

IBM WebSphere Application Server - Express Version 5.1



# Migrationshandbuch

#### **Hinweis**

Vor der Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen unter „**Bemerkungen**“ auf Seite 67 gelesen werden.

- Die IBM Homepage finden Sie im Internet unter: **ibm.com**
- IBM und das IBM Logo sind eingetragene Marken der International Business Machines Corporation.
- Das e-business-Symbol ist eine Marke der International Business Machines Corporation.
- Infoprint ist eine eingetragene Marke der IBM.
- ActionMedia, LANDesk, MMX, Pentium und ProShare sind Marken der Intel Corporation in den USA und/oder anderen Ländern.
- C-bus ist eine Marke der Corollary, Inc. in den USA und/oder anderen Ländern.
- Java und alle auf Java basierenden Marken und Logos sind Marken der Sun Microsystems, Inc. in den USA und/oder anderen Ländern.
- Microsoft Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.
- PC Direct ist eine Marke der Ziff Communications Company in den USA und/oder anderen Ländern.
- SET und das SET-Logo sind Marken der SET Secure Electronic Transaction LLC.
- UNIX ist eine eingetragene Marke der Open Group in den USA und/oder anderen Ländern.
- Marken anderer Unternehmen/Hersteller werden anerkannt.

#### **Sechste Ausgabe (Dezember 2003)**

Diese Veröffentlichung ist eine Übersetzung des Handbuchs  
*IBM WebSphere Application Server - Express Version 5.1 Migration Guide*,  
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2000, 2003  
© Copyright IBM Deutschland GmbH 2000, 2003

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:  
SW TSC Germany  
Kst. 2877  
Dezember 2003

---

# Inhaltsverzeichnis

## Kapitel 1. WebSphere Application Server - Express Migration - Übersicht . . . . . 1

## Kapitel 2. Ihren Produktionsserver migrieren . . . . . 3

Migration . . . . .	3
Migration und Koexistenz . . . . .	4
Migrationstools . . . . .	5
Befehl WASPreUpgrade. . . . .	5
Befehl WASPostUpgrade . . . . .	7
Konfigurationszuordnung während der Migration . . . . .	8
Konfigurationsdaten manuell migrieren . . . . .	11
Migration von V3.5.x auf V5.1 . . . . .	12
Migration von V3.5.x auf eine ferne V5.1-Maschine . . . . .	14
Migration von V5.0.x auf V5.1 . . . . .	16
Migration von V5.0.x auf eine ferne V5.1-Maschine . . . . .	17
Migration von einem nicht unterstützten Betriebssystem . . . . .	18

## Kapitel 3. Von IBM WebSphere Studio Site Developer Version 5.1 migrieren . . . . . 21

Migrieren von J2EE-Projekten zur Nutzung der Server Targeting-Unterstützung. . . . .	21
Abwärtskompatibilität mit aktivierter Server Targeting-Unterstützung . . . . .	21
Für die Assistentengenerierung ist ein Java-Paket für JDK 1.4 erforderlich. . . . .	22

## Kapitel 4. Von IBM WebSphere Studio Site Developer Version 5 oder 5.0.1 migrieren. . . . . 23

WebSphere Studio Workbench in Version 5, Version 5.0.1 und Version 5.1 . . . . .	23
IBM WebSphere Studio Site Developer Version 5.1.1 mit dem Arbeitsbereich von Version 5.0 verwenden . . . . .	23
Migration der Java-Projekte von Version 5 oder Version 5.0.1 . . . . .	24
Projekte aus Version 5 oder Version 5.0.1 mit Version 5.1.1 unter Verwendung eines Quellcodeverwaltungssystems gemeinsam benutzen . . . . .	24
Webprojekte migrieren. . . . .	26
Konvertierung der Webprojekte auf Struts 1.1 . . . . .	27
Änderungen an Web-Service-Tools. . . . .	27
Änderungen an Profilermittlungstools . . . . .	28
Bekanntete Probleme des Schablonenassistenten mit Kompatibilität . . . . .	28

## Kapitel 5. Von IBM WebSphere Studio Site Developer Version 4.0.x migrieren . . . . . 29

Unterschiede zwischen IBM WebSphere Studio Site Developer Version 4.0.x und Version 5 . . . . .	29
Änderungen an WebSphere Application Server und Tools für die Servlet/JSP-Konvertierung. . . . .	30

Interne Änderungen gegenüber Version 4.0.3 . . . . .	31
Schleifenabhängigkeiten von Projekten werden nicht standardmäßig erstellt . . . . .	31
Kompatibilität der Quellenposition von Webprojekten aus Version 5 mit Version 4.0.3. . . . .	31
Strukturen von Webprojekten in IBM WebSphere Studio Site Developer . . . . .	31
Unterschiede zwischen statischen und dynamischen Webprojekten . . . . .	32
Unterschiede zwischen HTML und JSP . . . . .	32
Projekte mit einem System zur Softwarekonfigurationsverwaltung (SCM) migrieren . . . . .	33
Projekte mit CVS oder Rational ClearCase migrieren . . . . .	33
Absolute Pfadverweise für EAR und Serverkonfiguration nach der Migration entfernen . . . . .	35
Projekte mit anderen SCMs migrieren . . . . .	35
Projekte durch Export und Import migrieren . . . . .	36
Projekte mit einem vorhandenen Arbeitsbereich aus Version 4.0.x migrieren . . . . .	36
Absolute Pfadverweise für EAR und Serverkonfiguration nach der Migration entfernen . . . . .	37
Bekanntete Probleme und Einschränkungen . . . . .	37
Relationale Daten in 4.0.3-Webprojekten migrieren . . . . .	38
WSDL-Fehler nach Import einer Web-Service-datei von 4.0.x . . . . .	39
J2EE-Projektstrukturen und/oder J2EE-Spezifikationsstufen migrieren . . . . .	39

## Kapitel 6. Migration von WebSphere Studio "Classic" auf IBM WebSphere Studio Site Developer . . . . . 41

Neue Einzelserverumgebung für die Migration erstellen . . . . .	42
Deskriptordatei für Webkonfiguration erstellen . . . . .	42
JAR-Migrationsdatei erstellen . . . . .	42
JAR-Migrationsdatei in IBM WebSphere Studio Site Developer importieren. . . . .	43
Migrierte Anwendung auf einem Testserver testen . . . . .	44

## Kapitel 7. Migration von VisualAge für Java auf IBM WebSphere Studio Site Developer . . . . . 45

Unterschiede zwischen VisualAge für Java und IBM WebSphere Studio Site Developer . . . . .	45
Migration von VisualAge für Java . . . . .	46
Java-Dateien und Projektressourcendateien aus VisualAge für Java exportieren . . . . .	46
IBM WebSphere Studio Site Developer starten und neue Projekte erstellen, die Ihren Code enthalten sollen . . . . .	47
Java- und Ressourcendateien in IBM WebSphere Studio Site Developer importieren. . . . .	47

Korrekte Definition von Servlets unter Verwendung des Editors web.xml sicherstellen (nur bei Webprojekten) . . . . .	48
Projekt- und Arbeitsbereichseinstellungen migrieren . . . . .	48
Testumgebung in WebSphere V4 definieren und migrierte Anwendung(en) testen . . . . .	50
Ihre Anwendungen aus IBM WebSphere Studio Site Developer auf fernem WebSphere Application Server implementieren . . . . .	50
Projekteinstellungen für IBM WebSphere Studio Site Developer mehreren Entwicklern zur gemeinsamen Benutzung zur Verfügung stellen (nach der Migration) . . . . .	50
Teamunterstützung in IBM WebSphere Studio Site Developer . . . . .	50

**Kapitel 8. Migration von VisualAge für Java Visual Composition Editor auf Visual Editor für Java . . . . . 51**

Metadaten für Entwurfszeit aus VisualAge für Java sichern . . . . .	51
Migration (Import in WebSphere Studio) vollständig abschließen . . . . .	52

**Kapitel 9. Erstellungskonfiguration (Bibliothek, JARs, abhängige Projekt-JARs, Ant-Erstellungen) . . . . . 53**

Java-Bibliotheks-JARs und externe Dritthersteller-JARs. . . . .	53
Empfohlene Methode für die Verwendung von Dritthersteller-JARs in Webprojekten . . . . .	53
Empfohlene Methode für die Verwendung von Dritthersteller-JARs mit mehreren Webprojekten . . . . .	54
Alternative Methode für die Verwendung externer JAR-Dateien (globaler Erstellungs- und Serverklassenpfad) . . . . .	54

Erstellung mehrerer Projekte mit abhängigen Projekt-JARs optimieren . . . . .	55
Produktionserstellungen mit Ant automatisieren . . . . .	55

**Kapitel 10. Migrationsbeispiele . . . . . 57**

Beispiel für JSP/Servlet aus VisualAge für Java (LeapYear). . . . .	57
Dateien aus VisualAge für Java exportieren. . . . .	57
Neues Webprojekt für IBM WebSphere Studio Site Developer erstellen . . . . .	58
Java- und Projektressourcendateien in das Projekt von IBM WebSphere Studio Site Developer importieren . . . . .	58
Servlets definieren und Anwendungsänderungen wegen Umstrukturierung vornehmen. . . . .	59
Serverprojekt für IBM WebSphere Studio Site Developer erstellen . . . . .	59
Migrierte Anwendung LeapYear testen . . . . .	60
Beispiel für Webanwendung (YourCo) aus WebSphere Studio "Classic" Version 4.0 (Windows) . . . . .	60
WebSphere Studio "Classic" Version 4.0 starten und neue Migrationsumgebung erstellen . . . . .	61
Deskriptordatei für Webkonfiguration erstellen . . . . .	61
Migrationsdatei erstellen . . . . .	61
IBM WebSphere Studio Site Developer starten und die WAR-Datei importieren . . . . .	62
Serverprojekt für IBM WebSphere Studio Site Developer erstellen . . . . .	62
Migrierte Anwendung YourCo testen . . . . .	63

**Kapitel 11. Weiterführende Informationen . . . . . 65**

<b>Bemerkungen. . . . . 67</b>	
Informationen zu Programmierschnittstellen . . . . .	69
Marken und Dienstleistungsmarken . . . . .	69

---

# Kapitel 1. WebSphere Application Server - Express Migration - Übersicht

In dieser Version von IBM® WebSphere® Application Server - Express Version 5.1 können Sie den Code aus folgenden Programmen migrieren:

- VisualAge für Java
- IBM WebSphere Application Single Server Edition Version 3.5
- IBM WebSphere Application Server - Express Version 5.0, 5.0.1, 5.0.2
- WebSphere Studio "Classic"
- IBM WebSphere Studio Site Developer
- IBM WebSphere Studio Site Developer Version 5 (Beta-, Early Availability- oder General Availability-Versionen)
- IBM WebSphere Studio Site Developer Version 5.0.1
- IBM WebSphere Studio Site Developer Version 5.1

WebSphere Application Server - Express 5.1 setzt sich aus WebSphere Application Server 5.1 und WebSphere Studio Site Developer 5.1.1 zusammen. Im ersten nachfolgenden Kapitel wird die Migration der Serverfunktion von WebSphere Application Server - Express behandelt. Die übrigen Kapitel dieses Migrationshandbuchs befassen sich mit der Migration von Code aus verschiedenen Versionen von WebSphere Studio Site Developer.

## Wichtiger Hinweis hinsichtlich der Migration des Servers:

Eine Migration Ihrer Serverkonfiguration ist nur dann von Bedeutung, wenn Sie den Server unter Verwendung der Administrationskonsole verwalten, so wie in der Regel in einer Produktionsumgebung üblich. Bei dieser Betriebsart werden die Serverkonfiguration und die implementierten Anwendungen im Konfigurationsverzeichnis des Servers gespeichert. Der Migrationsprozess migriert diese Konfiguration und die Anwendungsdateien für Sie. Wenn Sie jedoch WebSphere Studio Site Developer zum Konfigurieren und Implementieren von Anwendungen auf Ihrem fernen Server verwenden, müssen Sie die Serverkonfigurationsdateien nicht migrieren. Die Konfigurations- und Anwendungsdateien werden in diesem Fall im Studio Site Developer-Arbeitsbereich verwaltet. Der Arbeitsbereich wird von Studio Site Developer migriert. Sie können anschließend ein neues Exemplar eines WebSphere Application Server - Express 5.1-Servers definieren und Ihre Anwendungen von Studio Site Developer aus implementieren.

Das vorliegende Handbuch enthält die folgenden Kapitel:

- Ihren Produktionsserver migrieren
- Migration von IBM WebSphere Studio Site Developer Version 5.1
- Migration von IBM WebSphere Studio Site Developer Version 5 oder Version 5.0.1
- Migration von IBM WebSphere Studio Site Developer von Version 4.0.x
-  Migration von WebSphere Studio "Classic" auf IBM WebSphere Studio Site Developer
- Migration von VisualAge für Java auf IBM WebSphere Studio Site Developer
- Migration von VisualAge für Java Visual Composition Editor auf Visual Editor für Java

- Erstellungskonfiguration (Bibliothek, JARs, abhängige Projekt-JARs, Ant-Erstellungen)
- Migrationsbeispiele
- Weiterführende Informationen

Informationen zur Verwendung von WebSphere Application Server - Express finden Sie im Handbuch *Erste Schritte* und in der Onlinehilfefunktion. Lesen Sie das *Installationshandbuch* bevor Sie WebSphere Application Server - Express installieren. Nachdem Sie WebSphere Application Server - Express erfolgreich installiert haben, lesen Sie das Handbuch *Erste Schritte*, und arbeiten Sie die Lernprogramme unter *Erste Schritte* durch. Die Lernprogramme stellen Ihnen die Workbench, die Java™-Entwicklung und die Web-Services vor. Nachdem Sie die Lernprogramme abgeschlossen haben, können Sie Ihre Anwendungsressourcen unter Anleitung des vorliegenden Handbuchs auf WebSphere Application Server - Express migrieren.

Die HTML- und Acrobat PDF-Versionen dieses Handbuchs finden Sie im Verzeichnis /readme. Der Inhalt beider Versionen ist identisch. Sie können migrate.html in einem beliebigen Web-Browser öffnen. Um migrate.pdf aufzurufen, müssen Sie die Acrobat Reader-Software installiert haben. Sie können diese unter der Adresse [www.adobe.com/products/acrobat/readstep2.html](http://www.adobe.com/products/acrobat/readstep2.html) herunterladen.

Im vorliegenden Handbuch *Migration* werden durchgängig Windows®-Konventionen verwendet. Die Angabe "WS\_Installdir\" unter Windows ist beispielsweise äquivalent zur Angabe "WS\_Installdir/" unter Linux.

Künftige Aktualisierungen dieses Handbuchs finden Sie unter der Adresse [www.ibm.com/websphere/developer/zones/studio/transition.html](http://www.ibm.com/websphere/developer/zones/studio/transition.html).

---

## Kapitel 2. Ihren Produktionsserver migrieren

---

### Migration

Die Migration ermöglicht die Weiterverwendung vorhandenen Materials. Migrationstasks und -tools helfen Ihnen bei der Ausführung von Upgrades für das Produkt und die erforderlichen Voraussetzungen, der Wiederverwendung vorhandener Anwendungskomponenten, sofern möglich, und der Übertragung von Verwaltungskonfigurationen aus Ihrer früheren Version in eine aktuelle Version.

Die Migration von WebSphere Application Server-Produkten bewirkt, dass die vorhandene Umgebung und vorhandene Anwendungen weiter verwendet werden können und stellt deren Kompatibilität mit der aktuellen Produktversion sicher.

Funktionen zur Produktmigration werden von den Migrationstools in IBM WebSphere Application Server - Express, Version 5.1 bereitgestellt. Die Migrationstools unterstützen die Migration von folgenden Versionen:

- IBM WebSphere Application Server - Express Version 5.0, 5.0.1, 5.0.2
- IBM WebSphere Application Single Server Edition Version 3.5

Der Produktinstallationsassistent stellt fest, ob bereits ältere Versionen von IBM WebSphere Application Server - Express vorhanden sind und gibt Ihnen die Möglichkeit, während der Installation von Version 5.1 die Migrationstools auszuführen. Zum Migrieren von IBM WebSphere Application Server Version 3.5 müssen Sie diese Tools direkt ausführen.

#### Redbook zur Migration

Die Veröffentlichung *Migrating to WebSphere V5: An End-to-End Migration Guide* ist auf der Website für Redbooks unter <http://www.ibm.com/redbooks> verfügbar. Um das Redbook auf der Website zu finden, suchen Sie nach der Dokumentnummer SG24-6910-00. Das Redbook enthält umfassendere Informationen als dieser Artikel, z. B. ausführlichere Informationen zur Planung der Anwendungsmigration und den Tools und Beispielen von WebSphere Studio Application Developer.

#### Migration von Version 3.5: Wechsel auf das J2EE-Modell

Benutzer von V3.5.x, die einen Upgrade auf V5 ausführen, wechseln damit auf eine Plattform, die auf den J2EE-Spezifikationen basiert. Die J2EE-Technologie zieht eine deutliche Trennung zwischen der Erstellung von Anwendungen und der Verwaltung und Implementierung von Anwendungen. Die Migration von V3.5 umfasst Änderungen in Anwendungsstrukturen, Entwicklung und Implementierung.

Die Migrationstools helfen beim Übergang von Version 3.5.x auf Version 5 durch Migration der Systemkonfiguration und der Erstellung von J2EE-Artefakten, einschließlich der J2EE-Sicherheitsberechtigungsklassenzuordnung. Die Migrationstools erstellen auf Basis der Konfigurationen von Version 3.5.x erste J2EE-Unternehmensanwendungen. Auf Grund der erheblichen Änderungen in den Anwendungsstrukturen sollten Sie unter Verwendung der Entwicklungs- und Implementierungstools die migrierten Anwendungen sorgfältig testen und optimieren, um genau festzustellen, wie die Anwendungen in der neuen Umgebung funktionieren.

Das J2EE-Modell ermöglicht es, Anwendungen unabhängig von ihrer endgültigen Implementierungsumgebung zu entwickeln. Diese Tasktrennung vereinfacht den Hochstufungsprozess einer Anwendung vom Beginn der Entwicklung über die gesamte Produktion bzw. das Versetzen einer Anwendung von einem Server auf einen anderen. Dadurch sollen jeweils nur die Implementierungsparameter der Anwendung und nicht der Anwendungscode geändert werden müssen.

---

## Migration und Koexistenz

Bevor Sie beginnen, stellen Sie fest, ob auf der Maschine, auf der Sie Ihr Produkt der Version 5.1 installieren wollen, bereits eine Version von WebSphere Application Server installiert ist. Ist dies der Fall, müssen Sie planen, ob Sie die Konfiguration und die Anwendungen der früheren Version in die neue Version kopieren wollen, d. h. ob Sie eine *Migration* vornehmen wollen. Bei der Migration wird die frühere Version nicht deinstalliert. Dieses frühere Release bleibt weiterhin funktionsfähig. Wenn Sie es gleichzeitig mit der Installation von Version 5.1 ausführen, besteht zwischen den beiden Versionen eine *Koexistenz*. Damit Sie beide Versionen zur selben Zeit ausführen können, müssen Sie die Ports so konfigurieren, dass es nicht zu Konflikten kommt. Beachten Sie, dass die Migrationsoperation die Portdefinitionen nur "wie vorhanden" migriert, sodass Sie für beide Versionen die selben Definitionen vorliegen haben. WebSphere Application Server enthält Migrationstools, die die gesamte Migrationsfunktionalität bereitstellen. Der Installationsassistent kann die Migrationstools aufrufen, aber Sie können die Tools auch zu einem späteren Zeitpunkt manuell aufrufen.

Im Überblick ist die Migration von IBM WebSphere Application Server - Express V5.0.x auf V5.1 eine Routineoperation. Sie können das Installationsprogramm zur Migration verwenden, sodass Sie anschließend nur wenige oder gar keine Optimierungsoperationen ausführen müssen. Sie können die Migrationstools aber auch manuell einsetzen, um die Konfigurationsdaten von V5.0.0, V5.0.1 oder V5.0.2 zu speichern, V5.0.0, V5.0.1 oder V5.0.2 zu deinstallieren, V5.1 zu installieren und können die Migrationstools erneut verwenden, um die Konfigurationsdaten wiederherzustellen.

Im Überblick umfasst die Migration von IBM WebSphere Application Server V3.5 auf IBM WebSphere Application Server - Express V5.1 wesentliche Änderungen hinsichtlich der Anwendungsstrukturen, der Entwicklung und der Implementierung. Die Migrationstools unterstützen Sie bei diesem Übergang durch Migration von Systemkonfigurationen und Erstellung von J2EE-Artefakten, einschließlich der Zuordnung von früheren Sicherheitseinstellungen zu J2EE-Sicherheitsberechtigungsklassen. Diese Sicherheitszuordnungen ermöglichen Ihnen während dieses Übergangs den Zugriff auf migrierte Ressourcen. Die Migrationstools erstellen auf Basis der Konfigurationen von V3.5.x erste J2EE-Unternehmensanwendungen. Auf Grund der erheblichen Änderungen in den Anwendungsstrukturen sollten Sie unter Verwendung der Entwicklungs- und Implementierungstools die migrierten Anwendungen jedoch sorgfältig testen und optimieren.

Bei der Migration werden die folgenden Dateien im Sicherungsverzeichnis gespeichert.

### Für Version 3.5.x

- bin/setupCmdLine.sh (oder bin/setupCmdLine.bat für Windows-Plattformen)
- classes (werden nicht für iSeries gespeichert)
- hosts

- properties
- servlets

#### **Für Version 5.0.x**

- classes (werden nicht für iSeries gespeichert)
- config
- installableApps
- installedApps
- properties

---

## **Migrationstools**

In diesem Abschnitt werden die Migrationstools vorgestellt, die mit WebSphere Application Server zur Verfügung gestellt werden. Alle Migrationstools befinden sich im Verzeichnis `/migration` auf der Produkt-CD-ROM. Es ist wichtig, dass Sie jeweils die Migrationstools für die Version von Application Server verwenden, die Sie installieren. Die Tools unterliegen Änderungen im Lauf der Zeit. Die Tools, die auf der Produkt-CD-ROM enthalten sind, bieten die erforderliche Funktionalität für die Migration von einem früheren Release von Application Server auf die Version der Produkt-CD-ROM. Die auf der CD-ROM mitgelieferten Tools entsprechen dem Produkt auf der CD-ROM. Wenn Sie Migrationstools eines früheren Releases von Application Server verwenden, treten bei der Migration höchstwahrscheinlich Probleme auf.

#### **WASPreUpgrade.sh (und WASPreUpgrade.bat)**

Speichert die Anwendungen und Konfigurationsdaten einer früheren Installation von WebSphere Application Server in einem Sicherungsverzeichnis. Das Script `WASPostUpgrade` stellt die Konfigurationsdaten aus dem Verzeichnis in der neuen Installation wiederher. Das Installationsprogramm ruft während der Installation das Script `WASPreUpgrade.sh` auf, wenn Sie die Migration auswählen. Sie können auch nach der Installation der neuen Version den Befehl zum Ausführen einer manuellen Migration verwenden.

#### **WASPostUpgrade.sh (und WASPostUpgrade.bat)**

Stellt die Konfigurationsdaten aus einem früheren Release wiederher. `WASPostUpgrade` liest die Daten aus dem Sicherungsverzeichnis, in dem das Script `WASPreUpgrade` die Daten gespeichert hat. Das Installationsprogramm ruft während der Installation das Script `WASPostUpgrade.sh` auf, wenn Sie die Migration auswählen. Sie können auch nach der Installation der neuen Version den Befehl zum Ausführen einer manuellen Migration verwenden.

---

## **Befehl WASPreUpgrade**

Das Script `WASPreUpgrade.sh` (oder `WASPreUpgrade.bat`) ist ein Migrationstool, mit dem die Konfiguration und Anwendungen früherer Versionen oder Releases auf Version 5.1 Application Server - Express migriert werden können.

Die Befehlsdatei befindet sich nach der Installation im Unterverzeichnis `AppServer/bin` des Installationsstammverzeichnisses. Sie steht auch direkt auf der CD zur Verfügung, im Unterverzeichnis `migration`.

## Syntax

```
WASPreUpgrade sicherungsverzeichnis aktuellesWASVerzeichnis  
               [adminKnotenname]  
               [-nameServiceHost hostname [-nameServicePort portnummer ]]  
               [-traceString trace_spezifikation [-traceFile dateiname ]]
```

## Parameter

Die ersten beiden Argumente sind erforderlich und positionsgebunden.

Es werden folgende Argumente unterstützt:

### **sicherungsverzeichnis**

Der erforderliche, positionsgebundene Name des Verzeichnisses, in dem das Tool WASPreUpgrade die gespeicherte Konfiguration und die Dateien speichert, aus denen das Tool WASPostUpgrade später die Konfiguration und die Dateien liest. Das Tool WASPreUpgrade erstellt dieses Verzeichnis, wenn es noch nicht vorhanden ist.

### **aktuellesWASVerzeichnis**

Der erforderliche, positionsgebundene Name des Installationsstammverzeichnisses für die aktuelle Installation von V3.5.x oder V5.0.x. Bei dieser Version kann es sich um WebSphere Application Server Standard Edition, V3.5.x oder WebSphere Application Server - Express V5.0.x handeln.

### **adminKnotenname**

Optionaler, positionsgebundener Name des Knotens, der den Administrationsserver für das aktuell installierte Produkt enthält. Der Wert dieses Arguments muss mit dem Knotennamen übereinstimmen, der in der Topologiebaumstruktur auf der Registerkarte Topologie der Administrationskonsole für das aktuell installierte Produkt angegeben ist. Das Tool WASPreUpgrade ruft das Tool XMLConfig mit diesem Parameter auf. Dieser Parameter ist nur erforderlich, wenn Sie einen Upgrade für WebSphere Application Server Standard Edition, Version 3.5.x ausführen.

### **-nameServiceHost -nameServicePort**

Wenn dieser Parameter angegeben ist, übergibt das Tool WASPreUpgrade diese optionalen Parameter an das Tool XMLConfig. Verwenden Sie diese Parameter, um die vom Tool XMLConfig verwendeten Standardwerte für Hostname und Portnummer zu überschreiben.

### **-traceString -traceFile**

Optionale Parameter, mit denen Trace-Informationen für den IBM Kundendienst gesammelt werden. Geben Sie als *trace\_spezifikation* den Wert "*\*=all=enabled*" an (mit Anführungszeichen), um alle Trace-Informationen zu sammeln.

## Protokollierung

Das Tool WASPreUpgrade zeigt während der Ausführung den Status auf dem Bildschirm an. Es speichert außerdem umfangreiche Protokolldaten im *sicherungsverzeichnis*. Sie können die Datei WASPreUpgrade.log mit einem Texteditor anzeigen.

---

## Befehl WASPostUpgrade

Das Script `WASPostUpgrade.sh` (oder `WASPostUpgrade.bat`) ist ein Migrationstool, mit dem die Konfiguration und Anwendungen früherer Versionen oder Releases auf Version 5.1 Application Server - Express migriert werden können.

Die Befehlsdatei befindet sich im Unterverzeichnis `AppServer/bin` des Installationsstammverzeichnisses.

Das Tool `WASPostUpgrade` installiert alle migrierten Anwendungen im Verzeichnis `AppServer/installedApps` für die Installation der Version 5.1. Das Tool enthält Anwendungen aus dem Sicherungsverzeichnis, das vom Tool `WASPreUpgrade` erstellt wird. Das Tool `WASPreUpgrade` kopiert die Anwendungen aus dem Verzeichnis `installedApps` und anderen Verzeichnissen der früheren Version oder des früheren Releases.

### Syntax

```
WASPostUpgrade sicherungsverzeichnis
                [-serverName servername]
                [-webModuleAdditionalClasspath klassenpfad]
                [-documentRootLimit nummer]
                [-substitute "schlüssel1=wert1[;schlüssel2=wert2;...]"]
                [-portBlock portstartnummer]
                [-backupConfig true | false]
                [-replacePorts true | false]
                [[-traceString trace_spezifikation [-traceFile dateiname]]
```

### Parameter

Das erste Argument ist erforderlich. Es werden folgende Argumente unterstützt:

#### **serverName**

Optionaler Serverexemplarname. Der Standardwert lautet `server1`.

#### **sicherungsverzeichnis**

Der erforderliche Name des Verzeichnisses, in dem das Tool `WASPreUpgrade` die gespeicherte Konfiguration und die Dateien speichert, aus denen das Tool `WASPostUpgrade` die Konfiguration und die Dateien liest. Das Tool `WASPreUpgrade` erstellt dieses Verzeichnis, wenn es noch nicht vorhanden ist.

#### **-backupConfig**

Optionaler Parameter, der zum Speichern der vorhandenen Konfiguration verwendet wird, bevor die Migrationstools die Konfiguration ändern. Der Standardwert lautet `"true"`, die Konfiguration wird gespeichert.

#### **-documentRootLimit**

Optionaler Parameter, mit dem die Anzahl Dateien angegeben werden kann, die das Programm aus dem Feld des Dokumentstammverzeichnisses der Webanwendung kopiert. Dieser Parameter ist nur für Upgrades für Version 3.5.x verfügbar. Wenn kein Wert angegeben wird, lautet der Standardwert 300.

#### **-portBlock**

Optionaler Parameter, mit dem der Anfangswert angegeben wird, der beim Erstellen von Ports verwendet werden soll.

#### **-substitute**

Optionales Argument, das an das Tool `XMLConfig` übergeben wird. Geben

Sie Werte für die zu ersetzenden Variablen ein (beispielsweise `-substitute "NODE_NAME=admin_knoten;APP_SERVER=standardserver"`).

In der XML-Dateneingabedatei wird jeder der zu ersetzenden Schlüssel als `$key$` angezeigt. Dieses Argument ersetzt in der XML-Eingabedatei alle Vorkommen von `$NODE_NAME$` durch `admin_knoten` und `$APP_SERVER$` durch `standardserver`.

Wenn die Ersatzzeichenfolge Semikolons enthält, verwenden Sie dafür `$semiColon$`, um sie vom Begrenzer `;` zu unterscheiden. Auf UNIX-Plattformen fügen Sie zu jedem Dollarzeichen (\$) in der Ersatzzeichenfolge ein Escapezeichen hinzu (beispielsweise `\$semiColon\`).

Dieser Parameter kann auf gespeicherte Konfigurationen von Advanced Edition, Version 3.5.x angewendet werden.

#### **-traceString -traceFile**

Optionale Parameter, mit denen Trace-Informationen für den IBM Kundendienst gesammelt werden. Geben Sie als `trace_spezifikation` den Wert `"*=all=enabled"` an (mit Anführungszeichen), um alle Trace-Informationen zu sammeln.

#### **-webModuleAdditionalClasspath**

Optionaler Parameter, mit dem der Pfad oder die Pfade und Dateinamen bestimmter Verzeichnisse oder Dateien angegeben werden können, die Sie nicht in die Webarchivdatei (WAR) kopieren wollen. Das Programm fügt die Pfade und Dateien stattdessen zum Attribut `additionalClasspath` der Webmodulerweiterung (`ibm-web-ext.xmi`) hinzu. Dieser Parameter ist nur bei Migration einer Installation von Version 3.5.x gültig.

### **Protokollierung**

Das Tool `WASPostUpgrade` zeigt während der Ausführung den Status auf dem Bildschirm an. Es speichert außerdem umfangreiche Protokolldaten im `protokollverzeichnis`. Sie können die Datei `WASPostUpgrade.log` mit einem Texteditor anzeigen.

---

## **Konfigurationszuordnung während der Migration**

In diesem Abschnitt werden die Änderungen während der Migration beschrieben, die immer eine einzelne Maschine, beispielsweise eine Entwicklungsumgebung oder eine Standalone-Maschine betrifft.

### **Migration von Version 3.5 auf Version 5.x**

Die Migrationstools helfen beim Übergang von Version 3.5.x auf Version 5 durch Migration der Systemkonfigurationen und der Erstellung von J2EE-Artefakten, einschließlich der J2EE-Sicherheitsberechtigungsklassenzuordnung. Die Migrationstools erstellen auf Basis der Konfigurationen von Version 3.5.x erste J2EE-Unternehmensanwendungen. Auf Grund der erheblichen Änderungen in den Anwendungsstrukturen sollten Sie unter Verwendung der Entwicklungs- und Implementierungstools die migrierten Anwendungen sorgfältig testen und optimieren, um genau festzustellen, wie die Anwendungen in Version 5 funktionieren.

Analysieren Sie die Datei `WASPostUpgrade.log`, wenn Sie genauere Informationen zur Migration von Enterprise-Beans erhalten wollen. Das J2EE-Programmiermodell gibt eine Architektur zur Erstellung und Implementierung von Anwendungen an. Da Anwendungen in Version 3.5.x nicht dieselbe Architektur haben, erstellt das Tool `WASPostUpgrade` Anwendungen

erneut. Es erstellt alle migrierten Webressourcen und Enterprise-Beans in J2EE-Anwendungen. Es ordnet alle Unternehmensanwendungen aus der Installation der Version 3.5.x zu J2EE-Anwendungen desselben Namens zu, die auf demselben Server implementiert werden.

Das Tool WASPostUpgrade ordnet Webressourcen, die nicht in einer Unternehmensanwendung enthalten sind, einer J2EE-Standardanwendung zu, die den Namen des Servers enthält. Das Tool ordnet Webanwendungen zu J2EE-WAR-Dateien zu. Das Tool fasst Ressourcen in einer J2EE-EAR-Datei zusammen und implementiert diese in der Konfiguration der Version 5.

### Details für die Migration von V3.5.x auf Version 5.x zuordnen

- **datasources.xml**

Sie können die Datei `datasources.xml` einer Version 3.5.x verwenden, um die Konfigurationseinstellungen für Datenquellen zu ergänzen. Version 3.5.x speichert die Datei im Verzeichnis `properties`. Die Migrationstools migrieren eine vorhandene Datei `datasources.xml`, indem die in der Datei enthaltenen Eigenschaften in die Konfiguration der Datenquelle und des JDBC-Treibers eingefügt werden.

- **Unternehmensanwendungen**

Die Einträge der Unternehmensanwendungen von Version 3.5.x sind optional, sie werden oft zum Gruppieren von Objekten für Sicherheitsdefinitionen verwendet. Die Enterprise-Bean- und Webanwendungsabschnitte der Unternehmensanwendung zeigen auf ihre jeweiligen Einträge in anderen Abschnitten der xml-Datei. Jede Unternehmensanwendung wird verarbeitet, um eine J2EE-Anwendung mit demselben Namen zu erstellen. Die Enterprise-Bean- und Webanwendungseinträge werden als Zeiger auf die Definitionen der Enterprise-Beans und Webanwendungen verwendet. Die Einzeldaten dieser Einträge werden anschließend verwendet, um eine J2EE-Anwendung zu erstellen.

Für Enterprise-Bean-Dateien wird die Definition der JAR-Datei dazu verwendet, die erneut zu implementierenden JAR-Dateien zu finden und zur J2EE-Anwendung hinzuzufügen. Die Dokumentstammverzeichniseinträge der Webanwendung werden verwendet, um die in der Webanwendung verwendeten Ressourcen zu finden (HTML, JSP-Seiten usw.). Diese Dateien werden in die WAR-Datei innerhalb der J2EE-Anwendung kopiert. Die Klassenpfadeinträge der Webanwendung werden verwendet, um Servlets und JAR-Dateien zu finden, die in die WAR-Datei innerhalb der J2EE-Anwendung kopiert werden.

Unternehmensanwendungen werden während der Migration von Version 3.5.x erstellt. Sie werden als J2EE 1.2-kompatible Unternehmensanwendungen erstellt und enthalten Module der Stufe Servlet 2.2 und JSP 1.1. Dies bietet die direkteste Kompatibilität und ermöglicht die Interoperabilität mit früheren Versionen von WebSphere Application Server.

- **J2EE-Sicherheit**

Das Sicherheitsberechtigungsmodell in Version 3.5.x basiert auf dem Konzept von Unternehmensanwendungen und Methodengruppen. Das übergreifende Produkt der Unternehmensanwendung und der Methodengruppen ist eine WebSphere Application Server-Berechtigung. Die J2EE-Spezifikation umfasst ein auf Berechtigungsklassen basierendes Berechtigungsmodell.

Wenn Sie vom Berechtigungsmodell von WebSphere Application Server Version 3.5.x in das auf Berechtigungsklassen basierende Berechtigungsmodell in Version 5 konvertieren wollen, erstellen die Migrationstools eine Eins-zu-Eins-Zuordnung zwischen einer WebSphere Application Server-Berechtigung und einer neuen Berechtigung unter dieser Anwendung. Daher erstellen die Migrationstools für jede Unternehmensanwendung und jede Methodengruppe in Version 3.5.x eine Berechtigungsklasse in Version 5, die im Implementierungsdeskriptor der J2EE-Anwendung enthalten ist. Die berechtigten Subjekte für jede Berechtigungsklasse sind in einer Berechtigungstabelle in der Bindung der J2EE-Anwendung enthalten.

Die J2EE-Spezifikation umfasst ein auf Berechtigungsklassen basierendes Berechtigungsmodell. WebSphere Application Server interpretiert die Berechtigungsklasse so, dass sie eine Reihe von Berechtigungen für den Zugriff auf eine Ressource darstellt. Wenn eine Enterprise-Bean-Methode aufgerufen wird, wird die Berechtigung für den Zugriff auf die Methode für eine bestimmte Bean von einer Methodenberechtigung angegeben. Diese Methodenberechtigung ist mindestens einer Berechtigungsklasse in der Bean-JAR-Datei zugeordnet.

Beim Zugriff auf Webressourcen wird die Berechtigung für den Zugriff auf eine Web-URI und den Aufruf einer HTTP-Methode für diese URI mittels Webressourcensammlungen und Sicherheitsbedingungen in der J2EE-Spezifikation angegeben. Der Implementierungsdeskriptor der WAR-Datei der Webanwendung enthält die Sicherheitsbedingungen und Webressourcensammlungen.

- **JSP-Stufen**

Version 5 führt Objekte der Stufe JSP 1.0 und 1.1 als JSP 1.2-Objekte aus; dies ist die einzige unterstützte Stufe.

- **Servlet Redirector**

Version 5 bietet keine Unterstützung für Servlet Redirector aus früheren Versionen. Die Migrationstools ignorieren diese Objekte.

- **Änderungen von Servletpaketnamen bei der Migration von V3.5.x auf V5.x**

Wenn die Version 3.5.x-Konfiguration das Servlet SimpleFileServlet definiert, wird das Servlet nicht migriert. Die Migrationstools setzen das Attribut FileServingEnabled in der Webmoduldatei ibm-web-ext.xmi auf true.

Wenn die Version 3.5-Konfiguration das Servlet InvokerServlet definiert, wird das Servlet nicht migriert. Die Migrationstools setzen das Attribut ServeServletsByClassnameEnabled in der Webmoduldatei ibm-web-ext.xml auf true.

Wenn die Version 3.5.x-Konfiguration das Servlet DefaultErrorReporter definiert, wird das Servlet in die Webmoduldatei web.xml migriert. Die Migration verwendet das neue Paket zum Festlegen des Klassennamens.

- **Transporte**

Der Standardtransporttype der Servlet-Steuerkomponente in Version 3.5.x ist Open Servlet Engine (OSE). Da Version 5 für OSE-Transport keine Unterstützung mehr bietet, ordnen die Migrationstools diese Transporte zu HTTP-Transporten zu und verwenden dabei dieselben Portzuordnungen. Sie müssen für jeden Port manuell Aliasnamenseinträge für VirtualHost hinzufügen.

---

## Konfigurationsdaten manuell migrieren

Sie können administrative Konfigurationen mit dem Installationsassistenten oder manuell migrieren, wie in dieser Task beschrieben. Wenn Sie sich für die manuelle Migration entscheiden, wählen Sie nicht das Markierungsfeld für Migration Fenster des Installationsassistenten aus.

Wenn Sie eine frühere Version von WebSphere Application Server verwenden, ist es möglich, dass der Systemadministrator verschiedene Anwendungs- und Servereinstellungen für Ihre Umgebung optimiert hat. Es ist wichtig, dass Sie sich eine Strategie zurechtlegen, um diese Einstellungen mit möglichst hoher Effizienz und möglichst geringen Verlusten zu migrieren.

Sie können eine schrittweise manuelle Migration ausführen, indem Sie die Migrationstools mehrmals hintereinander aufrufen und jedes Mal eine andere Konfigurationsdatei angeben. Es können aus verschiedenen Gründen mehrere Konfigurationsdateien vorhanden sein. Unabhängig von diesen Gründen können Sie beim Migrieren von jeweils nur einer Konfigurationsdatei pro Schritt die Anwendungen schrittweise testen, bevor Sie mit der nächsten Konfigurationsdatei fortfahren.

Bevor Sie die Migrationstools verwenden, lesen Sie das Dokument mit den Release-Informationen zu V5.x, um sich einen Überblick über die für frühere Versionen erforderlichen Programmkorrekturen zu verschaffen. Wenn Sie Programmkorrekturen auf eine frühere Version anwenden, werden davon möglicherweise auch Dateien betroffen, die für die Migration von Bedeutung sind. Wenden Sie alle erforderlichen Programmkorrekturen an, um möglichst die effektivste Migration von Konfigurationen und Anwendungen zu erzielen.

Manuelle Migration bietet einen übersichtlicheren, schrittweisen Migrationsverlauf als die vom Installationsassistent ausgeführte vollständige Migration. IBM stellt eine Reihe von Migrationstools zur Verfügung, mit denen administrative Konfigurationen für das Produkt WebSphere Application Server - Express von einer beliebigen Version von V3.5.x oder V5.0.x ausgeführt werden können. Der gesamte Migrationsprozess umfasst das Speichern der aktuellen Konfiguration und der erforderlichen Dateien mit dem Migrationstool WASPreUpgrade, das Deinstallieren des früheren Releases, das Installieren des Produkts der Version 5 ohne Auswahl der Option für automatische Migration sowie das Wiederherstellen der Konfiguration aus dem früheren Release mit dem Migrationstool WASPostUpgrade.

Wählen Sie eines dieser Migrationsszenarios aus, um Informationen zur Migration von Konfigurationsdaten auf einen WebSphere Application Server-Basisknoten abzurufen:

- „Migration von V3.5.x auf V5.1“ auf Seite 12
- „Migration von V3.5.x auf eine ferne V5.1-Maschine“ auf Seite 14
- „Migration von V5.0.x auf V5.1“ auf Seite 16
- „Migration von V5.0.x auf eine ferne V5.1-Maschine“ auf Seite 17
- „Migration von einem nicht unterstützten Betriebssystem“ auf Seite 18

---

## Migration von V3.5.x auf V5.1

Sie können die Migrationstools zum Migrieren von Konfigurationsdaten von Version 3.5 von WebSphere Application Server auf Version 5.1 von WebSphere Application Server - Express verwenden.

Sie würden in der Regel die Migrationstools WASPreUpgrade und WASPostUpgrade aus V5.1 von WebSphere Application Server zur Migration von V3.5 auf V5.1 auf *derselben* Maschine verwenden. Wenn Ihr Szenario die Migration einer V3.5-Konfiguration auf einer Maschine auf WebSphere Application Server - Express V5.1 auf einer anderen Maschine beinhaltet, verwenden Sie die alternative Prozedur, die in „Migration von V3.5.x auf eine ferne V5.1-Maschine“ auf Seite 14 beschrieben ist.

In diesem Abschnitt wird beschrieben, wie Sie die V5.1-Migrationstools für folgende Migration verwenden:

- WebSphere Application Server Single Server Edition, V3.5

Das Tool WASPreUpgrade speichert die vorhandene V3.5-Konfiguration in einem migrationsspezifischen Sicherungsverzeichnis *migration-specific-backup*. Das Tool WASPostUpgrade verwendet dieses Verzeichnis, um die alten Konfigurationseinstellungen zur neuen V5.1-Umgebung hinzuzufügen.

Schritte für diese Task

1. Beschaffen Sie sich die Produkt-CD-ROM von V5.1.

Auf dieser CD befindet sich ein Verzeichnis *migration/bin*. Dieses Verzeichnis enthält eine Sonderumgebung, die Sie zur Ausführung des Tools WASPreUpgrade verwenden können, ohne V5.1 zu installieren.

2. Speichern Sie die aktuelle Konfiguration mit dem Script WASPreUpgrade aus dem Verzeichnis */migration/bin* der V5.1-Produkt-CD-ROM.

Speichern Sie die Konfiguration im Verzeichnis *migration-specific-backup*:

```
WASPreUpgrade /usr/tmp/migration-specific-backup /usr/websphere/appserver IhrKnotenname
```

Stellen Sie sicher, dass der Administrationsserver der vorhandenen Umgebung aktiv ist. Das Tool WASPreUpgrade gibt den Status auf den Bildschirm und in Protokolldateien im Verzeichnis *migration-specific-backup* aus. ASCII-Protokolldateinamen beginnen mit dem Text WASPreUpgrade und enthalten eine Datumszeitmarke.

Das Tool WASPreUpgrade speichert alle Dateien aus den folgenden Verzeichnissen der vorhandenen V3.5.x-Konfiguration im Sicherungsverzeichnis:

**Für Version 3.5.x**

- bin
- classes
- hosts
- properties
- servlets

Das Tool WASPreUpgrade speichert die ausgewählten Dateien aus dem Verzeichnis */bin* der V3.5.x. Es exportiert auch die vorhandene Application Server-Konfiguration aus dem Repository von V3.5.x. Das Tool WASPreUpgrade ruft XMLConfig auf, um das vorhandene V3.5-Repository in die Datei *websphere\_backup.xml* im Verzeichnis *migration-specific-backup* zu exportieren.

Wenn während der Ausführung des Tools WASPreUpgrade Fehler auftreten, müssen Sie gegebenenfalls Programmkorrekturen auf die V3.5-Installation anwenden, um den Exportschritt erfolgreich abschließen zu können. Die neuesten, gegebenenfalls erforderlichen Programmkorrekturen finden Sie auf der Seite der IBM Unterstützungsfunktion. Wenn Sie diese Informationen über das InfoCenter ansehen, klicken Sie auf **Support**, um eine Verbindung zur Seite der IBM Unterstützungsfunktion herzustellen.

3. Installieren Sie V5.1 des Produkts WebSphere Application Server - Express. Wählen Sie die Migrationsoption nicht aus, falls sie angezeigt wird. Überprüfen Sie nach jeder Verwendung von WASPostUpgrade die V5-Porteinstellungen in den folgenden beiden Dateien:
  - Überprüfen Sie die Portzuordnung `BOOTSTRAP_ADDRESS` für `server1` in der Datei `serverindex.xml`

Wenn der Port `BOOTSTRAP_ADDRESS` der früheren Version 900 lautete, ordnet die Migration ihm den Wert 7809 zu. Wenn der Port `BOOTSTRAP_ADDRESS` der früheren Version nicht 900 lautete, ordnet die Migration ihm in einer Migration der Advanced Edition den Wert `server1` zu bzw. in einer Migration der Advanced Single Server Edition den tatsächlichen Servernamen.
  - Überprüfen Sie die Portzuordnungen für HTTP Transport in der Datei `server.xml`

Die Verarbeitung von WASPostUpgrade fügt die HTTP-Transportports der früheren Version zur Datei `server.xml` der Version 5 hinzu. Dies bedeutet, dass `server1` doppelte HTTP-Transport-Portzuordnungen enthält, aus der Koexistenzanzeige und dem *Standardserver* der früheren Version.
4. Migrieren Sie die frühere Konfiguration auf die neue Installation mit dem Tool WASPostUpgrade im Verzeichnis `AppServer/bin` des Installationsstammverzeichnisses der V5.1.

Das Tool WASPostUpgrade migriert die V3.5.x-Konfigurationsdaten, die vom Tool WASPreUpgrade erstellt wurden, auf die Installation von V5.1. Da das V5.1-Produkt das J2EE-Programmiermodell beachtet und V3.5.x dies nicht tut, sind erhebliche Änderungen erforderlich, um die V3.5.x-Konfiguration auf eine V5.1-Installation anwenden zu können.

Das Tool WASPostUpgrade migriert keine Beispiele oder die Administrationskonsolenanwendung, da in V5.1 bereits Beispiele und eine Administrationskonsolenanwendung vorhanden sind.

Das Tool WASPostUpgrade zeichnet zu jeder Enterprise-Bean, die es implementiert, ausführliche Informationen in der Datei `WASPostUpgrade.log` auf.
5. Stoppen Sie den Administrationsserver der früheren Version, falls er aktiv ist, bevor Sie den Knoten der Version 5 ausführen.
6. Das Konfigurieren von WebSphere Application Server nach der Migration ist eine Möglichkeit, die Ergebnisse der Migrationstools zu überprüfen. Sie können auch während der Migration die Konfigurationszuordnung verwenden, um die Ergebnisse der Migration zu prüfen. Der Abschnitt enthält eine ausführliche Beschreibung, wie die Migrationstools Objekte migrieren und was Sie überprüfen sollten.

---

## Migration von V3.5.x auf eine ferne V5.1-Maschine

Sie können mit den Migrationstools eine manuelle Migration zwischen zwei Maschinen ausführen.

Sie würden in der Regel die Migrationstools WASPreUpgrade und WASPostUpgrade aus V5.1 von WebSphere Application Server - Express zur Migration von V3.5 auf V5.1 auf *derselben* Maschine verwenden.

Es gibt jedoch Situationen, in denen Sie die V3.5-Konfiguration auf einer Maschine auf V5.1 auf eine andere Maschine migrieren müssen. Eines dieser Szenarios ist das Installieren neuer Maschinen für Ihre neueste V5.1-Umgebung, wobei Sie die vorhandenen V3.5-Konfigurationen auf andere Maschinen migrieren müssen.

In diesem Abschnitt wird beschrieben, wie Sie die V5.1-Migrationstools für folgende Migration verwenden:

- WebSphere Application Server Single Server Edition, V3.5

Das Tool WASPreUpgrade speichert die vorhandene V3.5-Konfiguration in einem migrationsspezifischen Sicherungsverzeichnis *migration-specific-backup*. Das Tool WASPostUpgrade verwendet dieses Verzeichnis, um die alten Konfigurationseinstellungen zur neuen V5.1-Umgebung hinzuzufügen.

Schritte für diese Task

1. Beschaffen Sie sich die Produkt-CD-ROM von V5.1.

Auf dieser CD befindet sich ein Verzeichnis *migration/bin*. Dieses Verzeichnis enthält eine Sonderumgebung, die Sie zur Ausführung des Tools WASPreUpgrade verwenden können, ohne V5.1 zu installieren.

2. Speichern Sie die aktuelle Konfiguration mit dem Script WASPreUpgrade aus dem Verzeichnis */migration/bin* der V5.1-Produkt-CD-ROM, die Sie an die V3.5-Maschine anhängen müssen.

Speichern Sie die Konfiguration im Verzeichnis *migration-specific-backup*:

```
WASPreUpgrade /opt/tmp/migration-specific-backup /opt/websphere/appserver adminKnotenname
```

Stellen Sie sicher, dass der Administrationsserver der vorhandenen Umgebung aktiv ist.

Das Tool WASPreUpgrade gibt den Status auf den Bildschirm und in Protokolldateien im Verzeichnis *migration-specific-backup* aus. ASCII-Protokolldateinamen beginnen mit dem Text WASPreUpgrade und enthalten eine Datumszeitmarke. Das Tool WASPreUpgrade speichert die ausgewählten Dateien aus dem Verzeichnis */bin* der V3.5.x. Es exportiert auch die vorhandene Application Server-Konfiguration aus dem Repository von V3.5.x. Das Tool WASPreUpgrade ruft XMLConfig auf, um das vorhandene V3.5-Repository in die Datei *websphere\_backup.xml* im Verzeichnis *migration-specific-backup* zu exportieren.

Wenn während der Ausführung des Tools WASPreUpgrade Fehler auftreten, müssen Sie gegebenenfalls Programmkorrekturen auf die V3.5-Installation anwenden, um den Exportschritt erfolgreich abschließen zu können. Die neuesten, gegebenenfalls erforderlichen Programmkorrekturen finden Sie auf der Seite der IBM Unterstützungsfunktion. Wenn Sie diese Informationen über das InfoCenter ansehen, klicken Sie auf **Support**, um eine Verbindung zur Seite der IBM Unterstützungsfunktion herzustellen.

3. Kopieren Sie das Verzeichnis /migration-specific-backup von der V3.5-Maschine auf die V5.1-Maschine.

Verwenden Sie ftp, gemeinsam benutzten Speicher oder einen anderen Mechanismus, um die Datei auf die neue Maschine zu kopieren.

Führen Sie auf der Maschine mit V5.1 von WebSphere Application Server - Express die folgenden Schritte aus.

4. Kopieren Sie die Datei /migration-specific-backup/websphere\_backup.xml oder die Datei /migration-specific-backup/config/server-cfg.xml, und speichern Sie sie an einer Position Ihrer Wahl, um die Kopie als Archiv aufzubewahren.

Sie müssen die Datei kopieren, da Sie das Original im nächsten Schritt bearbeiten.

5. Bearbeiten Sie die Datei /migration-specific-backup/websphere\_backup.xml oder die Datei /migration-specific-backup/config/server-cfg.xml, um die maschinenabhängigen Einstellungen zu korrigieren.

Nehmen Sie an der Datei die folgenden Änderungen vor:

- a. **Ändern Sie den Knotennamen** in der Datei /migration-specific-backup/websphere\_backup.xml. In der Datei /migration-specific-backup/config/server-cfg.xml ist kein Knotenname vorhanden.

Wenn Sie denselben Knotennamen für die V5.1-Maschine verwenden, den Sie für die ursprüngliche V3.5-Konfiguration verwendet haben, müssen Sie den Namen nicht ändern. Andernfalls **müssen** Sie alle Vorkommen des V3.5-Knotenname in den Knotennamen ändern, den Sie für WebSphere Application Server V5.1 verwenden. Der Knotenname kommt in vielen XML-Zeilengruppen in der gesamten Datei vor. Wenn Sie nicht alle Vorkommen ändern, ist die Migration der Daten nicht vollständig.

- b. **Ändern Sie die Pfadnamen** in der Datei /migration-specific-backup/websphere\_backup.xml oder der Datei /migration-specific-backup/config/server-cfg.xml.

Die Konfigurationsdatei bezieht sich auf Pfadnamen in vielen XML-Zeilengruppen in der gesamten Datei. Ändern Sie alle Verweise auf eine Datei außerhalb der V3.5-Verzeichnisstruktur auf das entsprechende Verzeichnis auf der neuen Maschine, auch wenn Sie dieses Verzeichnis vielleicht erst erstellen müssten. Als Folge der Konfiguration einer übereinstimmenden Umgebung müssen Sie das Originalverzeichnis gegebenenfalls auf die V5.1-Maschine kopieren. Oder Sie müssen die entsprechende Software installieren.

- c. **Korrigieren Sie Spezifikationsstile** für Pfadnamen, die vom Betriebssystem abhängig sind.

Sie müssen die Pfadspezifikationen korrigieren, wenn Sie sich von den auf der V5.1-Maschine verwendeten Spezifikationen unterscheiden. Wenn Sie beispielsweise von V3.5 auf einer Windows-Plattform auf V5.1 auf einer Linux-Plattform migrieren, müssen Sie alle Windows-spezifischen Pfade in der Konfigurationsdatei so ändern, dass Sie dem Linux-Pfadstil entsprechen. Ändern Sie "c:\mystuff\somepath" in "/opt/mystuff/somepath".

- d. **Ändern Sie Benutzer-IDs und Kennwörter entsprechend den Sicherheitsbestimmungen**

Sie müssen Benutzer-IDs und Kennwörter ändern, wenn Sie nicht mit den auf der V5.1-Maschine verwendeten identisch sind.

Wenn Sie ein verschlüsseltes Kennwort in ein Klartextkennwort ändern wollen, ändern Sie <password>{xor}LCoxayht</password> in <password>mypassword</password>.

**e. Ändern Sie andere maschinenspezifische Informationen.**

Die Konfiguration enthält eventuell Verweise auf andere Softwareprodukte oder Konfigurationen, die auf der neuen Maschine nicht vorhanden sind. Auf der alten Maschine könnte eventuell eine Datenbank vorhanden sein. Die V5.1-Konfiguration soll möglicherweise weiter auf die Datenbank auf der alten Maschine verweisen. Ändern Sie die Datenquelle, sodass sie auf die Datenquelle auf der V3.5-Maschine zeigt.

6. Installieren Sie V5.1 von WebSphere Application Server, ohne die Migrationsoption auszuwählen.
7. Fügen Sie die V3.5-Konfiguration zur V5.1-Konfiguration hinzu.

Verwenden Sie das Tool WASPostUpgrade im Verzeichnis AppServer/bin des V5.1-Installationsstammverzeichnisses, um die V3.5-Konfiguration zur V5.1-Konfiguration hinzuzufügen.

```
WASPostUpgrade /opt/tmp/migration-specific-backup
```

Das Tool WASPostUpgrade zeichnet zu jeder Enterprise-Bean, die es implementiert, ausführliche Informationen in der Datei /migration-specific-backup/WASPostUpgrade.log auf.

---

## Migration von V5.0.x auf V5.1

Sie können das V5.1-Installationsprogramm verwenden, um die Migration von WebSphere Application Server - Express V5.0.x auf V5.1 auszuführen.

Schritte für diese Task:

1. Stoppen Sie den Application Server V5.0.x.

Verwenden Sie dazu das Script stopServer.sh (oder stopServer.bat) aus dem Verzeichnis AppServer/bin des Installationsstammverzeichnisses:

```
stopServer.sh server1
```

Sie können einen V5.0.x-Knoten migrieren, ohne ihn zu stoppen. Es ist zum Migrieren seiner Konfiguration jedoch nicht erforderlich, dass der Knoten aktiv ist. Die Migrationstools können alle Konfigurationsdaten abrufen, auch wenn der Knoten gestoppt wurde. Sie müssen den Knoten jedoch stoppen, bevor Sie den V5.1-Knoten starten, den Sie installieren. Sie können den Knoten jetzt stoppen.

2. Installieren Sie das V5.1-Produkt.

Wählen Sie die Migrationsoption aus, wenn sie angezeigt wird.

3. Bestätigen Sie die Installation von Application Server V5.1.

Verwenden Sie das Tool für Erste Schritte (First Steps), wenn es am Ende der Produktinstallation angezeigt wird, oder führen Sie den Funktionstest für die Installation selbst aus, falls das Tool eventuell nicht angezeigt werden sollte.

Sie können den V5.0.x-Server nach Bedarf deinstallieren.

---

## Migration von V5.0.x auf eine ferne V5.1-Maschine

Sie können mit den Migrationstools eine Migration zwischen zwei Maschinen ausführen.

### Vorbereitungen

Sie würden in der Regel die Migrationstools WASPreUpgrade und WASPostUpgrade aus V5.1 von WebSphere Application Server zur Migration von V5.0.x auf V5.1 auf *derselben* Maschine verwenden.

Es gibt jedoch Situationen, in denen Sie die V5.0.x-Konfiguration auf einer Maschine auf V5.1 auf eine andere Maschine migrieren müssen. Eines dieser Szenarios ist das Installieren neuer Maschinen für Ihre neueste V5.1-Umgebung, wobei Sie die vorhandenen V5.0.x-Konfigurationen auf andere Maschinen migrieren müssen.

Diese Task beschreibt, wie Sie die V5.1-Migrationstools zur Ausführung der Migration verwenden.

Das Tool WASPreUpgrade speichert die vorhandene V5.0.x-Konfiguration in einem migrationspezifischen Sicherungsverzeichnis *migration-specific-backup*. Das Tool WASPostUpgrade verwendet dieses Verzeichnis, um die alten Konfigurationseinstellungen zur neuen V5.1-Umgebung hinzuzufügen.

### Schritte für diese Task

1. Beschaffen Sie sich die Produkt-CD-ROM von V5.1.  
Auf dieser CD befindet sich ein Verzeichnis *migration/bin*. Dieses Verzeichnis enthält eine Sonderumgebung, die Sie zur Ausführung des Tools WASPreUpgrade verwenden können, ohne V5.1 zu installieren.
2. Speichern Sie die aktuelle Konfiguration mit dem Script WASPreUpgrade aus dem Verzeichnis */migration/bin* der V5.1-Produkt-CD-ROM, die Sie an die V5.0.x-Maschine anhängen müssen.

Speichern Sie die Konfiguration im Verzeichnis *migration-specific-backup*:  
`WASPreUpgrade /opt/tmp/migration-specific-backup /opt/websphere/appserver`

Das Tool WASPreUpgrade gibt den Status auf den Bildschirm und in Protokolldateien im Verzeichnis *migration-specific-backup* aus. ASCII-Protokolldateinamen beginnen mit dem Text "WASPreUpgrade" und enthalten eine Datumszeitmarke.

3. Kopieren Sie das Verzeichnis */migration-specific-backup* von der V5.0.x-Maschine auf die V5.1-Maschine.  
Verwenden Sie `ftp`, gemeinsam benutzten Speicher oder einen anderen Mechanismus, um die Datei auf die neue Maschine zu kopieren.
4. Installieren Sie V5.1 von WebSphere Application Server, ohne die Migrationsoption auszuwählen.
5. Fügen Sie die V5.0.x-Konfiguration zur V5.1-Konfiguration hinzu.

Verwenden Sie das Tool WASPostUpgrade im Verzeichnis *AppServer/bin* des V5.1-Installationsstammverzeichnisses, um die V5.0.x-Konfiguration zur V5.1-Konfiguration hinzuzufügen.

`WASPostUpgrade /opt/tmp/migration-specific-backup`

Das Tool WASPostUpgrade zeichnet zu jeder Enterprise-Bean, die es implementiert, ausführliche Informationen in der Datei /migration-specific-backup/WASPostUpgrade.log auf.

6. Ändern Sie die Konfiguration mit den Verwaltungsschnittstellen von WebSphere Application Server 5.1.

Nehmen Sie die folgenden Änderungen vor:

- a. **Ändern Sie Benutzer-IDs und Kennwörter entsprechend den Sicherheitsbestimmungen**

Sie müssen Benutzer-IDs und Kennwörter ändern, wenn Sie nicht mit den auf der V5.1-Maschine verwendeten identisch sind.

- b. **Ändern Sie andere maschinenspezifische Informationen.**

Die Konfiguration enthält eventuell Verweise auf andere Softwareprodukte oder Konfigurationen, die auf der neuen Maschine nicht vorhanden sind. Auf der alten Maschine könnte eventuell eine Datenbank vorhanden sein. Ändern Sie die Datenquelle, sodass sie auf die Datenquelle auf der alten Maschine zeigt.

7. Sie können den V5.0.x-Server nach Bedarf deinstallieren.

---

## Migration von einem nicht unterstützten Betriebssystem

Sie können ein Release einer früheren Version von WebSphere Application Server Version 3.5.x oder der Version 5.0.x migrieren, das auf einem Betriebssystem ausgeführt wird, das Version 5.1 nicht unterstützt.

Schritte für diese Task

1. Starten Sie WebSphere Application Server Version 3.5.x oder Version 5.0.x Administrative Server.
2. Führen Sie das Befehlszeilenmigrationstool **WASPreUpgrade** aus.

Es stehen zwei Optionen zur Auswahl. Sie können den Befehl im Verzeichnis migration\bin (oder migration/bin) im *plattformstammverzeichnis* der Version 5.1-CD-ROM ausführen. Alternativ können Sie die Dateien aus dem Verzeichnis auf der CD-ROM in ein Verzeichnis auf Ihrem Festplattenlaufwerk kopieren.

Geben Sie das Release der Version 3.5.x oder 5.0.x an sowie ein Sicherungsverzeichnis, in dem der Befehl die Konfigurationsdateien und migrierten Anwendungen aus der früheren Version speichert. Informationen zur Befehlsyntax finden Sie im Abschnitt zu *WASPreUpgrade*.

- a. Führen Sie den Befehl im Verzeichnis migration\bin (oder migration/bin) im *plattformstammverzeichnis* der Version 5.1-CD-ROM aus.

Geben Sie das Sicherungsverzeichnis und die Position der Konfigurationsdateien an.

```
CD_Laufwerk\WASPreUpgrade_sicherungsverzeichnis_dateipfad\WebSphere
\AppServer_IhrKnotenname
```

Wenn dies funktioniert, fahren Sie mit Schritt 4 fort. Wenn dies nicht funktioniert, führen Sie die Schritte 2B bis 2F aus.

- b. Erstellen Sie ein Verzeichnis mit dem Namen **migration** auf Ihrem Festplattenlaufwerk.
- c. Kopieren Sie die Dateien WASPreUpgrade.bat (oder WASPreUpgrade.sh) und setupCmdLine.bat (oder setupCmdLine.sh) aus dem Verzeichnis migration\bin\ (oder migration/bin/) im *plattformstammverzeichnis* der Version 5.1 CD-ROM in das Verzeichnis, das Sie auf Ihrem Festplattenlaufwerk erstellt haben.

- d. Editieren Sie die Datei `setupCmdLine.bat` (oder `setupCmdLine.sh`) in Ihrem neuen Verzeichnis.

Ändern Sie die folgenden Variablen:

- **WAS\_HOME** muss auf den vollständig qualifizierten Pfad zum Migrationsverzeichnis zeigen, das Sie erstellt haben.
- **JAVA\_HOME** muss auf den vollständig qualifizierten Pfad zu Ihrem IBM Developer Kit oder dem Java-Verzeichnis zeigen

- e. Stellen Sie sicher, dass das ausführbare Bit für die Dateien `setupCmdLine.sh` und `WASPreUpgrade.sh` im Verzeichnis `migration/bin` im *UNIX-basiertes-plattformstammverzeichnis* der Version 5.1-CD-ROM aktiviert ist, falls Sie eine UNIX-basierte Installation speichern.

- f. Führen Sie den Befehl in dem Verzeichnis `migration` aus, das Sie erstellt haben.

Geben Sie das Sicherungsverzeichnis und die Position der Konfigurationsdateien an.

```
IhrVerzeichnisMigration\WASPreUpgrade sicherungsverzeichnis dateipfad  
\WebSphere\AppServer IhrKnotenname
```

3. Beenden Sie WebSphere Application Server Version 3.5.x oder Version 5.0.x.
4. Komprimieren Sie das Sicherungsverzeichnis (mit TAR oder ZIP), und übertragen Sie es per FTP auf ein anderes System.
5. Installieren Sie das neue Betriebssystem, und behalten Sie denselben Hostnamen bei.

Falls möglich, verwenden Sie auch erneut den alten Systemnamen und die zugehörigen Kennwörter des alten Systems. Stellen Sie alle Datenbankdateien, die zu Anwendungen gehören, die Sie migrieren, wieder in denselben Pfad, wie auf dem alten System. Versuchen Sie allgemein, die gleichen Pfade zu verwenden. Installieren Sie jedoch nicht Version 5.1 in dem Verzeichnis der vorherigen Version. Wenn Sie Pfade und Namen ändern, können Sie die XML-Konfigurationsdateien bearbeiten, um die Änderungen entsprechend nachzuvollziehen. Nehmen Sie solche Änderungen vor, bevor Sie den nachfolgend angegebenen Befehl **WASPostUpgrade** ausführen.

6. Rufen Sie per FTP das Sicherungsverzeichnis vom anderen System ab, und entpacken Sie die komprimierte Datei.
7. Installieren Sie WebSphere Application Server- Express, Version 5.1. Wählen Sie die Migrationsoption nicht aus, falls sie angezeigt wird.
8. Führen Sie das Befehlszeilenmigrationstool **WASPostUpgrade** im Verzeichnis `bin` des Installationsstammverzeichnisses der Version 5.1 aus.

Geben Sie das Sicherungsverzeichnis (und alle vom Standard abweichenden Konfigurationsdateinamen) in dem Verzeichnis an, das der Befehl **WASPreUpgrade** erstellte. Informationen zur Befehlssyntax finden Sie im Abschnitt zu *WASPostUpgrade*.

```
installationsstammverzeichnis\bin\WASPostUpgrade WAS_HOME\migration
```



---

## Kapitel 3. Von IBM WebSphere Studio Site Developer Version 5.1 migrieren

Dieses Kapitel behandelt die folgenden Migrationsaspekte:

- Migrieren von J2EE-Projekten zur Nutzung der Server Targeting-Unterstützung
- Für die Assistentengenerierung ist ein Java-Paket für JDK 1.4 erforderlich.

---

### Migrieren von J2EE-Projekten zur Nutzung der Server Targeting-Unterstützung

In IBM WebSphere Studio Site Developer Version 5.1.1 wurde eine neue Server Targeting-Funktion hinzugefügt. Diese Funktion ist standardmäßig deaktiviert. Sie müssen sie auf der Benutzervorgaben-Seite von J2EE mit **Fenster > Benutzervorgaben > J2EE** aktivieren. Details zur Server Targeting-Funktion finden Sie in der Produktdokumentation zu IBM WebSphere Studio Site Developer. Wenn die Funktion aktiviert ist, können Sie einen bestimmten Anwendungsserver ansteuern. Dieses Feature wurde zur Unterstützung von JDK 1.4 implementiert. JDK 1.4 ist das JRE für WebSphere Application Server Version 5.1, das mit IBM WebSphere Studio Site Developer Version 5.1.1 ausgeliefert wird. J2EE-Projekte, welche die Server Targeting-Unterstützung nutzen, sind nicht abwärtskompatibel mit früheren Versionen von IBM WebSphere Studio Site Developer. Sie können daher *nicht* gemeinsam mit Benutzern früherer Versionen von IBM WebSphere Studio Site Developer (z.B. IBM WebSphere Studio Site Developer Version 5.1, IBM WebSphere Studio Site Developer Version 5.0.1) genutzt werden. IBM WebSphere Studio Site Developer bietet eine Möglichkeit, Abwärtskompatibilität zu erhalten, wenn dieses Feature aktiviert ist. Dieser Vorgang ist in „Abwärtskompatibilität mit aktivierter Server Targeting-Unterstützung“ auf Seite 21 beschrieben. Der Grund für die Inkompatibilität ist, dass die Server Targeting-Funktion die Datei .classpath für ein J2EE-Projekt ändert und die neuen Einträge in .classpath *nicht* von älteren Versionen von WebSphere Application Server - Express erkannt werden können.

### Abwärtskompatibilität mit aktivierter Server Targeting-Unterstützung

Wenn die Server Targeting-Unterstützung aktiviert ist, kann die Zielunterstützung von J2EE-Projekten rückgängig gemacht werden, so dass diese Projekte die Server Targeting-Unterstützung nicht nutzen. Dies kann mit Hilfe der Option **No target server specified** im Dialog Modify Target Server eingestellt werden. Der Dialog Modify Target Server kann vom Kontextmenü (**Zielserver > Ändern**) eines J2EE-Projekts im Ressourcennavigators oder in der Sicht J2EE Perspektive gestartet werden. Der Zielserver kann auch von der J2EE-Eigenschaftenseite (**Eigenschaften > J2EE**) für EAR-, EJB-, Anwendungsclient- und Connector-Projekte in **No target server specified** umgeändert werden. Für ein Webprojekt befindet sich die Einstellung für den Zielserver auf der Seite mit Webeeigenschaften (**Eigenschaften > WEB**). Details zur Änderung der Server Targeting-Funktion finden Sie in der Dokumentation zu IBM WebSphere Studio Site Developer. Wenn die Option **No target server specified** ausgewählt wird, wird erneut die Darstellung der Datei .classpath verwendet, die mit älteren Versionen von IBM WebSphere Studio Site Developer kompatibel ist, und die Datei .server wird aus dem Projekt entfernt.

**Anmerkung:** Nur zielgruppenspezifische J2EE-Projekte können unter WebSphere Application Server Version 5.1 implementiert werden und die JDK 1.4-Unterstützung nutzen.

---

## **Für die Assistentengenerierung ist ein Java-Paket für JDK 1.4 erforderlich.**

Aufgrund einer Änderung im JDK 1.4 muss der Benutzer ein Java-Paket angeben, wenn er mit Hilfe der Assistenten für Datenbank-Webseiten und JavaBeans-Webseiten Seiten generiert, die unter IBM WebSphere Studio Site Developer Version 5.1.1 ausgeführt werden. Dieses Problem tritt dann auf, wenn die View Bean-Schablone für den Assistenten für JavaBeans-Webseiten oder das IBM Database Access Java Beans-Master-Detailmuster verwendet wird. Dies gilt auch für Projekte mit Seiten und .java-Dateien, die zuvor ohne Angabe eines Pakets mit Hilfe dieser Assistenten generiert wurden. Versetzen Sie bei zuvor generiertem Code die .java-Dateien in ein Paket. Aktualisieren Sie dann die .jsp-Dateien, die Importanweisungen und die Klasseninformationen. Aktualisieren Sie in der Projektdatei web.xml den Servlet-klasseneintrag.

---

## Kapitel 4. Von IBM WebSphere Studio Site Developer Version 5 oder 5.0.1 migrieren

Dieses Kapitel behandelt die folgenden Migrationsaspekte:

- WebSphere Studio Workbench (WSWB) in Version 5, Version 5.0.1 und Version 5.1
- Die Verwendung von WebSphere Studio Version 5.1.1 mit dem Version 5.0-Arbeitsbereich
- Migration der Java-Projekte von Version 5 oder Version 5.0.1
- Projekte aus Version 5 oder Version 5.0.1 mit Version 5.1 unter Verwendung eines Quellcodeverwaltungssystems gemeinsam benutzen
- Migration der Webprojekte
- Konvertierung der Webprojekte auf Struts 1.1
- Migrationsfunktion, die zur J2EE-Projektmigration hinzugefügt wurde
- Änderungen an Web-Service-Tools
- Änderungen an Profilermittlungstools
- Bekannte Probleme des Schablonenassistenten mit Kompatibilität

---

### WebSphere Studio Workbench in Version 5, Version 5.0.1 und Version 5.1

IBM WebSphere Studio Site Developer Version 5.1.1 basiert auf der neuen Eclipse-Basis WebSphere Studio Workbench (WSWB) 2.1.2. Es bestehen einige Unterschiede zwischen den Versionen 2.1.2 und 2.0.3 bzw. 2.0.2. Um detaillierte Informationen zu den Unterschieden zu erhalten, lesen Sie die Readme-Datei, die sich im Verzeichnis *WS\_installationsverzeichnis\ eclipse\ readme* befindet. Hierbei ist *WS\_installationsverzeichnis* der Pfad, in dem Sie IBM WebSphere Studio Site Developer installiert haben.

IBM WebSphere Studio Site Developer Version 5.0 beruhte auf der Eclipse-Basis WSWB 2.0.2, und IBM WebSphere Studio Site Developer Version 5.0.1 beruhte auf der Eclipse-Basis WSWB 2.0.3. Es gibt zwischen den Versionen 2.0.2 und 2.0.3 keine bedeutenden Unterschiede. Das Release IBM WebSphere Studio Site Developer Version 5.0.1 war ein Update Manager-Fixpack, das beim Installieren IBM WebSphere Studio Site Developer Version 5.0 überlagerte.

---

### IBM WebSphere Studio Site Developer Version 5.1.1 mit dem Arbeitsbereich von Version 5.0 verwenden

Wenn IBM WebSphere Studio Site Developer Version 5.1.1 zum ersten Mal mit Hilfe eines vorhandenen Arbeitsbereichs von IBM WebSphere Studio Site Developer Version 5.0 gestartet wird, wird ein Dialogfenster angezeigt, das eine Möglichkeit der Migration von Version 5.0 beschreibt. Klicken Sie auf **OK**, um vom Arbeitsbereich der Version 5.0 zu migrieren, oder klicken Sie auf **Abbruch**, um das Starten von IBM WebSphere Studio Site Developer zu verhindern.

Wenn der Arbeitsbereich migriert wurde, können Sie weiterhin den Arbeitsbereich mit Version 5.0 verwenden, da die Metadaten der neuen Projektmerkmale ignoriert werden und von Version 5.0 gelesen werden können. Sie können in Version 5.0

keine Änderungen an den Projekten im Arbeitsbereich vornehmen, die Auswirkungen auf die Metadaten haben oder die Metadaten der neuen Projektmerkmale überschreiben könnten.

---

## Migration der Java-Projekte von Version 5 oder Version 5.0.1

Die Migration der Java-Projekte von Version 5 oder Version 5.0.1 ist unkompliziert und geschieht automatisch. Sobald die Projekte in den Arbeitsbereich der Version 5.1.1 geladen sind, treten keine Änderungen an den Metadaten in den .classpath-Dateien oder den .project-Dateien auf, es sei denn, Sie versuchen, eine der in Version 5.1.1 verfügbaren neuen Funktionen zu verwenden.

---

## Projekte aus Version 5 oder Version 5.0.1 mit Version 5.1.1 unter Verwendung eines Quellcodeverwaltungssystems gemeinsam benutzen

Besondere Aufmerksamkeit ist erforderlich, wenn ein Projekt aus einem Team-Repository von Entwicklern mit Hilfe von IBM WebSphere Studio Site Developer Version 5 und IBM WebSphere Studio Site Developer Version 5.1.1 geladen und betrieben wird. Das allgemeine Problem besteht darin, dass das Vorhandensein, der Inhalt und die Interpretation von Metadateien in Arbeitsbereichen für eine bestimmte Funktion oder Plug-in-Version spezifisch sein können und sich in den einzelnen Versionen unterscheiden. Die Arbeitsbereichskompatibilität garantiert nur die wichtigsten Fälle, in denen alle Entwickler ihre Arbeitsbereiche von IBM WebSphere Studio Site Developer in Sperrschritten upgraden. In diesen Fällen sollten keine Probleme mit gemeinsam benutzten Metadaten auftreten. Wenn jedoch einige Entwickler in IBM WebSphere Studio Site Developer Version 5.1.1 und andere zur selben Zeit in IBM WebSphere Studio Site Developer Version 5 arbeiten, kann keine Garantie gegeben werden. Dieser Abschnitt bietet Ihnen Informationen dazu, wie Sie vorgehen bzw. nicht vorgehen können.

Der Benutzer von IBM WebSphere Studio Site Developer Version 5.1.1 erhält den typischen Störungsmodus. Die Version 5.1.1-Metadaten gehen verloren, wenn ein Benutzer der Version 5 Änderungen speichert und anschließend die aktualisierten Metadatendateien dem Repository übergibt. Nachfolgend sind einige Dinge aufgeführt, die zu Problemen führen können:

- Ein Benutzer, der mit IBM WebSphere Studio Site Developer Version 5.1.1 arbeitet, erstellt oder modifiziert ein Projekt so, dass eine gemeinsam benutzte Metadatendatei geändert wird, diese Änderungen beruhen auf Version 5.1.1-spezifischen Informationen. Der Benutzer übergibt dann die aktualisierten Projektdateien, einschließlich der gemeinsam benutzten Metadatendatei, dem gemeinsam benutzten Repository.
- Ein anderer Benutzer, der mit IBM WebSphere Studio Site Developer Version 5 arbeitet, benutzt dieses Projekt gemeinsam aus demselben Repository. Die 5.1.1-spezifischen Informationen in der gemeinsam benutzten Metadatendatei werden von IBM WebSphere Studio Site Developer Version 5 nicht verstanden und im Allgemeinen ohne Warnung gelöscht oder ignoriert. Der Benutzer modifiziert das Projekt so, dass die gemeinsam benutzte Metadatendatei geändert wird, wodurch die gemeinsam benutzte Metadatendatei ohne die 5.1.1-spezifischen Informationen erneut geschrieben wird. Der Benutzer übergibt die aktualisierten Projektdateien, einschließlich der gemeinsam benutzten Metadatendatei, dem gemeinsam benutzten Repository. Der Benutzer merkt in der Regel nicht, dass gemeinsam benutzte Informationen aufgrund seiner Aktion gerade verloren gegangen sind.

- Ein Benutzer, der mit IBM WebSphere Studio Site Developer Version 5.1.1 arbeitet, wählt die Änderungen an einem Projekt aus dem gemeinsam benutzten Repository aus, einschließlich der aktualisierten gemeinsam benutzten Metadatei. Der Benutzer hat möglicherweise nicht gemerkt, dass die Änderungen gerade zurückgesetzt worden sind, bis es später zu Störungen kommt.

Nachfolgend finden Sie eine Liste mit Punkten, die zu beachten sind, wenn ein Projekt gemeinsam von Benutzern der IBM WebSphere Studio Site Developer Version 5.1.1 und Benutzern der Version 5 oder Version 5.0.1 benutzt werden soll:

- **Verknüpfte Ordner und Dateien**

Diese Unterstützung wurde in IBM WebSphere Studio Site Developer Version 5.1.1 hinzugefügt. Informationen zu verknüpften Ressourcen werden in der .project-Datei aufgezeichnet.

**Empfehlung:** Nicht verwenden. Am besten inaktivieren Sie die verknüpften Ressourcen über die Seite mit den Benutzervorgaben, wählen Sie hierzu **Workbench > Verlinkte Ressourcen** aus.

- **Erstellungsprogramme für externe Tools (Ant)**

Informationen zum Erstellungsprogramm für externe Tools werden in der .project-Datei aufgezeichnet. Das Format der Informationen wurde zwischen Version 5 und Version 5.1.1 geändert. Erstellungsprogramme, die in Version 5.1.1 erstellt oder geändert wurden, verwenden das neue Format, das von einem Version 5-Arbeitsbereich nicht verstanden wird. Erstellungsprogramme, die in Version 5 erstellt wurden, verwenden das alte Format und sind in Version 5.1.1 funktionsfähig.

**Empfehlung:** Erstellen oder editieren Sie Erstellungsprogramme für externe Tools über einen Version 5-Arbeitsbereich.

- **Optionale Ausschlussmuster bei Java-Quelleneinträgen im Erstellungsklassenpfad**

Diese Unterstützung wurde hinzugefügt. Diese Informationen werden in der .classpath-Datei aufgezeichnet.

**Empfehlung:** Geben Sie keine Ausschlussmuster an. Am besten inaktivieren Sie die Ausschlussmuster über die Seite mit den Benutzervorgaben, wählen Sie hierzu **Java > Compiler > Erstellungspfad** aus.

- **Ausgabeordner, die Java-Quelleneinträgen im Erstellungsklassenpfad zugeordnet sind**

Diese Unterstützung wurde hinzugefügt. Diese Informationen werden in der .classpath-Datei aufgezeichnet.

**Empfehlung:** Geben Sie im gesamten Projekt nichts anderes als den Standardausgabeordner ein. Am besten inaktivieren Sie die verschiedenen Ausgabepositionen über die Seite mit den Benutzervorgaben, wählen Sie hierzu **Java > Compiler > Erstellungspfad** aus.

- **Stammverzeichnispfad für Quellenanhänge, die Java-Bibliothekseinträgen im Erstellungsklassenpfad zugeordnet sind**

Wenn Sie eine Quellen-ZIP-Datei an eine JAR-Bibliothek im Java-Erstellungspfad anhängen, wird das Quellenrootpfadpräfix automatisch gewonnen. Dies hat sich ab Version 5 geändert, wo das Präfix noch explizit mit der Benutzerschnittstelle festgelegt und explizit in der .classpath-Datei aufgezeichnet werden konnte. Folglich könnte ein Java-Projekt, das in einem 5.1.1-Arbeitsbereich erstellt wurde, nicht die angehängte Quelle finden.

Verwenden Sie Version 5, um den Stammverzeichnispfad für Quellenanhänge anzugeben. In Version 5.1.1 wird eine zusätzliche Flexibilität für Quellenanhänge zur Verfügung gestellt. Sie können einen Ordner anstelle einer JAR- oder ZIP-Datei als einen Quellenanhang bereitstellen, und Sie können eine Quelle an einen Klassendateiordner anhängen. Diese Funktionalität ist in Version 5 nicht verfügbar, dort werden die Version 5.1.1-Informationen ignoriert.

- **PDE-Klassenpfadcontainer für abhängige Plug-ins**

Die Verwendung der Klassenpfadcontainer durch PDEs wurde hinzugefügt. Klassenpfadcontainer werden in der .classpath-Datei aufgezeichnet. Wenn PDE-Klassenpfadcontainer verwendet werden, dann wird ein Version 5-Arbeitsbereich über unaufgelöste Klassenpfadeinträge verfügen und daher werden die meisten Java-Funktionen (einschließlich der Kompilierung, Suche, Ausführung und Fehlerbehebung) nicht die gewünschten Ergebnisse erzielen.

**Empfehlung:** Stellen Sie sicher, dass die Einstellung auf der Seite mit den Benutzervorgaben **Plug-in-Entwicklung > Java-Erstellungspfadsteuerung** für die Verwendung von Klassenpfadcontainern inaktiviert ist, bevor Sie neue Plug-in- oder Fragmentprojekte erstellen.

---

## Webprojekte migrieren

Die Ordnernamen lauten **Java-Quelle** und **Webinhalt**. Die Standardordnernamen für neue Webprojekte sind über eine Benutzervorgabenseite konfigurierbar, wählen Sie hierzu **Fenster > Benutzervorgaben > Web-Tools > Neues Projekt** aus. Die Standardnamen lauten jetzt **Java-Quelle** und **Webinhalt**. Diese Standardnamen werden nur für neue Webprojekte verwendet. Webprojekte, die in Versionen vor diesem Release erstellt wurden, verwenden weiterhin die alten Namen. Dasselbe gilt für statische Webprojekte.

Wenn Sie sich dafür entscheiden, die Quellordnernamen für 4.0.x- und 5.0-Projekte in Version 5.1.1 zu ändern, verwenden Sie die Kontextmenüaktion **Umbenennen** in der Sicht "Navigator". Die Kontextmenüaktion **Umbenennen** benennt die Ordnernamen um und legt den Java-Erstellungspfad für die 4.0.x- und 5.0.x-Webprojekte fest.

Für den Java-Quellenordner **JavaSource** funktioniert die Kontextmenüaktion **Umbenennen** in den Sichten "Projektnavigator" und "Paket-Explorer". Für den Ordner **Webinhalt** funktioniert die Kontextmenüaktion **Umbenennen** in den Sichten "Ressourcennavigator" und "Projektnavigator".

Falls Sie ein Webprojekt aus einer Version vor diesem Release in einem SCM-Repository speichern und dann in dieses Release laden, behält dieses Projekt die alte Struktur mit dem Quellen- und dem Webanwendungsordner (**source** und **webApplication**) bei. Beide Strukturen werden korrekt erstellt.

**Anmerkung:** Wenn die Benutzer sich dafür entscheiden, die Namen für den Java-Quellenordner **Java Source** und den Webinhaltsordner **Web Content** umzubenennen, dann müssen sie jedes automatisierte Erstellungsscript manuell aktualisieren; sie müssen die Scripts so ändern, dass sie die neuen Ordnernamen verwenden.

---

## Konvertierung der Webprojekte auf Struts 1.1

Die Laufzeit der Struts-Tools wurde von Version 1.1 Beta 2 in Version 5 auf Version 1.1 heraufgesetzt. In IBM WebSphere Studio Site Developer Version 5 (General Availability) haben Sie die Option bei Erstellung eines Webprojekts, Ihrem Projekt die Struts-Unterstützung hinzuzufügen. Sie können zwischen Struts 1.0.2 oder Struts 1.1 Beta 2 auswählen. In IBM WebSphere Studio Site Developer Version 5.1.1 wurde die letzte Auswahlmöglichkeit durch Struts 1.1 ersetzt. Wenn Sie Webprojekte von Struts 1.1 Beta 2 in IBM WebSphere Studio Site Developer Version 5 erstellt haben, können Sie diese in Struts 1.1 konvertieren, dies ist aber nicht erforderlich, da Struts 1.1 Beta 2 noch unterstützt wird. Wenn Sie über Webprojekte von Struts 1.1 Beta 2 verfügen, die Sie in Struts 1.1 konvertieren wollen, müssen Sie hierzu wie folgt vorgehen:

1. Laden Sie Ihre Struts 1.1 Beta 2-Projekte in einen Arbeitsbereich von IBM WebSphere Studio Site Developer Version 5.1.1.
2. Erstellen Sie ein neues Webprojekt von Struts 1.1 mit dem Namen **Struts11**. Dadurch wird ausreichend Zugriff auf die Artefakte von Struts 1.1 zur Verfügung gestellt, der benötigt wird, während die tatsächlichen Projekte konvertiert werden. Sie können dieses Projekt löschen, sobald Sie fertig sind.
3. Für jedes Projekt von Struts 1.1 Beta 2, das Sie in Struts 1.1 konvertieren wollen, müssen Sie Folgendes ausführen:
  - a. Löschen Sie die folgenden .JAR-Dateien aus dem Projektverzeichnis WebContent/WEB-INF/lib: commons-\*.jar und struts.jar.
  - b. Kopieren Sie die folgenden .JAR-Dateien aus dem Verzeichnis Struts11/WebContent/WEB-INF/lib in Ihr Projektverzeichnis WebContent/WEB-INF/lib: commons-\*.jar und struts.jar.
  - c. Löschen Sie die folgenden .TLD-Dateien aus Ihrem Projektverzeichnis WebContent/WEB-INF: struts-\*.tld.
  - d. Kopieren Sie die folgenden .TLD-Dateien aus dem Verzeichnis Struts11/WebContent/WEB-INF in Ihr Projektverzeichnis WebContent/WEB-INF: struts-\*.tld.

Alle obigen Informationen gelten ebenfalls, wenn Sie ein Webprojekt von Struts 1.1 Beta 3 in IBM WebSphere Studio Site Developer Version 5.0.1 auf Struts 1.1 heraufsetzen.

---

## Änderungen an Web-Service-Tools

Die Web-Service-Tools haben zwei neue Laufzeitprotokolle von IBM WebSphere Application Server Version 5.0.2 hinzugefügt, die nur unter WebSphere Application Server Version 5.0.2 laufen. Es sollte zu keiner obligatorischen Migration kommen, da IBM WebSphere Studio Site Developer Version 5.1.1 und WebSphere Application Server Version 5.0.2 sowohl die alten als auch die neuen Laufzeitprotokolle unterstützen. IBM WebSphere Studio Site Developer wird drei Laufzeitprotokolle der Web-Service-Artefakte generieren und implementieren: das alte Laufzeitprotokoll "IBM SOAP", das unter WebSphere Application Server Version 4.x und Version 5.x läuft, und das neue Laufzeitprotokoll "IBM WebSphere 5.0.2-Laufzeit", das nur unter WebSphere Application Server Version 5.0.2 läuft; und das neue Laufzeitprotokoll "Apache Axis 1.0", das nur unter WebSphere Application Server Version 5.0.2 läuft.

Benutzer sollten in der Lage sein, ihre auf Version 5-Projekte und Web-Services bezogenen EAR- und WAR-Dateien ohne Änderungen in Version 5.1.1 wiederzuverwenden. Damit sie ihre Web-Services und Clients auf das neue Laufzeitprotokoll von IBM WebSphere 5.0.2 konvertieren und die JSR 101, 109, WS-I und WS-Sicherheitsstandards nutzen können, müssen sie diese über den Assistenten "Web-Services" erneut generieren. Der Web-Services-Explorer fährt automatisch fort die Favoriten des Benutzers zu lesen, obwohl die physische Datendatei automatisch an eine neue Speicherposition versetzt wird.

---

## Änderungen an Profilermittlungstools

Wenn Sie einen Arbeitsbereich von Version 5 migrieren, dann erhalten Sie eine Fehlermeldung "Beim Wiederherstellen der Workbench sind Fehler aufgetreten". Diese Nachricht wird angezeigt, wenn die Perspektive "Profilermittlung" während der Migration geöffnet ist und wenn die Sichten "Informationen zum Heapspeicher" oder "Exemplarstatistik" in der Perspektive "Profilermittlung" sichtbar waren. Das kommt daher, weil die Sichten "Informationen zum Heapspeicher" und "Exemplarstatistik", die in Version 5 vorkamen, entfernt worden sind. Diese Nachricht wird auch angezeigt, wenn die Perspektive "Profilermittlung" während der Migration im Arbeitsbereich geöffnet ist. Die Fehlermeldung kann bedenkenlos ignoriert werden, indem Sie auf **OK** klicken.

---

## Bekannte Probleme des Schablonenassistenten mit Kompatibilität

Um eine in Version 5 erstellte, angepasste Schablone zu verwenden, sollten Sie die angepasste Schablone laden, die Verbindung zwischen ihr und der Datenbank wiederherstellen und sie speichern. Wenn Sie die gespeicherte angepasste Schablone das nächste Mal erneut laden, wird die Verbindung geprüft.

Die generierten J2EE 1.2-Artefakte, die in diesem Release erstellt wurden, können unter Umständen von IBM WebSphere Studio Site Developer Version 4.0.3 nicht gelesen und von Testumgebungen der Version 4.0.3 nicht ausgeführt werden. Da der Schablonenassistent nicht mit Version 4.0.3 ausgeliefert wurde, wird für diese Version keine Abwärtskompatibilität bereitgestellt.

Schablonenanwendungen, die in diesem Release generiert wurden, können unter Version 5 ausgeführt werden, wenn in den Benutzervorgaben der Webprojekte der Java-Quellenordner "Java Source" und der Webinhaltsordner "Web Content" genannt wird.

---

## Kapitel 5. Von IBM WebSphere Studio Site Developer Version 4.0.x migrieren

Das folgende Kapitel behandelt die Migration von IBM WebSphere Studio Site Developer Version 4.0.x auf Version 5. .

Es gibt zwei unterstützte Methoden, mit denen Sie Ihre Projekte von IBM WebSphere Studio Site Developer Version 4.0.x auf Version 5 migrieren können. Diese beiden Methoden werden im Folgenden ausführlich beschrieben.

- Sie können ein System zur Softwarekonfigurationsverwaltung (Software Configuration Management - SCM) wie beispielsweise CVS (Concurrent Versioning System) oder Rational® ClearCase® verwenden. Diese Methode wird empfohlen.
- Sie können Ihre Projekte aus Version 4.0.x exportieren und dann in die vorliegende Version importieren. Bei dieser Methode werden alle Elemente mit Ausnahme der Informationen zum Projekterstellungspfad migriert.
- Sie können Ihren vorhandenen Arbeitsbereich aus Version 4.0.x verwenden. Aus Gründen, die an späterer Stelle noch erläutert werden, wird diese Methode nicht unterstützt.

Beachten Sie, dass durch die Migration von Version 4 auf Version 5 *nicht* automatisch die J2EE-Stufe von Projekten geändert wird, da Version 5 noch immer Erstellungs- und Deploymentoperationen für WebSphere Application Server Version 4 durchführen kann. Alle J2EE-Projekttypen, einschließlich der Webprojekte, können mit Hilfe des J2EE-Migrationsassistenten migriert werden, der in IBM WebSphere Studio Site Developer verfügbar ist. Wenn Sie auf den J2EE-Migrationsassistenten zugreifen möchten, klicken Sie mit der rechten Maustaste auf ein Projekt des Typs J2EE, und wählen Sie anschließend **Migrieren > J2EE-Migrationsassistent** aus.

---

### Unterschiede zwischen IBM WebSphere Studio Site Developer Version 4.0.x und Version 5

Die folgende Liste enthält einen Ausschnitt der Funktionserweiterungen, die seit Version 4.0.x aufgenommen wurden:

- IBM WebSphere Studio Site Developer Version 5 kann Code für Version 4.0 oder für Version 5 von WebSphere Application Server generieren und enthält die Testumgebungen von WebSphere Application Server Version 4.0.4 und Version 5.
- Die J2EE-Stufe für Unternehmensanwendungsarchive (EARs) wurde für Projekte in WebSphere Application Server Version 5 von 1.2 in 1.3 geändert.
  - J2EE 1.2-EARs können entweder auf WebSphere Application Server Version 4.0.x oder auf WebSphere Application Server Version 5 ausgeführt werden.
- Die Spezifikationsstufe für EJBs (Enterprise Java Beans) wurde für EARs in WebSphere Application Server Version 5 von 1.1 in 2.0 geändert.
  - wird weiterhin unterstützt. Diese Dateien können Bestandteil von J2EE 1.2-EARs von WebSphere Application Server Version 4.0.x oder von J2EE 1.3-EARs von Version 5 sein.

- Die J2EE-Stufe für Webanwendungen (WARs) wurde für Projekte in WebSphere Application Server Version 5 von 1.2 in 1.3 geändert.
  - Die JSP-Stufe wurde von 1.1 in 1.2 geändert. Die Servletstufe wurde von 2.2 in 2.3 geändert.
  - Dynamische 1.2-Webprojekte werden weiterhin unterstützt und können in J2EE 1.2-EARs von WebSphere Application Server Version 4.0.x oder in J2EE 1.3-EARs von Version 5 verwendet werden.
- In Version 5 können Sie sowohl statische Webprojekte als auch dynamische Webprojekte erstellen. In einem statischen Webprojekt können Sie lediglich Inhalt erstellen, der durch einen herkömmlichen HTTP-Server bereitgestellt wird (HTML, JavaScript™, Images, Text usw.).
- Die zu Grunde liegende Workbench, die auf dem Open Source-Projekt Eclipse basiert, wurde von Version 1.0 in Version 2.0 geändert.
  - Es gibt ein neues und erweitertes Java-Erstellungsprogramm.
  - Es gibt eine neue und optimiertere VCM-Schnittstelle (VCM - Version Control Management - Versionssteuerungsverwaltung) für SCM-Softwareanbieter.

---

## Änderungen an WebSphere Application Server und Tools für die Servlet/JSP-Konvertierung

Das WebSphere-InfoCenter

[[www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/index.html](http://www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/index.html)] enthält folgende Informationen:

- Unterschiede zwischen WebSphere Application Server Version 3.5 und 4.0  
[[www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/was/03.html](http://www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/was/03.html)].
- Weitere Informationen zur Migration auf WebSphere Application Server Version 5.0 von Version 3.5 oder Version 4.0 finden Sie im InfoCenter für WebSphere unter der Adresse [www.ibm.com/software/webservers/appserv/infocenter.html](http://www.ibm.com/software/webservers/appserv/infocenter.html), indem Sie die folgenden Schritte ausführen, gelangen Sie auf die Migrationsseite.
  1. Wählen Sie im ersten Link **Application Server for distributed operating systems** eine Sprache Ihrer Wahl aus.
  2. Wählen Sie den Link **Alle Abschnitte nach Aktivität** aus, der sich oben links befindet.
  3. Wählen Sie den Link **Migration** aus.

*Migrating to WebSphere V5.0 An End-to-End Migration Guide* stellt eine nützliche Informationsquelle für die Migration von Version 3.5 und Version 4.0 auf Version 5 dar [[www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246910.pdf](http://www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246910.pdf)].

Auf der Seite mit den Downloads für WebSphere Application Server

[[www14.software.ibm.com/webapp/download/product.jsp?s=p&id=TDUN-49EVRT&type=s&dt=DIAGNOSTIC+TOOL](http://www14.software.ibm.com/webapp/download/product.jsp?s=p&id=TDUN-49EVRT&type=s&dt=DIAGNOSTIC+TOOL)] finden Sie Tools, die Sie bei der Konvertierung von Anwendungen unterstützen:

- Das Tool "MigrateWC" konvertiert eine .91- oder 1.0-JSP in eine 1.1-JSP. Außerdem kann es ein 2.1-Servlet in ein 2.2-Servlet konvertieren.
- Das Tool "XMLconvert" konvertiert XML-Konfigurationsdateien aus Release 3.02x oder Release 3.5x in das Format von Release 4.0.

---

## Interne Änderungen gegenüber Version 4.0.3

### Schleifenabhängigkeiten von Projekten werden nicht standardmäßig erstellt

Wenn Ihre Projekte Schleifenabhängigkeiten aufweisen, meldet Version 5 einen Erstellungsfehler. In diesem Fall können Sie die Optionen **Fenster > Benutzervorgaben > Java > Compiler** auswählen und dann auf der Registerkarte **Erstellungspfad** das Markierungsfeld **Erstellung bei Erstellungspfadfehlern abbrechen** abwählen. Beachten Sie, dass die Erstellung dadurch nicht mehr gestoppt wird, jedoch ein oder mehrere Schleifenabhängigkeitsfehler bei der Erstellung noch immer auftreten, die in der Sicht "Tasks" angezeigt werden (selbst dann, wenn die Erstellung ansonsten erfolgreich verläuft). In diesem Fall können Sie diese Fehler in Warnungen ändern, indem Sie die Registerkarte **Andere** auswählen und dann die Vorgabe in der Dropdown-Liste **Schleifenabhängigkeiten** ändern.

### Kompatibilität der Quellenposition von Webprojekten aus Version 5 mit Version 4.0.3

In IBM WebSphere Studio Site Developer Version 5 gibt es Änderungen der internen Projektstruktur gegenüber Version 4.0.3. Eine J2EE 1.2-Web-WAR aus Version 5 kann, wenn sie mit Java-Quelle exportiert wird, in IBM WebSphere Studio Site Developer Version 4 importiert werden. Der Quellcodeordner wird automatisch in den richtigen Namen umgewandelt und problemlos erstellt. Das Webprojekt wird in WebSphere Application Server Version 4 genauso ordnungsgemäß ausgeführt, wie wenn ein Projekt aus Version 4 in Version 5 importiert wird, da der Name des Quellcodeordners automatisch in den richtigen Namen umgewandelt wird. Weitere Informationen zum Ändern von Ordnernamen finden Sie im Abschnitt „Strukturen von Webprojekten in IBM WebSphere Studio Site Developer“ auf Seite 31.

**Anmerkung:** Die obige Aussage trifft nicht zu, wenn die Webprojekte gemeinsam von Version 5 und Version 4 über ein System zur Softwarekonfigurationsverwaltung (SCM) verwendet werden. Die Projekte der Version 4 müssen auf eine Projektstruktur der Version 5 migriert werden und können nach der Migration von einem SCM-System nicht erneut in die Version 4 geladen werden.

### Strukturen von Webprojekten in IBM WebSphere Studio Site Developer

Die interne Webprojektstruktur in IBM WebSphere Studio Site Developer Version 5 unterscheidet sich von IBM WebSphere Studio Site Developer Version 4.0.x. Dieser Unterschied ist nicht auf die Abweichungen zwischen J2EE 1.2 und J2EE 1.3 zurückzuführen, sondern beruht auf einer Änderung, die die Benutzerfreundlichkeit des Tools verbessert.

In Version 4 waren Webprojekte standardmäßig dynamische Webprojekte und wurden in der Sicht "Navigator" mit einem Quellenordner (**source**) und einem Webanwendungsordner (**webApplication**) angezeigt. Wenn Sie in Version 5 ein dynamisches Webprojekt erstellen, dann wird es anstatt des Quellenordners (**source**) mit einem Java-Quellenordner (**Java Source**) und anstatt des Webanwendungsordners (**webApplication**) mit einem Webinhaltsordner (**Web Content**) angezeigt.

Falls Sie jedoch ein Webprojekt aus Version 4 in einem SCM-Repository speichern und dann in Version 5 laden, behält dieses Projekt die alte Struktur mit dem Quellen- und dem Webanwendungsordner (**source** und **webApplication**) bei. Beide Strukturen werden in Version 5 korrekt erstellt.

## Unterschiede zwischen statischen und dynamischen Webprojekten

In Version 5 können Sie sowohl statische als auch dynamische Webprojekte erstellen.

Statische Webprojekte enthalten nur statische Ressourcen wie HTML, Java-Scripts, Images, Text usw., aber keinen dynamischen Inhalt. Statische Webprojekte können von einem traditionellen HTTP-Web-Server ausgeführt und bedient werden; sie benötigen keinen Webanwendungsserver.

Dynamische Webprojekte enthalten neben den statischen Ressourcen noch dynamische J2EE-Ressourcen, wie z. B. Servlets, JSPs, Filter und zugeordnete Metadaten. Wenn Sie dynamische Webprojekte erstellen, können Sie gestaffelte Formatvorlagen und JSP-Tagbibliotheken mit einschließen, so dass Sie die Entwicklung mit einem umfangreicheren Satz Projektressourcen beginnen können. Dynamische Webprojekte sind immer in Unternehmensanwendungsprojekten eingebettet und werden nur auf Webanwendungsservern ausgeführt.

## Unterschiede zwischen HTML und JSP

- In Version 4.0.x wurden HTML-Dateien und JSP-Dateien von Page Designer identisch behandelt, beispielsweise konnten in einer HTML-Datei JSP-Tags verwendet werden. Dies ist nicht mehr zutreffend. Im aktuellen Release wird zwischen JSP-Dateien und HTML-Dateien unterschieden. Daher können JSP-Tags nicht mehr in einer HTML-Datei verwendet werden.
- Diese Unterscheidung wirkt sich auf die Codierung von JSP-Dateien in anderen Sprachen als Englisch aus. In den früheren Versionen wurden die HTML-Codierungsregeln auch bei JSP-Dateien verwendet, um die in einer Datei benannte Codierung zu bestimmen. Dies bedeutet, dass nach dem Attribut für den Inhaltstyp des Metatags gesucht wurde (`<META http-equiv="Content-Type" content="text/html; charset=UTF-8">`). In der vorliegenden Version wird die in einer JSP-Datei verwendete Codierung anhand der JSP-Codierungsregeln bestimmt. Dies bedeutet, dass nach der Seitendirektive der JSP-Datei gesucht wird (`<%@page contentType="text/html; charset=UTF-8"%>`). Die Codierung von HTML-Dateien hat sich gegenüber den früheren Versionen nicht geändert.

---

## Projekte mit einem System zur Softwarekonfigurationsverwaltung (SCM) migrieren

### Projekte mit CVS oder Rational ClearCase migrieren

Diese Methode wird beim Versetzen von Arbeitsbereichen aus Version 4.0.x in IBM WebSphere Studio Site Developer Version 5 empfohlen, denn nur diese Methode migriert alle Daten inklusive der Informationen zum Projekterstellungspfad.

1. Als Sicherheitsmaßnahme sollten Sie alle Projekte aus Version 4 im SCM-Repository speichern. Anschließend schreiben Sie alle anstehenden Änderungen fest (Freigabe).
2. Wenn Sie in den Versionen 4 und 5 von IBM WebSphere Studio Site Developer arbeiten wollen, dann speichern Sie Ihre Arbeit erneut in einer neuen Verzweigung (Datenstrom) für Version 5. Diese Verzweigung verwenden Sie anschließend, wenn Sie mit Version 5 arbeiten.
3. Installieren Sie Version 5.
4. Schließen Sie IBM WebSphere Studio Site Developer Version 4, und starten Sie IBM WebSphere Studio Site Developer Version 5.

**Tipp:** In Version 4 war das Arbeitsbereichsverzeichnis standardmäßig ein Unterverzeichnis des Installationsverzeichnisses. In Version 5 wurde diese Standardeinstellung in ein Arbeitsbereichsverzeichnis (**workspace**) geändert, das sich im Verzeichnis "Eigene Dateien" (**My Documents**) befindet. Wenn Sie die Speicherposition für Ihre Arbeit überschreiben wollen, verwenden Sie hierzu beim Starten der Workbench die Option `-data` des Befehls .

**Anmerkung:** Verwenden Sie die Option `-data` nicht, um auf einen vorhandenen Arbeitsbereich aus Version 4 zu verweisen, da es sich hierbei um eine andere Migrationsmethode handelt, die nicht unterstützt wird. (Weitere Informationen finden Sie unter the „Projekte mit einem vorhandenen Arbeitsbereich aus Version 4.0.x migrieren“ auf Seite 36.)

5. Inaktivieren Sie die Option **Fenster > Benutzervorgaben > Workbench > Erstellung bei Ressourcenänderung automatisch ausführen**. Auf diese Weise verhindern Sie Erstellungsfehler, die beim Laden einzelner, abhängiger Projekte entstehen.
6. **Für CVS:** Laden Sie alle Projekte, mit denen Sie arbeiten wollen, aus dem SCM-Repository in IBM WebSphere Studio Site Developer Version 5.  
**Für ClearCase:** Verwenden Sie einen leeren Arbeitsbereich von Version 5, und wählen Sie dann für jedes zu ladende Projekt **Datei > Importieren > Vorhandenes WebSphere Studio 4.x-ClearCase-Projekt in Arbeitsbereich** aus.
7. Stellen Sie die gewünschte Einstellung für **Fenster > Benutzervorgaben > Workbench > Erstellung bei Ressourcenänderung automatisch ausführen** wieder her.
8. Ändern Sie den Namen des Quellenordners (**source**) von **source** in **Java Source** (Java-Quelle) und den Webanwendungsordner **webApplication** in **Web Content** (Webinhalt) für die Webprojekte, wenn Sie eine vollständige Erstellung benötigen. Andernfalls wird die alte Ordnerstruktur beibehalten und die Webprojekte werden *nicht* vollständig erneut erstellt.

9. Nehmen Sie eine vollständige Neuerstellung (**Projekt > Alle neu erstellen**) vor, und speichern Sie die resultierenden Projekte wieder im Repository des neuen Datenstroms für Version 5. (Diese Ressourcen dürfen *nicht* mit dem aktuellen Datenstrom aus Version 4 gemischt werden.)

**Anmerkung:** Diese Projekte sind jetzt Projekte aus Version 5 und *können nicht* mit IBM WebSphere Studio Site Developer Version 4.0.x verwendet werden.

Hinweise für das Vorgehen nach der Migration:

- In IBM WebSphere Studio Site Developer Version 4.0 wurden die Dateien eines Projekts in binärem Format in CVS gespeichert. In binärem Format werden Zeilenschaltungen und Zeilenvorschübe in Dateien *nicht* für andere Plattformen übersetzt und können daher in UNIX<sup>®</sup>/Linux-Umgebungen *nicht* erkannt werden. Wenn Sie in einer Umgebung mit mehreren Plattformen arbeiten (z.B. DOS/Windows plus UNIX/Linux), kann es sinnvoll sein, Quelldateien jetzt als Textdateien zu markieren (Optionen **Team > CVS > ASCII/Binäreigenschaft ändern**) und sie dann in CVS zu speichern.
- Bei Webprojekten aus einem CVS-Repository der Version 4 muss die Einstellung **Fenster > Benutzervorgaben > Team > CVS > Leere Verzeichnisse bereinigen** inaktiviert werden (in der Standardeinstellung ist sie aktiviert). Wenn Sie diese Einstellung *nicht* inaktivieren und ein Webprojekt mit einem leeren Quellenordner **source** (z. B. das Webbeispiel "MyHomePage") laden, werden bei der Rückgabe die folgenden Fehler ausgegeben:  
Das Projekt wurde nicht erstellt, da es sich in einem Zyklus befindet oder da es Probleme mit dem Klassenpfad hat.  
Fehlender erforderlicher Quellenordner: /MyHomePageExample403/source.
- Bei Webprojekten, die in einem ClearCase-Repository gespeichert und aus diesem geladen wurden, müssen Sie eine Datei entnehmen, bevor Sie sie im Editor öffnen können. Wird die Datei nicht entnommen, erhalten Sie Fehler mit der Angabe Fehler beim Aktivieren dieser Sicht (Ausnahmebedingung für Nullverweis in Protokollen für Page Designer). Wenn Sie im Editor xml eine Datei web.xml bearbeiten wollen, müssen Sie zuvor die Dateien web.xml, ibm-web-bnd.xmi und ibm-web-ext.xmi entnehmen. (Die Statuszeile enthält Meldungen darüber, dass Sie diese Dateien entnehmen müssen und dass die Dateien schreibgeschützt sind. Diese Meldungen werden jedoch leicht übersehen.)
- Wenn Ihre Projekte Schleifenabhängigkeiten aufweisen, meldet Version 5 einen Erstellungsfehler. In diesem Fall können Sie die Optionen **Fenster > Benutzervorgaben > Java > Compiler** auswählen und dann auf der Registerkarte **Erstellungspfad** das Markierungsfeld **Erstellung bei Erstellungspfadfehlern abbrechen** abwählen.
- Die Dateien .vcm\_meta (oder .cc\_meta) könnten zu diesem Zeitpunkt aus dem Projekt von Version 5 gelöscht werden, weil sie von Version 5 nicht verwendet werden (stattdessen wird eine neue Datei .project verwendet) und weil Sie für diese Projekte von Version 5 eine neue Repositoryverzweigung (Datenstrom) verwenden. Bitte beachten Sie, dass diese Dateien in der aktuellen Verzweigung von Version 4 (Datenstrom) weiterhin benötigt werden.

## Absolute Pfadverweise für EAR und Serverkonfiguration nach der Migration entfernen

EAR-IBM Anwendungserweiterungsdateien und Serverkonfigurationsdateien von Version 4 haben absolute Pfadverweise enthalten. Nach der Migration dieser Dateien auf Version 5 müssen Sie sie mit ihrem Editor öffnen (dadurch werden ihre alten absoluten Pfadverweise in neue relative Verweise geändert).

1. Klicken Sie für jedes EAR-Projekt in der Sicht "Navigator" mit der rechten Maustaste auf **META-INF/application.xml > Öffnen mit > Editor für den Deploymentdeskriptor**.
  - a. Ein Dialogfenster mit der folgenden Nachricht wird angezeigt:  
Die IBM Erweiterungsdatei enthält veraltete absolute Pfade.  
Dies kann automatisch korrigiert werden und sollte gespeichert werden.  
Diese Operation entfernt die Pfade aus der Datei und muss lediglich einmal ausgeführt werden.  
Soll die automatische Korrektur ausgeführt werden?
  - b. Klicken Sie auf **Ja**.
  - c. Speichern Sie, und schließen Sie dann das Editorfenster.

**Anmerkung:** Alternativ hierzu können Sie mit dem J2EE-Migrationsassistenten die Projektstruktur für nur ein EAR-Projekt migrieren. Wenn Sie auf den J2EE-Migrationsassistenten zugreifen möchten, klicken Sie mit der rechten Maustaste auf das EAR-Projekt, und wählen Sie anschließend **Migrieren > J2EE-Migrationsassistent** aus.

2. Klicken Sie für jede Serverkonfiguration in der Sicht "Serverkonfiguration" der Perspektive "Server" mit der rechten Maustaste auf den Server, und wählen Sie dann **Öffnen** aus.
  - a. Daraufhin wird ein ähnlicher Dialog für die automatische Korrektur angezeigt.
  - b. Klicken Sie auf **Ja**.
  - c. Speichern Sie, und schließen Sie dann das Editorfenster.

## Projekte mit anderen SCMs migrieren

SCM-Plug-ins für IBM WebSphere Studio Site Developer werden auch von anderen SCM-Lieferanten zur Verfügung gestellt. Eine Liste aller Lieferanten finden Sie unter [www.ibm.com/software/ad/studioappdev/partners/scm.html](http://www.ibm.com/software/ad/studioappdev/partners/scm.html). Im Rahmen der Validierung *Ready for IBM WebSphere Studio Software* [[www.developer.ibm.com/websphere/ready.html](http://www.developer.ibm.com/websphere/ready.html)] haben alle SCM-Lieferanten, die ein Plug-in für Version 4 bereitgestellt haben, versichert, dass die vorstehend beschriebenen Migrationsschritte (Speichern aus Version 4 im SCM-Repository, Laden aus dem Repository in Version 5) auch bei ihren Systemen verwendet werden können.

---

## Projekte durch Export und Import migrieren

1. Exportieren Sie Ihre Projekte in IBM WebSphere Studio Site Developer Version 4.0.x in eine WAR-Datei, eine EAR-Datei oder eine JAR-Datei (**Datei > Exportieren**).
2. Importieren Sie in IBM WebSphere Studio Site Developer Version 5 Ihre WAR-Datei, EAR-Datei oder JAR-Datei (**Datei > Importieren**).

**Anmerkung:** Dies ist keine vollständige Migration, da die Informationen zum Projekterstellungspfad nicht erhalten bleiben.

---

## Projekte mit einem vorhandenen Arbeitsbereich aus Version 4.0.x migrieren

Diese Methode wird teilweise unterstützt und führt zu einer unvollständigen Migration. Einstellungen für die Benutzerschnittstelle, Debugereinstellungen und die meisten Benutzervorgaben gehen hierbei verloren. Projektnamen, Projektquellendateien und Java-Projekterstellungspfade (Klassenpfade) bleiben zwar erhalten, aber die Erhaltung weiterer Daten kann nicht garantiert werden. Diese Methode sollte nur dann verwendet werden, wenn kein unterstütztes SCM-System eingesetzt wird und eine Beibehaltung der Informationen zum Projekterstellungspfad von wesentlicher Bedeutung ist (diese Informationen gehen verloren, wenn Sie Projekte importieren, die aus Version 4 exportiert wurden). Sie können den vorhandenen Arbeitsbereich aus Version 4.0.x verwenden, indem Sie folgendermaßen vorgehen:

1. Schreiben Sie alle anstehenden Änderungen fest (Freigabe).
2. Schließen Sie alle Perspektiven, und fahren Sie IBM WebSphere Studio Site Developer Version 4 herunter.
3. Sichern Sie den Inhalt des *arbeitsbereichsverzeichnisses*. Die Variable *arbeitsbereichsverzeichnis* steht für den vollständig qualifizierten Namen des Verzeichnisses, das den Arbeitsbereich aus Version 4.0.x enthält. In der Standardeinstellung befindet sich das Unterverzeichnis für den Arbeitsbereich aus Version 4.0.x in dem gleichen Verzeichnis, in dem das Produkt installiert ist. Diese Sicherung benötigen Sie, wenn Sie erneut mit IBM WebSphere Studio Site Developer Version 4.0.x arbeiten möchten. Nachdem Sie aus einer IDE der Version 5 einen Verweis auf einen Arbeitsbereich aus Version 4.0.x erstellt haben, können Sie in IBM WebSphere Studio Site Developer Version 4.0.x nicht mehr mit diesem Arbeitsbereich arbeiten.
4. Installieren Sie IBM WebSphere Studio Site Developer Version 5.
5. Starten Sie IBM WebSphere Studio Site Developer Version 5 mit einem Arbeitsbereich aus Version 4.0.x über eine Eingabeaufforderung (d. h. geben Sie im Befehl mit der Option `-data` einen vollständig qualifizierten Pfad für das Arbeitsbereichsverzeichnis aus Version 4.0.x an), dies bewirkt ein Upgrade der `.metadaten`-Informationen.
6. Wenn Sie aufgefordert werden, die Konvertierung in das neue Benutzer-schnittstellenformat zu bestätigen, klicken Sie auf **OK**.

7. Bevor Sie Projekte, die sich im Arbeitsbereich befinden, erneut erstellen oder auswerten, wählen Sie in der Sicht "Navigator" der Perspektive "Ressource" alle Projekte aus, und wählen Sie dann die Option **Aktualisieren** im Kontextmenü aus. Auf diese Weise stellen Sie sicher, dass alle Dateien mit ihren entsprechenden Metadaten synchronisiert werden.
8. Öffnen Sie alle geschlossenen Projekte (weitere Informationen finden Sie unter "Bekannte Probleme").
9. Prüfen Sie die Klassenpfadvariablen (weitere Informationen finden Sie unter "Bekannte Probleme").
10. Manche Erstellungsprogramme und Prüfprogramme wurden in der vorliegenden Version 5 hinzugefügt, entfernt oder geändert. Um sicherzustellen, dass die korrekten Fehler und Warnungen angezeigt werden, müssen Sie alle Projekte durch Auswahl von **Projekt > Alle erneut erstellen** erneut erstellen und dann für jedes Java-Projekt den Befehl **Gültigkeitsprüfung ausführen** verwenden.
11. Einige Benutzervorgaben bleiben unter Umständen erhalten, andere jedoch nicht. Prüfen Sie daher die Benutzervorgaben in Version 5, um sicherzustellen, dass die von Ihnen gewünschten Einstellungen verwendet werden.

## Absolute Pfadverweise für EAR und Serverkonfiguration nach der Migration entfernen

Die Anweisungen zur Vorgehensweise nach der Migration, wie sie in „Absolute Pfadverweise für EAR und Serverkonfiguration nach der Migration entfernen“ auf Seite 35 beschrieben sind, gelten auch hier.

## Bekannte Probleme und Einschränkungen

Die folgenden Probleme können auftreten, wenn Sie versuchen, die Migration durch das Öffnen eines Arbeitsbereichs aus Version 4.0 in IBM WebSphere Studio Site Developer Version 5 vorzunehmen.

### Falscher Wert in Klassenpfadvariable JRE\_LIB

Mit den folgenden Schritten können Sie die Klassenpfadvariable JRE\_LIB auf eine gültige Position zurücksetzen. *Führen Sie dies auch dann durch, wenn der Wert beim ersten Öffnen des Fensters "Benutzervorgaben" korrekt zu sein schien.*

1. Wählen Sie nacheinander die Optionen **Fenster > Benutzervorgaben > Java > Installierte JREs** aus.
2. Wählen Sie in der Liste das Markierungsfeld für die standardmäßige JRE-Position aus, die für die Variable JRE\_LIB festgelegt werden soll.
3. Wählen Sie **Bearbeiten** aus, und klicken Sie anschließend auf **OK**, um das Dialogfenster JRE bearbeiten zu schließen.

Wenn Sie diesen Arbeitsschritt nicht ausführen, ist der Wert für JRE\_LIB möglicherweise falsch, was viele Erstellungsfehler in Java-Dateien verursachen kann.

Als allgemeine Prüfmaßnahme sollten Sie die Werte aller Klassenpfadvariablen überprüfen.

## Menü "Team" enthält Option "Projekt gemeinsam benutzen" für frühere gemeinsam benutzte SCM-Projekte

Die Teamunterstützung wurde von Eclipse 1.0 auf Eclipse 2.0 in wesentlichen Punkten geändert. Auch die Methode für die gemeinsame Benutzung von Projekten über das Repository hat sich geändert.

- Wenn Sie mit der rechten Maustaste auf das Projekt klicken, und dann die Optionen **Team > Projekt gemeinsam benutzen** auswählen, werden Sie von einem Assistenten durch den Migrationsprozess geführt. Nach Abschluss der Migration wird das Projekt gemeinsam benutzt, und die Sicht "Synchronisieren" wird geöffnet. Alle Dateien mit Konflikten verursachenden Änderungen werden angezeigt. Dies liegt daran, dass die Speicherungsmethode für Synchronisierungsinformationen von Eclipse 1.0 auf Eclipse 2.0 geändert wurde.
- Falls keine abgehenden Änderungen vorhanden sind (dies sollte der Fall sein, wenn vor dem Upgrade alle abgehenden Änderungen wie empfohlen festgeschrieben worden sind), können Sie das Projekt einfach in der Sicht "Synchronisieren" auswählen und anschließend mit dem Befehl **Überschreiben und aktualisieren** den aktuellen Inhalt aus dem Server laden.
- Sind abgehende Änderungen vorhanden, erweitern Sie das durch ein Dreieckssymbol angegebene Menü in der Sicht "Synchronisieren", und wählen Sie die Option **Dateiinhalte vergleichen** aus. Nachdem Sie einige Arbeitsschritte ausgeführt haben, zeigt die Sicht "Synchronisieren" nur die Dateien an, die tatsächlich unterschiedlich sind. Anschließend können Sie die entsprechenden Konflikte in der Sicht "Synchronisieren" auflösen.

## Außerhalb des Arbeitsbereichsverzeichnisses erstellte Projekte

In der Standardeinstellung werden Projekte im Arbeitsbereichsverzeichnis erstellt. Falls Sie die Standardeinstellung überschrieben und Projekte in anderen Positionen erstellt haben, müssen Sie alle Ihre Projekte öffnen, bevor Sie die Workbench schließen. Hierdurch kann die Datei .project für ein solches Projekt in die richtige Position geschrieben werden. Wenn Sie ein geschlossenes Projekt, dessen Verzeichnis nicht zum Arbeitsbereich gehört, nicht öffnen, entsteht ein Projekt, das das eigentliche Projekt maskiert und das lediglich eine Datei .project enthält.

## JSP-Unterbrechungspunkte müssen zurückgesetzt werden

Sie müssen alle vorhandenen JSP-Unterbrechungspunkte entfernen und im migrierten Arbeitsbereich von Version 5 zurücksetzen.

---

## Relationale Daten in 4.0.3-Webprojekten migrieren

Gehen Sie folgendermaßen vor, um relationale Daten von Projekten in IBM WebSphere Studio Site Developer 4.0.3 zu migrieren:

1. Generieren Sie von einem Arbeitsbereich in IBM WebSphere Studio Site Developer 4.0.3 DDL-Dateien für jede verfügbare Datenbank.
2. Entfernen Sie die Datenbank mit Hilfe der Ansicht Datendefinition aus dem Ordner für Webprojektquellen/-datenbanken.
3. Öffnen Sie den 4.0.3-Arbeitsbereich mit IBM WebSphere Studio Site Developer Version 5.
4. Migrieren Sie die Webprojekte, für die Sie relationale Daten wiederherstellen möchten.
5. Klicken Sie auf **Datei > Importieren > Dateisystem**, und geben Sie die DDL-Dateien von Ihrem 4.0.3-Arbeitsbereich an.
6. Wählen Sie in der Ansicht Datendefinition der Datenperspektive **Gegen Lokal ausführen**, und geben Sie das Zielwebprojekt an.

Die Artefakte der relationalen Daten werden wiederhergestellt.

## WSDL-Fehler nach Import einer Web-Servicedatei von 4.0.x

Wenn Sie eine Web-Servicedatei von 4.0.x importiert haben, erhalten Sie möglicherweise folgende Fehlermeldungen:

Error The part 'result' has an invalid value 'anyElement' defined for its type. Type declarations must refer to valid values defined in a schema.  
(Fehler Für den Typ der Komponente 'result' ist ein ungültiger Wert 'anyElement' definiert. Typendeklarationen müssen sich auf gültige Werte beziehen, die in einem Schema definiert sind.)

Error The part 'return' has an invalid value 'findPatientResult' defined for its element.  
(Fehler Für das Element der Komponente 'return' ist ein ungültiger Wert 'findPatientResult' definiert.)  
Element declarations must refer to valid values defined in a schema. (Elementdeklarationen müssen sich auf gültige Werte beziehen, die in einem Schema definiert sind.)

Error The part 'response' has an invalid value 'findPatientResponse' defined for its element.  
(Fehler Für das Element der Komponente 'response' ist ein ungültiger Wert 'findPatientResponse' definiert.)  
Element declarations must refer to valid values defined in a schema. (Elementdeklarationen müssen sich auf gültige Werte beziehen, die in einem Schema definiert sind.)

Die Abhilfemaßnahme ist:

1. Löschen Sie die WSDL-Dateien.
2. Regenerieren Sie Ihre Web-Services, indem Sie den Web-Services-Assistenten wiederholen.

---

## J2EE-Projektstrukturen und/oder J2EE-Spezifikationsstufen migrieren

Um auf den J2EE-Migrationsassistenten in Version 5 zuzugreifen, führen Sie die folgenden Schritte aus:

1. Wählen Sie das Projekt aus.
2. Klicken Sie mit der rechten Maustaste darauf, und wählen Sie dann **Migrieren** > **J2EE-Migrationsassistent** aus. Befolgen Sie die Schritte im Assistenten, die Sie durch die Migration führen.
3. Wenn Ihr Projekt der Versionssteuerung unterliegt, speichern Sie das umstrukturierte Projekt in Ihrer SCM.



---

## Kapitel 6. Migration von WebSphere Studio "Classic" auf IBM WebSphere Studio Site Developer

Dieses Kapitel dokumentiert, wie Sie eine Migration von WebSphere Studio Version 4.0 (sowohl Advanced als auch Professional Edition) auf IBM WebSphere Studio Site Developer vornehmen. Die Migration von WebSphere Studio "Classic" Version 4.0 auf IBM WebSphere Studio Site Developer Version 5.0 umfasst die folgenden Schritte:

1. Neue Einzelserverumgebung für die Migration erstellen.
2. Deskriptordatei für die Webkonfiguration erstellen.
3. JAR-Migrationsdatei exportieren.
4. JAR-Migrationsdatei in IBM WebSphere Studio Site Developer importieren.
5. Server einrichten und migrierte Anwendung testen.

**Anmerkung:** Die folgenden Anweisungen gelten für die Migration von WebSphere Studio Version 4.0. Wenn Sie eine frühere Version von WebSphere Studio migrieren wollen, sollten Sie zunächst auf WebSphere Studio 4.0 migrieren und anschließend auf IBM WebSphere Studio Site Developer.

Die Komponente für die erweiterte Publikation (Zuordnung von Dateien zu Publikationsphasen) sowie die Komponente Page Detailer (Analyse von Webseiten) von WebSphere Studio "Classic" sind in IBM WebSphere Studio Site Developer nicht verfügbar. Einige andere Komponenten vom CD-Datenträgerpack von Version 4.0.x sind ebenfalls nicht mehr verfügbar. Dazu gehören beispielsweise die Komponente Page Detailer für die Analyse von Webdateien, die Komponente Hot-Media<sup>®</sup> für speicherintensive Datenträgertypen, der XML-Editor Voice (jetzt in WebSphere Everyplace<sup>™</sup> Toolkit und Portal Toolkit enthalten) sowie DataBaseWizard für Einheiten für Pervasive Computing.

Bitte beachten Sie die folgenden Einschränkungen, bevor Sie Daten aus WebSphere Studio migrieren:

- IBM WebSphere Studio Site Developer verwendet einen XML-basierten SQL-Editor. Daher können Sie Ihre .sql-Dateien nicht in diesem Editor verwenden.
- Projekt- und Phaseninformationen für Projekte können nicht migriert werden.
- Informationen zur Serverkonfiguration aus WebSphere Studio können nicht migriert werden.
- Informationen zur Versionssteuerung können nicht migriert werden.

Während des im Folgenden beschriebenen Migrationsprozesses erstellt WebSphere Studio eine JAR-Datei für einen Einzelserver, die alle Projektdateien, Publikationsangaben und Quelleninformationen enthält. Alle in der Sicht **Publikation** für den Standardserver angezeigten Dateien werden in der JAR-Datei gepackt. Sie können anschließend die JAR-Datei in IBM WebSphere Studio Site Developer importieren.

Wenn Sie vorhandene Projekte migrieren, gehen während der Migration alle Projekt- und Phaseninformationen dafür verloren. Falls Ihre Umgebung mit mehreren Servern arbeitet, bleiben nur die Dateien erhalten, die auf dem Standardserver publiziert wurden. Daher müssen Sie für die Migration eine neue Umgebung erstellen, die nur einen Server verwendet.

---

## Neue Einzelserverumgebung für die Migration erstellen

Falls Ihre aktuelle Umgebung mit mehreren Servern arbeitet, können Sie auf die folgende Weise eine neue Umgebung mit dem Namen "Migration" erstellen, die nur einen Server verwendet:

1. Klicken Sie auf **Projekt > Publikationsphase anpassen**.
2. Geben Sie in das Feld **Name der Phase** den Wert Migration ein.
3. Klicken Sie auf **Hinzufügen**.
4. Klicken Sie auf **OK**.
5. Klicken Sie auf **Projekt > Publikationsphase**, und wählen Sie in der Liste der verfügbaren Phasen den Eintrag **Migration** aus.
6. Klicken Sie in der Sicht **Publikation** auf **Einfügen > Server**.
7. Geben Sie einen Servernamen wie beispielsweise localhost ein.
8. Bei der Änderung des Servers oder der Publikationsphase werden die Servletzuordnungsinformationen für WebSphere Application Server Version 4.0 nicht weitergegeben. Rufen Sie die Sicht **Publikation** auf, klicken Sie für jedes Servlet auf **Eigenschaften > Publikation > Servletzuordnung**, und kopieren Sie anschließend die tatsächliche Servletzuordnung.

---

## Deskriptordatei für Webkonfiguration erstellen

1. Klicken Sie in der Sicht mit der Projektdatei auf **Projekt > Deskriptordatei für Webkonfiguration erstellen**.
2. Wählen Sie alle erforderlichen Servlets aus.
3. Wählen Sie alle erforderlichen TLD-Dateien (Tag Library Descriptor - Tagbibliothekdeskriptor) aus.
4. Klicken Sie auf **Erstellen**.

Der Standardname der Deskriptordatei für die Webkonfiguration (*serverName\_web.xml*) lautet im vorliegenden Szenario localhost\_web.xml. Sofern Sie keine andere Position angeben, wird die Datei .xml im Verzeichnis WEB-INF gespeichert.

---

## JAR-Migrationsdatei erstellen

1. Wählen Sie in der Sicht mit der Projektdatei den Server **localhost** aus, klicken Sie auf **Eigenschaften > Publikation > WebApp**, und geben Sie einen Webpfad (Stammverzeichnis für Kontext) wie z. B. myWebPath ein. Diese Angabe wird auch als WebSphere Application Server - Express-Projektname verwendet.
2. Wählen Sie in der Sicht mit der Projektdatei die Optionen **Projekt > Migrationsdatei erstellen** aus.
3. Prüfen Sie, ob der Server **localhost** ausgewählt ist.
4. Prüfen Sie, ob die Datei **localhost\_web.xml** als Deskriptordatei für die Webkonfiguration ausgewählt ist.
5. Klicken Sie auf **OK**.
6. Der Standardname für die JAR-Datei (*serverName.jar*) lautet im vorliegenden Szenario localhost.jar. Auf Wunsch können Sie die Datei umbenennen.
7. Speichern Sie die JAR-Datei.

---

## JAR-Migrationsdatei in IBM WebSphere Studio Site Developer importieren

1. Starten Sie IBM WebSphere Studio Site Developer.
2. Erstellen Sie ein Webprojekt (**Datei > Neu > Projekt > Webprojekt**).
3. Geben Sie in das Feld **Projektname** den Namen des Webprojekts ein. Dieser Name sollte mit dem Namen identisch sein, den Sie in Schritt 1 des vorherigen Abschnitts „JAR-Migrationsdatei erstellen“ auf Seite 42 angegeben haben.
4. Geben Sie den Namen eines neuen oder eines vorhandenen EAR-Projekts an, das das Webprojekt für die Implementierung enthalten soll.
5. Geben Sie im Feld **Stammverzeichnis für Kontext** den Webapp-Webpfadnamen ein, den Sie bei der Erstellung der JAR-Migrationsdatei in WebSphere Studio angegeben haben. Klicken Sie auf **Fertig stellen**.
6. Wählen Sie in der Sicht "Navigator" das soeben erstellte Webprojekt aus.
7. Importieren Sie die JAR-Datei.
  - a. Klicken Sie auf **Datei > Importieren**.
  - b. Klicken Sie auf **WAR-Datei**. Klicken Sie auf **Weiter**. Sie *müssen* die JAR-Datei mit der Option "WAR-Datei" importieren, da eine einwandfreie Funktionsweise andernfalls nicht gewährleistet ist.
  - c. Geben Sie den Pfad für die Datei localhost.jar im Feld **WAR-Datei** ein, oder klicken Sie auf **Durchsuchen**, um nach der Datei zu suchen. (Sie können nur nach Dateinamen mit der Erweiterung .WAR suchen, nicht nach Dateinamen mit der Erweiterung .JAR.)
  - d. Wählen Sie das vorhandene Webprojekt aus, das Sie erstellt haben. Im Feld **Stammverzeichnis für Kontext** wird automatisch der zuvor angegebene Wert eingesetzt.
  - e. Klicken Sie auf **Fertig stellen**. Ein Dialog wird angezeigt, in dem Sie Folgendes gefragt werden: "Die Ressource WEB-INF/web.xml ist bereits vorhanden. Soll sie überschrieben werden?".
  - f. Wählen Sie **Ja** aus. IBM WebSphere Studio Site Developer entpackt nun die Datei *localhost.jar*.
8. Möglicherweise werden mehrere Verweise nicht aufgelöst oder Importdateien fehlen. Diese werden in der Sicht "Tasks" angezeigt. Zur Behebung dieses Fehlers müssen Sie den Java-Erstellungspfad für das Webprojekt ändern:
  - a. Klicken Sie mit der rechten Maustaste auf **Eigenschaften > Java-Erstellungspfad**.
  - b. Klicken Sie auf die Registerkarte **Bibliotheken**. Klicken Sie auf **Externe JARs hinzufügen**.
  - c. Importieren Sie alle benötigten JARs aus den folgenden Verzeichnissen:
    - WS\_Installdir/runtimes/aes\_v4/lib und
    - WS\_Installdir/runtimes/base\_v4/lib
9. Klicken Sie in der Sicht "Navigator" mit der rechten Maustaste auf das Projekt, und wählen Sie **Projekt erneut erstellen** aus.

---

## Migrierte Anwendung auf einem Testserver testen

Jetzt können Sie Ihre Anwendung testen. Zum Testen der Anwendung auf dem Standardtestserver führen Sie die folgenden Schritte aus:

1. Klicken Sie mit der rechten Maustaste auf das EAR-Projekt.
2. Wählen Sie die Option **Auf Server ausführen** aus.

Informationen zum Testen Ihrer Anwendung in anderen Serverlaufzeitumgebungen finden Sie im Onlinehilfetext für die Komponente "Sertertools".

---

## Kapitel 7. Migration von VisualAge für Java auf IBM WebSphere Studio Site Developer

Dieses Kapitel erläutert die Migration von VisualAge® für Java Professional Edition oder VisualAge für Java Enterprise Edition auf IBM WebSphere Studio Site Developer.

**Anmerkung:** Die Anweisungen in diesem Kapitel gelten für die Migration von VisualAge für Java Version 3.5.3 oder 4.0 für Windows. Wenn Sie von einer früheren Version von VisualAge für Java auf IBM WebSphere Studio Site Developer migrieren wollen, sollten Sie zuerst von Ihrer früheren Version von VisualAge für Java auf Version 3.5.3 oder 4.0 für Windows migrieren, bevor Sie die Migration auf IBM WebSphere Studio Site Developer ausführen.

**Anmerkung:**  Der IBM Business Partner Instantiations, Inc. vertreibt ein Produkt namens CodePro Studio, das Produktivitätserweiterungen für VisualAge für Java und WebSphere Application Server - Express bereitstellt, darunter auch Funktionen für Migration und Koexistenz. Damit Kunden, die VisualAge für Java verwenden, mit der Migration beginnen können, bietet Instantiations im Rahmen seiner zeitbegrenzten Probeversion von CodePro Studio die Exportfunktion von VisualAge für Java nach IBM WebSphere Studio Site Developer kostenlos und ohne Verwendungsbegrenzung an. Diese Probeversion können Sie unter der Adresse [www.instantiations.com/vaj-migrate](http://www.instantiations.com/vaj-migrate) herunterladen. Weitere Informationen zur erweiterten Koexistenzunterstützung sowie zur erweiterten Migrationsunterstützung von Instantiations (inklusive vollständige bidirektionale Dateixporte und -importe, Erstellung von Export-/Importgruppen, Projektsynchronisierung und Taskautomatisierung) finden Sie bei Instantiations, Inc. unter [www.instantiations.com/codepro/ws](http://www.instantiations.com/codepro/ws).

---

### Unterschiede zwischen VisualAge für Java und IBM WebSphere Studio Site Developer

Die folgende Liste enthält einen Ausschnitt der Änderungen seit VisualAge für Java:

- Die Spezifikationsstufe für EJBs (Enterprise Java Beans) wurde von 1.0 in 1.1 geändert (EJB 2.0 wird außerdem bei Anwendungen unterstützt, die in WebSphere Application Server Version 5 implementiert werden).
- Bei Webanwendungen bleibt die JSP-Stufe bei 1.1 (1.2 für Anwendungen auf WebSphere Application Server Version 5).
- Bei Webanwendungen bleibt die Servletstufe bei 2.2 (2.3 für Anwendungen auf WebSphere Application Server Version 5).
- Die Stufe der unterstützten Java 2-Plattform wurde von 1.2 in 1.3 geändert. (Der Compiler kann zwar Code für 1.4 generieren, aber die Laufzeitumgebung von WebSphere Application Server ist weiterhin 1.3.)
- Visual Composition Editor wurde durch Visual Editor für Java ersetzt.

- Die Versionssteuerung und das Repository für proprietären Quellcode von VisualAge für wurden durch die Unterstützung von SCM-Plug-ins (Software Configuration Management - Softwarekonfigurationsverwaltung) ersetzt.
- Die Tool-API von VisualAge für Java wurde durch die Plug-in-Architektur der WebSphere Studio-Workbench ersetzt.
- Die XML-Tools von VisualAge für Java XML wurden durch die XML-Tools von IBM WebSphere Studio Site Developer ersetzt.
- Das Projektkonzept von VisualAge für Java wurde durch mehrere Projekttypen von IBM WebSphere Studio Site Developer ersetzt.

---

## Migration von VisualAge für Java

Die folgenden Schritte skizzieren die Migration von VisualAge für Java. Details zu diesen Schritten finden Sie im weiteren Verlauf dieses Kapitels.

1. Exportieren Sie Ihre Java-Dateien und Projektressourcendateien aus VisualAge für Java.
2. Starten Sie IBM WebSphere Studio Site Developer, und erstellen Sie neue Projekte, die Ihren Code enthalten sollen.
3. Importieren Sie Ihre Java- und Projektressourcendateien in IBM WebSphere Studio Site Developer.
4. Verwenden Sie den Editor web.xml, um sicherzustellen, dass Servlets korrekt definiert sind (nur bei Webprojekten).
5. Migrieren Sie Ihre Projekt- und Arbeitsbereichseinstellungen.
6. Richten Sie Ihren Server ein, und testen Sie Ihre migrierte(n) Anwendung(en).
7. Implementieren Sie Ihre Anwendungen aus IBM WebSphere Studio Site Developer in WebSphere Application Server.
8. Stellen Sie Ihre Projekteinstellungen für IBM WebSphere Studio Site Developer mehreren Entwicklern zur gemeinsamen Benutzung zur Verfügung (nach der Migration).

### Java-Dateien und Projektressourcendateien aus VisualAge für Java exportieren

Die zusammengefasste Migration aller versionierten Projekte und Ressourcen aus dem Repository von VisualAge für Java wird nicht unterstützt. Sie können lediglich Projekte und Ressourcen migrieren, die sich im Arbeitsbereich von VisualAge für Java befinden. Wenn Sie die versionierte Kopie eines Projekts oder einer Ressource auf IBM WebSphere Studio Site Developer migrieren wollen, müssen Sie sie in den Arbeitsbereich von VisualAge für Java stellen und anschließend migrieren.

**Anmerkung:** Wenn Ihr Projekt mehrere Typen von Daten enthält (z. B. Enterprise-Beans und Java-Quellcodedateien), sollten Sie die Daten nach Typ in mehrere JARs aufteilen.

So können Sie Ihre Projekte in eine JAR-Datei exportieren:

1. Wenn sich die Projekte, die Sie exportieren wollen, gegenwärtig nicht im Arbeitsbereich von VisualAge für Java befinden, fügen Sie sie jetzt zum Arbeitsbereich hinzu.
2. Wählen Sie im Fenster der Workbench von VisualAge für Java Ihr Projekt aus, klicken Sie mit der rechten Maustaste, und klicken Sie dann auf **Exportieren**.
3. Wählen Sie das Optionsfeld **JAR-Datei** aus, und klicken Sie auf **Weiter**.
4. Geben Sie den Namen der JAR-Datei an.

5. Wählen Sie das Markierungsfeld **.java** aus, um die Java-Dateien zu exportieren, sowie das Markierungsfeld **Ressourcen**, um die Ressourcendateien zu exportieren.
6. Füllen Sie die anderen Felder wie erforderlich aus. Weitere Informationen zu dieser Task können Sie in der Onlinehilfe von VisualAge für Java nachlesen.

## IBM WebSphere Studio Site Developer starten und neue Projekte erstellen, die Ihren Code enthalten sollen

Starten Sie IBM WebSphere Studio Site Developer, und erstellen Sie anschließend die entsprechenden Projekte. Die folgenden Angaben sind allgemeine Migrationsrichtlinien, die Ihnen bei der Entscheidung helfen, in welchen Projekttyp von IBM WebSphere Studio Site Developer Sie Ihre Dateien importieren sollten:

- Gehört der Code zu einer Webanwendung, müssen Sie den Code in ein Webprojekt importieren:
  - Importieren Sie alle Java-Dateien in das Verzeichnis für die Java-Quelle (**Java Source**) des Webprojekts (die korrekte Hierarchie wird automatisch gemäß der Paketanweisungen durch IBM WebSphere Studio Site Developer erstellt).
  - Importieren Sie alle Ressourcendateien in das Webinhaltsverzeichnis (**Web Content**) des Webprojekts.
- Wenn Ihr Code ausschließlich in Java geschrieben ist (beispielsweise eine eigenständig ausgeführte Anwendung), müssen Sie den Code in ein Java-Projekt importieren.

**Anmerkung:** Die vorstehenden Angaben sind nur allgemeine Migrationsrichtlinien, die Ihnen bei der Auswahl des zu verwendenden Projekttyps von IBM WebSphere Studio Site Developer helfen sollen. Vor dem Erstellen von Projekten und dem Importieren von Code empfiehlt es sich, die Onlinehilfe von IBM WebSphere Studio Site Developer durchzulesen und sich mit den unterschiedlichen Projekttypen von WebSphere Application Server - Express vertraut zu machen.

## Java- und Ressourcendateien in IBM WebSphere Studio Site Developer importieren

1. Öffnen Sie IBM WebSphere Studio Site Developer, und wechseln Sie in die Ressourcenperspektive.
2. Klicken Sie auf **Datei > Importieren > Zip-Datei**. Klicken Sie auf **Weiter**.
3. Suchen Sie nach der gewünschten JAR-Datei.
4. Wählen Sie die zu importierenden Dateien sowie das Projekt oder den Ordner aus, in das/den die Dateien importiert werden sollen.

**Anmerkung:**

- Der Dateisystemimport kann ebenso anstelle des ZIP-Imports verwendet werden, allerdings wird der ZIP-Import häufiger verwendet. Sie können dieselben Anweisungen auch für einen Dateiimport verwenden und auf einen Ordner zeigen anstatt auf eine ZIP-Datei im Assistenten für Dateiimport.
- Wenn Sie Ihre Dateien in IBM WebSphere Studio Site Developer importieren, sollten Sie sicherstellen, dass die Dateien in das entsprechende Verzeichnis gestellt werden. Vor dem Importieren von Code empfiehlt es sich, die Onlinehilfe von IBM WebSphere Studio Site Developer durchzulesen und sich mit den unterschiedlichen Projekttypen von IBM WebSphere Studio Site Developer vertraut zu machen. Diese Angaben helfen Ihnen dabei, die jeweiligen Ordner für einen bestimmten Codetyp zu ermitteln.

## Korrekte Definition von Servlets unter Verwendung des Editors web.xml sicherstellen (nur bei Webprojekten)

Falls Ihre Anwendung Servlets verwendet, müssen Sie die URL-Zuordnungen für die Servlets in der Datei web.xml definieren. Hierzu gehen Sie folgendermaßen vor:

1. Öffnen Sie in der Perspektive "Web" die Datei web.xml, die sich im Unterverzeichnis "Web Content/WEB-INF" des Webprojekts befindet.
2. Klicken Sie auf die Registerkarte **Servlets**.
3. Klicken Sie auf **Hinzufügen**, und wählen Sie das Optionsfeld **Servlet** aus.
4. Geben Sie den Servletnamen ein, und klicken Sie auf **OK**.
5. Klicken Sie auf **Durchsuchen**, um den Wert für die **Servletklasse** in den gewünschten Paketnamen zu ändern.
6. (Optional) Der Anzeigename ist ein Kurzname für das Servlet. Geben Sie im Feld **Anzeigename** einen Kurznamen für das Servlet ein.
7. Eine URL-Zuordnung definiert ein Servlet und ein URL-Muster. Klicken Sie auf die Schaltfläche **Hinzufügen** neben dem Feld **URL-Zuordnungen**, und geben Sie anschließend den Namen der URL-Zuordnung ein.
8. Speichern Sie die Änderungen (**Datei > web.xml speichern**), und schließen Sie die Datei web.xml.

## Projekt- und Arbeitsbereichseinstellungen migrieren

Die folgenden Einstellungen von VisualAge für Java müssen Sie notieren und in IBM WebSphere Studio Site Developer konfigurieren:

- Projektklassenpfad
- Ressourcenzuordnungen
- Codeformatierung
- WTE-Konfiguration
- Java-Dateien und Projektressourcendateien

### Projektklassenpfad

In VisualAge für Java legen Sie den Projektklassenpfad auf der Seite **Ressourcen** des Fensters **Optionen** fest (**Fenster > Optionen > Ressourcen**). Nachdem Sie Ihre Projekte auf IBM WebSphere Studio Site Developer migriert haben, können Sie den Klassenpfad eines Projekts im Fenster **Eigenschaften** des Projekts festlegen. (Klicken Sie hierzu mit der rechten Maustaste auf das Projekt, und wählen Sie **Eigenschaften > Java-Erstellungspfad** aus.) Klicken Sie dann auf die Registerkarte **Bibliotheken**.) Sie können auch Klassenpfadvariablen im Fenster **Benutzervorgaben** festlegen (**Fenster > Benutzervorgaben > Java > Klassenpfadvariablen**.)

### Ressourcenzuordnungen

Wenn Sie eine Zuordnung zwischen einem Dateityp und einer ausführbaren Datei definieren, können Sie eine Datei, die sich außerhalb der Workbench befindet, aus der Workbench heraus öffnen.

In VisualAge für Java legen Sie die Ressourcenzuordnungen im Fenster **Optionen** fest (**Fenster > Optionen > Ressourcen > Ressourcenzuordnungen**).

Nachdem Sie die Ressourcendateien auf IBM WebSphere Studio Site Developer migriert haben, können Sie die Ressourcenzuordnungen im Fenster **Benutzervorgaben** definieren (**Fenster > Benutzervorgaben > Workbench > Dateizuordnungen**).

### Codeformatierungsprogramm

In VisualAge für Java legen Sie Ihre Codeformatierungsoptionen auf der Seite **Formatierungsprogramm** des Fensters **Optionen** fest (**Fenster > Optionen > Codierung > Formatierungsprogramm**). Nachdem Sie Ihren Code auf IBM WebSphere Studio Site Developer migriert haben, können Sie die Codeformatierung im Fenster **Benutzervorgaben** definieren (**Fenster > Benutzervorgaben > Java > Codeformatierungsprogramm**).

### WTE-Konfiguration

In VisualAge für Java sind die Laufzeiteinstellungen für die WebSphere-Einheiten-testumgebung und für WebSphere Application Server in unterschiedlichen Dateien des folgenden Verzeichnisses gespeichert: *VisualAge-Installationsverzeichnis\ide\project\_resources\IBM WebSphere Test Environment\properties*. Die Variable *VisualAge-Installationsverzeichnis* steht für das von Ihnen verwendete Produktinstallationsverzeichnis.

Falls Sie beispielsweise in der Eigenschaftendatei session.xml das erneute Schreiben der URL aktiviert haben, indem Sie die Eigenschaft durch die Angabe `<url-rewriting-enabled>true</url-rewriting-enabled>` auf "true" gesetzt haben, können Sie diese Eigenschaft in der Testumgebung von IBM WebSphere Studio Site Developer Version 4.0 konfigurieren. (Öffnen Sie in der Perspektive **Server** die Sicht **Serverkonfiguration**, klicken Sie mit der rechten Maustaste auf den gewünschten Server, und klicken Sie auf **Öffnen**. Klicken Sie auf die Registerkarte **Web**, und wählen Sie das Markierungsfeld **Erneutes Schreiben der URL aktivieren** aus.)

### Java-Dateien und Projektressourcendateien

Die Eigenschaftendatei default.servlet\_engine enthält das Stammverzeichnis für den Kontext (im Format `<root-uri>`) der Webanwendungen von VisualAge für Java. Beim Erstellen eines Webprojekts in IBM WebSphere Studio Site Developer enthält der Dialog **Webprojekt erstellen** ein Feld **Stammverzeichnis für Kontext** für diese Daten.

Webanwendungseinstellungen in Dateien wie *VisualAgeInstallationsverzeichnis\ide\project\_resources\IBM WebSphere Test Environment\hosts\default\_host\default\_app\servlets\default\_app.webapp*, die Sie selbst angepasst haben, sollten auf die Datei *Ihr\_Webprojekt\Web Content\WEB-INF\web.xml* in IBM WebSphere Studio Site Developer migriert werden. Wenn Sie beispielsweise Servletnamen und Servletpfade in der Datei default\_app.webapp geändert haben, müssen Sie die entsprechenden Änderungen auch in der Datei web.xml vornehmen.

## Testumgebung in WebSphere V4 definieren und migrierte Anwendung(en) testen

Wenn es sich bei der Anwendung um ein Java-Projekt handelt, verwenden Sie zum Testen die normale Unterstützung **Ausführen** oder **Debug** für Java-Projekte von IBM WebSphere Studio Site Developer.

Verwendet die Anwendung WebSphere Application Server , verwendet Sie zum testen den integrierten WebSphere Application Server. Hierzu muss ein Standard-testserver erstellt und gestartet werden. Bei einem Webprojekt klicken Sie mit der rechten Maustaste auf die HTML-Hauptseite, und wählen Sie die Option **Auf Server ausführen** aus, um den Web-Browser zu starten.

Informationen zum Testen anderer Projekttypen finden Sie in der Onlinehilfe.

## Ihre Anwendungen aus IBM WebSphere Studio Site Developer auf fernem WebSphere Application Server implementieren

Wenn Sie WebSphere Application Server als Laufzeitumgebung verwenden, implementieren Sie Ihre Anwendung mit der Komponente "Server-Tools" von IBM WebSphere Studio Site Developer.

## Projekteinstellungen für IBM WebSphere Studio Site Developer mehreren Entwicklern zur gemeinsamen Benutzung zur Verfügung stellen (nach der Migration)

IBM WebSphere Studio Site Developer-Projekte (und ihre zugeordneten Einstellungen) können von Entwicklern gemeinsam benutzt werden. Hierzu speichern Sie ein Projekt im SCM-Server von IBM WebSphere Studio Site Developer. Anschließend extrahieren Sie es unter IBM WebSphere Studio Site Developer für ein anderes Teammitglied.

---

## Teamunterstützung in IBM WebSphere Studio Site Developer

Informationen zur Teamunterstützung in IBM WebSphere Studio Site Developer Version 4.0 finden Sie unter [www.ibm.com/websphere/developer/library/techarticles/0108\\_karasiuk/0108\\_karasiuk.html](http://www.ibm.com/websphere/developer/library/techarticles/0108_karasiuk/0108_karasiuk.html)

Informationen zur Teamunterstützung in IBM WebSphere Studio Site Developer finden Sie außerdem im Installationshandbuch und in der Onlinehilfefunktion.

---

## Kapitel 8. Migration von VisualAge für Java Visual Composition Editor auf Visual Editor für Java

Dieses Kapitel enthält Anweisungen für die Migration von Anwendungen, die mit der Funktion "Visual Composition Editor" von VisualAge für Java erstellt wurden, auf Visual Editor für Java in WebSphere Application Server - Express:

- Metadaten für Entwurfszeit aus VisualAge für Java sichern
- Migration (Import in WebSphere Studio) vollständig abschließen

---

### Metadaten für Entwurfszeit aus VisualAge für Java sichern

Dieser Schritt ist optional, wird jedoch aus den folgenden Gründen sehr empfohlen (insbesondere dann, wenn Ihre Anwendung über Verbindungen verfügt).

- VisualAge für Java speichert keine Informationen über die Platzierung von Beans der höchsten Ebene (das heißt, Beans, die nicht in anderen Beans enthalten sind) im freien Entwurfsbereich. Im Unterschied dazu speichert WebSphere Studio diese Informationen immer als Kommentar in der Codezeile, die die Bean deklariert, und stellt die Bean bei jedem Öffnen des Visual Editor auf dieser Position im Entwurfsbereich wieder her. Sie haben die Möglichkeit, diese Informationen der Entwurfszeit vor der Migration in VisualAge für Java zu speichern. Wenn Sie sie nicht speichern, berechnet der neue Visual Editor für Java beim ersten Öffnen der .java-Dateien für Beans der höchsten Ebene (auch als Komponenten für den freien Entwurfsbereich bezeichnet) eine Standardposition, die Sie mit Hilfe einer Drag-and-Drop-Operation auf einfache Weise ändern können.
- Obwohl der neue Visual Editor für Java in WebSphere Application Server - Express den Verbindungsentwurf nicht unterstützt, wird möglicherweise in Zukunft eine entsprechende Funktion zu diesem Editor hinzugefügt. (**Hinweis:** Dies ist jedoch keine Produktankündigung, sondern lediglich eine denkbare Möglichkeit.)

Gehen Sie wie folgt vor, um die erweiterten Metadateninformationen vor der Migration zu speichern:

1. Rufen Sie die VisualAge für Java-Entwicklerdomäne auf [<http://www.software.ibm.com/vad/data/document4293.nsf/data/document4293>], und laden Sie das Dienstprogramm *IBM VCE Code Generation and Export Utility* herunter.
2. Führen Sie die Anweisungen in der Tool-Readme-Datei aus, fügen Sie das Tool zu VisualAge für Java hinzu, stoppen Sie VisualAge für Java, und starten Sie dieses Produkt anschließend erneut.
3. Versionieren Sie den aktuellen Anwendungscode im VisualAge für Java-Repository (damit Sie diese Version im Fall einer weiteren Entwicklung von VisualAge für Java wieder verwenden können).
4. Wählen Sie für jede Ihrer grafischen Anwendungen in VisualAge für Java eines oder mehrere der grafischen Programme (normalerweise XxxView) aus, klicken Sie mit der rechten Maustaste, und führen Sie anschließend die folgenden Schritte aus:
  - a. Klicken Sie auf **VCE-Codegenerierung/-export**, und belassen Sie die Option **Nach Codeneugenerierung in Verzeichnis exportieren** ausgewählt.
  - b. Klicken Sie auf **Fertig stellen**.

- c. Belassen Sie **Verzeichnis** als ausgewähltes Ziel für den Export, und klicken Sie auf **Weiter**.
- d. Wählen Ihr Zielverzeichnis aus, wobei Sie die Option **.class** abwählen und die Option **.java** auswählen (da der Quellcode benötigt wird), und klicken Sie anschließend auf **Fertig stellen**.
- e. Laden Sie Ihren VisualAge für Java-Code mit der zugehörigen vorangegangenen Version optional erneut (klicken Sie auf die rechte Maustaste, und wählen Sie dann **Ersetzen durch > Vorherige Edition** aus).

---

## Migration (Import in WebSphere Studio) vollständig abschließen

Ihre Klassen können jetzt in WebSphere Application Server - Express importiert werden. Lesen Sie die vorhergehenden Abschnitte in Kapitel 7, „Migration von VisualAge für Java auf IBM WebSphere Studio Site Developer“, auf Seite 45. Nachdem die früheren Quellenprogramme aus Visual Composition Editor in WebSphere Application Server - Express importiert wurden, können Sie sie mit Visual Editor für Java verwalten.

---

## Kapitel 9. Erstellungskonfiguration (Bibliothek, JARs, abhängige Projekt-JARs, Ant-Erstellungen)

Dieses Kapitel behandelt die folgenden Migrationsaspekte:

- Java-Bibliotheks-JARs und externe Dritthersteller-JARs
- Optimierung der Erstellung mehrerer Projekte durch abhängige Projekt-JARs
- Automatisierte Produktionserstellungen mit Ant

---

### Java-Bibliotheks-JARs und externe Dritthersteller-JARs

Eine ausführliche Erläuterung der involvierten Elemente können Sie im Artikel *J2EE Class Loading Demystified* ([www.ibm.com/websphere/developer/library/techarticles/0112\\_deboer/deboer.html](http://www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer.html)) (J2EE-Module und Klassenpfade) sowie zur Entwicklung von J2EE-Dienstprogramm-JARs ([www.ibm.com/websphere/developer/library/techarticles/0112\\_deboer/deboer2.html](http://www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer2.html)) (Java-JARs in J2EE-Modulen) nachlesen. Diese Artikel vermitteln auf hervorragende Weise den technischen Hintergrund und liefern wertvolle Ratschläge.

### Empfohlene Methode für die Verwendung von Dritthersteller-JARs in Webprojekten

Die empfohlene Methode für die Verwendung einer Dritthersteller-JAR in einem Webprojekt ist das Importieren der JAR (unter ihrer Beibehaltung als JAR-Datei) in den Bibliotheksordner des Webprojekts. Dies ist der einzige portierbare, von J2EE definierte Weg für die Verwendung einer JAR-Datei. Er stellt sicher, dass bei der Implementierung auf einem anderen Server keine Änderungen vorgenommen werden müssen.

Mit den folgenden Schritten können Sie eine externe JAR-Datei in einem einzelnen Webprojekt verwenden. Wenn Sie die JAR-Datei in mehreren Webprojekten verwenden wollen, führen Sie stattdessen die unter „Empfohlene Methode für die Verwendung von Dritthersteller-JARs mit mehreren Webprojekten“ auf Seite 54 beschriebenen Schritte aus. .

1. Wählen Sie die Optionen **Datei** > **Importieren** > **Dateisystem** aus. Klicken Sie auf **Weiter**. Sie müssen die Option **Dateisystem** - nicht **Komprimierte Datei (ZIP)** - auswählen, damit die JAR-Datei während des Imports nicht entpackt wird.
2. Klicken Sie auf **Durchsuchen**, und suchen Sie nach dem Verzeichnis der JAR-Datei.
3. Importieren Sie die Datei in den Ordner *Webprojekt*/WebContent/WEB-INF/lib. Hierbei steht *Webprojekt* für den Namen Ihres Webprojekts.
4. Klicken Sie auf **Fertig stellen**. Die JAR-Datei wird jetzt automatisch zum Java-Erstellungspfad hinzugefügt. Während der Laufzeit müssen keine weiteren Änderungen vorgenommen werden.

## Empfohlene Methode für die Verwendung von Dritthersteller-JARs mit mehreren Webprojekten

Die empfohlene Methode für die Verwendung einer Dritthersteller-JAR in zwei oder mehreren Webprojekten ist das Importieren der JAR (unter ihrer Beibehaltung als JAR-Datei) in das Unternehmensanwendungsprojekt (EAR). Dies ist der einzige portierbare, von J2EE definierte Weg für die Verwendung einer JAR-Datei. Er stellt sicher, dass bei der Implementierung auf einem anderen Server keine Änderungen vorgenommen werden müssen.

Mit den folgenden Schritten können Sie eine externe JAR-Datei in mehreren Webprojekten verwenden. Wenn Sie die JAR-Datei nur in einem einzelnen Webprojekt verwenden wollen, führen Sie die im vorangegangenen Abschnitt beschriebenen Schritte aus.

1. Wählen Sie die Optionen **Datei > Importieren > Dateisystem** aus. Klicken Sie auf **Weiter**. Sie müssen die Option **Dateisystem - nicht Komprimierte Datei (ZIP)** - auswählen, damit die JAR-Datei während des Imports nicht entpackt wird.
2. Klicken Sie auf **Durchsuchen**, und suchen Sie nach dem Verzeichnis der JAR-Datei.
3. Importieren Sie die JAR-Datei in das EAR-Projekt, das die Webprojekte enthält.
4. Klicken Sie auf **Fertig stellen**. Die JAR-Datei wird jetzt automatisch zum Java-Erstellungspfad hinzugefügt. Während der Laufzeit müssen keine weiteren Änderungen vorgenommen werden.
5. Führen Sie die Schritte im nächsten Abschnitt aus, um die JAR-Datei zu den Modulabhängigkeiten des Web-Projekts hinzuzufügen.

## Alternative Methode für die Verwendung externer JAR-Dateien (globaler Erstellungs- und Serverklassenpfad)

Sie können die JAR-Datei auch außerhalb von WebSphere Application Server - Express belassen und sie sowohl zum Java-Erstellungspfad als auch zum Klassenpfad des Serverexemplars hinzufügen. Diese Vorgehensweise ist jedoch nicht zu empfehlen, weil das Portieren der Anwendung in diesem Fall schwierig ist. Wenn Sie auf einen anderen Server wechseln, müssen Sie immer den Klassenpfad des Servers aktualisieren. Außerdem müssen Sie sicherstellen, dass Ihre Klassendateien keine Konflikte mit anderen Versionen ähnlicher Klassendateien verursachen, die bereits im Klassenpfad des Servers vorhanden sind und durch den Server oder dessen andere Anwendungen benötigt werden. Sollten Sie sich dennoch für diese Methode entscheiden, führen Sie die folgenden Schritte aus:

1. Fügen Sie die externe JAR-Datei zum Java-Erstellungspfad des Projekts hinzu, das die JAR-Datei benötigt.
  - a. Wählen Sie das Projekt aus, klicken Sie mit der rechten Maustaste auf das Projekt, und wählen Sie im Kontextmenü die Option **Eigenschaften** aus.
  - b. Klicken Sie auf **Java-Erstellungspfad**.
  - c. Klicken Sie auf die Registerkarte **Bibliotheken**.
  - d. Klicken Sie auf **Externe JARs hinzufügen**. Wählen Sie die JAR-Datei aus, und klicken Sie auf **Öffnen**.
  - e. Klicken Sie auf **OK**.

2. Fügen Sie die externe JAR-Datei zum Klassenpfad des Serverexemplars hinzu.
  - a. Öffnen Sie die Sicht "Serverkonfiguration", und erweitern Sie den Ordner **Server**.
  - b. Wählen Sie das Serverexemplar aus, in dem das Projekt implementiert wird. Klicken Sie mit der rechten Maustaste auf das Exemplar, und klicken Sie auf **Öffnen**.
  - c. Klicken Sie auf die Registerkarte **Pfade**.
  - d. Klicken Sie in ws.ext.dirs auf **Externe JARs hinzufügen**. Wählen Sie die JAR-Datei aus, und klicken Sie auf **Öffnen**. Beachten Sie, dass ws.ext.dirs für Anwendungs-JAR-Dateien und die Variable CLASSPATH für Server-JAR-Dateien verwendet wird.
  - e. Schließen Sie das Serverexemplar, und speichern Sie die Änderungen.

---

## Erstellung mehrerer Projekte mit abhängigen Projekt-JARs optimieren

WebSphere Application Server - Express enthält eine leistungsstarke Komponente für die automatische Erstellung, die die Erstellungsleistung bei komplexen Erstellungen für mehrere Projekte herabsetzen kann. Es gibt eine Reihe von Möglichkeiten, diese automatischen Erstellungen zu steuern (abhängige Dateien, aktive und inaktive Projekte sowie Projekte im Quellen- oder JAR-Format), aber diese Optionen können relativ kompliziert werden. Der folgende Artikel erläutert diese Optionen und zeigt, wie Sie die Erstellungsleistung optimieren können. Lesen Sie den Artikel "Optimizing Multi-Project Builds Using dependent Project JARs in WebSphere Studio Application Developer" in der WebSphere-Entwicklerdomäne ([www.ibm.com/websphere/developer/library/techarticles/0204\\_searle/searle.html](http://www.ibm.com/websphere/developer/library/techarticles/0204_searle/searle.html)).

---

## Produktionserstellungen mit Ant automatisieren

Wenn Sie Ant mit WebSphere Application Server - Express verwenden, können Sie die Produktionserstellungen automatisieren. Es gibt einen mehrteiligen Artikel, der die folgenden Aspekte erläutert:

- Beschreibung von Ant
- Ausführung von Ant innerhalb und außerhalb von WebSphere Application Server - Express
- Verwendung von Ant für Produktionserstellungen von J2EE-Elementen (EJB, WAR, EAR usw.)
- Erweiterung von Ant durch neue Erstellungstasks von WebSphere Application Server - Express

Den entsprechenden Artikel Using Ant with WebSphere Studio Application Developer finden Sie in der WebSphere-Entwicklerdomäne unter [www.ibm.com/websphere/developer/library/techarticles/0203\\_searle/searle1.html](http://www.ibm.com/websphere/developer/library/techarticles/0203_searle/searle1.html).



---

## Kapitel 10. Migrationsbeispiele

In diesem Kapitel können Sie anhand von Migrationsbeispielen mehr über die Migration von VisualAge für Java und WebSphere Studio "Classic" auf WebSphere Application Server - Express IBM WebSphere Studio Site Developer erfahren.

- Beispiel für JSP/Servlet aus VisualAge für Java (LeapYear)
- Beispiel für Webanwendung aus WebSphere Studio "Classic" (YourCo)

---

### Beispiel für JSP/Servlet aus VisualAge für Java (LeapYear)

#### Beschreibung

Das Beispiel FindTheLeapYears wird mit VisualAge für Java Version 4.0 zur Verfügung gestellt. Informationen zu diesem Beispiel finden Sie in der Onlinehilfe von VisualAge für Java (**Beispiele** > **Entwicklungsumgebung für JSP/Servlet**).

#### Migrationsübersicht

Bitte führen Sie die unten angegebenen Schritte für die Migration des Beispiels aus VisualAge für Java auf IBM WebSphere Studio Site Developer aus. Diese Schritte werden später noch ausführlich erläutert:

1. Java- und Projektressourcendateien aus VisualAge für Java exportieren.
2. Neues Webprojekt für IBM WebSphere Studio Site Developer erstellen.
3. Java- und Projektressourcendateien in das Projekt von IBM WebSphere Studio Site Developer importieren.
4. Servlets definieren und Anwendungsänderungen wegen Umstrukturierung vornehmen.
5. Serverprojekt für IBM WebSphere Studio Site Developer erstellen.
6. Migrierte Anwendung testen.

#### Dateien aus VisualAge für Java exportieren

1. Öffnen Sie VisualAge für Java.
2. Wählen Sie das Projekt **IBM JSP Examples** aus.
3. Klicken Sie mit der rechten Maustaste auf das Projekt, und wählen Sie die Option **Exportieren** aus. Wählen Sie das Optionsfeld **Verzeichnis** aus, und klicken Sie auf **Weiter**.
4. Geben Sie den Namen des Verzeichnisses ein, in das die Dateien exportiert werden sollen.
5. Wählen Sie das Markierungsfeld **.class** ab. Sie müssen diese Dateien nicht exportieren, da Sie das Projekt und diese Dateien in WebSphere Application Server - Express erneut erstellen werden.
6. Wählen Sie das Markierungsfeld **.java** aus, und klicken Sie auf **Details**. Wählen Sie nur die Dateien für LeapYear aus, und klicken Sie auf **OK**.
7. Wählen Sie das Markierungsfeld **resource** aus, und klicken Sie auf **Details**.
8. Wählen Sie die Dateien **LeapYearInput.html** und **LeapYearResults.jsp** aus, die sich in dem folgenden Verzeichnis befinden: **IBM WebSphere Test Environment\hosts\default\_host\default\_app\web\JSP\sample3**.
9. Klicken Sie auf **OK**.

10. Wählen Sie das Markierungsfeld **Manifestdatei erstellen** ab (die Erstellung einer Manifestdatei ist nicht erforderlich).
11. Klicken Sie auf **Fertig stellen**.
12. Schließen Sie VisualAge für Java.

## Neues Webprojekt für IBM WebSphere Studio Site Developer erstellen

1. Starten Sie IBM WebSphere Studio Site Developer.
2. Erstellen Sie ein neues Webprojekt namens LeapYear (**Datei > Neu > Projekt > Web > Webprojekt**).
3. Stellen Sie sicher, dass **Dynamisches Webprojekt** ausgewählt ist, und klicken Sie dann auf **Weiter**.
4. Wählen Sie **NEU** aus.
5. Ändern Sie den **Namen des Unternehmensanwendungsprojekts** in LeapYearEAR, und wählen Sie **J2EE level 1.2** aus. Sie könnten das Webprojekt zwar in ein vorhandenes Unternehmensanwendungsprojekt (EAR) stellen, aber in diesem Beispiel wird es in das Projekt LeapYearEAR gestellt.
6. Belassen Sie den Eintrag LeapYear im Feld **Stammverzeichnis für Kontext**.
7. Klicken Sie auf **Fertig stellen**.

## Java- und Projektressourcendateien in das Projekt von IBM WebSphere Studio Site Developer importieren

Importieren Sie folgendermaßen die Java-Quellendateien in das Quellenverzeichnis des Projekts LeapYear:

1. Erweitern Sie in der Perspektive "Web" das Projekt "LeapYear", und wählen Sie das Java-Quellenverzeichnis **JavaSource** aus.
2. Klicken Sie auf **Datei > Importieren > Dateisystem**, und klicken Sie auf **Weiter**. Suchen Sie nach dem Verzeichnis, in das Sie die Dateien exportiert haben, und klicken Sie auf **OK**.
3. Da Sie lediglich die Java-Quellendateien in das Java-Quellenverzeichnis **JavaSource** exportieren wollen, erweitern Sie im Dialog "Importieren" das Exportverzeichnis, und wählen Sie nur das Unterverzeichnis **com** aus (dieses Verzeichnis enthält die drei Java-Quellendateien).
4. Klicken Sie auf **Fertig stellen**. Daraufhin werden die Dateien "LeapYear\JavaSource\com\ibm\ivj\wte\samples\leapyear\LeapYearXXXX.java" erstellt. Java-Klassen werden automatisch in "LeapYear\WebContent\WEB-INF\classes" kompiliert.

Importieren Sie die Ressourcendateien wie folgt in das Projekt "LeapYear" unter dem Webinhaltsverzeichnis **WebContent**:

1. Erweitern Sie in der aktuellen Perspektive "Web" das Projekt "LeapYear", und wählen Sie das Webinhaltsverzeichnis **WebContent** aus.
2. Klicken Sie auf **Datei > Importieren > Dateisystem**, und klicken Sie auf **Weiter**. Suchen Sie nach dem Verzeichnis, in das Sie die Dateien exportiert haben, erweitern Sie das Exportverzeichnis bis zum Unterverzeichnis **sample3**, und klicken Sie dann auf **OK**.
3. Da Sie nur die Ressourcendateien in das Webinhaltsverzeichnis **WebContent** importieren wollen, wählen Sie im Dialog "Importieren" das Unterverzeichnis **sample3** aus, das die .JSP- und .HTML-Dateien enthält.

4. Klicken Sie auf **Fertig stellen**. Daraufhin werden die Dateien in das Webinhaltsverzeichnis **WebContent** importiert.

## Servlets definieren und Anwendungsänderungen wegen Umstrukturierung vornehmen

1. Jetzt müssen Sie ein Servlet erstellen. Wählen Sie das Projekt LeapYear aus, und erweitern Sie es bis zur Datei web.xml (**LeapYear > WebContent > WEB-INF**). Öffnen Sie die Datei web.xml.
2. Klicken Sie unten auf der Seite auf die Registerkarte **Servlets**.
3. Klicken Sie auf **Hinzufügen**.
4. Vergewissern Sie sich, dass das Optionsfeld **Servlet** ausgewählt ist.
5. Wählen Sie die Klasse **LeapYear** aus, und klicken Sie anschließend auf **OK**.
6. Wählen Sie **URL-Zuordnung > Hinzufügen** aus, und geben Sie LeapYear ein.
7. Speichern Sie die Änderungen (**Datei > web.xml speichern**), und schließen Sie die Datei web.xml.

Nun müssen Sie alle Anwendungsänderungen vornehmen, die auf Grund der leicht geänderten Quellen-/Anwendungsstruktur erforderlich sind.

1. In der Sicht **Tasks** werden zwei Fehler angezeigt. Der erste Fehler wurde für die Datei LeapYearInput.html festgestellt, der zweite für LeapYearResults.jsp.
2. Öffnen Sie die Datei LeapYearResults.jsp. Ersetzen Sie die Angabe JSP/index.html durch LeapYearInput.html.
3. Öffnen Sie die Datei LeapYearInput.html. Ersetzen Sie die Angabe /servlet/com.ibm.ivj.wte.samples.leapyear.LeapYear durch LeapYear.
4. Speichern Sie die Änderungen, und schließen Sie die Dateien LeapYearResults.jsp und LeapYearInput.html.
5. Zur Verhinderung eines Laufzeitfehlers öffnen Sie die Datei LeapYear.java, die sich im Unterverzeichnis "JavaSource\com\ibm\ivj\wte\samples\leapyear" befindet.
6. Gehen Sie zu Zeile 118, und ändern Sie die Angabe für getRequestDispatcher von "/JSP/Sample3/LeapYearResults.jsp" in "LeapYearResults.jsp"
7. Speichern Sie die Änderungen, und schließen Sie die Datei LeapYear.java.

Jetzt wurde das Beispiel auf IBM WebSphere Studio Site Developer migriert. Sie müssen nur noch ein Serverprojekt in IBM WebSphere Studio Site Developer erstellen und das Beispiel in der WebSphere-Testumgebung testen.

## Serverprojekt für IBM WebSphere Studio Site Developer erstellen

1. Klicken Sie auf **Datei > Neu > Projekt > Server > Serverprojekt**. Klicken Sie auf **Weiter**. Geben Sie im Feld **Projektname** den Wert newServer ein, und klicken Sie auf **Fertig stellen**. Daraufhin wird automatisch in die Perspektive "Server" umgeschaltet.
2. Klicken Sie mit der rechten Maustaste auf **newServer**, und klicken Sie auf **Neu > Server und Serverkonfiguration**.
3. Geben Sie im Feld **Servername** den Wert WSTestEnv ein. Wählen Sie im Feld **Typ des Serverexemplars** den Eintrag **WebSphere V4.0 Testumgebung** aus. Klicken Sie auf **Fertig stellen**.

Jetzt müssen Sie das EAR-Projekt in der Serverkonfiguration angeben:

1. Erweitern Sie in der Sicht "Serverkonfiguration" die Serverelemente, und klicken Sie auf **WSTestEnv**.
2. Klicken Sie mit der rechten Maustaste auf den Eintrag, und klicken Sie auf die Option **Hinzufügen > LeapYearEAR**.

## Migrierte Anwendung LeapYear testen

1. Wählen Sie die Datei LeapYearInput.html aus.
2. Klicken Sie mit der rechten Maustaste auf die HTML-Datei, und klicken Sie im Kontextmenü auf **Auf Server ausführen**.
3. Warten Sie, bis der Server gestartet wurde. Achten Sie auf die Seite **Konsole** (klicken Sie in der Sicht "Server" auf die Registerkarte **Konsole**), bis die Nachricht ausgegeben wird, dass der Standardserver für das e-business geöffnet wurde.
4. Sobald ein Browser geöffnet wird, geben Sie im Feld **Start Year** den Wert 2001 ein, und klicken Sie auf **Submit**.
5. In der Sicht "Konsole" wird die Nachricht LeapYear: init angezeigt. Warten Sie, bis die Liste der Schaltjahre angezeigt wird, und wählen Sie dann in der Sicht "Server" **WSTestEnv** aus. Klicken Sie mit der rechten Maustaste auf den Eintrag, und wählen Sie die Option **Stoppen** aus.

---

## Beispiel für Webanwendung (YourCo) aus WebSphere Studio "Classic" Version 4.0 (Windows)

### Beschreibung

Bei diesem Beispiel müssen Sie WebSphere Studio "Classic" Version 4.0.x verwenden.

Sie arbeiten in diesem Fall mit dem Beispiel **YourCo**. Auf dieses Beispiel greifen Sie durch Öffnen der Onlinehilfe zu (**Hilfe > WebSphere Studio 4.0 > Vorgehensweise > Mit Beispielen arbeiten > Übersicht**). Zum Laden des Beispiels befolgen Sie die Anweisungen unter **Studio-Beispiele öffnen** (für WebSphere Application Server 4.0), und laden Sie die Datei YourCo.war.

**Anmerkung:** Die migrierte Anwendung wird später zwar in IBM WebSphere Studio Site Developer ausgeführt, aber IBM WebSphere Studio Site Developer stellt gegenwärtig nicht alle Entwurfs- und Entwicklungskomponenten für Webseiten von WebSphere Studio Professional Edition bzw. Advanced Edition zur Verfügung.

### Vorbereitungen

- Vergewissern Sie sich, dass die Beispielanwendung YourCo in WebSphere Studio "Classic" geladen wurde.
- Stoppen Sie alle Exemplare von WebSphere Application Server (damit keine Konflikte mit IBM WebSphere Studio Site Developer verursacht werden).

### Migrationsschritte

Zur Migration dieses Beispiels aus WebSphere Studio "Classic" auf IBM WebSphere Studio Site Developer müssen Sie die folgenden Schritte ausführen. Jeder Schritt wird später noch detailliert erläutert.

1. (Optional) WebSphere Studio "Classic" starten und neue Migrationsumgebung erstellen.
2. Deskriptordatei für Webkonfiguration erstellen.
3. WAR-Migrationsdatei erstellen (mit Webpfad).
4. IBM WebSphere Studio Site Developer starten und die WAR-Datei importieren.
5. Serverprojekt für IBM WebSphere Studio Site Developer erstellen.
6. Migrierte Anwendung testen.

## WebSphere Studio "Classic" Version 4.0 starten und neue Migrationsumgebung erstellen

(Optional) Normalerweise erstellen Sie bei einer Migration eine neue Umgebung. Für das vorliegende Beispiel verwenden Sie jedoch die Testumgebung, die Bestandteil von WebSphere Studio "Classic" ist. Dank der Testumgebung müssen Sie viele Servletzuordnungen in Schritt 8 nicht manuell bearbeiten.

Informationen zur Erstellung einer neuen Umgebung für die Migration finden Sie im Kapitel "Migration von WebSphere Studio "Classic" auf IBM WebSphere Studio Site Developer".

## Deskriptordatei für Webkonfiguration erstellen

1. Klicken Sie in der Sicht mit der Projektdatei auf **Projekt > Deskriptordatei für Webkonfiguration erstellen**, und übernehmen Sie den Standardwert `WEB-INF\localhost_web.xml`.
2. Wählen Sie alle erforderlichen Servlets aus (d. h. alle Dateien mit einem anderen Namen als `xxxxBean`).
3. Für dieses Beispiel sind keine TLD-Dateien (Tag Library Descriptor - Tagbibliothekdeskriptor) vorhanden.
4. Klicken Sie auf **Erstellen**.

## Migrationsdatei erstellen

1. Wählen Sie in der Sicht mit der Projektdatei den Server `localhost` aus, klicken Sie auf **Eigenschaften > Publikation > WebApp**, und geben Sie einen neuen Webpfad (Stammverzeichnis für Kontext) namens "newStudioSample" ein. (Das Festlegen eines Webpfads wird auch in der endgültigen Produktversion von IBM WebSphere Studio Site Developer die empfohlene Methode sein.)
2. Wählen Sie in der Sicht mit der Projektdatei die Optionen **Projekt > Migrationsdatei erstellen** aus.
3. Prüfen Sie, ob der Server `localhost` ausgewählt ist.
4. Prüfen Sie, ob die Datei `localhost_web.xml` als Deskriptordatei für die Webkonfiguration ausgewählt ist.
5. Klicken Sie auf **OK**.
6. Der Standardname für die JAR-Datei lautet `X:\Studio40\projects\YourCo\localhost.jar`. Hierbei steht X für das Installationsverzeichnis von WebSphere Studio "Classic".
7. Klicken Sie auf **Speichern**.
8. Schließen Sie WebSphere Studio "Classic".
9. Benennen Sie die Datei `localhost.jar` in `localhost.war` um.

## IBM WebSphere Studio Site Developer starten und die WAR-Datei importieren

1. Starten Sie IBM WebSphere Studio Site Developer.
2. Klicken Sie auf **Datei > Importieren > WAR-Datei > Weiter**.

**Anmerkung:** Sie *müssen* die JAR-Datei mit der Option "WAR-Datei" importieren, da eine einwandfreie Funktionsweise andernfalls nicht gewährleistet ist.

3. Geben Sie den Pfad für die Datei localhost.war im Feld **WAR-Datei** ein, oder klicken Sie auf **Durchsuchen**, um nach der Datei zu suchen.
4. Wählen Sie im Feld **Webprojekt** die Option **Neu** aus, und geben Sie anschließend den Wert newStudioSample ein.
5. Wählen Sie im Feld **EAR-Projektname** die Option **Neu** aus, und geben Sie anschließend den Wert newStudioSampleEAR ein.
6. Klicken Sie auf **Fertig stellen**. IBM WebSphere Studio Site Developer packt nun die Datei localhost.war aus.
7. Anschließend werden viele nicht aufgelöste Verweise bzw. fehlende Importdateien gemeldet. Diese werden in der Sicht "Tasks" angezeigt.
  - a. com.ibm.db erfordert databeans.jar,
  - b. com.ibm.webtools.runtime erfordert webtlsrn.jar,
  - c. com.ibm.ejs.ns.jndi erfordert ns.jar,
  - d. com.ibm.webshpere.advanced.cm.factory erfordert cm.jar,
  - e. com.ibm.ejs.models.base.extensions.webappext.ServletExtensions erfordert ws-base-extensions.jar

Zur Behebung dieses Fehlers müssen Sie den Java-Erstellungspfad für das Webprojekt ändern.

- a. Klicken Sie mit der rechten Maustaste auf das Projekt, und klicken Sie auf **Eigenschaften > Java-Erstellungspfad**.
  - b. Klicken Sie auf die Registerkarte **Bibliotheken**. Klicken Sie auf **Externe JARs hinzufügen**.
  - c. Importieren Sie die JAR-Dateien **databeans.jar**, **webtlsrn.jar**, **ns.jar**, **cm.jar** und **ws-base-extensions.jar** aus dem Verzeichnis MyInstall\runtimes\aes\_v4\lib.
  - d. Anschließend sind noch 24 Warnungen vorhanden. Diese Warnungen können Sie jedoch ignorieren.
8. Klicken Sie mit der rechten Maustaste auf das Projekt **newStudioSample**, und klicken Sie auf **Projekt erneut erstellen**.

Jetzt wurde das Beispiel auf IBM WebSphere Studio Site Developer migriert. Sie müssen nur noch ein Serverprojekt in IBM WebSphere Studio Site Developer erstellen und das Beispiel in der WebSphere-Testumgebung testen.

## Serverprojekt für IBM WebSphere Studio Site Developer erstellen

1. Wechseln Sie in die Perspektive "Server".
2. Klicken Sie auf **Datei > Neu > Projekt > Server > Serverprojekt**. Klicken Sie auf **Weiter**. Geben Sie im Feld **Projektname** den Wert newServer ein, und klicken Sie auf **Fertig stellen**.
3. Klicken Sie in der Sicht "Navigator" mit der rechten Maustaste auf **newServer**, und klicken Sie auf **Neu > Server und Serverkonfiguration**.

4. Geben Sie im Feld **Servername** den Wert **WSTestEnv** ein. Wählen Sie im Feld **Typ des Serverexemplars** den Eintrag **WebSphere V4.0 > Testumgebung** aus. Klicken Sie auf **Fertig stellen**.

Jetzt müssen Sie das EAR-Projekt in der Serverkonfiguration angeben:

1. Klicken Sie in der Sicht Serverkonfiguration auf **Server > WSTestEnv**.
2. Klicken Sie mit der rechten Maustaste auf den Eintrag, und klicken Sie auf die Option **Hinzufügen > newStudioSampleEAR**.

**Anmerkung:** (Optional) Klicken Sie mit der rechten Maustaste auf das Projekt **newStudioSample**, wählen Sie **Eigenschaften > Servereinstellungen > Immer auf folgendem Server ausführen** und dann **WSTestEnv** aus. Klicken Sie danach auf **Anwenden > OK**. (Dieser Schritt ist nur dann erforderlich, wenn weitere Server verwendet werden.)

## Migrierte Anwendung YourCo testen

1. Wählen Sie die Datei **YourCoIntro.html** aus, die sich im Verzeichnis **WebContent\StudioSamples** des neuen Projekts **newStudioSample** befindet.
2. Klicken Sie mit der rechten Maustaste auf die Datei **YourCoIntro.html**, klicken Sie im Kontextmenü auf **Auf Server ausführen**, und wählen Sie dann **WSTestEnv** aus.
3. Warten Sie, bis der Server gestartet wurde. Achten Sie auf die Seite **Konsole** (klicken Sie in der Sicht "Server" auf die Registerkarte **Konsole**), bis die Nachricht ausgegeben wird, dass der Standardserver für das e-business geöffnet wurde.
4. Falls Sie dieses Beispiel in WebSphere Studio "Classic" noch nie ausgeführt haben, müssen Sie die Datenbank konfigurieren. Hierzu klicken Sie auf **Datenbankkonfiguration**.
5. Sobald ein Browser geöffnet wird, blättern Sie vor, und klicken Sie auf die Option für die **Ausführung dieses Beispiels**.
6. Warten Sie, bis im Browser die Startseite **Welcome** angezeigt wird, und klicken Sie dann auf **Employee Center**.

**Anmerkung:** Wenn Sie diese Anwendung zum ersten Mal ausführen, empfangen Sie auf der Seite "Konsole" den folgenden Fehler:  
Datenquelle nicht gefunden. Versuchen Sie, einen neuen Datenquellennamen zu erstellen: jdbc/yourco  
Datenquelle nicht gefunden. Versuchen Sie, einen neuen Datenquellennamen zu erstellen: jdbc/studio.  
Diese Fehler korrigieren sich selbst. Sie können sie daher ignorieren.

7. Sobald Sie fertig sind, schließen Sie das Browserfenster und die Sicht "Web-Browser". Anschließend klicken Sie in der **Systemsteuerung des Servers** mit der rechten Maustaste auf **WSTestEnv** und dann auf **Stoppen**.
8. (Optional) Schließen Sie IBM WebSphere Studio Site Developer.



---

## Kapitel 11. Weiterführende Informationen

Aktuelle Informationen zur Migration und zu anderen Themen finden Sie unter [www.ibm.com/websphere/developer/zones/studio/transition.html](http://www.ibm.com/websphere/developer/zones/studio/transition.html).

Die folgenden Veröffentlichungen und Webseiten bieten allgemeine Informationen, die bei der Arbeit mit WebSphere Application Server - Express hilfreich sein können:

- *JSR-000053 Java Servlet 2.3 and JavaServer Pages 1.2 Specifications:*  
[java.sun.com/aboutJava/communityprocess/first/jsr053/index.html](http://java.sun.com/aboutJava/communityprocess/first/jsr053/index.html)
- IBM WebSphere Application Server Version 4 InfoCenter:  
[www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/index.html](http://www.ibm.com/software/webservers/appserv/doc/v40/aes/infocenter/index.html)
- *WebSphere Version 4 Application Development Handbook:*  
[www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246134.pdf](http://www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246134.pdf)
- *Programming J2EE APIs with WebSphere Advanced:*  
[www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246124.pdf](http://www.redbooks.ibm.com/pubs/pdfs/redbooks/sg246124.pdf)
- *WebSphere Application Server Version 3.5 auf 4.x - Migrationshinweise & Tipps:*  
[www7b.software.ibm.com/wsdd/library/techarticles/0208\\_wright/wright.html](http://www7b.software.ibm.com/wsdd/library/techarticles/0208_wright/wright.html)
- *WebSphere Studio Application Developer Service Portal:*  
[www.ibm.com/software/ad/studioappdev/support/](http://www.ibm.com/software/ad/studioappdev/support/)
- *WebSphere Studio Application Developer FAQ Frequently Asked Questions (Häufig gestellte Fragen):*  
[www.ibm.com/support/search.wss?rs=457&tc=SSBRLP&dc=D800](http://www.ibm.com/support/search.wss?rs=457&tc=SSBRLP&dc=D800)
- *WebSphere Application Server Service Portal:*  
[www.ibm.com/software/software/webservers/appserv/support.html](http://www.ibm.com/software/software/webservers/appserv/support.html)
- *WebSphere Application Server FAQ Frequently Asked Questions (Häufig gestellte Fragen):* [www.ibm.com/support/search.wss?rs=180&tc=SSEQTP&dc=D800](http://www.ibm.com/support/search.wss?rs=180&tc=SSEQTP&dc=D800)

Weitere interessante Literatur:

- Artikel zur Verwendung von Ant mit WebSphere Studio Application Developer (inklusive Erstellung/Export von J2EE-Projekten):  
[www.ibm.com/websphere/developer/library/techarticles/0203\\_searle/searle1.html](http://www.ibm.com/websphere/developer/library/techarticles/0203_searle/searle1.html)
- Artikel zur Optimierung komplexer Erstellungen in WebSphere Studio Application Developer:  
[www.ibm.com/websphere/developer/library/techarticles/0204\\_searle/searle.html](http://www.ibm.com/websphere/developer/library/techarticles/0204_searle/searle.html)
- Artikel zum Laden von J2EE-Klassen (J2EE-Module und -Klassenpfade) in WebSphere Studio Application Developer:  
[www.ibm.com/websphere/developer/library/techarticles/0112\\_deboer/deboer.html](http://www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer.html)
- Artikel zum Entwickeln von J2EE-Dienstprogramm-JARs (Java-JARs in J2EE-Modulen) in WebSphere Studio Application Developer:  
[www.ibm.com/websphere/developer/library/techarticles/0112\\_deboer/deboer2.html](http://www.ibm.com/websphere/developer/library/techarticles/0112_deboer/deboer2.html)
- Artikel zur Teamunterstützung in WebSphere Studio Application Developer:  
[www.ibm.com/websphere/developer/library/techarticles/0108\\_karasiuk/0108\\_karasiuk.html](http://www.ibm.com/websphere/developer/library/techarticles/0108_karasiuk/0108_karasiuk.html)
- Artikel zur Migration von Enterprise Access Builder-Komponenten von VisualAge für Java auf WebSphere Studio Application Developer:  
[www.ibm.com/websphere/developer/techjournal/0201\\_minocha/minocha.html](http://www.ibm.com/websphere/developer/techjournal/0201_minocha/minocha.html)

- Artikel zum EJB-Anwendungsentwurf unter Verwendung von Session Facade für die Kommunikation mit CMPs:  
[www.ibm.com/websphere/developer/library/techarticles/0106\\_brown/sessionfacades.html](http://www.ibm.com/websphere/developer/library/techarticles/0106_brown/sessionfacades.html)
- Artikel zu den empfohlenen Methoden für WebSphere Application Server:  
[www.ibm.com/software/webservers/appserv/ws\\_bestpractices.pdf](http://www.ibm.com/software/webservers/appserv/ws_bestpractices.pdf)
- Artikel im Bereich der empfohlenen Methoden für WebSphere:  
[www.ibm.com/websphere/developer/zones/bp/](http://www.ibm.com/websphere/developer/zones/bp/)
- Hauptwebseite der WebSphere-Entwicklerdomäne:  
[www.ibm.com/websphere/developer](http://www.ibm.com/websphere/developer)
- Technische Artikel der WebSphere-Entwicklerdomäne:  
[www.ibm.com/websphere/developer/techjournal/](http://www.ibm.com/websphere/developer/techjournal/)
- Informationen zur WebSphere Studio-Produktfamilie sowie den Komponenten und Zielsetzungen von WebSphere Studio Application Developer:  
[www.ibm.com/websphere/developer/zones/studio/transition.html](http://www.ibm.com/websphere/developer/zones/studio/transition.html)
- Alle Informationen zur IBM WebSphere Studio-Produktfamilie von Entwicklungstools:  
[www.ibm.com/websphere/developer/library/techarticles/0108\\_studio/studio\\_beta.html](http://www.ibm.com/websphere/developer/library/techarticles/0108_studio/studio_beta.html)
- Externe Newsgroup für Anwendungsentwickler:  
[news://news.software.ibm.com/ibm.software.websphere.studio.application-site-developer](mailto:news://news.software.ibm.com/ibm.software.websphere.studio.application-site-developer)
- Externe Newsgroup für die Eclipse-Workbench:  
[news://news.software.ibm.com/ibm.software.websphere.studio.workbench](mailto:news://news.software.ibm.com/ibm.software.websphere.studio.workbench)
- Externe Newsgroup für WebSphere Application Server:  
[news://news.software.ibm.com/ibm.software.websphere.application-server](mailto:news://news.software.ibm.com/ibm.software.websphere.application-server)
- Artikel zur Implementierung einer J2EE-Anwendung aus WebSphere Studio Application Developer in WebSphere Application Server:  
[www.software.ibm.com/vad.nsf/Data/Document3584](http://www.software.ibm.com/vad.nsf/Data/Document3584)
- Lieferanten von Application Developer Software Configuration Management (Source Code Management):  
[www.ibm.com/software/ad/studioappdev/partners/scm.html](http://www.ibm.com/software/ad/studioappdev/partners/scm.html)
- Migration von Anwendungen auf Application Developer von Entwicklungstools anderer Hersteller:  
[www.ibm.com/websphere/developer/zones/studio/migration.html](http://www.ibm.com/websphere/developer/zones/studio/migration.html)
- Migration von VisualCafé WebLogic-Anwendungen auf Application Developer (Implementierung noch immer in WebLogic):  
[www.ibm.com/websphere/developer/library/techarticles/0209\\_searle/searle1.html](http://www.ibm.com/websphere/developer/library/techarticles/0209_searle/searle1.html)
- Eclipse.org: [www.eclipse.org](http://www.eclipse.org)
- Plug-in-Zentrale der WebSphere-Entwicklerdomäne:  
[www.ibm.com/websphere/developer/downloads/plugin/](http://www.ibm.com/websphere/developer/downloads/plugin/)
- Plug-ins für die Eclipse-Workbench (nicht auf Eclipse.org enthalten): [www.eclipse-workbench.com/jsp/plugins.jsp](http://www.eclipse-workbench.com/jsp/plugins.jsp)
- Eclipse-Plug-ins (nicht auf Eclipse.org enthalten): [www.eclipse-plugins.2y.net/eclipse/plugins.jsp](http://www.eclipse-plugins.2y.net/eclipse/plugins.jsp)

---

## Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. Möglicherweise bietet IBM die in diesem Dokument beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Dienstleistungen von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Dienstleistungen können auch andere ihnen äquivalente Produkte, Programme oder Dienstleistungen verwendet werden, solange diese keine gewerblichen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremdservices liegt beim Kunden.

Für in diesem Dokument beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Dokuments ist keine Lizenzierung dieser Patente verbunden. Lizenzanfragen sind schriftlich und auf Englisch an folgende Adresse zu richten:

*IBM Europe  
Director of Licensing  
92066 Paris La Defense  
Cedex,  
France*

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen bzw. neuen Editionen der Veröffentlichung bekannt gegeben. IBM kann jederzeit ohne vorherige Ankündigung Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

*Lab Director  
IBM Canada Ltd. Laboratory  
8200 Warden Avenue  
Markham, Ontario, Canada L6G 1C7*

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der IBM Kundenvereinbarung, in der internationalen IBM Vereinbarung für Programmlizenzen oder einer äquivalenten Vereinbarung.

Informationen über Produkte anderer Hersteller als IBM wurden von den Herstellern dieser Produkte zur Verfügung gestellt bzw. aus von ihnen veröffentlichten Ankündigungen oder anderen öffentlich zugänglichen Quellen entnommen. IBM hat diese Produkte nicht getestet und übernimmt im Hinblick auf Produkte anderer Hersteller keine Verantwortung für einwandfreie Funktion, Kompatibilität oder andere Ansprüche. Fragen hinsichtlich des Leistungsspektrums von Produkten anderer Hersteller als IBM sind an den jeweiligen Hersteller des Produkts zu richten.

Verweise in dieser Veröffentlichung auf Websites anderer Anbieter dienen lediglich als Benutzerinformationen und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des täglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden, Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

#### COPYRIGHT-LIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispielprogramme wurden nicht unter allen denkbaren Bedingungen getestet. Die in diesem Handbuch aufgeführten Beispiele sollen lediglich der Veranschaulichung und zu keinem anderen Zweck dienen. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der IBM Anwendungsprogrammierschnittstelle konform sind, für die diese Beispielprogramme geschrieben werden.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

(C) (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corp. abgeleitet. (C) Copyright IBM Corp. 2000, 2003. All rights reserved. Copyright IBM Deutschland 2000, 2003. Alle Rechte vorbehalten.

---

## Informationen zu Programmierschnittstellen

Diese Informationen sollen Ihnen bei der Erstellung von Programmen unter Verwendung dieses Produkts helfen.

Allgemeine Programmierschnittstellen ermöglichen es Ihnen, Programme zu schreiben, die die Services der Tools dieses Programms verwenden.

Diese Informationen dokumentieren möglicherweise auch Informationen zur Diagnose, Modifizierung und Anpassung. Solche Informationen sollen Ihnen beim Debugging Ihrer Programme helfen.

**Achtung:** Verwenden Sie die Informationen zur Diagnose, Modifizierung und Anpassung nicht als Programmierschnittstelle, da sie ständig Änderungen unterliegen.

---

## Marken und Dienstleistungsmarken

Die folgenden Namen sind in gewissen Ländern Marken oder eingetragene Marken der International Business Machines Corporation:

- AIX
- CICS
- Cloudscape
- DB2
- DB2 Extenders
- DB2 Universal Database
- e-business
- IBM
- iSeries
- OS/390
- S/390
- VisualAge
- WebSphere
- z/OS

Java und alle Java-basierten Marken und Logos sind in gewissen Ländern Marken oder eingetragene Marken von Sun Microsystems, Inc.

ActiveX, Microsoft, Windows, Windows NT und das Windows-Logo sind in gewissen Ländern Marken oder eingetragene Marken der Microsoft Corporation.

UNIX ist eine eingetragene Marke von The Open Group.

Andere Unternehmens-, Produkt- und Dienstleistungsamen, die durch doppelte Sterne (\*\*) gekennzeichnet sind, können Marken oder Dienstleistungsmarken anderer Unternehmen sein.