	129
3.3	Entwicklungsmöglichkeiten	130
3.3.1	Eigene Felder (Fields)	133
3.3.2	(Externe) Inhaltsdatentypen	133
3.3.3	Masterpages und Seitenvorlagen (Page Layouts)	133
3.3.4	Menüband-Erweiterung (Ribbon)	134
3.3.5	Ereignisempfänger (Event Receiver)	134
3.3.6	Web Parts	134
3.3.7	Anwendungsseiten	135
3.3.8	Zeitgesteuerte Aufträge (Timer Jobs)	135
3.3.9	Dienstanwendungen	136

3.3.10 Full Trust Proxies	136
3.4 Komponenten identifizieren	136
3.4.1 Beispiel: Dokumentenverwaltung	138
3.4.2 Beispiel: Urlaubsantrag	139
3.5 Application Lifecycle Management (ALM)	140
3.5.1 Entwicklungsumgebung	141
3.5.2 Erweiterungs- und Änderungsverfahren	142
3.5.3 ALM-Beispiel	143
4 Konfigurationswerte	145
4.1 Allgemeine Unterscheidungsmerkmale	145
4.1.1 Speicherobjekte und Hierarchien	146
4.1.2 (Konfigurations-)Speicherobjekte verwenden	147
4.1.3 Sicherheitsmodell	155
4.2 Wahl des optimalen Speicherobjekts	156
4.2.1 Speicherort: web.config	156
4.2.2 Speicherort: Property Bag	157
4.2.3 SharePoint-Liste	158
4.2.4 SPPersistedObject	159
5 Realisierung von Web Parts	161
5.1 Seiten- und Web-Part-Struktur	162
5.2 Möglichkeiten eigener Web Parts	165
5.3 Web-Part-Entwicklungsvarianten	166
5.4 Einbinden und Testen eigener Web Parts	167
5.5 Visuelle Web Parts	169
5.5.1 Zugriff auf Listendaten	172
5.5.2 Web-Part-Verbindungen: Master-Detail	174
5.6 Standard-Web-Parts	185
5.6.1 Northwind-Bestellungen	187
5.7 Web Parts als Sandkastenlösung	192
5.7.1 Full Trust Proxy	193
5.8 Visuelle Web Parts als Sandkastenlösung	197
5.8.1 Funktionsweise	198
5.9 Model View Presenter Web Part	199
5.9.1 Umsetzung eines MVP (Passive View) Web Parts	201
5.9.2 MVP und Testbarkeit	206
5.10 SPGridView: Probleme und Lösungen	209

6	User Interface	217
6.1	Erweiterung des Ribbon Menu (Menüband)	218
6.1.1	Anatomie und Erweiterung des Ribbons	218
6.1.2	Hinzufügen einer Schaltfläche	219
6.1.3	CustomAction	220
6.1.4	Hinzufügen einer Gruppe	224
6.1.5	Hinzufügen eines neuen Tabs	230
6.1.6	Finden der korrekten Location/Sequence	231
6.2	Ribbon – Ereignisse und Steuerung	232
6.2.1	Clientseitige Funktionsaufrufe	232
6.2.2	Kontextabhängige Ribbon-Funktionen	235
6.2.3	Serverseitige Funktionsaufrufe	246
6.3	Dialogframework	248
6.3.1	Statusbar	250
6.3.2	Benachrichtigungen	251
6.3.3	Dialoge	253
7	Inhaltstypen (Content Types)	263
7.1	Strukturierung über Content Types	265
7.2	Struktur und Hierarchie	267
7.2.1	Eingebaute Content Types	268
7.2.2	Basisinformationen eines Content Types	270
7.2.3	Content-Types-Felder (Fields)	271
7.2.4	Site und List Content Types	273
7.3	Erstellung von Content Types	275
7.3.1	Deklarative Erstellung	275
7.3.2	Programmatische Erstellung	287
7.4	Erweiterte Felder	292
7.4.1	SPField-Klasse	294
7.4.2	Erstellung eines einfachen Feldes	296
7.4.3	Erstellung eines komplexen Feldes	302
7.5	Listendefinitionen	312
7.5.1	Das Schema einer Liste	312
7.5.2	Erstellen einer Listendefinition	314

8	Reagieren auf Ereignisse	321
8.1	Event Receivers – eine Übersicht	321
8.1.1	Ereignistypen	322
8.1.2	SPEventReceiverBase	323
8.1.3	SPEventPropertiesBase	324
8.2	Reagieren auf Listenereignisse	325
8.2.1	SPItemEventProperties	326
8.2.2	Umsetzen eines SPItemEventReceiver	328
8.3	Reagieren auf Eintragsänderungen	339
8.3.1	SPListEventProperties	340
8.3.2	Umsetzen eines SPListEventReceiver	341
8.4	Websiteereignisse	349
8.4.1	Umsetzen eines SPWebEventReceiver	351
8.5	Workflow-Ereignisse	355
8.5.1	SPWorkflowEventProperties	355
8.6	Feature-Ereignisse	356
8.6.1	SPFeatureReceiverProperties	357
8.6.2	Umsetzen eines SPFeatureReceiver	358
8.7	Rekursive Ereignisse vermeiden	361
8.8	Charakteristika von Ereignisempfängern	361
8.8.1	Bindungen von Ereignissen	362
8.8.2	Ereignisquellen deklarativ festlegen	365
8.8.3	Ereignisquellen programmatisch festlegen	367
9	Workflow	369
9.1	Logischer Aufbau von Workflows	371
9.2	Arten von Workflows	373
9.3	Technische Basis	376
9.3.1	Konfigurationsmöglichkeiten	380
9.3.2	Technische Architektur	383
9.3.3	Lebenszyklus	384
9.3.4	Workflow-Komponenten	385
9.3.5	Die Entwicklungsumgebung	390

9.4	Ein einfacher Workflow	394
9.4.1	Aktivierung	395
9.4.2	Schreiben in die Historienliste (History List)	396
9.4.3	Anlage einer Aufgabe (Task)	397
9.4.4	Aufgabenstatus auswerten	398
9.4.5	Testen	399
9.5	Korrelations-Token (Correlation Token)	401
9.6	Eigene Workflow-Formulare	402
9.6.1	Workflow: Bildbearbeitung	403
9.6.2	Assoziationsformular	405
9.6.3	Startformular	409
9.6.4	Workflow implementieren	410
9.7	Einsatz von InfoPath	413
9.7.1	Das Formular	413
9.7.2	Workflow erweitern	417
9.8	Eigene Aktivitäten (Custom Activities)	420
9.8.1	Implementierung	421
9.8.2	Konfigurationsdatei *.actions definieren	423
9.8.3	Installationsprojekt hinzufügen	424
9.8.4	Verwendung der Aktivität	425
10	Anwendungsbeispiele	427
10.1	Silverlight	427
10.1.1	Implementierung	428
10.1.2	Bereitstellen	435
10.1.3	Testen	435
10.2	Authentifizierungsanbieter	435
10.2.1	Umsetzung eines eigenen MembershipProviders	437
10.2.2	Konfigurationseinträge	439
10.2.3	Provider testen	440
10.3	Anwendungsseiten	441
10.3.1	Technische Basis von Anwendungsseiten	443
10.3.2	Northwind Demo	444
10.4	Timer und Jobs	453
10.4.1	Umsetzung eines Timer Jobs	454
10.4.2	Testen	456
10.4.3	Debuggen	457
Stichwortverzeichnis		459

Vorwort

Das vorliegende Buch SharePoint Server 2010 – Das Entwicklerbuch richtet sich an alle, die den SharePoint Server als Plattform für eigene Lösungen und Erweiterungen einsetzen möchten. Der Autor geht davon aus, dass die Oberfläche sowie die allgemeine Arbeitsweise des SharePoint Servers bekannt sind.

Bei der hier behandelten SharePoint-Version 2010 handelt es sich mittlerweile um die vierte Produktversion. Damit zunächst deutlich wird, wie das Produkt „SharePoint“ entstanden ist, folgt eine kurze Darstellung der historischen Entwicklung. Abbildung 1 zeigt sie als schematische Übersicht.

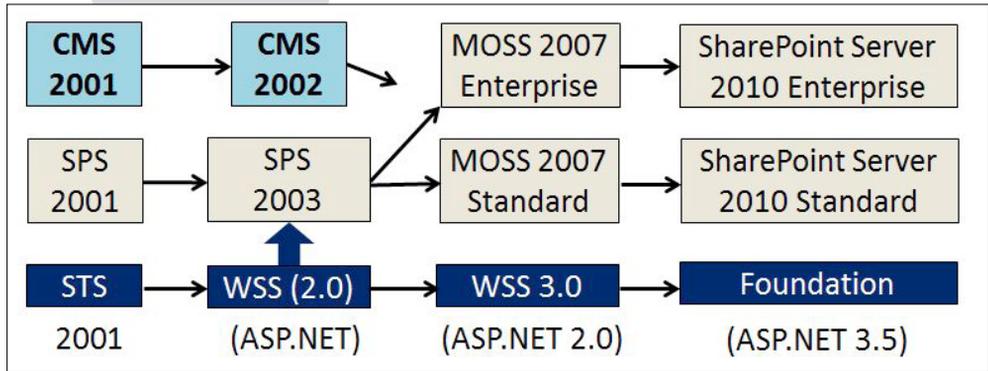


Abbildung 1: SharePoint-Entwicklungshistorie

Den Startpunkt dieser Serverplattform bildete der SharePoint Server 2001, der allerdings nur begrenzte Möglichkeiten besaß. Initial wurde die erste SharePoint-Version vom Office-Exchange-Entwicklungsteam ins Leben gerufen. Parallel zu dieser Version entstanden zwei weitere Produkte, die später den Entwicklungsprozess des SharePoint Servers beeinflussten: der Content Management Server 2001 (CMS 2001) und die SharePoint Team Services (STS). Der CMS 2001 wurde ursprünglich nicht von Microsoft entwickelt, sondern Microsoft kaufte das Produkt und übernahm es in seine eigene Produktlinie. Technologisch basiert das Produkt auf ASP 3.0 und passte daher sehr gut in das Microsoft-Produktportfolio. Die STS entstanden innerhalb der Front-Page-Entwicklergruppe und stellten eine Erweiterung der Front Page Extension dar. Für die nächste Version der Produkte nahm Microsoft eine Neuausrichtung vor und brachte die bis dahin unabhängigen Lösungen zusammen. So wurde der SharePoint Server 2003 in zwei Schichten (Layer) aufgeteilt. Die Basisschicht bilden hierbei die Windows SharePoint Services 2.0 (WSS 2.0),

die im Grunde aus einer Neuimplementierung der STS auf Basis von ASP.NET hervorgingen. Der SPS 2001 wurde ebenfalls erweitert und verwendet als Basis die neuen WSS 2.0. In der Zwischenzeit wurde der gekaufte CMS 2001 ebenfalls aktualisiert und die technologische Basis auf ASP.NET gehoben. Die wesentlichste Aktualisierung wurde in der nachfolgenden Version durchgeführt. Die WSS 2.0 wurden stringent erweitert und die WSS 3.0 basieren nun auf ASP.NET 2.0. Das gilt ebenfalls für den aktualisierten SharePoint Server, dessen offizielle Bezeichnung in Microsoft Office SharePoint Server (MOSS) geändert wurde, da er weiterhin auf den WSS 3.0 basiert. Zusätzlich wurde der bis dahin unabhängige CMS 2002 in die SharePoint-Produktfamilie integriert. Neben diesen technischen Änderungen wurde der SharePoint Server in zwei Lizenzen gesplittet. Für die aktuellste Version 2010 wurden die WSS 3.0 auf ASP.NET 3.5 aktualisiert. Der offizielle Name ist nun nicht mehr WSS 4.0 – wie man vielleicht annehmen könnte – sondern SharePoint Foundation. Der SharePoint Server selbst wurde ebenfalls wieder umbenannt und trägt nun die Bezeichnung SharePoint Server 2010 (SPS 2010). Technologisch wurde die neueste Version um viele Funktionen erweitert, die nun allesamt auf ASP.NET 3.5 basieren.

Aufgrund der erreichten Produktreife und der damit verbundenen Funktionsvielfalt können nicht alle Entwicklerthemen in einem Buch beschrieben werden. Dieses Buch behandelt daher die wichtigsten und am häufigsten benötigten Techniken, um eigene Lösungen basierend auf der SharePoint-Technologie umsetzen zu können.

Verwendete Begriffe und Sprache

Für Erklärungen und Erläuterungen werden in der Regel englischsprachige Bezeichner verwendet. Auch für die Objekte aus dem SharePoint-Objektmodell werden bevorzugt die englischsprachigen Ausdrücke herangezogen. Generell existieren für alle Bezeichnungen auch deutsche Übersetzungen, die jedoch nicht in allen Fällen eindeutig und sofort verständlich sind. Ebenfalls werden die Beispiele innerhalb einer englischsprachigen Entwicklungsumgebung umgesetzt. Das Gleiche gilt für die verwendete SharePoint-Oberfläche.

Onlinere Ressourcen zum Buch

Alle gezeigten Beispiele und Listings können über das zugehörige Onlineportal des Buches aus der Lösungsgalerie heruntergeladen werden. Neben den verfügbaren Visual-Studio-Projektmappen stehen weitere Ressourcen bereit, z. B. Webcasts zu verschiedenen Themenfeldern. Das Portal ist nicht statisch, sondern wird sukzessiv um weitere Inhalte erweitert. Der integrierte Blog berichtet über Neuigkeiten und zusätzliche Informationsangebote, z. B. über neue Einträge in der Lösungsgalerie.

Das Portal erreichen Sie unter dem URL: <http://zhou.sharepoint.com>

Technologisch basiert das Portal auf Office 365 – also auf SharePoint online. Weitere Informationen dazu finden Sie ebenfalls auf den Portalseiten unter dem Punkt *Über diese Seite*.

1

Entwicklungsumgebung

Der wichtigste Schritt für eine erfolgreiche und effiziente SharePoint-Entwicklung besteht im Aufbau und der Konfiguration einer robusten und stabilen Entwicklungsumgebung. Generell gibt es mehrere Möglichkeiten, wie eine solche Umgebung aufgebaut werden kann. Am effektivsten ist es erfahrungsgemäß, die Entwicklung auf einer lokalen Maschine durchzuführen, auf der eine SharePoint-Instanz lokal installiert und eingerichtet ist. Diese Umgebung kann am einfachsten und schnellsten eingerichtet werden, indem die Installation des SharePoint 2010 Servers auf einer Microsoft-Windows-7- bzw. Windows-Vista-Clientmaschine durchgeführt wird. Eine etwas komplexere, dafür aber optimalere Umgebung wird erreicht, wenn als Betriebssystem Microsoft Windows Server 2008 R2 verwendet wird. Geht man diesen Schritt und wechselt zu einem Serverbetriebssystem, können innerhalb des Windows Servers 2008 R2 über Hyper-V mehrere SharePoint-Instanzen verwaltet werden. Zusätzlich besteht natürlich die Möglichkeit, eine Umgebung mit mehreren physikalischen Servern für die Entwicklung bereitzustellen. Somit stehen ein dedizierter Active Directory Domain Controller, ein dedizierter Microsoft SQL Server sowie dedizierte SharePoint Server zur Verfügung. Da diese Variante jedoch eher die Ausnahme darstellt, wird darauf hier nicht weiter eingegangen. Ist es nicht möglich, den SharePoint 2010 Server auf einer lokalen Entwicklungsmaschine zu installieren, ist auch eine Remote-Entwicklung möglich. Die Entscheidung, welche Entwicklungsumgebung verwendet wird, ist abhängig von mehreren Faktoren. In kleinen Unternehmen mit wenigen, meist nur einem, SharePoint-Entwickler eignet sich die Entwicklung unter einem Clientbetriebssystem wie Windows 7 sehr gut. Müssen jedoch mehrere SharePoint-Projekte, teilweise für unterschiedliche (Kunden-)Projekte, verwaltet werden, eignet sich die Windows-Server-Variante besser. Für erste Test- und Entwicklungszwecke kann auch auf eine vorhandene SharePoint-Umgebung zurückgegriffen werden. Microsoft stellt für Partner eine komplett installierte Demonstrationsumgebung zur Verfügung. Die Umgebung wird auf einer portablen 500-GB-Festplatte zur Verfügung gestellt und beinhaltet die SharePoint-Installation der imaginären Firma Contoso. Somit steht sofort ein SharePoint Server bereit, der schon eine Seitenstruktur und Benutzer besitzt. Im Wesentlichen stellt das Image folgende Testversionen bereit:

- Windows Server 2008 R2 Standard Edition x64, inkl. Active Directory, DNS und WINS
- Microsoft SQL Server 2008 R2 Enterprise Edition
- Microsoft Visual Studio 2010
- Microsoft SharePoint Server 2010 Enterprise Edition
- Microsoft FAST Search for SharePoint 2010

- Microsoft Office Professional Plus 2010
- und weitere

Die folgenden Kapitel beschreiben die verschiedenen Varianten im Detail.

1.1 Allgemeine Anforderungen

Bevor auf die einzelnen Installationsvarianten näher eingegangen wird, werden im Folgenden die allgemeinen Hard- und Softwareanforderungen beschrieben. Da es sich bei dem SharePoint Server um ein Serversystem handelt, muss vor allem die Hardware bestimmte Voraussetzungen erfüllen. Auch wenn lediglich ein Entwicklungssystem eingerichtet wird, sollte die vorhandene Hardware entsprechend gut ausgelegt werden. Andernfalls sinkt die Entwicklungsproduktivität erheblich, dafür steigt die Entwicklerfrustration. Am Ende dieses Kapitels werden beispielhaft zwei optimale Entwicklungsumgebungen aus Hardwaresicht beschrieben.

1.1.1 Hardware

Die folgenden Erläuterungen gelten für eine typische SharePoint-Installation für Entwicklungszwecke. Alle Komponenten werden also auf einer Maschine bereitgestellt. Weitere Informationen über Hard- und Softwareanforderungen für eine produktive (Farm-) Bereitstellung sind unter folgendem Link abrufbar: *Hardware and software requirements (SharePoint Server 2010)* (<http://technet.microsoft.com/en-us/library/cc262485.aspx>)

Eingesetzt werden sollte eine 64-bit CPU ab 2.5 GHz mit mindestens vier Kernen. Die Mindestanforderung an den Arbeitsspeicher liegt bei 4 GB RAM. Für produktive Maschinen werden 8 GB RAM empfohlen. Aufgrund der Preise für Arbeitsspeicher sind aber auch für eine Entwicklungsmaschine 8 GB RAM empfehlenswert. Die Anforderung an Festplattenspeicherplatz beginnt bei 80 GB. Dieser Wert ist natürlich abhängig davon, welche Installationsvariante verwendet wird. Wird Windows Server 2008 R2 verwendet, um über Hyper-V mehrere virtuelle SharePoint-Instanzen zu verwalten, ergibt sich ein entsprechend höherer Wert. Neben diesen technischen Anforderungen sollte jeder Entwicklerarbeitsplatz über mindestens zwei Bildschirme verfügen, um eine effektive Entwicklung zu unterstützen.

1.1.2 Software

Wie in der Einleitung schon deutlich wurde, kann der SharePoint Server 2010 auf folgenden 64-bit-Betriebssystemen installiert werden:

- Windows 7/64-bit
- Windows Vista Service Pack 1/64-bit

- Windows Vista Service Pack 2/64-bit
- Windows Server 2008 Service Pack 2/64-bit
- Windows Server 2008 R2/64-bit

HINWEIS: Windows 7 und Vista sind ausschließlich für Entwicklungsumgebungen verwendbar. Eine produktive Bereitstellung darüber ist nicht möglich.

Die folgenden Datenbanksysteme werden unterstützt:

- Eingebaute MS-SQL-Express-Variante (vollständige Installation)
- Microsoft SQL Server 2008 R2/64-bit
- Microsoft SQL Server 2008 Service Pack 1/64-bit
- Microsoft SQL Server 2005 Service Pack 3/64-bit

Die beiden letztgenannten SQL-Datenbanksysteme benötigen darüber hinaus die Installation kumulativer Updates. Weitere Informationen dazu sind unter <http://technet.microsoft.com/en-us/library/cc262485.aspx#section4> abrufbar.

Wie erkennbar ist, werden nur 64-bit-Systeme unterstützt. Eine Installation basierend auf einem existierenden Windows Server 2003 ist nicht möglich.

1.2 SharePoint 2010 unter Windows 7

Findet die SharePoint-Entwicklung innerhalb eines Unternehmens bzw. ausschließlich für ein Unternehmen statt, kann als Betriebssystemplattform für die SharePoint-Entwicklung Windows 7 verwendet werden. Das ist gegenüber den Vorgängervarianten, z. B. dem SharePoint Server 2007, eine erhebliche Verbesserung. Frühere SharePoint-Versionen ließen nur eine Installation auf einem Serverbetriebssystem, z. B. Windows Server 2003, zu.

HINWEIS: Eine auf Windows 7 basierende SharePoint-2010-Server-Installation darf jedoch nicht für den produktiven Einsatz verwendet werden. Daher muss neben der eigentlichen SharePoint-Entwicklungsumgebung eine Testumgebung vorhanden sein, die der späteren Produktionsumgebung entspricht.

Um effektiv auf einer Windows-7-Maschine entwickeln zu können, muss eine 64-bit-kompatible CPU zum Einsatz kommen. Werden ausschließlich Lösungen basierend auf der SharePoint Foundation Plattform entwickelt, sollte die Maschine über mindestens 2 GB (empfohlen 4 GB) Arbeitsspeicher verfügen. Sollen Lösungen für den SharePoint Server umgesetzt werden, verdoppeln sich die beiden genannten Größen für den Arbeitsspeicher.

Microsoft weist des weiteren darauf hin, dass unter Windows 7 lediglich eine vollständige SharePoint-Installation durchgeführt werden sollte. Eine Farm-Installation sollte vermieden werden. Lediglich der Microsoft SQL-Server kann als eigenständiger Server betrieben werden.

Bevor die eigentliche SharePoint-Installation gestartet werden kann, müssen einige Vorarbeiten bzw. Vorbereitungen getroffen werden. Zunächst müssen benötigte Softwarekomponenten auf der Maschine installiert werden. Die dafür eigentlich vorhandene Installationsroutine *PrerequisiteInstaller.exe* kann unter Windows 7 nicht verwendet werden, somit müssen die notwendigen Pakete manuell installiert werden.

HINWEIS: Die hier beschriebene Vorgehensweise kann auch unter Windows Vista durchgeführt werden. Dabei müssen lediglich einige zusätzliche Vorarbeiten und Konfigurationen durchgeführt werden. Da ich davon ausgehe, dass die wenigsten Entwickler Windows Vista einsetzen, wird hierauf nicht weiter eingegangen. Der MSDN-Artikel *Setting Up the Development Environment for SharePoint 2010 on Windows Vista, Windows 7, and Windows Server 2008* beschreibt die einzelnen Schritte im Detail.

1.2.1 Installation vorbereiten

Wie bereits angedeutet wurde, müssen vor der eigentlichen SharePoint-Server-Installation noch einige Vorbereitungen getroffen werden. Für die Installation ist eine Anmeldung an der lokalen Maschine mit Administrationsrechten notwendig. Danach müssen die folgenden Schritte durchgeführt werden:

1. Kopieren der Installationsdatei(en) in ein lokales Verzeichnis, z. B. *c:\SPS-InstallFiles*, auf dem Rechner.
2. Liegt die Installationsdatei als komprimierte Datei vor, muss sie zunächst extrahiert werden. Hierbei ist zu unterscheiden, ob lediglich die SharePoint Foundation oder der SharePoint Server installiert werden soll. Im Fall einer SharePoint-Foundation-Installation werden die Dateien über den Befehl *c:\SPS-InstallFiles\SharePoint /extract:c:\SharePointFiles* entpackt. Für den SharePoint Server ist hingegen folgender Aufruf notwendig: *c:\SPS-InstallFiles\OfficeServer /extract:c:\SharePointFiles*.
3. Installation des Microsoft Filter Pack 2.0. Das dafür notwendige Installationspaket ist unter *c:\SPS-InstallFiles\PrerequisiteInstallerFiles\FilterPack\FilterPack.msi* zu finden.
4. Zusätzlich müssen die folgenden Komponenten installiert werden:
 - Microsoft Sync Framework
 - SQL Server Native Client
 - Windows Identity Foundation (Windows6.1-KB974405-x64.msu)

- Die folgenden Komponenten sind nur notwendig, wenn der SharePoint Server installiert wird. Für die Installation der SharePoint Foundation werden die Komponenten nicht benötigt.
 - Chart Controls
 - SQL Server Analysis Services – ADOMD.Net
- 5. Windows-7-Rollen aktivieren. Die Aktivierung der notwendigen Rollen kann wahlweise über ein Skript oder manuell über die Oberfläche erfolgen. Der Weg der manuellen Aktivierung erfolgt über das Einschalten der entsprechenden Windows-Funktionen. Dazu wird innerhalb der *Systemsteuerung der Punkt Programme und Funktionen* geöffnet. Über die Auswahl des Punkts *Windows-Funktionen aktivieren oder deaktivieren* können die Rollen verwaltet werden. Abbildung 1.1 zeigt die für die Installation notwendigen Rollen.

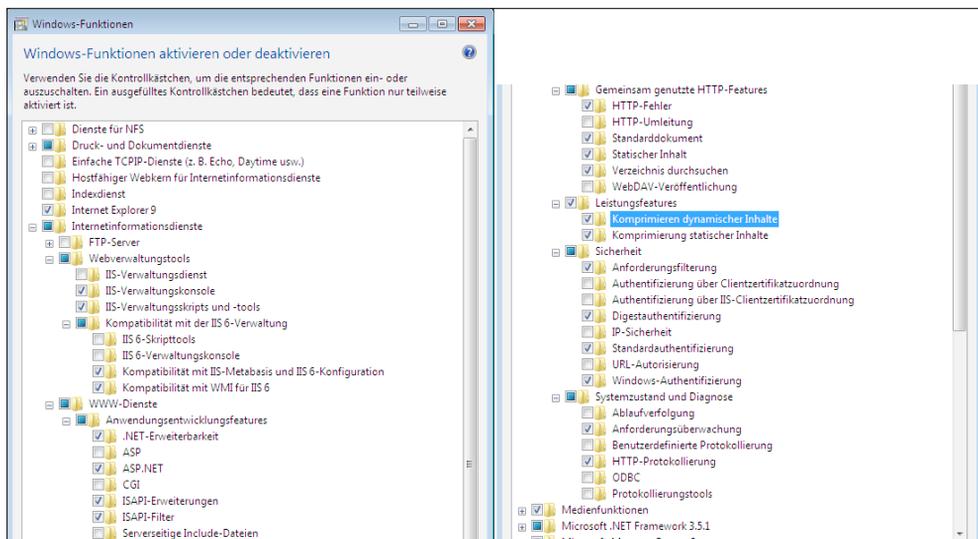


Abbildung 1.1: Die notwendigen Windows-Funktionen

Nun muss die SharePoint-Installationsroutine noch für die Installation auf einer Clientmaschine vorbereitet werden. Dazu muss der Datei *setup.config*, die sich im Verzeichnis *C:\Install SPS 2010\Files\Setup* befindet, ein Eintrag hinzugefügt werden. Zu ergänzen ist die Einstellung: `<Setting Id="AllowWindowsClientInstall" Value="True"/>`. Nach der Änderung sollte die Datei den in Listing 1.1 gezeigten Inhalt besitzen. Wird lediglich die SharePoint Foundation installiert, ist die Datei etwas kürzer.

```
<Configuration>
  <Setting Id="AllowWindowsClientInstall" Value="True"/>
  <Package Id="sts">
```

```
<Setting Id="LAUNCHEDFROMSETUPSTS" Value="Yes"/>
</Package>
<Package Id="spswfe">
  <Setting Id="SETUPCALLED" Value="1"/>
</Package>
<Logging Type="verbose" Path="%temp%"
  Template="SharePoint Server Setup(*).log"/>
<!--<PIDKEY Value="Enter Product Key Here" />-->
<Setting Id="SERVERROLE" Value="SINGLESERVER"/>
<Setting Id="USINGUIINSTALLMODE" Value="1"/>
<Setting Id="SETUPTYPE" Value="CLEAN_INSTALL"/>
<Setting Id="SETUP_REBOOT" Value="Never"/>
</Configuration>
```

Listing 1.1: Inhalt der SharePoint-2010-Installationskonfigurationsdatei

Wurden die genannten Schritte allesamt erfolgreich durchgeführt, kann die eigentliche Installation gestartet werden.

1.2.2 Installation durchführen

Die eigentliche Installation ist nun vergleichbar mit einer normalen Installation auf einem Serversystem. Die Installation wird über die *setup.exe* aus dem lokalen Verzeichnis *c:\SPS-InstallFiles* heraus gestartet. Als Installationstyp ist für eine lokale und vollständige Bereitstellung der Punkt *Standalone*, wie es Abbildung 1.2 zeigt, zu wählen.

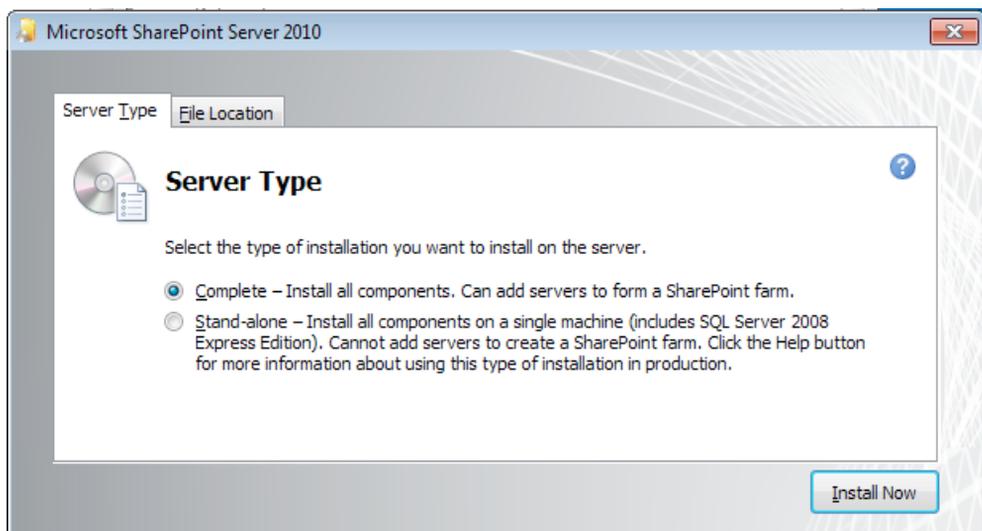


Abbildung 1.2: Auswahl der Installationsvariante

HINWEIS: Gemäß den Microsoft-Empfehlungen ist eine Farm-Installation unter Windows 7 nicht optimal. Dennoch kann der notwendige MS SQL Server ausgelagert werden, bzw. es kann eine vorhandene Instanz verwendet werden. Um das zu ermöglichen, müssen zuvor einige Einstellungen manuell durchgeführt werden. Weitere Informationen dazu am Ende dieses Kapitels.

Im ersten Schritt werden die SharePoint-Installationsdateien kopiert und registriert. Nachdem dieser Schritt erfolgreich beendet wurde, kann im Anschluss der SharePoint-Konfigurationsassistent gestartet werden. Da es sich um eine vollständige Installation handelt, werden alle SharePoint-Komponenten installiert, die vorhandenen Einflussmöglichkeiten sind daher begrenzt. Falls es während der Konfigurationsphase zu Problemen kommt, liegt das oft an zuvor nicht installierten Softwarekomponenten. In der Regel stoppt der Assistent die Konfiguration und berichtet den Fehler. Falls aus der Fehlermeldung die Ursache nicht abgeleitet werden kann, befindet sich innerhalb des Dialogs ein Link zu der Protokolldatei. Die Protokolldatei enthält detaillierte Meldungen und unterstützt somit bei der Fehleranalyse.

Während der Konfigurationsphase müssen lediglich zwei Entscheidungen getroffen werden. Zunächst muss die Vorlage ausgewählt werden, die für das initiale Portal verwendet werden soll, Abbildung 1.3 zeigt die Auswahlmöglichkeiten. Hier sollte, wenn nichts dagegen spricht, aus dem Register *Enterprise* die Vorlage *Veröffentlichungsportal* ausgewählt werden.

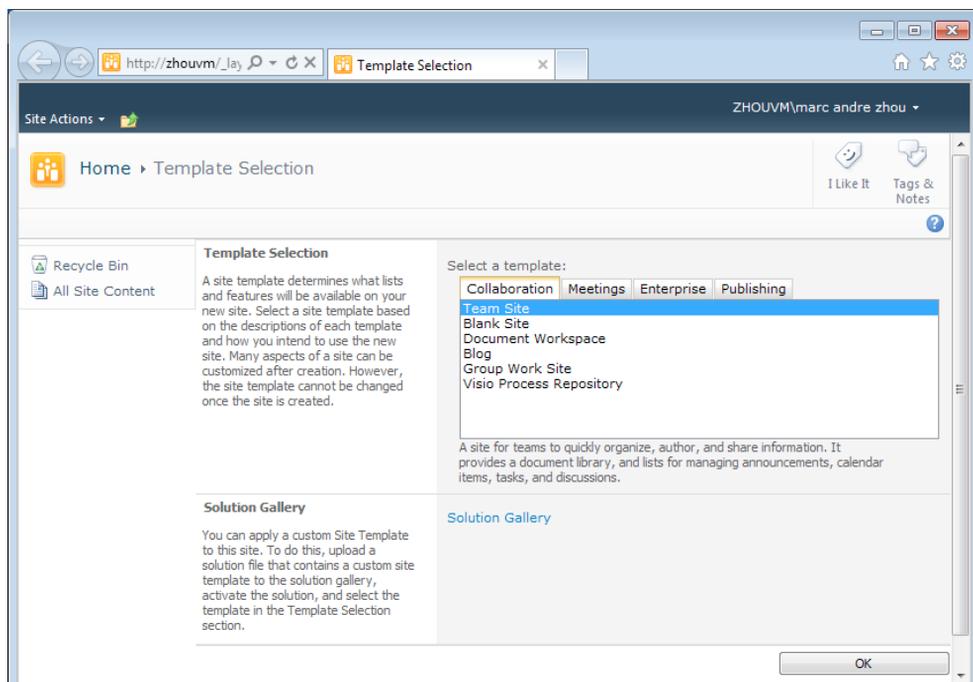


Abbildung 1.3: Auswahl der Portalvorlage

Nachdem die Vorlage ausgewählt wurde, müssen die Sicherheitsgruppen definiert werden. Hier können die vorhandenen SharePoint-Gruppen verwendet oder auch neue angelegt werden.

Wurden die beiden Entscheidungen getroffen, wird das Portal angelegt und die Startseite wird geladen und angezeigt. Somit ist die SharePoint-Installation unter Windows 7 beendet.

1.2.3 Verwendung einer vorhandenen MS-SQL-Server-Instanz

Im vorigen Kapitel wurde die vollständige SharePoint-Installation unter Windows 7 beschrieben. Bei einer vollständigen so genannten *Standalone-Installation* wird die Microsoft-SQL-Express-Datenbank installiert und als Datenbanksystem verwendet. Oft ist jedoch bereits ein Microsoft SQL Server vorhanden und die bestehende Instanz soll verwendet werden. In diesem Fall muss zu Beginn der Installation zwingend der Punkt *Server Farm* gewählt werden. Hierbei gibt es allerdings ein Problem, da bei einer Farm-Installation eine Active-Directory-Domäne bzw. ein AD-Benutzerkonto benötigt wird, ansonsten kommt es bei der Ausführung des SharePoint-Konfigurationsassistenten zu einer Fehlermeldung, wie es Abbildung 1.4 zeigt.

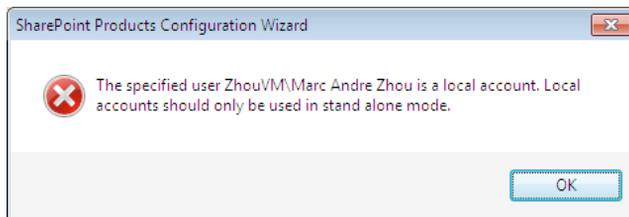


Abbildung 1.4: Fehlermeldung bei der Verwendung eines lokalen Benutzerkontos

Das beschriebene Problem tritt auf, wenn der Assistent die Zugangsdaten für die MS-SQL-Server-Instanz abfragt. Um dennoch eine Server-Farm-Installation durchführen zu können, muss daher einfach die notwendige Konfigurationsdatenbank manuell per PowerShell angelegt werden, das geschieht über den Befehl *New-SPConfigurationDatabase*. Der Befehl benötigt folgende Informationen:

- Datenbankname
- Datenbankserver
- Farm-Zugangsdaten
- Passphrase

Um den Befehl ausführen zu können, muss zunächst über *Programme -> Microsoft SharePoint 2010 Produkt -> SharePoint 2010 Management Shell* die SharePoint PowerShell geöffnet werden. Danach kann, wie auch in Abbildung 1.5 zu sehen ist, der Befehl abgesetzt werden.

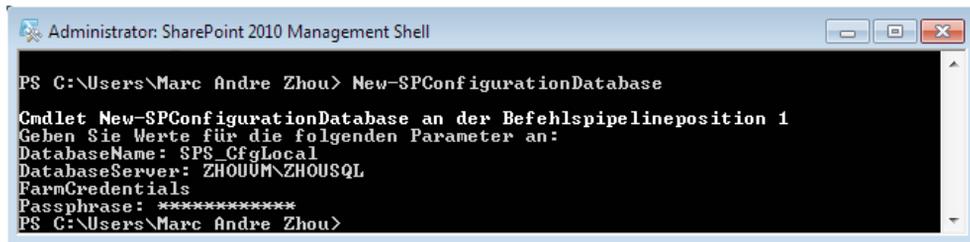


Abbildung 1.5: Ausführen des „New-SPConfigurationDatabase“-Befehls

PROFITIPP: Kommt es bei der Ausführung zu einer Fehlermeldung der Art „Invalid-Data“, sollte sichergestellt werden, dass die PowerShell mit Administrationsrechten ausgeführt wird. Das kann erreicht werden, indem bei der Auswahl des Menüpunkts *SharePoint 2010 Management Shell* über das Kontextmenü der Punkt *Als Administrator ausführen* gewählt wird.

Wurde die Konfigurationsdatenbank erfolgreich erstellt, kann danach wieder der SharePoint-Konfigurationsassistent gestartet werden. Da nun bereits eine Datenbank vorhanden ist, werden die gefundenen Datenbankinformationen im ersten Schritt des Assistenten angezeigt, wie es auch Abbildung 1.6 zeigt. An dieser Stelle kann der Assistent fortgesetzt werden.

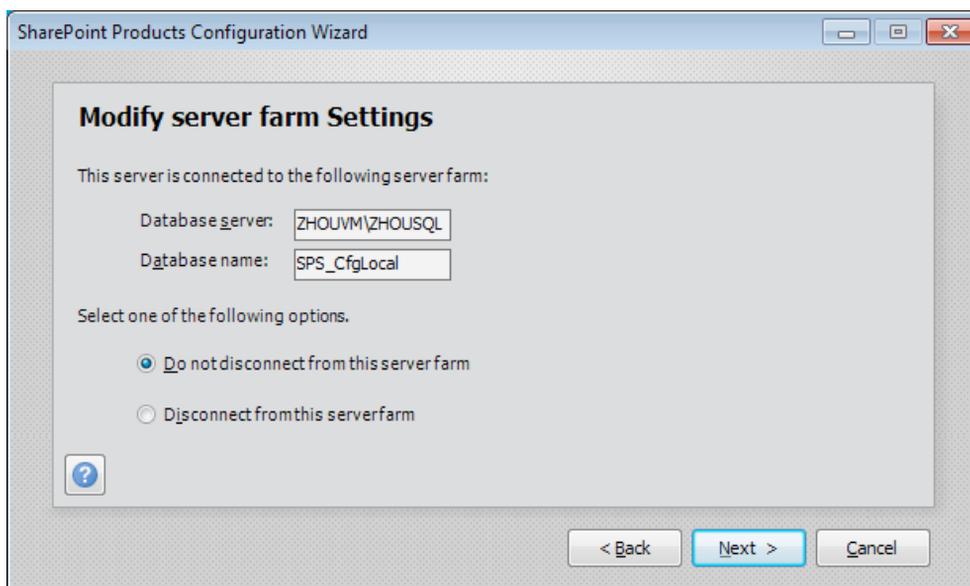


Abbildung 1.6: Anzeige der aktuellen Konfigurationsdatei

Nachdem der Assistent durchgelaufen ist und die Konfiguration erfolgreich beendet wurde, ist die initiale Installation abgeschlossen. Über die Zentraladministration kann nun die neue SharePoint-Instanz verwaltet werden.

PROFITIPP: Ist es innerhalb der Zentraladministration nicht möglich, eine neue Webapplikation zu erstellen, oder andere Funktionen sind unerklärlich deaktiviert, sollte die Benutzerkontosteuerung abgeschaltet und die Maschine neu gestartet werden. Danach sind alle Aktionen wieder durchführbar.

1.3 SharePoint Server 2010 unter Windows Server 2008 R2

Die optimale Betriebssystemgrundlage für den SharePoint Server 2010 stellt Windows Server 2008 R2 dar. Die beste Installationsvariante besteht darin, das Betriebssystem direkt auf der Hardware – also ohne Virtualisierung – zu installieren. Bei dieser Variante ändert sich zwangsweise auch das Betriebssystem der Entwicklungsmaschine. Dennoch muss man als Entwickler nicht gänzlich auf die Vorteile eines Clientbetriebssystems, z. B. Windows 7, verzichten. Es gibt eine Vielzahl von Blogbeiträgen die beschreiben, wie Windows Server 2008 R2 als Workstation-Betriebssystem konfiguriert werden kann. Eine empfehlenswerte Beschreibung ist unter <http://www.win2008workstation.com> zu finden.

Wird Windows Server 2008 R2 als Betriebssystem für die SharePoint-Entwicklungsmaschine verwendet, ergibt sich zusätzlich ein weiterer Vorteil. Da diese Kombination weitestgehend der späteren produktiven Umgebung entspricht, werden im Vorfeld eventuelle Probleme vermieden, die aufgrund unterschiedlicher Systemplattformen auftreten könnten.

Bevor jedoch der SharePoint Server auf dem System installiert werden kann, muss der Windows Server die folgenden Rollen bereitstellen:

- Windows Active Directory Service
- Domain Name Service (DNS)
- Anwendungsserver (ASP.NET, IIS usw.)

Die folgende Beschreibung geht von einer vollständigen Bereitstellung aller Komponenten auf einer Maschine aus. Besteht bereits eine eingerichtete Active-Directory-Struktur oder ist bereits eine MS-SQL-Server-Instanz verfügbar, kann natürlich auch die vorhandene Infrastruktur verwendet werden.

Die vollständige Installation aller notwendigen Komponenten hat allerdings den Vorteil, dass keinerlei Abhängigkeiten zu einem bestimmten Netzwerk bestehen. Das ist beson-

ders dann interessant, wenn auch offline entwickelt bzw. Präsentationen durchgeführt werden müssen.

HINWEIS: Da bei einer vollständigen Installation viele Komponenten auf einer Maschine installiert und betrieben werden, sollte die Hardware entsprechend großzügig dimensioniert werden.

1.3.1 Installation und Konfiguration des Serverbetriebssystems

Zunächst kann das Betriebssystem normal und im Standardmodus installiert werden. Für eine clientbasierte Installation reicht die Windows Server 2008 R2 Standard Edition aus. Auf die Verwendung der Enterprise-Version kann also verzichtet werden. Die hier verwendete Wahl der Version hat auch keinen Einfluss auf die spätere SharePoint-Version (Standard oder Enterprise).

HINWEIS: Wird das System in einer virtuellen Umgebung installiert, z. B. unter der kostenlosen VMWare-Server-Variante, sollte als Basisfestplatte eine Mindestgröße von 80 GB vorgesehen werden.

Nach der Basisinstallation ist es ratsam, per Windows Update alle neuen und gefundenen Aktualisierungen zu installieren. So wird sichergestellt, dass das System dem neuesten Patch-Level entspricht.

Ist das Basissystem bereit, können die notwendigen Rollen aktiviert werden. Als Erstes ist es empfehlenswert, die Active-Directory-Rolle zu konfigurieren, da für weitere Installationen Benutzerkonten benötigt werden.

1.3.2 Windows-Server-2008-Rollen

Um eine vollständige Umgebung zu installieren, müssen der neuen Serverinstanz zunächst folgende Rollen hinzugefügt werden:

- Active Directory Domain Services: Required .NET Framework 3.5.1
- Application Server: Required Process Model/ .NET-Umgebung/Configuration APIs
- Webserver

Wie bereits beschrieben, sollte zunächst die Active-Directory-Rolle hinzugefügt werden. Dabei ist zu beachten, dass nicht alle notwendigen Rollen gleichzeitig hinzugefügt werden können. Bei dem Versuch, neben der Active-Directory-Rolle weitere Rollen hinzuzufügen, erscheint die in Abbildung 1.7 gezeigte Fehlermeldung.



Abbildung 1.7: Die Active-Directory-Rolle kann nur separat installiert werden

Daher wird im ersten Schritt ausschließlich die Active-Directory-Rolle hinzugefügt und konfiguriert. Im abschließenden zweiten Schritt können dann die beiden Rollen *Application Server* und *Web Server* hinzugefügt werden.

Das Hinzufügen einer Rolle wird über den Server Manager, der in Abbildung 1.8 abgebildet ist, in Windows Server 2008 initiiert. Im Register ist der Link *Add Roles* zum Hinzufügen einer Rolle verfügbar.

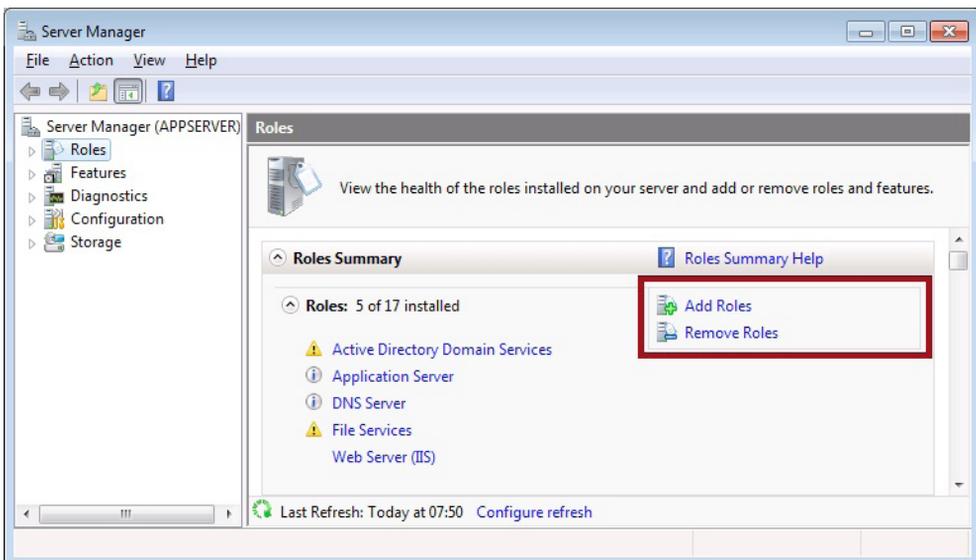


Abbildung 1.8: Server Manager

Nach Klick auf den Punkt *Add Roles* erscheint der in Abbildung 1.9 dargestellte Assistent, über den die gewünschten Rollen ausgewählt werden können. Die Active-Directory-Rolle wird über den Punkt *Active Directory Domain Services* hinzugefügt.

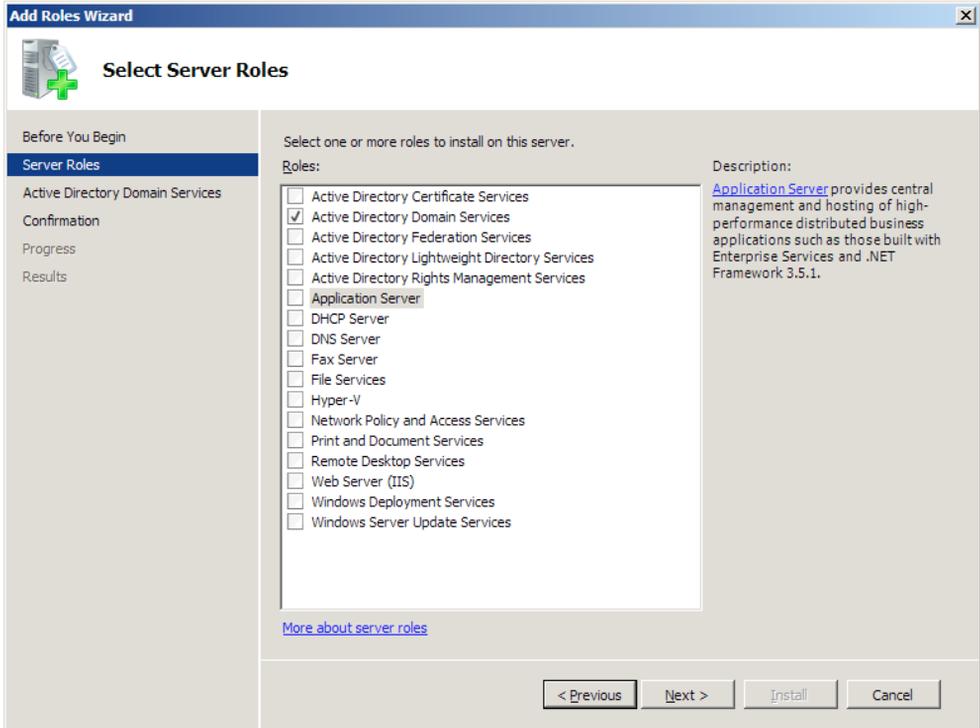


Abbildung 1.9: Add Role Wizard

Wurden die notwendigen Dateien für den Active-Directory-Dienst installiert, muss das Verzeichnis konfiguriert werden. Dazu kann am einfachsten der Konfigurationsassistent *dcpromo.exe* gestartet werden. Der Assistent führt durch die Active-Directory-Konfiguration und richtet das Verzeichnis ein. Der Assistent besteht im wesentlichen aus 8 Schritten:

1. Begrüßung.
2. Entscheidung: Neuanlage eines Verzeichnisses oder bestehendes Verzeichnis verwenden. Hier ist der Punkt *Create a new domain in a new forest* zu wählen.
3. Festlegung des vollständigen Namens des Verzeichnisses, z. B. *spsdemo.appserver.de*.
4. Einstellung des Kompatibilitätsmodus. Da in der Regel kein weiterer Client auf den Server zugreift bzw. keine weitere ältere Domäne hinzugefügt wird, kann hier der neueste Modus *Windows Server 2008 R2* gewählt werden.
5. Hinzufügen weiterer Funktionen. Da es sich um den ersten Domänen-Controller handelt, wird automatisch der Global Catalog installiert. Zusätzlich sollte auch der Domain Name Service (DNS-Dienst) installiert und konfiguriert werden.

6. Angabe der Verzeichnisse für: Datenbank, Logdateien und das SYSVOL.
7. Festlegung eines Passworts für den Wiederherstellungsadministrator.
8. Abschluss des Assistenten.

Nach der Einrichtung des Active Directory sollte der Server neu gestartet werden. Danach werden die beiden verbleibenden Rollen *Application Server* und *Web Server (IIS)* dem Server hinzugefügt. Um diese Rollen hinzuzufügen, wird der Punkt *Add Roles* im Server Manager aktiviert. Danach startet der Assistent, der die notwendigen Informationen sammelt. Auf der zweiten Seite, die auch in Abbildung 1.10 abgebildet ist, sind die beiden Punkte *Application Server* und *Web Server (IIS)* auszuwählen.

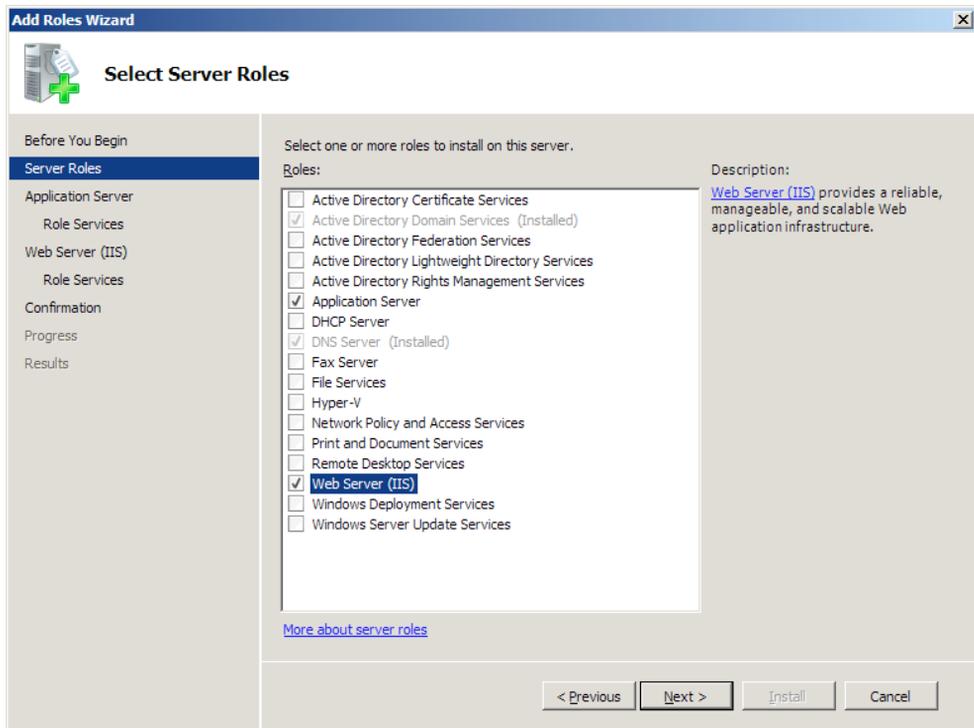


Abbildung 1.10: Serverrollen hinzufügen

Die ausgewählte Rolle Web Server (IIS) benötigt einige weitere Rollen und Features, wie es in Abbildung 1.11 zu sehen ist. Diese weiteren Funktionen müssen über die Schaltfläche *Add Required Role Service* zusätzlich installiert werden.

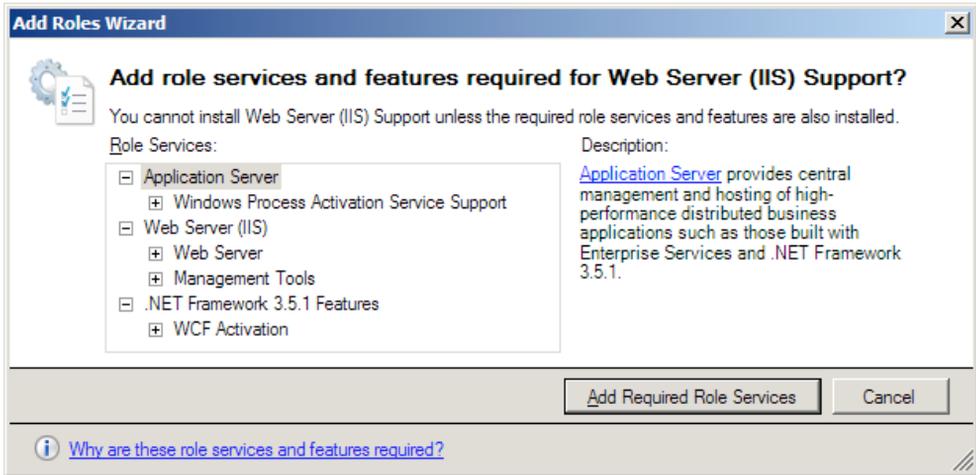


Abbildung 1.11: Abhängige Rollen und Features

Danach werden die erforderlichen Dienste (Services) für die Webserver-Rolle abgefragt. Hier sollte sichergestellt sein, dass folgende Punkte ausgewählt sind:

- Common HTTP Features
- Static Content
- Default Document
- Directory Browsing
- HTTP Errors
- Application Development
- ASP.NET
- .NET Extensibility
- ISAPI Extensions
- ISAPI Filters
- Health and Diagnostics
- HTTP Logging
- Logging Tools
- Request Monitor
- Tracing
- Security

- Basic Authentication
- Windows Authentication
- Digest Authentication
- Request Filtering
- Performance
- Static Content Compressions
- Dynamic Content Compressions
- Management Tools
- IIS Management Console
- IIS 6 Management Compatibility
- IIS 6 Metabase Compatibility

Auf der nächsten Seite des Assistenten erscheint noch einmal eine Zusammenfassung der zu installierenden Rollen mitsamt den getroffenen Einstellungen. Nach der erfolgreichen Installation sollte ein Neustart des Servers durchgeführt werden, bevor mit der eigentlichen SharePoint-Installation fortgefahren wird.

1.3.3 SharePoint-2010-Installation

Wurde der Windows Server 2008 eingerichtet, wie in den vorherigen Abschnitten erläutert, und konfiguriert, kann mit der SharePoint-Installation begonnen werden. Bevor jedoch die eigentliche SharePoint-Installation gestartet werden kann, müssen die folgenden Softwarepakete und Updates installiert werden:

- Microsoft SQL Server 2008 Native Client
- Hotfix for Microsoft Windows (KB976462)
- Windows Identity Foundation (KB974405)
- Microsoft Sync Framework Runtime v1.0 (x64)
- Microsoft Chart Controls for Microsoft .NET Framework 3.5
- Microsoft Filter Pack 2.0
- Microsoft SQL Server 2008 Analysis Services ADOMD.NET
- Microsoft Server Speech Platform Runtime (x64)
- Microsoft Server Speech Recognition Language - TELE(en-US)
- SQL 2008 R2 Reporting Services SharePoint 2010 Add-in

Die aufgezählten Punkte müssen jedoch nicht separat und einzeln installiert werden. Der SharePoint-Installationsassistent beinhaltet dafür den so genannten *Prerequisites* Installer. Um ihn zu starten, kann am einfachsten die SharePoint *setup.exe* vom Installationsmedium ausgeführt werden. Daraufhin wird der in Abbildung 1.12 gezeigte Willkommensbildschirm angezeigt. Über *Install software prerequisites* werden die benötigten Komponenten automatisch installiert.



Abbildung 1.12: Installieren benötigter Funktionen

Im Anschluss der Vorinstallation kann die eigentliche SharePoint-Installation über *Install SharePoint Server* (Abbildung 1.12) gestartet werden. Als Installationsvariante steht in dieser Konstellation lediglich die vollständige Bereitstellung via Farm-Installation zur Auswahl.

HINWEIS: Eine *Standalone-Installation* wird nicht angeboten, da die Expressvariante des SQL Servers nicht auf einem Active Directory Controller unterstützt wird.

Im Fall einer Farm-Installation muss zunächst sichergestellt werden, dass eine Microsoft-SQL-Server-Instanz verfügbar ist. Hier kann entweder auf eine vorhandene Instanz zurückgegriffen werden, oder der MS SQL Server wird zusätzlich auf dem Server installiert. Beachtet werden muss allerdings die verwendete Version des MS SQL Servers, da nicht alle Versionen unterstützt werden. Damit der SQL Server von SharePoint Server 2010 verwendet werden kann, ist der Einsatz folgender Editionen möglich:

- Microsoft SQL Server 2008 R2 in der 64-bit-Version
- Microsoft SQL Server 2008 in der 64-bit-Version mit Service Pack 1 und kumulativen Aktualisierungspaket 2 für SQL Server 2008 mit Service Pack 1. Weitere Informationen dazu sind unter <http://go.microsoft.com/fwlink/?LinkId=165962> zu finden.
- Microsoft SQL Server 2005 in der 64-bit-Version mit Service Pack 3 und dem kumulativen Aktualisierungspaket 3 für SQL Server 2005 mit Service Pack 3. Weitere Informationen dazu sind unter <http://go.microsoft.com/fwlink/?LinkId=165748> zu finden.

HINWEIS: An dieser Stelle wird nicht detailliert auf die Installation eines MS SQL Servers eingegangen. Weitere Informationen dazu sind im Artikel *Installieren von SQL Server 2008 R2 (Setup)* unter der Adresse <http://msdn.microsoft.com/de-de/library/ms143219.aspx> abrufbar.

Um die Installation durchführen zu können, sollte ein eigenes Benutzerkonto angelegt und verwendet werden. Dieses Benutzerkonto wird typischerweise als „Installationskonto“ (engl.: Setup User Account) bezeichnet und für die initiale Installation verwendet. Dieses neue Benutzerkonto muss wie folgt konfiguriert werden, damit die Installation unter dem Benutzer problemlos durchgeführt werden kann:

- Das Benutzerkonto muss Domänenbenutzerberechtigungen besitzen
- Das Benutzerkonto muss Mitglied der lokalen Administratorgruppe sein
- Werden unter dem Benutzer PowerShell-Kommandos ausgeführt, die sich auf Datenbanken auswirken, muss das Benutzerkonto die Datenbankrolle *db_owner* besitzen.
- Während der Installation muss das Benutzerkonto die Datenbankrollen *securityadmin* und *dbcreator* zugewiesen bekommen.

Neben dem Installationskonto muss noch ein Benutzerkonto für den Farm-Dienst (Farm Service) im Vorfeld erstellt werden. Dieses Konto muss lediglich Domänenbenutzer-Kontoberechtigungen besitzen.

Steht eine unterstützte MS-SQL-Server-Instanz bereit und die notwendigen Benutzerkonten wurden eingerichtet, kann die SharePoint-Installation gestartet werden.

HINWEIS: Für die Installation muss das zuvor eingerichtete Installationskonto verwendet werden. Daher ist vor Beginn der Installation sicherzustellen, dass der korrekte Benutzer angemeldet ist.

Hierzu wird einfach wieder die *setup.exe* vom Installationsmedium gestartet. Wurde der Lizenzschlüssel eingegeben und die Lizenzbedingungen akzeptiert, können noch die