

Peregrine

Get-Services 4.0.1

Installationshandbuch

Für Windows, Solaris, AIX und Linux

Copyright © 2003 Peregrine Systems, Inc. oder Niederlassungen von Peregrine Systems, Inc. Alle Rechte vorbehalten.

Die Angaben in diesem Handbuch sind Eigentum von Peregrine Systems, Inc. und dürfen nur mit schriftlicher Genehmigung von Peregrine Systems, Inc. verwendet bzw. offengelegt werden. Dieses Handbuch darf weder ganz noch teilweise ohne vorherige schriftliche Genehmigung von Peregrine Systems, Inc. reproduziert werden. In diesem Handbuch werden eine Anzahl von Produkten mit ihren Handelsbezeichnungen aufgeführt. In den meisten, wenn nicht allen, Fällen werden diese Bezeichnungen als Warenzeichen oder eingetragene Warenzeichen von den jeweiligen Unternehmen beansprucht.

Peregrine Systems® und ServiceCenter® sind eingetragene Warenzeichen und Get-Services™ ist ein Warenzeichen von Peregrine Systems, Inc. oder von Niederlassungen von Peregrine Systems, Inc.

Dieses Produkt umfasst Software, die von der Apache Software Foundation (<http://www.apache.org/>) und von Advantys (<http://www.advantys.com>) entwickelt wurde. Darüber hinaus umfasst dieses Produkt auch Software, die von den folgenden Unternehmen oder Einzelpersonen entwickelt wurden: Sun Microsystems, Inc., Jean-Marc Lugin, Netscape Communications Corporation und Original Reusable Objects, Inc.

Dieses Handbuch und die hierin beschriebene Software werden im Rahmen einer Lizenzvereinbarung oder einer Nichtoffenlegungsvereinbarung zur Verfügung gestellt und dürfen ausschließlich gemäß der Bedingungen dieser Vereinbarungen verwendet bzw. vervielfältigt werden. Die Informationen in diesem Dokument können ohne vorherige Ankündigung geändert werden und stellen keine Verpflichtung seitens Peregrine Systems, Inc. dar. Wenden Sie sich bitte an Peregrine Systems, um anhand des Datums sicherzustellen, dass Sie über die neueste Version dieses Dokuments verfügen.

Die in der Beispieldatenbank und in Beispielen in diesem Handbuch verwendeten Bezeichnungen von Unternehmen und Einzelpersonen sind frei erfunden und dienen zur Veranschaulichung der Verwendungen der Software. Etwaige Ähnlichkeiten mit derzeit oder früher tatsächlich existierenden Unternehmen oder Personen wären rein zufällig.

Wenn Sie technische Unterstützung benötigen oder die Benutzerdokumentation für ein Produkt anfordern möchten, für das Sie über eine Lizenz verfügen, senden Sie eine E-Mail an den Kundendienst von Peregrine Systems, Inc. Die E-Mailadresse lautet support@peregrine.com.

Wenn Sie Kommentare oder Vorschläge zu dieser Benutzerdokumentation haben, wenden Sie sich bitte unter doc_comments@peregrine.com an den Kundendienst von Peregrine Systems, Inc.

Diese Ausgabe gilt für Version 4.0.1 des lizenzierten Programms.

Peregrine Systems, Inc.
Weltweite Unternehmenszentrale
3611 Valley Centre Drive, San Diego, CA 92130
Tel +1 (0) 800 638 5231 oder 858 481 5000
Fax +1 (0) 858 481 1751
www.peregrine.com



Inhalt

	Über dieses Handbuch	7
	Zielgruppe des Handbuchs	7
	Zugehörige Dokumentation	8
	Integrierbare Anwendungen	8
	Terminologie.	8
	Typografische Regeln	9
	Hinweise.	10
	Aufbau des Handbuchs	10
	Kontaktaufnahme mit dem Kundendienst	11
	Peregrine CenterPoint-Website	11
	Dokumentations-Website	12
Kapitel 1	Übersicht über die Get-Services-Installation	13
	Installationsanforderungen	14
	Installationsarten	15
	Backend-Systeme	16
Kapitel 2	Installation unter Windows	17
	Auswählen einer Installationsumgebung	18
	Entwicklungsumgebung	19
	Produktionsumgebung	20
	Get-Services-Migration von früheren Versionen	21
	Neuerstellen von Anpassungen in Get-Services 4.0	21
	Konfigurieren einer vorhandenen Backend-Datenbank für Get-Services 4.0	24
	Konfigurieren alternativer Anwendungsserver	24

Bestehende Tomcat- und Apache-Server	25
Tomcat 4.1.12 mit einer Verbindung zu IIS 5.0	27
WebSphere 4.0.2	35
Installieren von WebSphere Portal Server.	41
Empfohlene Konfiguration für WebSphere Portal Server	41
Alternative Konfiguration für WebSphere Portal Server.	44
Konfigurieren des WebSphere-Übersetzungsservers für Get-Services	52
WebLogic 6.1 SP3 oder SP4	56
JRun 3.1.	62
Installationsoption „Typisch“	72
Komponenten der typischen Installation	72
Verfahren für die typische Installation	73
Installationsoption „Benutzerdefiniert“	78
Komponenten einer benutzerdefinierten Installation	78
Verfahren für die benutzerdefinierte Installation.	79
Deinstallieren von Get-Services	87
Testen der Installation	88
Kapitel 3 Installation unter AIX, Linux oder Solaris	89
Auswählen einer Installationsumgebung	90
Entwicklungsumgebung	91
Produktionsumgebung	92
Get-Services-Migration von früheren Versionen	93
Neuerstellen von Anpassungen in Get-Services 4.0	93
Konfigurieren einer vorhandenen Backend-Datenbank für Get-Services 4.0	96
Konfigurieren alternativer Anwendungsserver	96
Tomcat 4.1.12	97
WebSphere 4.0.2	99
Installieren von WebSphere Portal Server.	105
Empfohlene Konfiguration für WebSphere Portal Server	105
Alternative Konfiguration für WebSphere Portal Server.	107
Konfigurieren des WebSphere-Übersetzungsservers für Get-Services	115
WebLogic 6.1 SP3 oder SP4	119
JRun 3.1.	121

Installationsoption „Typisch“	131
Komponenten der typischen Installation	132
Verfahren für die typische Installation	133
Installationsoption „Benutzerdefiniert“	138
Komponenten einer benutzerdefinierten Installation	139
Verfahren für die benutzerdefinierte Installation.	140
Deinstallation – AIX, Linux oder Solaris	151
Kapitel 4 Lastenausgleich	153
Lastenausgleich für Anwendungsserver	154
Erstellen mehrerer Tomcat-Instanzen für Apache	156
Kopieren des Tomcat-Verzeichnisses.	157
Bearbeiten der Datei „workers.properties“	158
Bearbeiten der Datei „mod_jk.conf-auto“	160
Bearbeiten der Datei „httpd.conf“	161
Bearbeiten der Dateien „server.xml“ für Apache	161
Bearbeiten der Dateien „jk2.properties“ für Apache	163
Installieren von Tomcat-Instanzen als Dienste für Apache.	163
Testen des Lastenausgleichs für Apache.	164
Erstellen mehrerer Tomcat-Instanzen für IIS	165
Kopieren des Tomcat-Verzeichnisses.	166
Konfigurieren des ISAPI-Plug-Ins für IIS	167
Erstellen und Konfigurieren eines virtuellen Verzeichnisses „jakarta“ in IIS	168
Konfigurieren von „isapi_redirector2.dll“ als ISAPI-Filter	168
Erstellen und Konfigurieren eines virtuellen Verzeichnisses „oaa“ in IIS	169
Bearbeiten der Datei „workers2.properties“ für IIS	170
Bearbeiten der Dateien „server.xml“ für IIS	171
Bearbeiten der Dateien „jk2.properties“ für IIS	173
Installieren von Tomcat-Instanzen als Dienste für IIS.	174
Testen des Lastenausgleichs für IIS	174
Kapitel 5 Verwalten von ServiceCenter	177
Konfigurieren von ServiceCenter für Get-Services	178
Aktualisieren von ServiceCenter	178
Anwenden von Entladedateien auf ServiceCenter.	179

	ServiceCenter 4.x-Entladefdateien mit Get-Services 4.0	180
	ServiceCenter 5.0-Entladefdateien mit Get-Services 4.0	181
	ServiceCenter 5.0-Entladefdateien mit Get-Services 4.0 Change Management	182
Kapitel 6	Konfigurieren des Adapters	185
	Zugreifen auf das Peregrine Portal-Verwaltungsmodul	186
	Aktivitätenmenü	188
	Verwenden der Systemsteuerung	188
	Verwenden der Seite Einstellungen	189
	Konfigurieren von Verbindungen zum Backend-System.	190
	Einstellen des ServiceCenter-Adapters	190
	Einstellen des Portal-DB-Adapters.	193
	Einstellen des Datenbankadapters für die Webanwendung	194
	Einstellen der Service Desk-Parameter	195
	Incident Management	198
Kapitel 7	Fehlerbehebung	201
	Beheben von Fehlern beim Apache-Webserver für Windows.	203
	Der Webserver antwortet nicht	203
	Benutzer können nicht auf den Webserver zugreifen, obwohl er ausgeführt wird und die Netzwerk- und Internetverbindungen aktiviert sind	204
	Beheben von Fehlern beim Apache-Webserver für Unix.	206
	Der Webserver antwortet nicht	206
	Überprüfen der Apache-Protokolldateien auf weitere Fehler.	207
	Beheben von Fehlern beim Tomcat-Dienst	207
	Überprüfen von Tomcat-Anschlusskonflikten.	208
	Ermitteln von Tomcat-Fehlern	208
	Beheben von OAA-Fehlern	210
	OAA-Backend-Konfiguration.	210
	OAA-Protokolldateien.	211
	Beheben von Fehlern beim ServiceCenter-Server.	211
	Überprüfen des Autorisierungscode und der Anschlusseinstellung von ServiceCenter.	212
	Überprüfen des ServiceCenter-Protokolls.	213
	Index	215

Über dieses Handbuch

Get-Services ist eine Anwendung, die eine webbasierte Schnittstelle zu Peregrine ServiceCenter® bereitstellt, über die Benutzer Probleme, die innerhalb ihrer Arbeitsumgebung auftreten, melden können, indem Sie Problem-Tickets im entsprechenden Backend-System öffnen.

In diesem Handbuch finden Sie Schritt-für-Schritt-Anweisungen zur Installation von Get-Services. Mit Hilfe dieses Handbuchs können Sie folgende Aufgaben durchführen:

- Installieren der Peregrine OAA-Plattform und Installieren von Get-Services.
- Konfigurieren von Get-Services für ServiceCenter.

Zielgruppe des Handbuchs

Dieses Handbuch richtet sich an Get-Services-Verwalter, die die Anwendung konfigurieren und warten. Um dieses Handbuch effektiv verwenden zu können, müssen Sie über Kenntnisse auf den folgenden Gebieten verfügen:

- XML und ECMAScript (oder JScript/JavaScript)
- Betriebs- und Referenzhandbücher sowie sonstiges Dokumentationsmaterial für die eingesetzte PC-Hardware und das Betriebssystem
- Verwaltung und Funktionen von ServiceCenter

Zugehörige Dokumentation

In den folgenden Dokumentationsmaterialien finden Sie zusätzliche Informationen:

- *Get-Services-Verwaltungshandbuch* beschreibt die Peregrine OAA-Plattform und die Verwaltung von Get-Services.
- *Get-Services-Versionshinweise*: Hier finden Sie aktuelle Informationen und Hinweise zu bekannten Problemen für Get-Services. Die Versionshinweise werden kontinuierlich aktualisiert und auf der Kundendienst-Website zur Verfügung gestellt. Informationen zum Zugriff auf die Kundendienst-Website finden Sie unter *Kontaktaufnahme mit dem Kundendienst* auf Seite 11.

Integrierbare Anwendungen

Dieses Handbuch enthält keine Informationen zu Produkten, die zusammen mit Get-Services verwendet werden können, z. B. Peregrine OAA, ServiceCenter oder Password Management. Informationen zur Installation, Konfiguration und Verwendung dieser Anwendungen finden Sie in der jeweiligen Produktdokumentation.

Hinweis: ServiceCenter muss vor der Installation und Konfiguration von Get-Services installiert und konfiguriert werden. Peregrine OAA wird mit Get-Services installiert und in diesem Handbuch wird lediglich die Installation von Peregrine OAA für Get-Services beschrieben.

Terminologie

Die in diesem Handbuch und auf der Get-Services-Oberfläche verwendete Terminologie basiert auf ServiceCenter 4.x und 5.x.

Typografische Regeln

In diesem Handbuch werden typografische Konventionen verwendet, um bestimmte Begriffe und Aktionen hervorzuheben. Diese Konventionen und ihre Bedeutung werden im Folgenden beschrieben:

Regel	Bedeutung
Fett	Daten, die genau so eingegeben werden müssen, wie sie angezeigt werden, sind fett formatiert. Die Namen von Schaltflächen, Menüs, Menüoptionen, Dateien und Ordnern sind ebenfalls fett formatiert.
<i>Kursiv</i>	Variablen und Werte, die Sie angeben müssen, sind <i>kursiv</i> formatiert. Neue Begriffe sind ebenfalls <i>kursiv</i> formatiert.
Festbreitenschriftart	Code- oder Skriptbeispiele, Systemausgaben und -meldungen werden in einer Festbreitenschriftart dargestellt. <pre>var msgTicket = new Message("Problem"); ... msgTicket.set("_event", "epmc");</pre> <p>Eine Klammer (...) wird verwendet, um anzuzeigen, dass Teile eines Skripts ausgelassen wurden, da sie für das aktuelle Thema nicht relevant sind. Bei Codebeispielen wird nicht der gesamte Dateiinhalt wiedergegeben, sondern nur der für das betreffende Thema relevante Teil des Codes.</p>
Serifenlose Schriftart	Dateinamen werden in einer serifenlosen Schriftart dargestellt (z. B. <code>login.asp</code>).

Hinweise

Um Ihnen das Auffinden von Informationen zu erleichtern, werden in diesem Handbuch spezielle Elemente verwendet. Diese Elemente und ihre Bedeutung sind in der folgenden Tabelle aufgeführt:

Hinweis	Benutzung
Wichtig:	Informationen, die benötigt werden, um eine Aufgabe durchzuführen
Hinweis:	Informationen, die für die Zielgruppe von allgemeinem Interesse sind
Hinweis:	Informationen, mit denen eine Aufgabe besonders problemlos oder schnell durchgeführt werden kann
Warnung:	Informationen zu einem möglichen Datenverlust

Aufbau des Handbuchs

Die folgende Tabelle bietet eine Übersicht über die in diesem Handbuch bereitgestellten Informationen.

Abschnitt	Enthaltene Informationen
<i>Kapitel 1, Übersicht über die Get-Services-Installation</i>	Installationsanforderungen, Installationsarten und Backend-Datenbanken.
<i>Kapitel 2, Installation unter Windows</i>	Installieren und Konfigurieren von Anwendungs- und Webservern unter einem Windows-Betriebssystem.
<i>Kapitel 3, Installation unter AIX, Linux oder Solaris</i>	Installieren und Konfigurieren von Anwendungs- und Webservern unter einem Unix-Betriebssystem.
<i>Kapitel 4, Lastenausgleich</i>	Erstellen und Konfigurieren mehrerer Serverinstanzen.
<i>Kapitel 5, Verwalten von ServiceCenter</i>	Heraufladen von Dateien und Konfigurieren von ServiceCenter für Get-Services.

Abschnitt	Enthaltene Informationen
<i>Kapitel 6, Konfigurieren des Adapters</i>	Konfigurieren des Get-Services-Verwaltungsmoduls für ServiceCenter.
<i>Kapitel 7, Fehlerbehebung</i>	Beheben von Installationsproblemen beim Apache-Webserver, Tomcat, OAA und ServiceCenter.

Kontaktaufnahme mit dem Kundendienst

Wenn Sie weitere Informationen zu dieser Version oder Unterstützung benötigen, wenden Sie sich an den Kundendienst von Peregrine Systems.

Peregrine CenterPoint-Website

Aktuelle Informationen über örtliche Kundendienstzentren stehen über Ihren Hauptansprechpartner oder auf der CenterPoint-Website von Peregrine unter der nachfolgend angegebenen Adresse zur Verfügung:

Für den Zugriff auf diese Webseite benötigen Sie Ihren gültigen Benutzernamen und Ihr Kennwort.

So setzen Sie sich mit dem Kundendienst von Peregrine in Verbindung:

- 1 Melden Sie sich mit Ihrem Benutzernamen und Ihrem Kennwort an der Website an.
- 2 Klicken Sie neben dem Bereich für die CenterPoint-Unterstützung auf **Go** (Weiter).
- 3 Wählen Sie dann unter **Contents (Inhalt)** auf der linken Seite die Option **Whom Do I Call? (An wen soll ich mich wenden?)** aus, um die Informationen zu den **Peregrine Ansprechpartnern** anzuzeigen, die weltweit Unterstützung bieten.

Dokumentations-Website

Eine vollständige Liste der verfügbaren Benutzerdokumentation finden Sie auf der CenterPoint-Website von Peregrine unter

Wichtig: Die Versionshinweise werden auch nach Veröffentlichung des Produkts kontinuierlich aktualisiert. Besuchen Sie die Kundendienst-Website von Peregrine, um sicherzustellen, dass Sie stets über die neueste Version der Versionshinweise verfügen.

1 KAPITEL Übersicht über die Get-Services-Installation

Dieses Kapitel behandelt folgende Themen für Get-Services:

- *Installationsanforderungen* auf Seite 14
- *Installationsarten* auf Seite 15
- *Backend-Systeme* auf Seite 16

Installationsanforderungen

In diesem Abschnitt wird die empfohlene Mindestkonfiguration für die ordnungsgemäße Installation und Konfiguration von Get-Services. Stellen Sie vor Beginn der Installation sicher, dass die folgenden Anforderungen erfüllt werden.

Anforderung	für Windows	für Unix
Java-Laufzeitumgebung	Java 2 SDK Standard Edition v1.3.1_05. <i>Verfügbar auf der Get-Services-Installations-CD.</i>	Java 2 SDK Standard Edition v1.3.1_05. <i>Verfügbar auf der Get-Services-Installations-CD.</i>
Anwendungsserver	Einen der folgenden Server: <ul style="list-style-type: none"> ■ Tomcat 4.1.12 <i>Verfügbar auf der Get-Services-Installations-CD.</i> ■ WebSphere Application Server (WAS) 4.0 Fix Pack 2 oder höher ■ WebLogic 6.1 SP3 oder SP4 ■ JRun 3.1 	Einen der folgenden Server: <ul style="list-style-type: none"> ■ Tomcat 4.1.12 <i>Verfügbar auf der Get-Services-Installations-CD.</i> ■ WebSphere 4.0 Fix Pack 2 oder höher ■ WebLogic 6.1 SP3 oder SP4 ■ JRun 3.1
Backend-Datenbank(en)	ServiceCenter 4.0.x oder 5.0.x	ServiceCenter 4.0.x oder 5.0.x
Betriebssysteme	<ul style="list-style-type: none"> ■ Windows 2000 Server SP2 	<ul style="list-style-type: none"> ■ AIX 5.1 ■ Red Hat Linux 7.3 ■ Solaris 2.7 oder Solaris 2.8

Anforderung	für Windows	für Unix
Webserver	Einen der folgenden Server: <ul style="list-style-type: none"> ■ Apache 2.0.43 <i>Verfügbar auf der Get-Services-Installations-CD.</i> ■ Microsoft IIS Server 5.0 ■ IBM HTTP Server 1.3.19 <i>Verfügbar auf der WebSphere-Installations-CD und auf der Support-Website von IBM</i> 	Einen der folgenden Server: <ul style="list-style-type: none"> ■ Apache 2.0.43 <i>Verfügbar auf der Get-Services-Installations-CD.</i> ■ IBM HTTP Server 1.3.19 <i>Verfügbar auf der WebSphere-Installations-CD und auf der Support-Website von IBM</i>
Systemprozessor	Pentium, 400 MHz oder schneller	Linux: Pentium, 400 MHz oder schneller AIX: POWER 3, 375 MHz oder schneller Solaris: Ultra SPARC II, 300 MHz oder schneller
RAM	Mindestens 512 MB	Mindestens 512 MB
Festplattenspeicher	100 MB für Get-Services	100 MB für Get-Services

Installationsarten

Das Get-Services-Installationsprogramm bietet zwei grundlegende Installationsarten:

- Typische Installation
- Benutzerdefinierte Installation

Bei einer *typischen Installation* wird Get-Services in einer bestimmten Konfiguration auf einem Server installiert. Typische Installationen werden zum Einrichten von Entwicklungsumgebungen eingesetzt (siehe unten).

Bei einer *benutzerdefinierten Installation* können Sie angeben, welche Komponenten auf dem jeweiligen Server installiert werden sollen. Benutzerdefinierte Installationen sind für den Einsatz alternativer Anwendungsserver und Webserver bzw. zur Einrichtung einer Produktionsumgebung vorgesehen.

Eine benutzerdefinierte Installation von Get-Services kann für zwei Arten von Umgebungen optimiert werden:

- Entwicklungsumgebung
- Produktionsumgebung

Bei einer Installation für eine *Entwicklungsumgebung* werden alle benötigten Softwarekomponenten und sämtliche Daten auf einem Server installiert bzw. gespeichert. Diese Installationsart wird verwendet, wenn bei der Implementierung von Get-Services zunächst die Funktionen der Anwendung und der Anpassungen getestet werden sollen, bevor eine Produktionsumgebung bereitgestellt wird. Standardmäßig verwendet das Get-Services-Installationsprogramm die Installationsart „Entwicklungsumgebung“.

Installationen für eine *Produktionsumgebung* sind für eine optimale Leistung und Skalierbarkeit ausgelegt. Die einzelnen Software-Funktionen (z. B. Anwendungsserver und Webserver) werden auf verschiedenen Servern installiert. Darüber hinaus können von allen Komponenten der grundlegenden Get-Services-Umgebung mehrere Instanzen eingerichtet werden. Um die Vielfalt der Installationsmöglichkeiten ausschöpfen zu können, muss eine Produktionsumgebung manuell eingerichtet werden.

Backend-Systeme

Um Get-Services einsetzen zu können, müssen Sie über ein korrekt konfiguriertes Backend-System verfügen. Get-Services verwendet ServiceCenter als Backend-Datenbank: Get-Services kann zusammen mit ServiceCenter 4.x oder ServiceCenter 5.x installiert werden. Get-Services verwendet das Backend-System für folgende Aufgaben:

- Authentifizieren von Benutzern und Festlegen der Zugriffsrechte
- Verarbeiten von Anwendungs-Workflows und Speichern von Daten
- Speichern von Personalisierungseinstellungen für die Webanwendung

Eine vollständige Liste der mit Get-Services kompatiblen Versionen finden Sie unter *Installationsanforderungen* auf Seite 14.

2 Installation unter Windows

KAPITEL

Dieses Kapitel behandelt folgende Themen:

- *Auswählen einer Installationsumgebung* auf Seite 18
- *Get-Services-Migration von früheren Versionen* auf Seite 21
- *Konfigurieren alternativer Anwendungsserver* auf Seite 24
- *Installationsoption „Typisch“* auf Seite 72
- *Installationsoption „Benutzerdefiniert“* auf Seite 78
- *Deinstallieren von Get-Services* auf Seite 87
- *Testen der Installation* auf Seite 88

Auswählen einer Installationsumgebung

Sie können Get-Services in einer der folgenden Umgebungen installieren:

- Entwicklungsumgebung
- Produktionsumgebung

In der Entwicklungsumgebung von Get-Services können Sie die Funktionen des Produkts testen und die Installation ggf. anpassen, bevor Sie das Produkt in einer Produktionsumgebung bereitstellen. In einer Entwicklungsumgebung wird die gesamte für Get-Services benötigte Software auf einem Computersystem installiert.

Sie haben zwei Auswahlmöglichkeiten für die Entwicklungsumgebung:

- Typische Installation
 - Apache 2.0-Webserver
 - Get-Services auf Tomcat 4.1.12-Anwendungsserver
- Benutzerdefinierte Installation
 - Auswahl des Webservers
 - Auswahl des Anwendungsservers für Get-Services

Die Produktionsumgebung von Get-Services ist für eine optimale Serverleistung und Skalierbarkeit ausgelegt und ermöglicht die Bereitstellung aller Anpassungen. In einer Produktionsumgebung werden die einzelnen Komponenten von Get-Services auf verschiedenen Servern installiert, um eine besonders hohe Leistung zu gewährleisten.

Sie haben zwei Auswahlmöglichkeiten für die Produktionsumgebung:

- Typische Installation
 - Apache 2.0-Webserver
 - Get-Services auf mehreren Instanzen des Tomcat 4.1.12-Anwendungsservers
- Benutzerdefinierte Installation
 - Auswahl des Webservers
 - Auswahl des Anwendungsservers für Get-Services

Entwicklungsumgebung

In den folgenden Verfahren wird die Installation von Get-Services in einer Entwicklungsumgebung beschrieben.

So installieren Sie Get-Services in einer typischen Entwicklungsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank.
- Schritt 3** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Typisch**. Siehe *Installationsoption „Typisch“* auf Seite 72.
- Schritt 4** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer.

So installieren Sie Get-Services in einer benutzerdefinierten Produktionsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank.
- Schritt 3** Installieren Sie alternative Anwendungs- und Webserver.
- Schritt 4** Konfigurieren Sie den alternativen Anwendungsserver für Get-Services. Siehe *Konfigurieren alternativer Anwendungsserver* auf Seite 24.
- Schritt 5** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 78.
- Schritt 6** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer.

Produktionsumgebung

In den folgenden Verfahren wird die Installation von Get-Services in einer Produktionsumgebung beschrieben.

So installieren Sie Get-Services in einer typischen Produktionsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank auf einem separaten Server.
- Schritt 3** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Typisch**. Siehe *Installationsoption „Typisch“* auf Seite 72.
- Schritt 4** Konfigurieren Sie mehrere Instanzen von Tomcat für den Lastenausgleich auf dem Apache-Webserver.
- Schritt 5** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer.

So installieren Sie Get-Services in einer benutzerdefinierten Produktionsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank.
- Schritt 3** Installieren Sie den alternativen Anwendungsserver und Webserver auf separaten Servern.
- Schritt 4** Konfigurieren Sie den alternativen Anwendungsserver für Get-Services. Siehe *Konfigurieren alternativer Anwendungsserver* auf Seite 24.
- Schritt 5** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 78.
- Schritt 6** Konfigurieren Sie die Anwendungs- und Webserver für den Lastenausgleich.
- Schritt 7** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer.

Get-Services-Migration von früheren Versionen

Für die Migration von früheren Versionen von Get-It oder Get-Services in Get-Services 4.0 muss ein manueller Migrationsprozess für die Daten durchgeführt werden und die von Ihnen vorgenommenen Oberflächenanpassungen müssen neu erstellt werden. In den folgenden Schritten wird der Migrationsprozess beschrieben.

Wichtig: Erstellen Sie ein Backup aller Get-Services-Daten und aller Daten des Backend-Systems, bevor Sie den Migrationsprozess durchführen.

So führen Sie eine Migration früherer Versionen in Get-Services 4.0 durch:

- Schritt 1** Überprüfen Sie die in der früheren Version vorgenommenen Anpassungen und bestimmen Sie, welche Anpassungen in Get-Services 4.0 neu erstellt werden müssen. Siehe *Neuerstellen von Anpassungen in Get-Services 4.0* auf Seite 21.
- Schritt 2** Installieren Sie Get-Services 4.0 auf einem neuen System. Siehe *Auswählen einer Installationsumgebung* auf Seite 18.
- Schritt 3** Nehmen Sie die erforderlichen Änderungen an der Backend-Datenbank vor, die in Get-Services 4.0 migriert werden soll. Siehe *Konfigurieren einer vorhandenen Backend-Datenbank für Get-Services 4.0* auf Seite 24.

Neuerstellen von Anpassungen in Get-Services 4.0

Anpassungen, die in früheren Versionen implementiert wurden, können nicht direkt in Get-Services 4.0 migriert werden. Stattdessen müssen Sie die Änderungen mit den in Get-Services 4.0 verfügbaren Funktionen und Methoden neu erstellen.

In den folgenden Abschnitten wird beschrieben, wie die in früheren Versionen implementierten Anpassungen neu erstellt werden.

Keine Anpassungen

Wenn Sie keine Anpassungen für Get-Services erstellt haben, können Sie Get-Services 4.0 problemlos auf einem neuen System installieren und die Daten aus der vorhandenen Backend-Datenbank migrieren.

Benutzerdefinierte JSP-Dateien

In früheren Versionen mussten JSP-Dateien direkt geändert werden, um bestimmte Funktionen hinzuzufügen oder zu entfernen. In der folgenden Tabelle wird beschrieben, wie häufig an JSP-Dateien vorgenommene Änderungen neu erstellt werden.

JSP-Datei, Änderung	Neues Verfahren
Entfernen der Option zur Benutzerregistrierung von der Anmeldeseite	Aktivieren oder Deaktivieren der Option zur Benutzerregistrierung über die Seite Admin-Einstellungen
Entfernen der Option zur Kennwortänderung von der Anmeldeseite	Aktivieren oder Deaktivieren der Option zur Kennwortänderung über die Seite Admin-Einstellungen

Personalisierte Seiten

Get-Services 4.0 bietet eine Vielzahl weiterer Seiten, die Sie direkt über die Webschnittstelle personalisieren können. Wenn Sie in einer früheren Version personalisierte Seiten eingerichtet haben, müssen Sie diese in Get-Services 4.0 mit DocExplorer neu erstellen.

Sie können die Personalisierung für folgende Aufgaben verwenden:

- Hinzufügen oder Entfernen von Feldern auf einer Seite
- Speichern personalisierter Suchergebnisse oder Details auf der Portalseite

Benutzerdefinierte Darstellungen, Formatvorlagen und Designs

Get-Services In 4.0 werden alle Bilder und Formatvorlagen für die Benutzeroberfläche in Designs zusammengefasst. Die Benutzer haben nicht mehr die Möglichkeit, einzelne Darstellungen und Formatvorlagen auszuwählen. Die neuen Designs umfassen Darstellungen (die wiederum Bilddateien, Rahmendefinitionen und Ebenendateien enthalten), CSS-Definitionen und XSL-Vorlagen.

Sie können benutzerdefinierte Designs aus einer früheren Version in Get-Services 4.0 kopieren, allerdings können dann aufgrund der neuen Bilder, CSS-Definitionen, Rahmendefinitionen und Ebenen Anzeigefehler auftreten. Daher sollten Sie alle benutzerdefinierten Designs auf der Basis der Get-Services 4.0-Version des klassischen Designs neu erstellen.

Alternative Anmeldeseiten und Authentifizierungsmethoden

Wenn Sie in einer früheren Version eine benutzerdefinierte Anmeldeseite oder eine alternative Authentifizierungsmethode verwendet haben, können Sie diese Anpassungen mit der aktualisierten Anweisung wieder verwenden oder neu erstellen. Informationen zu alternativen Sicherheitsmethoden finden Sie im *Get-Services-Verwaltungshandbuch*.

Mit einem früheren Tailoring Kit erstellte Anpassungen

Viele Anpassungen, für die in früheren Versionen ein Tailoring Kit benötigt wurde, können jetzt direkt über die Webschnittstelle von Get-Services erstellt werden. In der folgenden Tabelle wird beschrieben, wie bestimmte häufig mit einem Tailoring Kit vorgenommene Änderungen neu erstellt werden.

Tailoring Kit-Änderung	Neues Verfahren
Hinzufügen oder Entfernen von Feldern in einem Formular.	Hinzufügen oder Entfernen von Feldern über Personalisierung.
Hinzufügen neuer Sprachen oder Ländereinstellungen zur Get-Services-Oberfläche.	Direktes Erstellen und Bearbeiten von Dateien mit sprachspezifischen Zeichenfolgen. Sie haben auch die Möglichkeit, offiziell unterstützte Sprachpakete von Peregrine Systems zu erwerben.
Vornehmen von Änderungen am allgemeinen Paket, am Portalpaket oder am Peregrine Studio-Paket.	Diese Pakete können nicht mehr mit Tailoring-Verfahren geändert werden, die meisten Schnittstelleneinstellungen können jetzt jedoch über die Seite Admin-Einstellungen angepasst werden.
Vornehmen von Änderungen an Schemas oder serverseitigen ECMA-Skripten.	Überprüfen der neuen Funktionen und Entscheiden, ob die benutzerdefinierten Skripten und Schemas noch benötigt werden. Ist dies der Fall, müssen Sie sie in der aktuellen Version des Tailoring Kit von Get-Services neu erstellen.

Konfigurieren einer vorhandenen Backend-Datenbank für Get-Services 4.0

In der folgenden Tabelle werden die für die Datenmigration verfügbaren Optionen aufgeführt.

Migration von Get-Services 2.3 in Get-Services 4.0

Backend-Version	Erforderliche Migration
ServiceCenter 3.0	Aktualisieren auf ServiceCenter 4.x oder 5.0.x
ServiceCenter 4.x	Anwenden von Entladefdateien aus Get-Services 4.0 auf vorhandenes ServiceCenter 4.x
ServiceCenter 5.0.x	Anwenden von Entladefdateien aus Get-Services 4.0 auf vorhandenes ServiceCenter 5,0.x

Konfigurieren alternativer Anwendungsserver

Sie müssen einen Java-aktivierten Anwendungsserver zur Unterstützung der Peregrine-Webanwendungen installieren. Peregrine OAA unterstützt die folgenden alternativen Anwendungsserver:

- *Bestehende Tomcat- und Apache-Server*
- *Tomcat 4.1.12 mit einer Verbindung zu IIS 5.0*
- *WebSphere 4.0.2*
- *WebLogic 6.1 SP3 oder SP4*
- *JRun 3.1*

Wenn Sie Get-Services mit der Installationsoption **Typisch** installieren, wird Tomcat 4.1.12 installiert und mit einem Apache 2.0-Webserver verbunden. Sie können Tomcat 4.1.12 jedoch auch bei Auswahl der Installationsoption **Benutzerdefiniert** verwenden.

Wichtig: Wenn Sie einen anderen Anwendungsserver als Tomcat 4.1.12 verwenden möchten, müssen Sie die Anwendungs- und Webserver konfigurieren, *bevor* Sie das Get-Services-Installationsprogramm ausführen.

In den folgenden Abschnitten finden Sie Anweisungen zur Konfiguration alternativer Anwendungsserver für Get-Services.

Bestehende Tomcat- und Apache-Server

Wenn Sie die Installationsoption **Typisch** verwenden, konfiguriert das Get-Services-Installationsprogramm Tomcat für den Apache-Webserver. Sie können Ihre bestehenden Tomcat- und Apache-Server so konfigurieren, dass eine Verbindung zwischen ihnen hergestellt wird; das gilt auch für Server, die NICHT vom Installationsprogramm installiert werden.

So konfigurieren Sie einen bestehenden Tomcat-Server für die Verbindung mit einem Apache-Server:

- 1 Kopieren Sie die folgenden Dateien in die unten angegebenen Verzeichnisse.

Hinweis: Diese Dateien befinden sich auf dem Computer, auf dem Tomcat und Apache vom Installationsprogramm bei Auswahl der Option **Typisch** installiert werden.

Kopieren Sie die Datei von einem anderen Computer mit einer typischen Installation

- **mod_jk.conf** auf Ihren Computer in das Verzeichnis `\conf` der Tomcat-Installation. Der Standardpfad lautet:
C:\Program Files\Apache Group\Tomcat 4.1\conf
- **workers.properties** auf Ihren Computer in das Verzeichnis `\conf` der Tomcat-Installation. Der Standardpfad lautet:
C:\Program Files\Apache Group\Tomcat 4.1\conf
- **mod_jk.dll** auf Ihren Computer in das Verzeichnis `\modules` der Apache-Installation. Der Standardpfad lautet:
C:\Program Files\Apache Group\Apache2\modules

- 2 Öffnen Sie die Dateien `mod_jk.conf` und `workers.properties`, die sich im Verzeichnis `\conf` der Tomcat-Installation befinden, in einem Texteditor.

Der Standardpfad lautet:

C:\Program Files\Apache Group\Tomcat 4.1\conf

- a Bearbeiten Sie alle Vorkommen des Tomcat-Pfads so, dass diese dem aktuellen Tomcat 4.1-Installationspfad entsprechen.
- b Bearbeiten Sie alle Vorkommen des JDK-Pfads so, dass diese dem aktuellen JDK-Installationspfad entsprechen.

- 3 Öffnen Sie die Datei `http.conf`, die sich im Pfad `C:\Program Files\Apache Group\Apache2\conf` befindet, in einem Texteditor.
 - a Fügen Sie die folgenden Zeilen am Ende der Datei ein:


```
### Tomcat4.1 Connector ###
include "<your Tomcat installation>/conf/mod_jk.conf"

# PRGNREDIRECT #
RewriteEngine on

RewriteRule ^/$ http://%{SERVERNAME}:%{SERVERANSCHLUSS}/oaa/login.jsp
[R=permanent]
RewriteRule ^/oaa/$
http://%{SERVERNAME}:%{SERVERANSCHLUSS}/oaa/login.jsp [R=permanent]
```
 - b Überprüfen Sie im Abschnitt **modules**, dass sich die folgende Zeile in der Datei befindet und NICHT auskommentiert ist:


```
LoadModule rewrite_module modules/mod_rewrite.so
```
- 4 Öffnen Sie die Datei `server.xml`, die sich im Ordner `\conf` des Tomcat-Installationspfads befindet, in einem Texteditor.

Der Standardpfad lautet: `C:\Program Files\Apache Group\Tomcat 4.1\conf`.

 - a Fügen Sie die folgenden Zeilen unterhalb des Abschnitts **Services** hinzu:


```
<Connector className="org.apache.jk.tomcat4.Ajp13Connector"
port="8009" minProcessors="5" maxProcessors="75"
acceptCount="10" debug="0"/>
```
 - b Speichern Sie die Datei.
- 5 Halten Sie sowohl den Tomcat- als auch den Apache-Server an.
- 6 Installieren Sie Get-Services mit Hilfe des Installationsprogramms und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 78.
- 7 Starten Sie Tomcat.
- 8 Starten Sie Apache.
- 9 Starten Sie die Get-Services-Anwendung.

Tomcat 4.1.12 mit einer Verbindung zu IIS 5.0

Sie können den Tomcat-Anwendungsserver mit dem Get-Services-Installationsprogramm installieren. Wenn Sie die Installationsoption *Typisch* verwenden, konfiguriert das Get-Services-Installationsprogramm Tomcat für den Apache-Webserver. Zur Konfiguration von Tomcat für den IIS-Webserver müssen Sie eine benutzerdefinierte Installation durchführen und IIS anhand der folgenden Anweisungen konfigurieren.

Hinweis: Mit dem folgenden Verfahren wird Tomcat zur Verwendung einer einzelnen Java Virtual Machine (JVM) eingerichtet. Im Kapitel *Lastenausgleich* des *Installationshandbuchs* finden Sie Informationen zur Installation mehrerer JVMs.

So konfigurieren Sie Tomcat für die Verbindung mit einem IIS 5.0 Webserver:

- Schritt 1** Starten Sie das Get-Services-Installationsprogramm. Siehe *Ausführen des Installationsprogramms* auf Seite 28.
- Schritt 2** Konfigurieren Sie das ISAPI-Plug-In für IIS. Siehe *Konfigurieren des ISAPI-Plug-Ins für IIS* auf Seite 28.
- Schritt 3** Konfigurieren Sie IIS zur Verwendung von `isapi_redirector2.dll` als ISAPI-Filter. Siehe *Konfigurieren von „isapi_redirector2.dll“ als ISAPI-Filter* auf Seite 29.
- Schritt 4** Erstellen und konfigurieren Sie ein virtuelles Verzeichnis `jakarta` in IIS. Siehe *Konfigurieren eines virtuellen jakarta-Verzeichnisses in IIS* auf Seite 29.
- Schritt 5** Erstellen und konfigurieren Sie ein virtuelles Verzeichnis `oaa` in IIS. Siehe *Konfigurieren eines virtuellen oaa-Verzeichnisses in IIS* auf Seite 30.
- Schritt 6** Bearbeiten Sie die Datei `server.xml`, um Leistungseinstellungen hinzuzufügen und alternative Kommunikationsanschlüsse zu konfigurieren (optional). Siehe *Bearbeiten der Datei „server.xml“ für IIS* auf Seite 31.
- Schritt 7** Installieren Sie Tomcat mit Hilfe von `installservice.bat` als Dienst (optional). Diese Datei befindet sich im Verzeichnis `Tomcat\bin`. Siehe *Installieren von Tomcat als Dienst* auf Seite 33.
- Schritt 8** Kompilieren Sie JSP-Dateien für die Produktionsumgebung vor. Siehe *Vorkompilieren von JSP-Dateien für eine Tomcat-Produktionsumgebung* auf Seite 34.

Ausführen des Installationsprogramms

Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 78.

Konfigurieren des ISAPI-Plug-Ins für IIS

Das ISAPI-Plug-In für IIS stellt die Verbindung zwischen Tomcat und dem IIS-Webserver her. Bevor Sie IIS für die Verwendung dieses Connectors konfigurieren, müssen Sie in der Registrierungsdatei den Eintrag für den Connector aktualisieren, um sicherzustellen, dass dort die richtigen Pfade für den Tomcat-Anwendungsserver angegeben werden.

Das Get-Services-Installationsprogramm legt im folgenden Ordner automatisch eine Kopie des ISAPI-Plug-Ins für IIS ab:

```
c:\Program Files\Peregrine\Common\Tomcat4\bin
```

Führen Sie die folgenden Verfahren durch, um das Plug-In für Ihre Intranetumgebung zu konfigurieren.

So konfigurieren Sie das ISAPI-Plug-In für IIS:

- 1 Öffnen Sie die Datei `jk2.reg` in einem Texteditor. Der Standardpfad lautet:
`C:\Program Files\Peregrine\Common\Tomcat4\conf`
- 2 Stellen Sie sicher, dass in den Werten `serverRoot` und `workersFile` der richtige Installationspfad von Tomcat angegeben wird. Standardmäßig lauten diese Werte:

```
"ServerRoot"="C:\\Program Files\\Peregrine\\Common\\Tomcat4"  
"workersFile"="C:\\Program Files\\Peregrine\\Common\\Tomcat4\\conf\\  
workers2.properties"
```

Hinweis: Wenn Sie diese Datei im standardmäßigen Installationsverzeichnis installiert haben, müssen Sie keine Änderungen vornehmen.

- 3 Speichern und schließen Sie die Datei `jk2.reg`.
- 4 Doppelklicken Sie im Windows-Explorer auf die Datei `jk2.reg`.
Windows übernimmt die Einstellungen in die Windows-Registrierung.

Konfigurieren von „isapi_redirector2.dll“ als ISAPI-Filter

Um eine Verbindung zwischen Tomcat und IIS herzustellen, müssen Sie isapi_redirector2.dll als ISAPI-Filter installieren.

So installieren Sie „isapi_redirect2.dll“ als ISAPI-Filter:

- 1 Öffnen Sie in **Windows-Systemsteuerung > Verwaltung** die Managementkonsole für Internetdienste.
- 2 Klicken Sie mit der rechten Maustaste auf den Knoten **Standardwebsite** und klicken Sie dann auf **Eigenschaften**.
- 3 Klicken Sie auf das Register **ISAPI-Filter**.
- 4 Klicken Sie auf **Hinzufügen**.
- 5 Geben Sie die nachfolgend aufgeführten Daten ein:
 - a **Filtername:** jakarta. Der Filtername muss dem in der Registrierungsdatei jk2.reg festgelegten Namen entsprechen. Der standardmäßige Name des Filters ist jakarta.
 - b **Ausführbare Datei:** isapi_redirector2.dll. Der Standardpfad lautet:
C:\Program Files\Peregrine\Common\Tomcat4\bin\isapi_redirector2.dll
- 6 Klicken Sie auf **OK**.

Hinweis: Damit die Änderungen wirksam werden, müssen Sie den IIS-Dienst anhalten und neu starten. Sie müssen außerdem Peregrine Tomcat neu starten.
- 7 Klicken Sie in der Managementkonsole für Internetdienste mit der rechten Maustaste auf den Knoten **Standardwebsite** und wählen Sie dann erneut **Eigenschaften > ISAPI-Filter** aus.
Neben dem ISAPI-Filter in IIS weist ein grüner Statuspfeil darauf hin, dass dieser ausgeführt wird.
- 8 Schließen Sie die Managementkonsole für Internetdienste.

Konfigurieren eines virtuellen jakarta-Verzeichnisses in IIS

Um das ISAPI-Plug-In für IIS ausführen zu können, wird ein bestimmtes virtuelles Verzeichnis benötigt. Verwenden Sie die folgenden Richtlinien zum Erstellen eines virtuellen Verzeichnisses auf der Standardwebsite. Spezifische Anweisungen zum Konfigurieren von IIS finden Sie in der Windows-Hilfe.

So konfigurieren Sie ein virtuelles jakarta-Verzeichnis in IIS:

- 1 Verwenden Sie die folgenden Richtlinien zum Erstellen eines virtuellen Verzeichnisses auf der Standardwebsite.

Anforderungen für ein virtuelles jakarta-Verzeichnis

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	jakarta
Zuordnung zu physischem Pfad	<Tomcat>\bin
Verzeichniszugriffsrechte	Schreiben, Skriptzugriff, Ausführen

- 2 Geben Sie für <Tomcat> den Pfad zur Tomcat-Installation ein. Der Standardpfad lautet:

C:\Program Files\Peregrine\Common\Tomcat4

Konfigurieren eines virtuellen oaa-Verzeichnisses in IIS

Um Get-Services aus IIS ausführen zu können, müssen Sie ein virtuelles Verzeichnis erstellen, das eine Zuordnung zum Tomcat-Bereitstellungsordner festlegt. Spezifische Anweisungen zum Konfigurieren von IIS finden Sie in der Windows-Hilfe.

So konfigurieren Sie ein virtuelles oaa-Verzeichnisses in IIS:

- 1 Verwenden Sie die folgenden Richtlinien zum Erstellen des virtuellen Verzeichnisses.

Anforderungen für ein virtuelles oaa-Verzeichnis

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	<ooa>
Zuordnung zu physischem Pfad	<Tomcat>\webapps\<ooa>
Verzeichniszugriffsrechte	Lesen, Skriptzugriff

- 2 Geben Sie für `<aaa>` den Namen des virtuellen Verzeichnisses ein, das für Get-Services verwendet werden soll. Der empfohlene Name für das virtuelle Verzeichnis lautet `aaa`. Wenn Sie einen anderen Namen verwenden möchten, müssen Sie diesen an den folgenden Stellen eingeben:
 - Benennen Sie den Ordner `<Tomcat>\webapps\aaa` in `<Tomcat>\webapps\<Neuer Name>` um
 - Benennen Sie die Zuordnungen `[uri]` in `workers2.properties` von `aaa` in den neuen Namen des virtuellen Verzeichnisses um.
 - Benennen Sie alle `aaa`-Kontexteinträge in `mod_jk2.conf` von `aaa` in den neuen Namen des virtuellen Verzeichnisses um.
 - Benennen Sie den Pfad `<Context>` und die Attribute `docBase` in `server.xml` von `aaa` in den neuen Namen des virtuellen Verzeichnisses um.

Wichtig: Der von Ihnen gewählte Name für das virtuelle Verzeichnis wird Bestandteil des URL, den Benutzer zum Herstellen einer Verbindung mit Get-Services eingeben. Beispiel:
`http://Servername/<Neuer Name>/login.jsp`

- 3 Geben Sie für `<Tomcat>` den Pfad zur Tomcat-Installation ein. Der Standardpfad lautet:

`C:\Program Files\Peregrine\Common\Tomcat4`

Bearbeiten der Datei „server.xml“ für IIS

Eine Standardinstallation von Tomcat ist für die meisten Get-Services-Installationen ausreichend. Sollten jedoch Leistungsprobleme oder Konflikte bei den Kommunikationsanschlüssen auftreten, müssen Sie die Tomcat-Datei `server.xml` bearbeiten.

Leistungseinstellungen

In der Tomcat-Datei `server.xml` können Sie festlegen, wie Get-Services-Dateien von Tomcat verarbeitet werden. Wenn Sie Probleme hinsichtlich der Leistung haben, können Sie die Einstellung `<Context>` für Get-Services ändern, um das erneute Laden von Seiten zu unterdrücken.

Hinweis: Erstellen Sie eine Sicherungskopie der Datei `server.xml`, bevor Sie sie bearbeiten.

So bearbeiten Sie die Leistungseinstellungen der Datei „server.xml“:

- 1 Öffnen Sie die Datei `server.xml` in einem Texteditor. Der Standardpfad lautet:
`C:\Program Files\Peregrine\Common\Tomcat4\conf`
- 2 Erstellen Sie einen Eintrag für das Element `<Context>` von Tomcat zum Get-Services-Bereitstellungsverzeichnis als Referenzpunkt für das Attribut `docBase`.
Fügen Sie den Eintrag direkt oberhalb des Eintrags `examples` des Elements `Context` ein.

Beispiel:

```
<Context path="/oaa"  
docBase="<Tomcat>/webapps/oaa"  
crossContext="false"  
debug="0"  
reloadable="false" >  
</Context>
```

Wenn Sie das Attribut `reloadable` auf `false` setzen, können JSP-Seiten schneller verarbeitet werden.

Geben Sie für `<Tomcat>` im Attribut `docBase` den absoluten Pfad der ersten bzw. Hauptinstanz von Tomcat an.

Einstellungen für die Kommunikationsanschlüsse

Werden von Ihrem Get-Services-Server die Kommunikationsanschlüsse 8005 und 8009 bereits verwendet, kommt es zu einem Anschlusskonflikt, wenn Sie Tomcat mit den Standardeinstellungen installieren. Um dies zu verhindern, müssen Sie in der Datei `server.xml` die von Tomcat verwendeten Kommunikationsanschlüsse ändern.

Wichtig: Sie müssen diese Schritte nicht durchführen, wenn die Standard-Kommunikationsanschlüsse von Tomcat auf Ihrem Server noch verfügbar sind.

Hinweis: Erstellen Sie eine Sicherungskopie der Datei `server.xml`, bevor Sie sie bearbeiten.

So bearbeiten Sie die Einstellungen für die Kommunikationsanschlüsse der Datei „server.xml“:

- 1 Öffnen Sie die Datei server.xml in einem Texteditor. Der Standardpfad lautet:

C:\Program Files\Peregrine\Common\Tomcat4\conf

- 2 Weisen Sie dem Anschlussnummerattribut des Elements <Server> einen freien Kommunikationsanschluss zu.

Hinweis: Standardmäßig verwendet Tomcat Anschluss 8005 für Anforderungen zum Herunterfahren.

Beispiel:

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

- 3 Weisen Sie dem Attribut port des Coyote Connector-Elements <Connector> einen freien Kommunikationsanschluss zu.

Hinweis: Standardmäßig verwendet Tomcat 8009 für den Coyote-Connector.

Beispiel:

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector" port="8009"
minProcessors="5" maxProcessors="75" enableLookups="true" redirectPort="8443"
acceptCount="10" debug="0" connectionTimeout="20000"
useURIVValidationHack="false"
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler" />
```

- 4 Speichern Sie die Datei server.xml.
- 5 Starten Sie Tomcat, damit die neuen Einstellungen wirksam werden.

Installieren von Tomcat als Dienst

Wenn Sie die Tomcat-Dateien bearbeitet haben, können Sie Tomcat mit Hilfe von installservice.bat als Windows-Dienst installieren.

So installieren Sie Tomcat als Dienst:

- 1 Öffnen Sie eine DOS-Eingabeaufforderung und wechseln Sie zum Tomcat-Verzeichnis bin.
- 2 Geben Sie für jede zu erstellende Instanz von Tomcat den folgenden Befehl ein:

```
installservice <Dienstname> <Tomcat_Basis> <jvm_dll_Pfad>
```

Hierbei gilt: *<Dienstname>* ist der Name, den Sie dem Tomcat-Dienst zuweisen, *<Tomcat_Basis>* ist das Installationsverzeichnis der Instanz, für die Sie den Dienst erstellen, und *<jvm_dll_Pfad>* ist das Installationsverzeichnis von Java SDK.

Wenn Sie bereits die Umgebungsvariablen CATALINA_HOME und JAVA_HOME eingestellt haben, ist der zweite und dritte Parameter optional.

Beispiel:

```
installservice Tomcat8009 C:\Program Files\Peregrine\Common\Tomcat4
C:\Program Files\Peregrine\Common\jdk1.3.1_05
```

- 3 Wiederholen Sie Schritt 1 bis Schritt 2 für jeden zu erstellenden Tomcat-Dienst.

Vorkompilieren von JSP-Dateien für eine Tomcat-Produktionsumgebung

Wenn Sie Tomcat 4.1.12 in einer Produktionsumgebung oder auf einem Mehrprozessorsystem verwenden möchten, müssen Sie die für Get-Services bereitgestellten JSP-Dateien vorkompilieren. Dieser zusätzliche Konfigurationsschritt ist aufgrund eines bekannten Tomcat-Fehlers erforderlich. Unter dem folgenden URL finden Sie detaillierte Informationen zu diesem Problem:

http://nagoya.apache.org/bugzilla/show_bug.cgi?id=14077

So kompilieren Sie JSP-Dateien für Tomcat 4.1.12 vor:

- 1 Halten Sie den Tomcat-Anwendungsserver an.
- 2 Löschen Sie den Tomcat-Zwischenspeicher. Der Standardpfad lautet:
C:\Program Files\Peregrine\Common\Tomcat4\work\Standalone\localhost\oaa
- 3 Öffnen Sie eine Eingabeaufforderung.
- 4 Geben Sie den folgenden Befehl ein:

Betriebssystem	Erforderlicher Befehl
Windows	set JASPER_HOME=<Tomcat-Verzeichnis>

Geben Sie für *<Tomcat-Verzeichnis>* den absoluten Pfad zur Tomcat-Installation ein.

- 5 Wechseln Sie zum Tomcat-Ordner bin.

Betriebssystem	Erforderlicher Befehl
Windows	cd %jasper_home%\bin

- 6 Führen Sie die Batchdatei `precompile` aus.

Betriebssystem	Erforderlicher Befehl
Windows	precompile <Name der Webanwendung> <Name der Tomcat-Instanz>

Geben Sie für <Name der Webanwendung> den Namen des Get-Services-Bereitstellungsordners ein. Wenn Sie den Standardordner `oaa` verwenden, müssen Sie diesen Namen nicht angeben.

Geben Sie für <Name der Tomcat-Instanz> den Namen der installierten Instanz von Tomcat an. Wenn Sie die Standardinstanz `Standalone` verwenden, müssen Sie diesen Namen nicht angeben.

Die Batchdatei zeigt den Status der Konvertierung an. Wenn die Konvertierung abgeschlossen ist, wird wieder die Eingabeaufforderung angezeigt.

Starten Sie den Tomcat-Anwendungsserver.

WebSphere 4.0.2

Führen Sie die folgenden Verfahren durch, um WebSphere zur Ausführung von Get-Services unter Windows zu konfigurieren.

So konfigurieren Sie WebSphere 4.0.2:

- Schritt 1** Installieren Sie WebSphere 4.0.2. Ihre Version von WebSphere 4.0.2 umfasst den IBM HTTP Server.
- Schritt 2** Stellen Sie die WAR-Datei des Portals für WebSphere bereit, so dass die für Get-Services benötigte Ordnerstruktur erstellt wird. Siehe *Bereitstellen der WAR-Datei des Portals für WebSphere* auf Seite 36.
- Schritt 3** Stellen Sie die JVM Java Heap-Größe für jede Instanz von WebSphere ein, die Get-Services ausführt. Siehe *Einstellen der Java Heap-Größe* auf Seite 39.

Schritt 4 Erstellen Sie auf dem Webserver das virtuelle Verzeichnis, das für Get-Services verwendet werden soll. Siehe *Konfigurieren eines virtuellen Verzeichnisses für IBM HTTP Server* auf Seite 40.

Schritt 5 Starten Sie das Get-Services-Installationsprogramm. Siehe *Ausführen des Installationsprogramms* auf Seite 41.

Wenn Sie einen WebSphere Portal Server oder einen WebSphere Translation Server einrichten möchten, finden Sie entsprechende Informationen unter *Installieren von WebSphere Portal Server* auf Seite 41 oder *Konfigurieren des WebSphere-Übersetzungsservers für Get-Services* auf Seite 52.

Bereitstellen der WAR-Datei des Portals für WebSphere

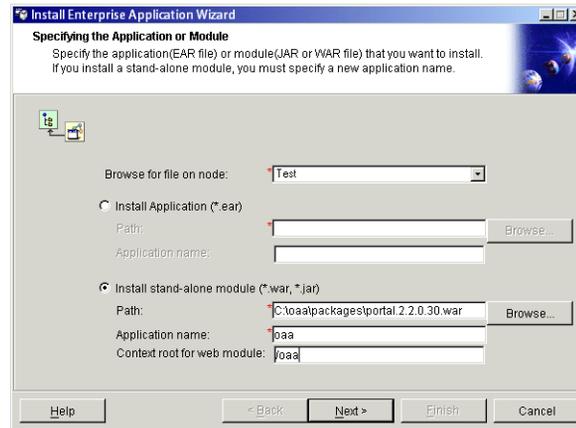
Die WAR-Datei des Portals erstellt die Ordnerstruktur, die zur Bereitstellung von Get-Services auf dem Anwendungsserver benötigt wird. Wenn Sie diese Datei für WebSphere bereitgestellt haben, können Sie das Get-Services-Installationsprogramm ausführen.

So stellen Sie die WAR-Datei des Portals für WebSphere bereit:

- 1 Stellen Sie sicher, dass der WebSphere Admin Server gestartet ist.
- 2 Öffnen Sie die Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere: **Start > Programme > IBM WebSphere > Application Server** (Anwendungsserver) > **Administrator's Console** (Verwaltungskonsole).
- 3 Klicken Sie im Menü auf der linken Seite der Konsole mit der rechten Maustaste auf **Enterprise Applications** (Unternehmensanwendungen) und wählen Sie **Install Enterprise Application** (Unternehmensanwendung installieren).
- 4 Führen Sie im angezeigten Bildschirm die folgenden Schritte durch:
 - a Wählen Sie **Install stand-alone module** (Standalone-Modul installieren).
 - b Geben Sie im Feld **Path** (Pfad) den Pfad zur Datei **Portal<Versionsnummer>.war** an. Der Standardpfad lautet: **<CD-ROM-Laufwerk>\portal<Versionsnummer>.war**.
Wählen Sie für <Versionsnummer> die neueste verfügbare Version aus (4.0.0.44 oder höher).
 - c Geben Sie im Feld **Application Name** (Anwendungsname) **oaa** ein.
 - d Geben Sie im Feld **Context Root** (Kontextstamm) den Namen des für Get-Services zu verwendenden virtuellen Webserver-Verzeichnisses an. Beispiel: **/oaa**.

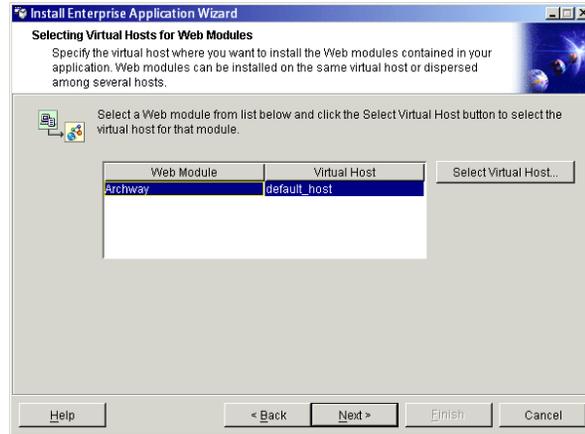
Wichtig: Sie müssen ein virtuelles Webserver-Verzeichnis erstellen, das dem unter **Context Root** (Kontextstamm) eingegebenen Verzeichnis entspricht.

Im folgenden Bildschirm wird das ausgefüllte Formular angezeigt.

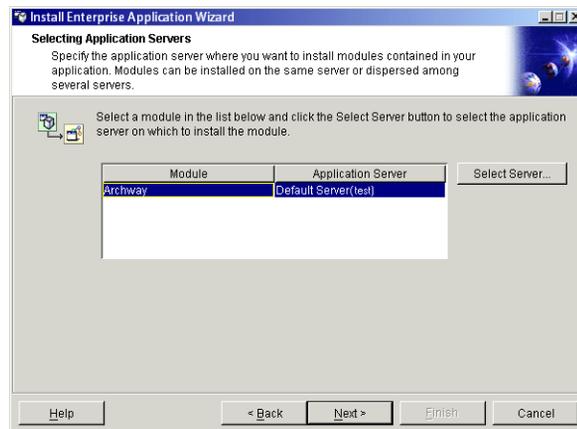


- 5 Klicken Sie auf **Weiter** (Next).
- 6 Klicken Sie in den folgenden Dialogfeldern auf **Next** (Weiter). Diese Dialogfelder werden nicht verwendet.
 - Mapping Users to Roles (Zuordnen von Benutzern zu Rollen)
 - Mapping EJB Run As Roles to Users (Zuordnen von EJB-Run-As-Rollen zu Benutzern)
 - Binding Enterprise Beans to JNDI Names (Binden von Unternehmens-Beans an JNDI-Namen)
 - Mapping EJB References to Enterprise Beans (Zuordnen von EJB-Referenzen zu Unternehmens-Beans)
 - Mapping Resource References to Resources (Zuordnen von Ressourcenreferenzen zu Referenzen)
 - Specifying the Default Datasource (Angaben der standardmäßigen Datenquelle)
 - Specifying Data Sources for Individual CMP Beans (Angaben der Datenquellen für einzelne CMP-Beans)

- Wählen Sie im Dialogfeld **Selecting Virtual Hosts for Web Modules** (Auswählen virtueller Hosts für Webmodule) die zu verwendende Serverinstanz von WebSphere aus und klicken Sie dann auf **Next** (Weiter).



- Wählen Sie im Dialogfeld **Selecting Application Servers** (Auswählen der Anwendungsserver) die zu verwendende Serverinstanz von WebSphere aus und klicken Sie dann auf **Next** (Weiter).



- Klicken Sie im angezeigten Dialogfeld auf **Finish** (Fertig stellen).

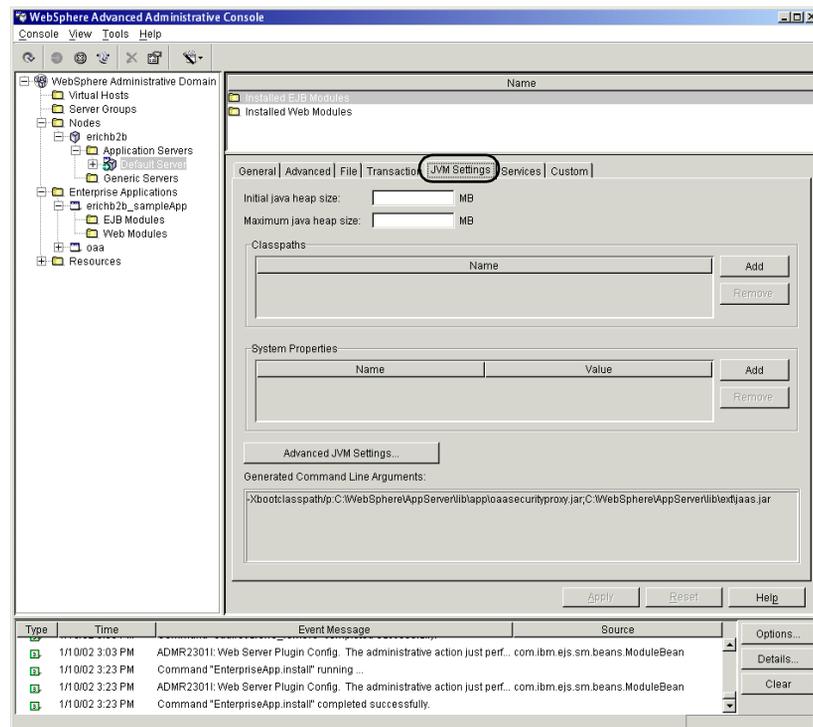
Einstellen der Java Heap-Größe

Sie können die Größe des für die Instanzen des Anwendungsservers verfügbaren Speichers konfigurieren. In den folgenden Anweisungen wird davon ausgegangen, dass Sie nur eine Instanz von WebSphere verwenden. Wenn Sie zum Lastenausgleich mit mehreren Instanzen von WebSphere arbeiten, müssen Sie die Heap-Größe entsprechend anpassen.

So stellen Sie die Java Heap-Größe ein:

- 1 Stellen Sie sicher, dass der WebSphere Admin Server gestartet ist.
- 2 Öffnen Sie die Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere: **Start > Programme > IBM WebSphere > Application Server (Anwendungsserver) > Administrator's Console (Verwaltungskonsole)**.
- 3 Klicken Sie auf **Nodes (Knoten) > <Systemname> > Application Servers (Anwendungsserver) > <Name des Anwendungsservers>**.

Die Seite für Servereinstellungen wird geöffnet.



- 4 Klicken Sie auf das Register **JVM Settings (JVM-Einstellungen)**.

- 5 Nehmen Sie die folgenden JVM-Einstellungen vor:
 - a **Initial java heap size** (Anfängliche Java Heap-Größe). Geben Sie 60 ein.
 - b **Maximum java heap size** (Maximale Java Heap-Größe). Geben Sie den gewünschten Wert für den Heap-Speicher ein. Dieser Wert sollte zwischen 225 MB und 512 MB liegen.

Hinweis: Stellen Sie sicher, dass der Höchstwert für die Heap-Größe kleiner ist als der für die Anwendungsserver verfügbare freie Arbeitsspeicher. Wenn Sie einen größeren Wert angeben als freier Arbeitsspeicher verfügbar ist, lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt. Für die meisten Systeme reicht die Einstellung von 256 MB aus.

Konfigurieren eines virtuellen Verzeichnisses für IBM

HTTP Server

Sie müssen auf dem Webserver ein virtuelles Verzeichnis für Get-Services konfigurieren. In den folgenden Anweisungen wird davon ausgegangen, dass Sie den in WebSphere integrierten Webserver verwenden, d. h. IBM HTTP Server. Wenn Sie einen anderen Webserver verwenden, finden Sie in der Dokumentation dieses Webserver Informationen zum Erstellen eines virtuellen Verzeichnisses.

So konfigurieren Sie ein virtuelles Verzeichnis für IBM HTTP Server:

- 1 Halten Sie den IBM HTTP Server an.
- 2 Öffnen Sie die Datei `httpd.conf` in einem Texteditor. Der Standardpfad lautet:

`C:\IBM HTTP Server\conf`

- 3 Fügen Sie die folgende Zeile am Ende der Datei ein:

```
Alias /oaa/ "C:/WebSphere/AppServer/installedApps/oaa.ear/portal.  
<Version>.war/"
```

Geben Sie für `<Version>` die Versionsnummer der installierten WAR-Datei ein.

Wichtig: Der hier angegebene Name des virtuellen Verzeichnisses muss dem in WebSphere unter **Context Root** (Kontextstamm) festgelegten Verzeichnis entsprechen.

- 4 Speichern Sie die Datei.
- 5 Starten Sie den IBM HTTP Server.

Ausführen des Installationsprogramms

Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 78.

Installieren von WebSphere Portal Server

Sie können Get-Services so einrichten, dass es auf einem WebSphere Portal Server in einer der folgenden zwei Konfigurationen angezeigt wird:

- Alle Komponenten von Get-Services und WebSphere werden auf einem einzigen System ausgeführt. Siehe *Empfohlene Konfiguration für WebSphere Portal Server* auf Seite 41.
- Die Komponenten von Get-Services werden auf einem System ausgeführt und die Komponenten von WebSphere auf einem anderen. Siehe *Alternative Konfiguration für WebSphere Portal Server* auf Seite 44.

Wichtig: Für beide Konfigurationen müssen Sie zunächst WebSphere Portal Server installieren. Weitere Informationen finden Sie in der Dokumentation zu WebSphere Portal Server.

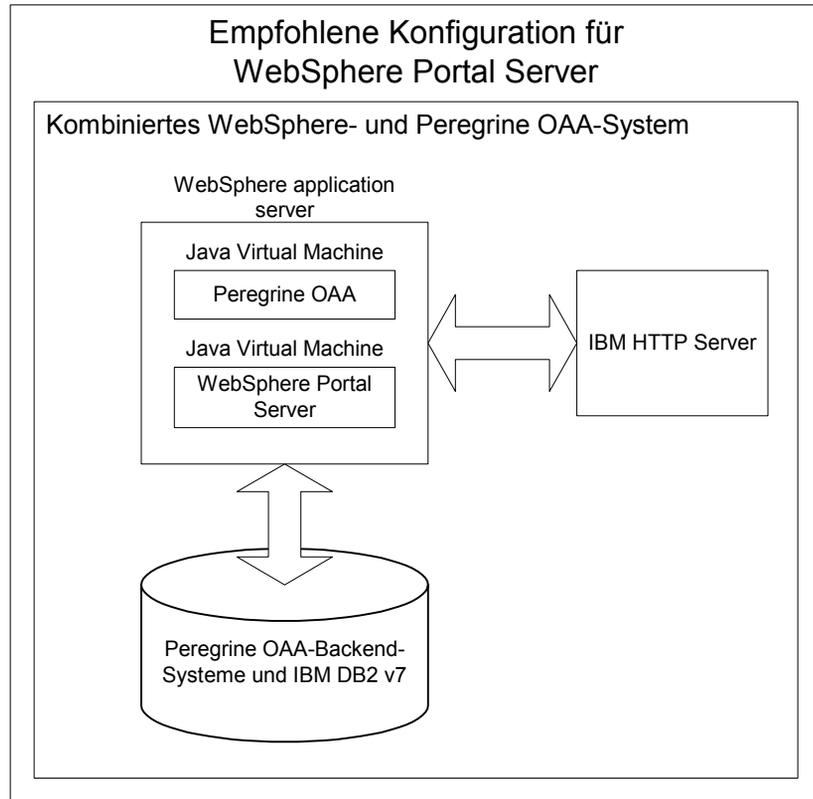
Empfohlene Konfiguration für WebSphere Portal Server

Führen Sie die folgenden Schritte durch, um Get-Services gemäß der für WebSphere Portal Server empfohlenen Konfiguration einzurichten:

- Schritt 1** Prüfen Sie die Installationsanforderungen für WebSphere Portal Server. Siehe *Installationsanforderungen für WebSphere Portal Server* auf Seite 46.
- Schritt 2** Erstellen Sie eine Get-Services-WAR-Datei mit den Portalkomponenten, die in WebSphere Portal Server angezeigt werden sollen. Siehe *Erstellen einer Get-Services-WAR-Datei* auf Seite 46.
- Schritt 3** Melden Sie sich am Get-Services-Server an und halten Sie den WebSphere-Anwendungsserver an.

- Schritt 4** Bearbeiten Sie die Datei `archway.xml`, um die HTTP-Authentifizierungsmethode von Basisauthentifizierung in alternative Authentifizierung zu ändern. Siehe *Ändern der Datei „archway.xml“* auf Seite 47.
- Schritt 5** Ändern Sie die Datei `web.xml`, um das `AuthController`-Servlet zu aktivieren. Siehe *Ändern der Datei „web.xml“* auf Seite 48.
- Schritt 6** Setzen Sie in der Datei `ibm-web-ext.xmi` den Parameter `fileServingEnabled`. Siehe *Ändern der Datei „ibm-web-ext.xmi“* auf Seite 48.
- Schritt 7** Starten Sie den WebSphere-Anwendungsserver. Siehe *Starten des WebSphere-Anwendungsservers* auf Seite 49.
- Schritt 8** Stellen Sie die Get-Services-WAR-Datei für WebSphere Portal Server bereit. Siehe *Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server* auf Seite 49.
- Schritt 9** Erstellen Sie in WebSphere Portal Server Orte und Seiten für die Anzeige von Get-Services-Portlets. Siehe *Konfigurieren von Orten und Seiten in WebSphere Portal Server* auf Seite 50.
- Schritt 10** Aktivieren Sie die Bearbeitungsrechte für Get-Services-Portlets. Siehe *Aktivieren der Bearbeitungsrechte für Get-Services-Portlets* auf Seite 50.

Wenn Sie diese Schritte durchgeführt haben, weist die Installation die folgende Konfiguration auf:



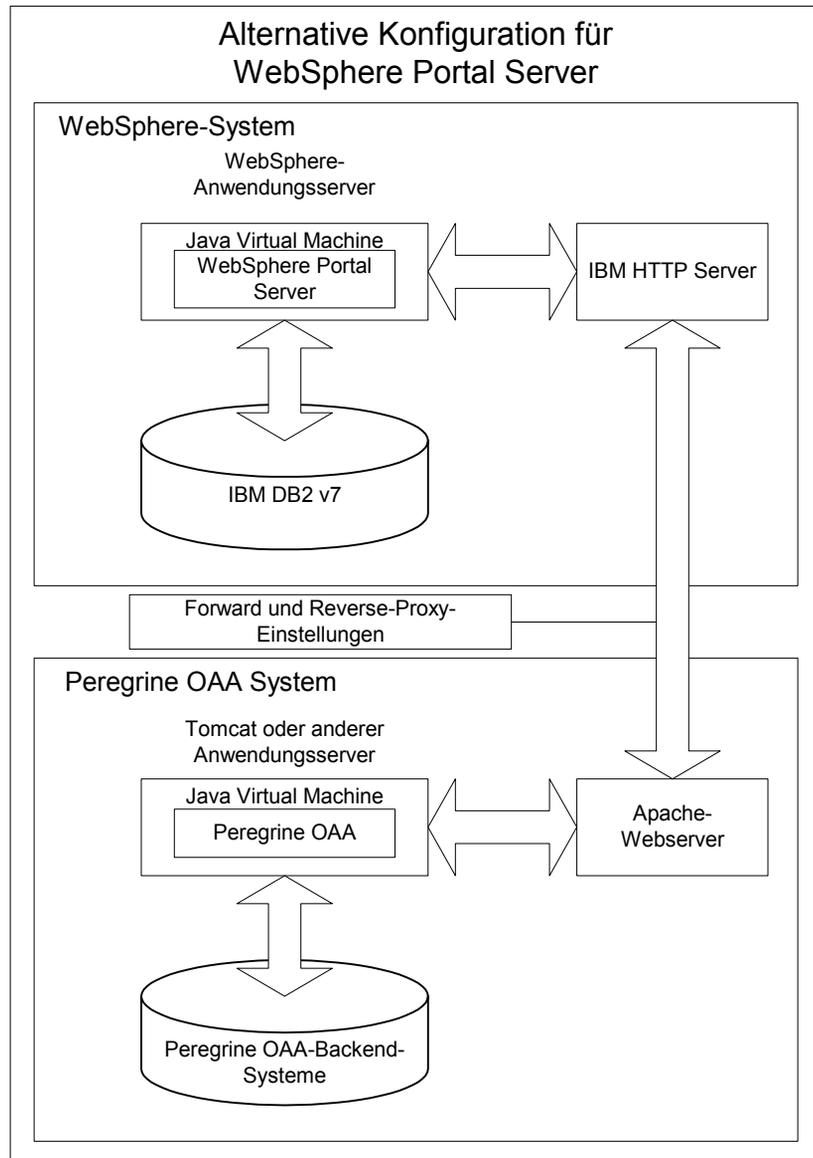
Alternative Konfiguration für WebSphere Portal Server

Führen Sie die folgenden Schritte durch, um Get-Services gemäß der alternativen Konfiguration für WebSphere Portal Server einzurichten:

- Schritt 1** Prüfen Sie die Installationsanforderungen für WebSphere Portal Server. Siehe *Installationsanforderungen für WebSphere Portal Server* auf Seite 46.
- Schritt 2** Erstellen Sie eine Get-Services-WAR-Datei mit den Portalkomponenten, die in WebSphere Portal Server angezeigt werden sollen. Siehe *Erstellen einer Get-Services-WAR-Datei* auf Seite 46.
- Schritt 3** Melden Sie sich am Get-Services-Server an und halten Sie den WebSphere-Anwendungsserver an. Siehe *Anhalten des WebSphere-Anwendungsservers* auf Seite 47.
- Schritt 4** Bearbeiten Sie die Datei `archway.xml`, um die HTTP-Authentifizierungsmethode von Basisauthentifizierung in alternative Authentifizierung zu ändern. Siehe *Ändern der Datei „archway.xml“* auf Seite 47.
- Schritt 5** Ändern Sie die Datei `web.xml`, um das AuthController-Servlet zu aktivieren. Siehe *Ändern der Datei „web.xml“* auf Seite 48.
- Schritt 6** Setzen Sie in der Datei `ibm-web-ext.xmi` den Parameter `fileServingEnabled`. Siehe *Ändern der Datei „ibm-web-ext.xmi“* auf Seite 48.
- Schritt 7** Ändern Sie `setDomain.js`, so dass die Funktion `SetDomain` aufgerufen wird. Siehe *Ändern der Datei „setDomain.js“* auf Seite 49.
- Schritt 8** Starten Sie den WebSphere-Anwendungsserver. Siehe *Starten des WebSphere-Anwendungsservers* auf Seite 49.
- Schritt 9** Stellen Sie die Get-Services-WAR-Datei für WebSphere Portal Server bereit. Siehe *Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server* auf Seite 49.
- Schritt 10** Erstellen Sie in WebSphere Portal Server Orte und Seiten für die Anzeige von Get-Services-Portlets. Siehe *Konfigurieren von Orten und Seiten in WebSphere Portal Server* auf Seite 50.
- Schritt 11** Aktivieren Sie die Bearbeitungsrechte für Get-Services-Portlets. Siehe *Aktivieren der Bearbeitungsrechte für Get-Services-Portlets* auf Seite 50.

Schritt 12 Ändern Sie die Datei `httpd.conf` von IBM HTTP Server, um Forward- und Reverse-Proxy-URLs hinzuzufügen. Siehe *Ändern der Datei „httpd.conf“ für IBM HTTP Server* auf Seite 51.

Wenn Sie diese Schritte durchgeführt haben, weist die Installation die folgende Konfiguration auf:



Installationsanforderungen für WebSphere Portal Server

Für die empfohlene Konfiguration von WebSphere Portal Server müssen die folgenden Komponenten auf demselben Server installiert sein:

- WebSphere-Anwendungsserver 4.0.2
- IBM HTTP Server 1.3.19
- IBM DB2 v7-Datenbankservers
- WebSphere Portal Server
- Eine benutzerdefinierte Installation von Get-Services, bei der WebSphere als Anwendungsserver ausgewählt ist

Für die alternative Konfiguration von WebSphere Portal Server müssen die folgenden Komponenten auf mindestens zwei Servern installiert sein:

- Server 1
 - WebSphere-Anwendungsserver 4.0.2
 - IBM HTTP Server 1.3.19
 - IBM DB2 v7-Datenbankservers
 - WebSphere Portal Server
- Server 2
 - Get-Services-kompatibler Anwendungsserver
 - Webserver
 - Backend-Datenbank für Get-Services
 - Eine Installation von Get-Services

Erstellen einer Get-Services-WAR-Datei

Um Get-Services in WebSphere Portal Server anzuzeigen, müssen Sie die Portalkomponenten von Get-Services zunächst als WAR-Datei exportieren. Anschließend können Sie diese WAR-Datei in WebSphere Portal Server importieren und die Portalkomponenten auswählen, die als WebSphere Portal Server-Portlets angezeigt werden sollen.

So erstellen Sie eine Get-Services-WAR-Datei:

- 1 Melden Sie sich bei der Get-Services-Verwaltungsseite an (`admin.jsp`).
- 2 Klicken Sie auf **Integration des IBM-Portals WebSphere**.

- 3 Geben Sie die folgenden Konfigurationsdaten ein:
 - a **Ausgangspfad.** Geben Sie den vollständigen Pfad zur Datei `WebSphere.war` im Ordner `package` von Get-Services an. Standardmäßig handelt es sich hierbei um den folgenden Ordner:


```
<WebSphere>/oaa/packages
```
 - b **Zielpfad.** Geben Sie den vollständigen Pfad und den Dateinamen an, der für die erstellte Get-Services-WAR-Datei verwendet werden soll.
 - c **Standard-URL.** Geben Sie den vollständigen URL des Get-Services-Bereitstellungsverzeichnis ein. Standardmäßig handelt es sich hierbei um den folgenden URL:


```
http://<Server>:<Anschluss>/oaa/servlet/basicauth
```
- 4 Klicken Sie auf **WAR-Datei generieren**.
Get-Services erstellt eine neue WAR-Datei mit dem in Schritt 3 unter **Zielpfad** angegebenen Pfad und Namen.

Anhalten des WebSphere-Anwendungsservers

Melden Sie sich beim Get-Services-Server an und halten Sie den WebSphere-Anwendungsserver an, bevor Sie die Konfiguration ändern.

Ändern der Datei „archway.xml“

Um die Anmeldung über WebSphere Portal Server zu ermöglichen, müssen Sie Get-Services zur Verwendung einer alternativen HTTP-Authentifizierungsmethode konfigurieren.

So ändern Sie die Datei „archway.xml“:

- 1 Öffnen Sie die Datei `archway.xml` in einem Texteditor. Sie finden die Datei unter:


```
<Anwendungsserver>\webapps\oaa\WEB-INF\default.
```
- 2 Bearbeiten Sie die folgende Zeile:


```
<httpauthclass ...>HttpBasicAuthenticationManager</httpauthclass>
```
- 3 Ändern Sie den Wert `HttpBasicAuthenticationManager` in `HttpAlternateAuthenticationManager`.
- 4 Speichern Sie die Datei.

Ändern der Datei „web.xml“

Sie müssen das AuthController-Servlet zur Bereitstellung eines Proxy für die HTTP-Basisauthentifizierung aktivieren.

So ändern Sie die Datei „web.xml“:

- 1 Öffnen Sie die Datei `web.xml` in einem Texteditor. Diese Datei befindet sich unter:

```
<Anwendungsserver>\webapps\oaa\WEB-INF.
```

- 2 Fügen Sie die folgende Zeile am Ende der letzten `<Servlet>`-Definition ein:

```
<Servlet>
  <Servlet-name>AuthController</Servlet-name>
  <display-name>AuthController</display-name>
  <description>Ein Controller (Decorator)-Servlet, mit dem der
konfigurierbare Autorisierungsschutz für jede beliebige Ressource
aktiviert werden kann.</description>

  <Servlet-class>com.peregrine.oaa.archway.AuthControllerServlet
  </Servlet-class>
  <load-on-startup>2</load-on-startup>
</Servlet>

<Servlet-mapping>
  <Servlet-name>AuthController</Servlet-name>
  <url-pattern>/servlet/basicauth/*</url-pattern>
</Servlet-mapping>
<Servlet-mapping>
  <Servlet-name>AuthController</Servlet-name>
  <url-pattern>/servlet/auth/*</url-pattern>
</Servlet-mapping>
```

- 3 Speichern Sie die Datei.

Ändern der Datei „ibm-web-ext.xmi“

Sie müssen den Parameter `fileServingEnabled` auf `true` setzen, um statische Inhalte verarbeiten zu können.

So ändern Sie die Datei „ibm-web-ext.xmi“:

- 1 Öffnen Sie die Datei `ibm-web-ext.xmi` in einem Texteditor. Der Standardpfad lautet:

```
c:\WebSphere\AppServer\installedApps\getit.ear\getit.war\WEB-INF
```

- 2 Setzen Sie den Parameter `fileServingEnabled` auf `true`.

```
fileServingEnabled="true"
```

- 3 Speichern Sie die Datei.

Ändern der Datei „setDomain.js“

Um die alternative Konfiguration von WebSphere Portal Server verwenden zu können, müssen Sie die Funktion `setDomain` aktivieren.

Hinweis: Wenn Sie WebSphere Portal Server in der empfohlenen Konfiguration einrichten, können Sie diese Anweisungen überspringen.

So ändern Sie die Datei „setDomain.js“:

- 1 Melden Sie sich beim Get-Services-Server an.
- 2 Fahren Sie den Anwendungsserver herunter.
- 3 Öffnen Sie die Datei `setDomain.js` in einem Texteditor. Diese Datei befindet sich unter:


```
<Anwendungsserver>\webapps\oaa\js.
```
- 4 Fügen Sie die folgende Zeile am Ende der Datei ein:


```
setDomain();
```
- 5 Speichern Sie die Datei.

Starten des WebSphere-Anwendungsservers

Starten Sie den WebSphere-Anwendungsserver, damit die Änderungen wirksam werden.

Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server

Nach dem Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server können Sie die anzuzeigenden Portlets, die Einstellungen für die Anzeige und die Zugriffsrechte für jedes Portlet konfigurieren.

Ausführliche Anweisungen finden Sie in der Dokumentation zu WebSphere Portal Server.

So stellen Sie die Get-Services-WAR-Datei bereit:

- 1 Melden Sie sich als `wpsadmin` oder als ein anderer Benutzer mit Verwaltungsrechten bei WebSphere Portal an.
- 2 Wählen Sie im Menü **Places** (Orte) den Befehl **Portal Administration** (Portalverwaltung) aus.
- 3 Klicken Sie auf **Portlets > Install Portlets** (Portlets installieren).

- 4 Klicken Sie auf **Browse** (Durchsuchen) und navigieren Sie zu dem Zielpfad, den Sie beim Erstellen der Get-Services-WAR-Datei eingegeben haben.
- 5 Klicken Sie auf **Next** (Weiter), um die Get-Services-WAR-Datei zu laden. WebSphere Portal Server zeigt eine Liste der zu installierenden Portlets an.
- 6 Klicken Sie auf **Install** (Installieren).
WebSphere Portal Server installiert die Portlets und zeigt die folgende Meldung an: **Portlets successfully installed** (Portlets wurden erfolgreich installiert).

Konfigurieren von Orten und Seiten in WebSphere Portal Server

Sie können Get-Services-Portlets an allen Orten und auf allen Seiten bereitstellen, die den folgenden Anforderungen entsprechen.

Orte

Die Orte in WebSphere Portal Server müssen die folgende Eigenschaft aufweisen:

- HTML muss unterstützt werden

Seiten

Die Seiten in WebSphere Portal Server müssen die folgenden Eigenschaften aufweisen:

- HTML muss unterstützt werden
- Die folgende Einstellung muss für die Seite ausgewählt werden: **Allow all portlets that a user can access** (Alle Portlets zulassen, auf die ein Benutzer zugreifen kann)
- Der Benutzergruppe **All authenticated users** (Alle authentifizierten Benutzer) muss die minimale Berechtigung zum Bearbeiten aller auf einer Seite angezeigten Get-Services-Portlets zugewiesen werden

Aktivieren der Bearbeitungsrechte für Get-Services-Portlets

Benutzer von WebSphere Portal Server benötigen Bearbeitungsrechte für die Get-Services-Portlets, um sie anpassen und ihrer Portalseite hinzufügen zu können.

So aktivieren Sie Bearbeitungsrechte für Get-Services-Portlets:

- 1 Melden Sie sich als **wpsadmin** oder als ein anderer Benutzer mit Verwaltungsrechten bei WebSphere Portal an.
- 2 Wählen Sie im Menü **Places** (Orte) den Befehl **Portal Administration** (Portalverwaltung) aus.
- 3 Klicken Sie auf **Security** (Sicherheit) > **Access Control List** (Zugriffskontrollliste).
- 4 Wählen Sie in der Dropdown-Liste für die Option **Special groups** (Spezielle Gruppen) den Eintrag **All authenticated users** (Alle authentifizierten Benutzer).
- 5 Wählen Sie in der Dropdown-Liste **Select the objects for the permissions** (Auswählen der Objekte für die Berechtigungen) den Eintrag **portlet applications** (Portlet-Anwendungen).
- 6 Wählen Sie die Option **Search on** (Suchen nach) und geben Sie dann im Feld **Name contains** (Name enthält) **Peregrine** ein.
- 7 Klicken Sie auf **Go** (Los).
WebSphere Portal Server zeigt eine Liste der Portlets an, deren Name die Zeichenfolge **Peregrine** enthält.
- 8 Klicken Sie in der Spalte **Edit** (Bearbeiten) am Ende der Tabelle auf **Select All** (Alle auswählen).
- 9 Klicken Sie auf **Speichern**.
Die Benutzer können Get-Services-Portlets jetzt über WebSphere Portal Server anzeigen und bearbeiten.

Ändern der Datei „httpd.conf“ für IBM HTTP Server

Um die alternative Konfiguration von WebSphere Portal Server verwenden zu können, müssen Sie die von IBM HTTP Server verwendete Datei **httpd.conf** ändern, so dass der Remote-Instanz von Get-Services die Forward- und Reverse-Proxy-URLs hinzugefügt werden.

Hinweis: Wenn Sie WebSphere Portal Server in der empfohlenen Konfiguration einrichten, können Sie diese Anweisungen überspringen.

So ändern Sie die Datei „httpd.conf“ für IBM HTTP Server:

- 1 Melden Sie sich beim Get-Services-Server an.
- 2 Halten Sie den IBM HTTP Server an.
- 3 Öffnen Sie die Datei httpd.conf in einem Texteditor. Diese Datei befindet sich unter:

C:\IBM HTTP Server\conf

- 4 Fügen Sie die folgenden Zeilen am Ende der Datei ein:

```
ProxyPass /<aaa-Stammverzeichnis>/ http://<Server>:<Anschluss>/
<aaa-Stammverzeichnis>/servlet/basicauth/
ProxyPassReverse /<aaa-Stammverzeichnis>/
http://<Server>:<Anschluss>/
<aaa-Stammverzeichnis>/servlet/basicauth/
```

Geben Sie als <OAA-Stammverzeichnis> den Namen des von IBM HTTP Server verwendeten virtuellen OAA-Verzeichnisses ein. Standardmäßig lautet der Name dieses virtuellen Verzeichnisses oaa.

Geben Sie für <Server>:<Anschluss> den Servernamen und die Nummer des Kommunikationsanschlusses der Get-Services-Installation ein.

- 5 Speichern Sie die Datei.

Konfigurieren des WebSphere-Übersetzungsservers für Get-Services

Sie können Get-Services zur Verwendung eines WebSphere Translation Server konfigurieren, so dass Sie in Echtzeit auf Übersetzungen angezeigter Textdaten zugreifen können.

So konfigurieren Sie WebSphere Translation Server für Get-Services:

- Schritt 1** Kopieren Sie die Datei wts.jar in den Get-Services-Bereitstellungsordner. Siehe *Kopieren von „wts.jar“ in den Get-Services-Bereitstellungsordner* auf Seite 52.
- Schritt 2** Konfigurieren Sie Get-Services zur Verwendung des WebSphere Translation Server. Siehe *Konfigurieren von Get-Services zur Verwendung des WebSphere Translation Server* auf Seite 53.

Kopieren von „wts.jar“ in den Get-Services-Bereitstellungsordner

Nachfolgend wird beschrieben, an welchem Speicherort sich die Datei wts.jar befindet und in welchen Ordner sie kopiert werden muss.

So kopieren Sie „wts.jar“ in den Get-Services-Bereitstellungsordner:

- 1 Fahren Sie den Anwendungsserver herunter.
- 2 Wechseln Sie zum Installationsordner des WebSphere Translation Server.
- 3 Kopieren Sie in diesem Ordner die Datei **wts.jar**.
- 4 Fügen Sie die Datei **wts.jar** in den Get-Services-Bereitstellungsordner ein. Dieser Ordner befindet sich unter:

<Anwendungsserver-Installation>\WEB-INF\lib

- 5 Starten Sie den Anwendungsserver neu.

Konfigurieren von Get-Services zur Verwendung des WebSphere Translation Server

In den folgenden Anweisungen wird beschrieben, wie Get-Services zur Verwendung des WebSphere Translation Server konfiguriert wird.

So konfigurieren Sie Get-Services zur Verwendung des WebSphere Translation Server:

- 1 Melden Sie sich bei der Get-Services-Verwaltungsseite an (admin.jsp).
- 2 Klicken Sie auf das Register **Einstellungen > Allgemein**.

Die Seite für Verwaltungseinstellungen wird geöffnet.

Portal	MobileServer	AssetCenter	Allgemein	Service Desk	Portal-DB	Designs	Webanwendung
Protokollieren	ServiceCenter	Get-Resources	XSL	E-Mail			
Allgemeines Backend: portalDB				Adapterzielname zur Unterstützung allgemeiner Benutzervorgänge.			
Liste der Ziel-Aliasnamen: weblication;mail				Gibt eine Liste mit Ziel-Aliasnamen an, die von den Webanwendungen in diesem Paket verwendet werden. Die Einträge sind durch ein Semikolon getrennt.			
Admin-Name: Admin				Anmeldungsname für den Administrator. Diese Anmeldung ermöglicht den Zugriff auf die administrativen Funktionen ohne Authentifizierung gegen die Backendadapter.			
Admin-Kennwort: []				Admin-Kennwort			
Übersetzung							
Editorklasse des Übersetzungservers: com.peregrine.util.WTSLanguageTranslatorFactory				Die Java-Editorklasse, die eine eigene, dem Übersetzungsserver zugeordnete Klasse generiert.			
Sprache, aus der übersetzt wird: Englisch				Die Sprache, aus der übersetzt werden soll oder die Standardsprache, in der die Texte derzeit angezeigt werden sollen.			
IP-Adresse des Übersetzungservers: 10.3.128.181:1097				Die IP-Adresse des Übersetzungservers. Je nach den für den Übersetzungsserver definierten Anforderungen kann diese Adresse u.U. eine Portnummer enthalten. Klicken Sie für den Standardwert hier!			
Anwendungspfad: WEB-INF/apps/				Verzeichnisstandort für die Webanwendungen des Peregrine-Portals			
Ereigniswarteschlange: portalDB				Geben Sie den Namen des Adapters ein, der von der Warteschlangen-Engine des Peregrine-Portals verwendet werden soll. Beispiel: Um die ServiceCenter-Datenbank zu verwenden, geben Sie "sc" • ein. Um die AssetCenter-Datenbank zu verwenden, geben Sie "ac" ein.			

- 3 Geben Sie die folgenden Konfigurationseinstellungen ein:
 - a **Editorklasse des Übersetzungsservers:** Geben Sie die Java-Editorklasse für den Übersetzungsserver ein. Die standardmäßige Java-Editorklasse ist:
`com.peregrine.util.WTSLanguageTranslatorFactory`
 - b **Sprache, aus der übersetzt wird:** Geben Sie die Ausgangssprache für die Übersetzung ein. Dies ist standardmäßig **Deutsch**.
 - c **IP-Adresse des Übersetzungsservers:** Geben Sie die IP-Adresse und den Kommunikationsanschluss des Übersetzungsservers ein. Beispiel:
`10.3.128.181:1097`.
 - d **Anwendungspfad:** Geben Sie den relativen Pfad zum Verzeichnis der Serveranwendungen ein. Der Standardwert ist:
`WEB-INF/apps/`
 - e **Ereigniswarteschlange:** Geben Sie den Namen des Adapters ein, der von der Warteschlangen-Engine verwendet werden soll. Der Standardwert ist:
`PortalDB`
- 4 Klicken Sie auf **Speichern**.
Die Systemsteuerung wird geöffnet.
- 5 Klicken Sie auf **Server zurücksetzen**.

Übersetzen angezeigter Textdaten mit einem Übersetzungsserver

Wenn Sie Get-Services-Daten in verschiedenen Sprachen speichern möchten, können Sie Get-Services so konfigurieren, dass Daten an einen Übersetzungsserver gesendet und dort in Echtzeit übersetzt werden. Über diese Schnittstelle werden nur Textdaten übersetzt, die aus der Backend-Datenbank abgerufen oder manuell in Formulare eingegeben wurden. Wenn Sie eine übersetzte Benutzeroberfläche benötigen, können Sie ein Get-Services-Sprachpaket direkt bei Peregrine Systems erwerben.

So übersetzen Sie angezeigte Textdaten mit einem Übersetzungsserver:

- 1 Aktivieren Sie den Übersetzungsserver über die Seite **Verwaltung > Einstellungen** (siehe *Konfigurieren von Get-Services zur Verwendung des WebSphere Translation Server* auf Seite 53.).

Die Schaltfläche **Übersetzen** wird oben rechts auf der Symbolleiste angezeigt.

Die Schaltfläche
„Übersetzen“.



- 2 Klicken Sie auf die (aus der Backend-Datenbank abgerufenen oder in ein Formular eingegebenen) Textdaten, die übersetzt werden sollen.

Klicken Sie auf den zu
übersetzenden Text.

 A screenshot of a search form. The title bar says 'Geben Sie die Suchkriterien ein, und klicken Sie auf die Schaltfläche "Suchen".' The form has several fields: 'Name:' with a dropdown and a text input; 'Beschreibung:' with a dropdown and a text input containing 'The quick brown fox jumped over the lazy dog'; 'Standort-ID:' with a dropdown and a text input; 'Standortvorlage:' with two icons; 'Einheitenkategorie:' with two icons; and 'Kategorie:' with a dropdown and a text input. At the bottom are three buttons: 'Suchen', 'Alle anzeigen', and 'Neu'. A line points from the text 'Klicken Sie auf den zu übersetzenden Text.' to the 'Beschreibung:' text input field.

- 3 Klicken Sie auf die Schaltfläche **Übersetzen**.

Das Fenster **Übersetzung** wird geöffnet.

Wählen Sie in der
Dropdown-Liste die
gewünschte Zielsprache aus.



- 4 Wählen Sie in der Dropdown-Liste die Zielsprache aus, in die die ausgewählten Textdaten übersetzt werden sollen.

Die Übersetzung der ausgewählten Textdaten wird im Feld **Übersetzung** angezeigt.

WebLogic 6.1 SP3 oder SP4

Mit den folgenden Verfahren wird WebLogic zur Ausführung von Get-Services unter Windows konfiguriert.

So konfigurieren Sie WebLogic 6.3 SP3 oder SP4 mit IIS:

- Schritt 1** Halten Sie WebLogic und den Webserver an. *Anhalten der Server* auf Seite 56.
- Schritt 2** Bearbeiten Sie die Datei `startWebLogic.cmd`, um das Systemkennwort, die Speichereinstellungen und den Startmodus festzulegen. Siehe *Bearbeiten von „startWebLogic.cmd“* auf Seite 57.
- Schritt 3** Bearbeiten Sie die Datei `Server.Policy`, um die Debug-Option auf `true` zu setzen. Siehe *Bearbeiten der Datei „Server.Policy“* auf Seite 58.
- Schritt 4** Starten Sie das Get-Services-Installationsprogramm. Siehe *Ausführen des Installationsprogramms* auf Seite 58.
- Schritt 5** Verschieben Sie alle Dateien vom Typ `.jar` in den Ordner `ext` des Java Development Kit. Siehe *Verschieben von .jar-Dateien in den Ordner „ext“ des Java Development Kit* auf Seite 58.
- Schritt 6** Konfigurieren Sie IIS zur Verwendung von `iisforward.dll` als ISAPI-Filter und legen Sie eine Erweiterung fest. Siehe *Konfigurieren von „iisforward.dll“ und Festlegen einer Erweiterung* auf Seite 59.
- Schritt 7** Konfigurieren Sie IIS zur Verwendung von `iisproxy.dll` als Erweiterung. Siehe *Konfigurieren der Datei „iisproxy.dll“ als Erweiterung* auf Seite 60.
- Schritt 8** Erstellen Sie auf dem Webserver ein virtuelles Verzeichnis für Get-Services. Siehe *Erstellen eines virtuellen Verzeichnisses für Get-Services* auf Seite 61.
- Schritt 9** Starten Sie WebLogic und den Webserver neu. Siehe *Neustarten der Server* auf Seite 62.

Anhalten der Server

Bevor Sie mit der Konfiguration von WebLogic beginnen, müssen Sie den WebLogic-Server und Webserver herunterfahren.

So beginnen Sie mit der Konfiguration von WebLogic:

- 1 Halten Sie den WebLogic-Anwendungsserver an.
- 2 Halten Sie den Webserver an.

Bearbeiten von „startWebLogic.cmd“

So bearbeiten Sie „startWebLogic.cmd“:

- 1 Öffnen Sie die Datei startWebLogic.cmd in einem Texteditor. Der Standardpfad lautet:

`c:\bea\wlserver6.1\config\Eigene_Domäne\`

- 2 Wechseln Sie zu folgendem Abschnitt des Skripts:

```
echo *****
echo * To start WebLogic Server, use the password      *
echo * assigned to the system user. The system        *
echo * username and password must also be used to    *
echo * access the WebLogic Server console from a web  *
echo * browser.                                       *
echo *****
@rem Set WLS_PW equal to your system password for no password prompt
server startup.
set WLS_PW=password
```

- 3 Ersetzen Sie in der letzten Zeile das Wort password durch Ihr Systemkennwort für WebLogic.
- 4 Suchen Sie in der Datei nach der Parametereinstellung `-mx`. Ändern Sie diese Einstellung auf einen Wert zwischen 225 MB und 512 MB.

Hinweis: Stellen Sie sicher, dass der Höchstwert für die Heap-Größe kleiner ist als der für die Anwendungsserver verfügbare freie Arbeitsspeicher. Wenn Sie einen größeren Wert angeben als freier Arbeitsspeicher verfügbar ist, lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt. Für die meisten Systeme reicht die Einstellung von 256 MB aus.

- 5 Setzen Sie die Variable STARTMODE auf `STARTMODE=false`.
Wenn Sie WebLogic nach der Installation erstmals starten, müssen Sie den Entwicklungsmodus auswählen, damit es die bereitgestellten Webanwendungen erkennt.
- 6 Fügen Sie die folgende Zeile vor dem Ende des Eintrags `goto finish` ein:

```
“-Djava.security.auth.login.config==<Weblogic>\lib\server.policy"
weblogic.server
```

Geben Sie für `<Weblogic>` den Installationspfad für WebLogic ein. Standardmäßig handelt es sich hierbei um den folgenden Pfad:

`c:\bea\wlserver6.1`

- 7 Speichern Sie die Datei.

Bearbeiten der Datei „Server.Policy“

So bearbeiten Sie die Datei „Server.Policy“:

- 1 Öffnen Sie die Datei Server.Policy in einem Texteditor. Der Standardpfad lautet:

```
c:\bea\wlserver6.1\lib\
```

- 2 Fügen Sie die folgenden Zeilen am Ende der Datei ein:

```
ServerLoginModule
{
    weblogic.security.internal.ServerLoginModule required debug=true;
};
```

- 3 Speichern Sie die Datei.

Ausführen des Installationsprogramms

Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Verfahren für die benutzerdefinierte Installation* auf Seite 79.

Verschieben von .jar-Dateien in den Ordner „ext“ des Java Development Kit

So verschieben Sie .jar-Dateien:

- 1 Überprüfen Sie, ob das folgende Verzeichnis existiert. Erstellen Sie es ggf.:
c:\bea\jdk131\jre\lib\ext
- 2 Wechseln Sie zum Ordner lib der Peregrine OAA-Plattform (normalerweise bea\wlserver6.1\config*Eigene Domäne*\applications\oaa\WEB-INF\lib), wobei *<Eigene Domäne>* die WebLogic-Domäne des Systems ist, auf dem WebLogic installiert ist. Verschieben Sie die folgende Datei in den Ordner \bea\jdk131\jre\lib\ext:
log4j-1.2.6.jar
- 3 Stellen Sie sicher, dass sich die Datei oaasecurityproxy.jar im Ordner \bea\jdk131\jre\lib\ext befindet.

- 4 Wechseln Sie zum Ordner `external` der Peregrine OAA-Plattform (normalerweise `Peregrine\oaa\external`) und kopieren Sie die folgenden Dateien in den Ordner `\bea\jdk131\jre\lib\ext`:

```
jaas.jar
jai_codec.jar
jai_core.jar
jce1_2_1.jar
jcert.jar
jnet.jar
jsse.jar
local_policy.jar
mllibwrapper_jai.jar
sunjce_provider.jar
US_export_policy.jar
```

Konfigurieren von „issforward.dll“ und Festlegen einer Erweiterung

Um eine Verbindung zwischen WebLogic und IIS herzustellen, müssen Sie `issforward.dll` als ISAPI-Filter installieren.

So installieren Sie „issforward.dll“ als ISAPI-Filter und legen eine Erweiterung fest:

- 1 Öffnen Sie die Managementkonsole für Internetdienste.
- 2 Klicken Sie mit der rechten Maustaste auf `<Computername>` und dann auf **Eigenschaften**.

Hinweis: Dies ist keine der Websites; vielmehr handelt es sich hierbei um den übergeordneten Knoten in der Struktur für die Websites.

Ein Dialogfeld wird geöffnet.

- 3 Klicken Sie im Fensterausschnitt **Haupteigenschaften** auf **Bearbeiten**.
- 4 Klicken Sie auf das Register **ISAPI-Filter**.
- 5 Klicken Sie auf **Hinzufügen**.

- 6 Geben Sie die nachfolgend aufgeführten Daten ein:
 - a **Filtername:** iisforward.
 - b **Ausführbare Datei:** issforward.dll. Der Standardpfad lautet:
c:\bea\wlserver6.1\bin\issforward.dll
- 7 Klicken Sie auf **OK**.
- 8 Klicken Sie auf das Register **Basisverzeichnis**.
- 9 Klicken Sie auf **Konfiguration**.
Das Register **Anwendungskonfiguration** der Seite **Anwendungszuordnungen** wird geöffnet.
- 10 Überprüfen Sie, ob eine Zuordnung für die Erweiterung **wlforward** vorhanden ist. Ist dies nicht der Fall, klicken Sie auf **Hinzufügen**, um die Erweiterung **.wlforward** zuzuordnen.
- 11 Geben Sie die nachfolgend aufgeführten Daten ein:
 - a **Ausführbare Datei:** issforward.dll. Der Standardpfad lautet:
c:\bea\wlserver6.1\bin\issforward.dll
 - b **Erweiterung:** .wlforward.
- 12 Schließen Sie die Managementkonsole für Internetdienste.

Konfigurieren der Datei „iisproxy.dll“ als Erweiterung

Um eine Verbindung zwischen WebLogic und IIS herzustellen, müssen Sie die Datei **iisproxy.dll** als Erweiterung installieren.

So installieren Sie die Datei „iisproxy.dll“ als Erweiterung:

- 1 Öffnen Sie die Managementkonsole für Internetdienste.
- 2 Klicken Sie mit der rechten Maustaste auf den Knoten **Standardwebsite** und klicken Sie dann auf **Eigenschaften**.
- 3 Klicken Sie auf das Register **Basisverzeichnis**.
 - a Aktivieren Sie das Kontrollkästchen **Lesen**.
 - b Wählen Sie in der Dropdown-Liste **Ausführungsberechtigungen** den Eintrag **Skripts und ausführbare Dateien** aus.
- 4 Klicken Sie auf **Konfiguration**.
Das Register **Anwendungskonfiguration** der Seite **Anwendungszuordnungen** wird geöffnet.

- 5 Stellen Sie sicher, dass eine Zuordnung für die Erweiterung `.jsp` zu `c:\bea\wlserver6.1\bin\iisproxy.dll` angezeigt wird.
 - a Gehen Sie folgendermaßen vor, wenn keine Zuordnung für die Erweiterung `.jsp` vorhanden ist:
 - b Klicken Sie auf **Hinzufügen**.
 - c Geben Sie die nachfolgend aufgeführten Daten ein:
 - Ausführbare Datei: `iisproxy.dll`. Der Standardpfad lautet:
`c:\bea\wlserver6.1\bin\iisproxy.dll`
 - Erweiterung: `.jsp`.

Hinweis: Vergewissern Sie sich, dass sich die Datei `iisproxy.dll` in demselben Verzeichnis wie die Datei `iisforward.dll` befindet.
- 6 Schließen Sie die Managementkonsole für Internetdienste.

Erstellen eines virtuellen Verzeichnisses für Get-Services

Um Get-Services ausführen zu können, müssen Sie auf dem Webserver ein virtuelles Verzeichnis erstellen, das eine Zuordnung zum WebLogic-Bereitstellungsordner festlegt. Bei einer typischen Installation wird ein virtuelles Verzeichnis namens `oaa` erstellt; Sie können dem virtuellen Verzeichnis jedoch auch einen anderen Namen zuweisen.

So konfigurieren Sie ein virtuelles Verzeichnis:

- 1 Verwenden Sie die folgenden Richtlinien zum Erstellen eines virtuellen Verzeichnisses auf der Standardwebsite.

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	<code><oaa></code>
Zugriffsberechtigungen	Lesen, Skriptzugriff
Zuordnung zu physischem Pfad	<code><Weblogic>\applications\oaa</code>
Erteilen von Ausführungsberechtigungen für	Skripts und ausführbare Dateien

- a Geben Sie für `<oaa>` den Namen des virtuellen Verzeichnisses ein, das für Get-Services verwendet werden soll. Den hier gewählten Namen müssen Sie ebenfalls in der Konfiguration des Anwendungsservers verwenden.
- b Geben Sie für `<WebLogic>` den Pfad zur WebLogic-Installation ein. Der Standardpfad lautet:
`c:\bea\wlserver6.1\config\<Eigene_Domäne>\applications\oaa`

- 2 Klicken Sie mit der rechten Maustaste auf das neu erstellte virtuelle Verzeichnis unter **Standardwebsite**.
 - a Wählen Sie **Eigenschaften** aus.
 - b Stellen Sie sicher, dass die Berechtigung **Skripts und ausführbare Dateien** lautet.
- 3 Klicken Sie im Register **Basisverzeichnis** auf **Konfiguration**.
Das Register **Anwendungskonfiguration** der Seite **Anwendungszuordnungen** wird geöffnet.
- 4 Stellen Sie sicher, dass eine Zuordnung für die Erweiterung `.jsp` zu `c:\bea\wlserver6.1\bin\iisproxy.dll` angezeigt wird.

Neustarten der Server

Sie müssen die Server neu starten, damit die neuen WebLogic-Konfigurationen aktiviert werden.

So aktivieren Sie die WebLogic-Konfigurationen:

- 1 Starten Sie den Webserver neu.
- 2 Starten Sie den WebLogic-Server neu.
- 3 Starten Sie Get-Services.

JRun 3.1

Mit den folgenden Verfahren wird JRun zur Ausführung von Get-Services unter Windows konfiguriert.

So konfigurieren Sie JRun 3.1:

- Schritt 1** Installieren Sie eine Java-Laufzeitumgebung. Siehe *Installieren einer Java-Laufzeitumgebung* auf Seite 63.
- Schritt 2** Laden Sie JRun von der Macromedia-Website herunter und installieren Sie das Programm im Stammverzeichnis der Festplatte (zum Beispiel C:\). Siehe *Installieren von JRun* auf Seite 63.
- Schritt 3** Wenden Sie das aktuelle JRun-Update an. Siehe *Anwenden des aktuellen JRun-Updates* auf Seite 64.
- Schritt 4** Stellen Sie die WAR-Datei des Portals für JRun bereit, so dass die für Get-Services benötigte Ordnerstruktur erstellt wird. Siehe *Bereitstellen der WAR-Datei des Portals für JRun* auf Seite 64.

- Schritt 5** Starten Sie das Get-Services-Installationsprogramm. Siehe *Ausführen des Installationsprogramms* auf Seite 66.
- Schritt 6** Verschieben Sie `js.jar` in den Ordner `ext` des Java Development Kit. Siehe *Verschieben von „js.jar“ in das Java Development Kit* auf Seite 66.
- Schritt 7** Führen Sie den JRun Connector Wizard (JRun-Verbindungsassistent) aus, um eine Verbindung zwischen JRun und dem Webserver herzustellen. Siehe *Ausführen des JRun Connector Wizard (JRun-Verbindungsassistent)* auf Seite 67.
- Schritt 8** Konfigurieren Sie die JRun Java-Einstellungen. Siehe *Konfigurieren der Java-Einstellungen* auf Seite 67.
- Schritt 9** Konfigurieren Sie JRun für IIS und erstellen Sie auf Ihrem Webserver ein virtuelles Verzeichnis für Get-Services. Siehe *Konfigurieren von JRun für IIS* auf Seite 70.
- Schritt 10** Starten Sie JRun und den Webserver neu. Siehe *Neustarten der Server* auf Seite 71.

Installieren einer Java-Laufzeitumgebung

Das Get-Services-Installationsprogramm umfasst Java 2 SDK Standard Edition v1.3.1_05; Sie können jedoch auch JRE 1.3.1 verwenden, wenn diese Version bereits installiert ist. Siehe *Komponenten einer benutzerdefinierten Installation* auf Seite 78.

Installieren von JRun

Vor der Installation von Get-Services müssen Sie JRun in Ihrem Stammverzeichnis installieren.

So installieren Sie JRun:

- 1 Rufen Sie den folgenden URL auf:
<http://www.macromedia.com/software/jrun/>
- 2 Klicken Sie auf den Link **JRun 3.1 Available for Purchase** (JRun 3.1 verfügbar).
- 3 Folgen Sie den angezeigten Installationsanweisungen.

Anwenden des aktuellen JRun-Updates

Bevor Sie Get-Services installieren, müssen Sie das aktuelle Update für JRun 3.1 anwenden.

So installieren Sie das aktuelle JRun-Update:

- 1 Rufen Sie den folgenden URL auf:
http://www.macromedia.com/support/jrun/updates/3/updates_31.html
- 2 Klicken Sie auf den Link für die JRun-Edition (Enterprise, Advanced oder Professional) und das Betriebssystem des Servers.
- 3 Folgen Sie den angezeigten Installationsanweisungen.

Bereitstellen der WAR-Datei des Portals für JRun

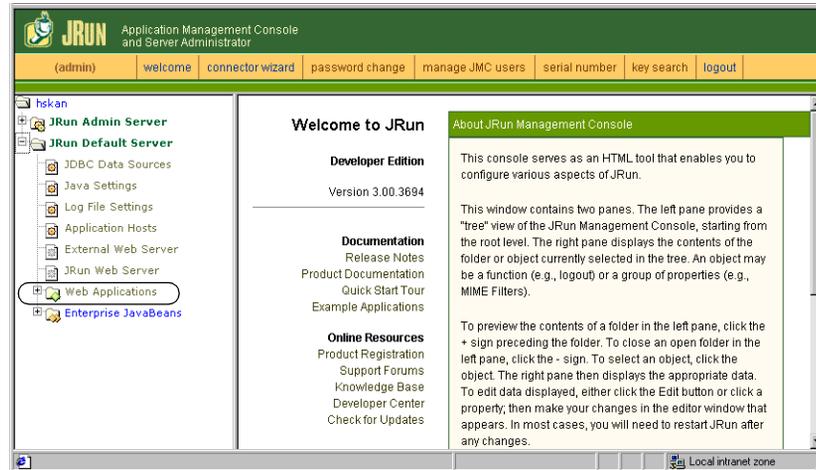
Die WAR-Datei des Portals erstellt die Ordnerstruktur, die zur Bereitstellung von Get-Services auf dem Anwendungsserver benötigt wird. Wenn Sie diese Datei für JRun bereitgestellt haben, können Sie das Get-Services-Installationsprogramm ausführen.

So stellen Sie die Get-Services-WAR-Datei des Portals für JRun bereit:

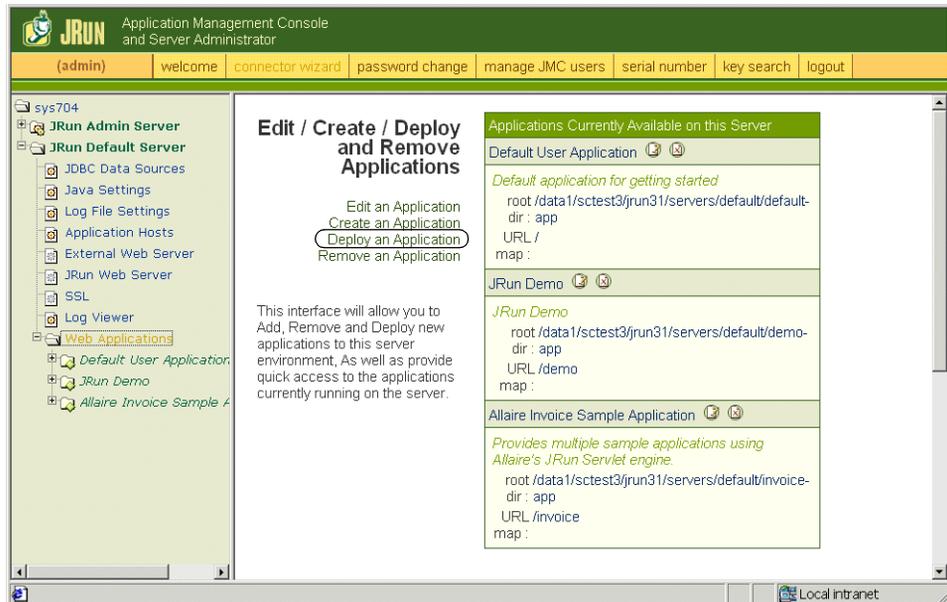
- 1 Öffnen Sie die JRun Management Console (JRun-Managementkonsole) und melden Sie sich an.



2 Wählen Sie JRun Default Server (JRun-Standardserver) > Web Applications (Webanwendungen).



Die Seite **Edit / Create / Deploy and Remove Applications** (Anwendungen bearbeiten/erstellen/bereitstellen und entfernen) wird geöffnet.



3 Klicken Sie auf den Link **Deploy an Application** (Anwendung bereitstellen).

- 4 Machen Sie in den Feldern auf der daraufhin angezeigten Seite folgende Angaben:
 - Servlet War File or Directory (Servlet-WAR-Datei oder -Verzeichnis):
Wechseln Sie zu
<CD-ROM-Laufwerk>:\oaa\packages\portal<Versionsnummer>.war.
Wählen Sie für <Versionsnummer> die aktuellste Version aus.
Wählen Sie diese Datei aus und klicken Sie auf **Accept** (Annehmen).
 - JRun Server Name (Name des JRun-Servers):
Wählen Sie **JRun Default Server** (JRun-Standardserver).
 - Application Name (Anwendungsname):
Geben Sie oaa ein.
 - Application URL (Anwendungs-URL):
Geben Sie /oaa ein.
 - Application Deploy Directory (Bereitstellungsverzeichnis der Anwendung):
Dieses Verzeichnis wird von JRun erstellt. Notieren Sie diesen Pfad. (Sie benötigen diese Angabe weiter unten in diesem Verfahren.) Beispiel:
c:\JRun\servers\default\oaa
- 5 Klicken Sie auf **Deploy** (Bereitstellen).
Es wird eine Meldung angezeigt, die darauf hinweist, dass OAA bereitgestellt wurde.

Ausführen des Installationsprogramms

Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Verfahren für die benutzerdefinierte Installation* auf Seite 79.

Verschieben von „js.jar“ in das Java Development Kit

Um JRun ausführen zu können, muss eine aktuelle Version von js.jar im Java Development Kit verfügbar sein.

So verschieben Sie „js.jar“ in das Java Development Kit:

- 1 Halten Sie alle ausgeführten JRun-Dienste an.
- 2 Suchen Sie js.jar. Der Standardpfad lautet:
<JRun-Installation>\servers\default\oaa\WEB-INF\lib

- 3 Schneiden Sie die Datei aus und fügen Sie sie im Ordner ext des Java Development Kit ein. Beispiel:
C:\Program Files\Peregrine\Common\jdk1.3.1_05\jre\lib\ext
- 4 Starten Sie JRun neu.

Ausführen des JRun Connector Wizard (JRun-Verbindungsassistent)

Der JRun Connector Wizard (JRun-Verbindungsassistent) stellt eine Verbindung zwischen JRun und dem Webserver her.

So führen Sie den JRun Connector Wizard (JRun-Verbindungsassistent) aus:

- 1 Melden Sie sich bei der JRun Management Console (JRun-Managementkonsole) an.
- 2 Klicken Sie auf **Connector Wizard** (Verbindungsassistent).
- 3 Wählen Sie unter **JRun Server Name** (Name des JRun-Servers) den Eintrag **JRun Default Server** (JRun-Standardserver).
- 4 Wählen Sie in der Dropdown-Liste den von Ihnen verwendeten Webserver aus.
- 5 Wenn der Webserver eine andere IP-Adresse als der JRun-Server verwendet, geben Sie unter **JRun Server IP Address** (IP-Adresse des JRun-Servers) die IP-Adresse des JRun-Servers ein.
- 6 Stellen Sie sicher, dass der Verbindungsanschluss des JRun-Servers nicht in Konflikt mit einem anderen auf diesem Server verwendeten Kommunikationsanschluss steht.
- 7 Geben Sie den Pfad zum Skriptverzeichnis ein. Für IIS 5.0 lautet dieser Wert:
C:\Inetpub\Scripts
- 8 Klicken Sie auf **Done** (Fertig).
Es wird eine Meldung angezeigt, dass JRun erfolgreich eine Verbindung zum Webserver hergestellt hat.

Konfigurieren der Java-Einstellungen

Nach der Installation von Get-Services müssen Sie die Java-Einstellungen konfigurieren, mit denen JRun die Webanwendung ausführt.

So konfigurieren Sie die Java-Einstellungen:

- 1 Melden Sie sich bei der JRun Management Console (JRun-Managementkonsole) an.
- 2 Klicken Sie auf **JRun Default Server** (JRun-Standardserver) > **Java Settings** (Java-Einstellungen).

Die Seite **Java Settings** (Java-Einstellungen) wird geöffnet.

- 3 Klicken Sie auf **Java Arguments** (Java-Argumente).

Klicken Sie auf **Java Arguments** (Java-Argumente).

The screenshot shows the JRun Management Console interface. The left sidebar contains a tree view with 'JRun Default Server' expanded, showing 'Java Settings' selected. The main content area displays 'JRun Default Server > Java Settings'. Below this, there is a table of settings for the Java Virtual Machine. The 'Classpath' and 'Java Arguments' rows are highlighted in blue. The 'Java Arguments' row has an 'edit' button next to it. At the bottom of the page, there is a copyright notice for Allaire Corporation.

Name	Value	Summary
Java Executable	C:\jdk131-1\bin\javaw.exe	Path to your JVM executable
System.out Log File	{run.rootdir}\logs\{run.server.name}-out.log	Location where System.out messages appear
System.err Log File	{run.rootdir}\logs\{run.server.name}-err.log	Location where System.err messages appear
JRun Control Port	53000	Port used by JRun to send server commands
Classpath	{default}	Additional classpath entries
Java Arguments		Additional command-line arguments passed to the Java Executable
Library Path	{servlet.jspath};{ejb.jspath}	Directory of native JNI

edit

add to welcome page

Allaire, JRun, JRun logo, and the Allaire logo are trademarks of Allaire Corporation in the USA and other countries.
 Portions of this software are copyrighted by Merant, Inc., 1991-2001
 This product includes code licensed from RSA Data Security
 All other products or name brands are the trademarks of their respective holders.
 © 1997-2001 Allaire Corporation. All rights reserved.

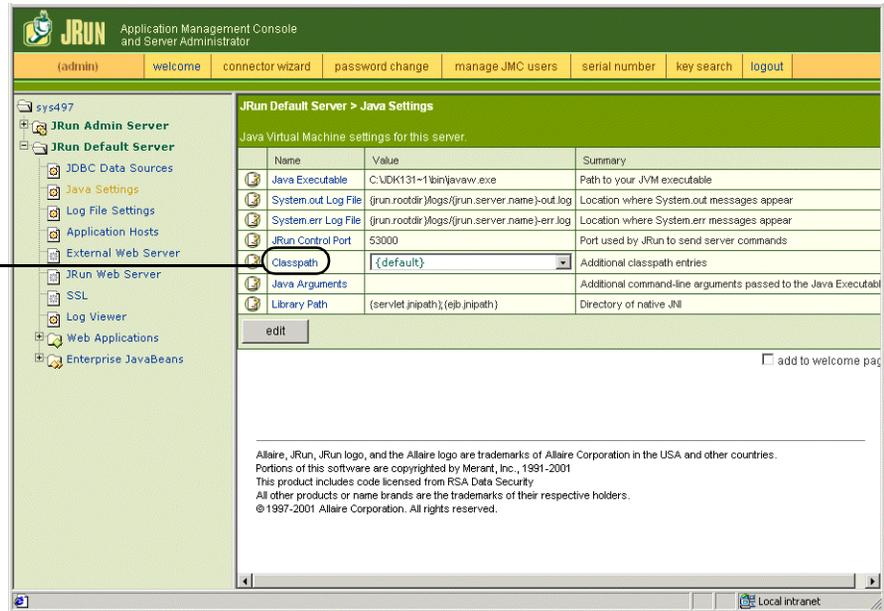
Das Bearbeitungsfenster wird geöffnet.

- 4 Geben Sie einen `-Xmx`-Wert ein, um eine Obergrenze für den Heap-Speicher des Systems festzulegen. Dieser Wert sollte zwischen 225 MB und 512 MB liegen.

Hinweis: Stellen Sie sicher, dass der Höchstwert für die Heap-Größe kleiner ist als der für die Anwendungsserver verfügbare freie Arbeitsspeicher. Wenn Sie einen größeren Wert angeben als freier Arbeitsspeicher verfügbar ist, lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt. Für die meisten Systeme reicht die Einstellung von 256 MB aus. Für Anwendungen, die dauerhaft verwendet werden, müssen Sie ggf. einen höheren Wert angeben.

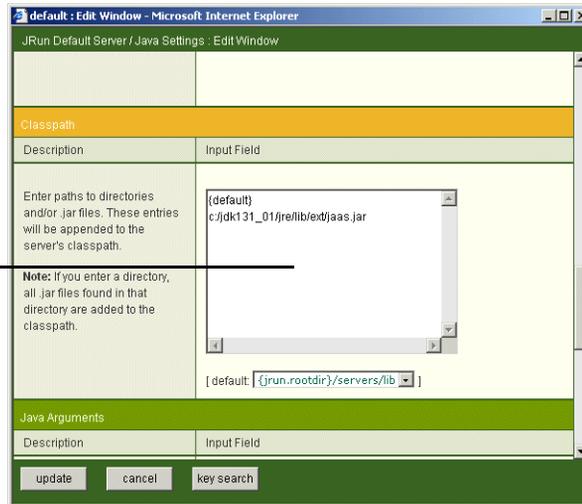
5 Klicken Sie auf der Seite Java Settings (Java-Einstellungen) auf Classpath (Klassenpfad).

Klicken Sie auf **Classpath** (Klassenpfad).



Das Bearbeitungsfenster wird geöffnet.

Geben Sie hier den Pfad zu den JAR-Dateien ein.



- 6 Geben Sie die folgenden Klassenpfade ein:
 - Ordner `ext` des Java Development Kit. Beispiel:
C:\Program Files\Peregrine\Common\jdk1.3.1_05\jre\lib\ext
- 7 Klicken Sie auf der Seite **Java Settings** (Java-Einstellungen) auf **Java Executable** (Ausführbare Java-Datei).
- 8 Stellen Sie sicher, dass der Pfad des Java Development Kit dem unter **Classpath** (Klassenpfad) angegebenen Pfad entspricht. Beispiel:
C:\Program Files\Peregrine\Common\jdk1.3.1_05\bin\javaw.exe
- 9 Klicken Sie auf **update** (Aktualisieren).
- 10 Melden Sie sich bei der JRun Management Console (JRun-Managementkonsole) ab.

Konfigurieren von JRun für IIS

Um eine Verbindung zwischen JRun und IIS herstellen zu können, müssen Sie die Verbindung zunächst überprüfen.

So konfigurieren Sie JRun für IIS:

- 1 Öffnen Sie die IIS Management Console.
- 2 Klicken Sie mit der rechten Maustaste auf `<Computername>` und dann auf **Eigenschaften**.

Hinweis: Dies ist keine der Websites; vielmehr handelt es sich hierbei um den übergeordneten Knoten in der Struktur für die Websites.

Ein Dialogfeld mit einer Schaltfläche **Bearbeiten** wird geöffnet.

- 3 Klicken Sie auf **Bearbeiten > ISAPI-Filter**.
- 4 Überprüfen Sie, ob der JRun-Eintrag korrekt ist und der Wert den vollständigen Pfad zur Datei `scripts/jrun.dll` enthält.
Filtername: Ausführbare Datei für JRun Connector-Filter: `/jrun.dll`
- 5 Überprüfen Sie die folgenden Einstellungen:
 - Das virtuelle Verzeichnis `SCRIPTS` ist in IIS definiert.
 - Es verweist auf das Verzeichnis `Inetpub/Scripts`.
 - Es verfügt über die Berechtigung **Skripts und ausführbare Dateien**.

Anschließend müssen Sie auf dem Webserver ein virtuelles Verzeichnis erstellen, das dem JRun-Bereitstellungsordner zugeordnet ist. Bei einer typischen Installation wird ein virtuelles Verzeichnis namens `oaa` erstellt; Sie können dem virtuellen Verzeichnis jedoch auch einen anderen Namen zuweisen.

So konfigurieren Sie ein virtuelles Verzeichnis:

- 1 Verwenden Sie die folgenden Richtlinien, um ein virtuelles Verzeichnis für `oaa` unter dem Knoten **Standardwebsite** zu erstellen.

Anforderungen für das virtuelle Verzeichnis für Get-Services

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	<code><oaa></code>
Zuordnung zu physischem Pfad	<code><JRun>\oaa</code>
Erteilen von Ausführungsberechtigungen für	Skripts und ausführbare Dateien

- 2 Geben Sie für `<oaa>` den Namen des virtuellen Verzeichnisses ein, das für Get-Services verwendet werden soll. Den hier gewählten Namen müssen Sie ebenfalls in der Konfiguration des Anwendungsservers verwenden.
- 3 Geben Sie für `<JRun>` den Pfad zur JRun-Installation ein. Der empfohlene Installationspfad ist:
`C:\JRun\servers\default\oaa`
- 4 Klicken Sie mit der rechten Maustaste auf das erstellte virtuelle Verzeichnis `oaa` und wählen Sie **Eigenschaften** aus.
- 5 Stellen Sie sicher, dass die Berechtigung **Skripts und ausführbare Dateien** lautet.

Neustarten der Server

Sie müssen die Server neu starten, damit die neuen JRun-Konfigurationen wirksam werden.

So aktivieren Sie die Jrun-Konfigurationen:

- 1 Starten Sie den Webserver neu.
- 2 Starten Sie den JRun-Standardserver neu.
- 3 Starten Sie Get-Services.

Installationsoption „Typisch“

Bei einer Get-Services-Installation mit der Option **Typisch** werden die am häufigsten verwendeten Komponenten des Produkts installiert und die Anwendungsdateien und -daten werden in den standardmäßigen Zielverzeichnissen gespeichert. Die meisten Benutzer verwenden die typische Installation.

Komponenten der typischen Installation

Es folgt eine kurze Beschreibung der Komponenten, die bei einer typischen Installation von Get-Services automatisch installiert werden:

Anwendungs- und Dateipfade

Get-Services-Komponente	Standardmäßiges Installationsverzeichnis
Apache-Webserver	C:\Program Files\Peregrine\Common\Apache2
Tomcat-Anwendungsserver	C:\Program Files\Peregrine\Common\Tomcat4
Java Development Kit	C:\Program Files\Peregrine\Common\jdk1.3.1_05
OAA-Plattform und Get-Services	C:\Program Files\Peregrine\oaa

Dienste

Das Installationsprogramm erstellt und startet außerdem die folgenden Dienste auf dem Windows-Server:

- Apache-Webdienst
- Peregrine Tomcat-Dienst

Wichtig: Wenn Sie bereits andere Web- oder Anwendungsserver auf dem Windows-Zielsystem ausführen, müssen Sie die Dienste der entsprechenden Anwendungen anhalten, bevor Sie das Setup-Programm von Get-Services ausführen. Das Setup-Programm wird zwar auch dann ordnungsgemäß beendet, wenn Sie diese Dienste vor der Installation nicht anhalten, die zum Ausführen von Get-Services benötigten Dienste Apache und Peregrine Tomcat werden jedoch u. U. nicht erstellt.

Kommunikationsanschlüsse

Get-Services verwendet bei einer typischen Installation die in der folgenden Tabelle aufgeführten Kommunikationsanschlüsse. Nach der Installation können Sie Get-Services zur Verwendung alternativer Kommunikationsanschlüsse konfigurieren, wenn die angegebenen Kommunikationsanschlüsse im lokalen Netzwerk bereits verwendet werden.

Standard-anschluss	Wird verwendet von	Alternativer Anschluss
80	Apache-Webserver	8081
8005	Verwaltung des Tomcat-Anwendungsservers	8015
8009	Worker-Datei des Tomcat-Anwendungsservers	8019
8011	Worker-Datei der Tomcat-Anwendung für Lastenausgleich (optional).	8021
8013	Worker-Datei der Tomcat-Anwendung für Lastenausgleich (optional).	8023
8015	Worker-Datei des Tomcat-Anwendungsservers für Lastenausgleich (optional).	8025

Hinweis: Wenn Sie Einstellungen dieser Komponenten ändern oder andere Komponenten installieren möchten, verwenden Sie die Installationsoption **Benutzerdefiniert** (Custom) für Get-Services.

Verfahren für die typische Installation

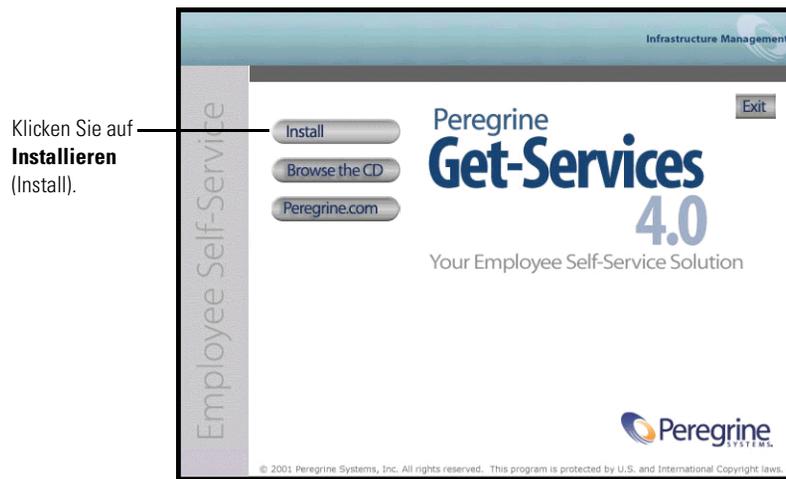
In diesem Abschnitt wird das Verfahren für die Installation von Get-Services mit einem Tomcat-Anwendungsserver und einem Apache-Webserver unter einem Windows-Betriebssystem erläutert.

So führen Sie eine typische Installation von Get-Services unter Windows durch:

- 1 Legen Sie die Get-Services-Installations-CD in das CD-ROM-Laufwerk des Computers ein. In der Regel wird das Installationsprogramm automatisch gestartet.

Wenn das Installationsprogramm nicht automatisch gestartet wird, verwenden Sie den Windows-Befehl **Start > Ausführen**, wechseln Sie zum CD-ROM-Laufwerk und öffnen Sie die Datei **Setup.exe**.

- 2 Klicken Sie im Hauptbildschirm des Installationsprogramms auf **Installieren** (Install).

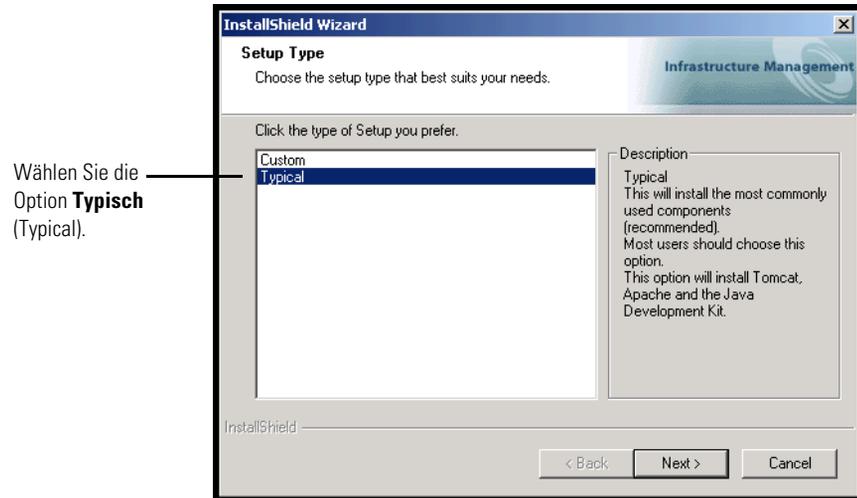


In Statusmeldungen wird darauf hingewiesen, dass das Setup-Programm den InstallShield-Assistenten vorbereitet.

Hinweis: Das Setup-Programm hält den Apache- und Peregrine Tomcat-Dienst an. Folgen Sie zum Fortfahren den Eingabeaufforderungen.

- 3 Wählen Sie im Bildschirm **Setup-Typ** (Setup Type) die Option **Typisch** (Typical) und klicken Sie dann auf **Weiter** (Next).

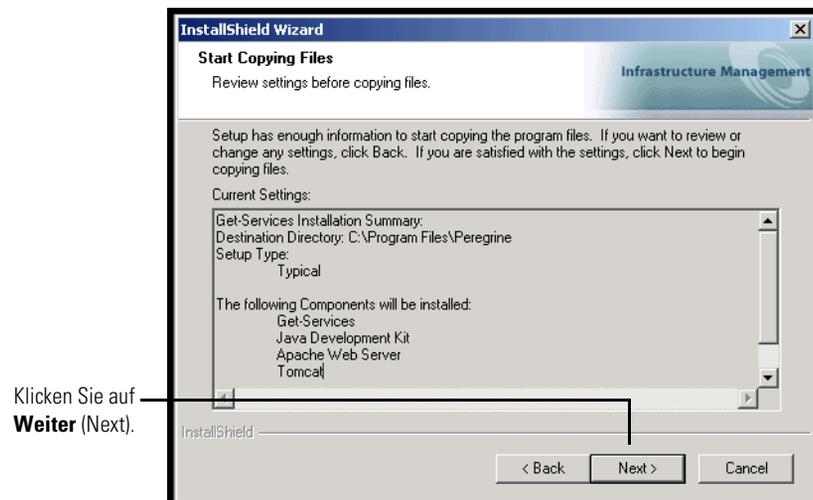
Hinweis: Dieser Bildschirm wird während einer Wartungs- oder Upgrade-Installation nicht angezeigt.



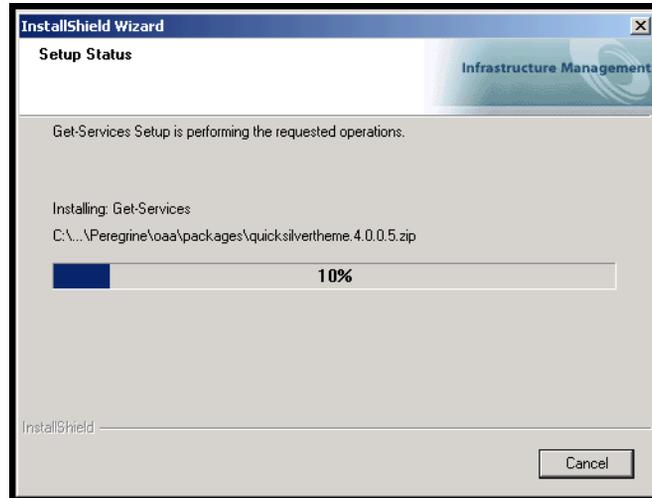
Das Installationsprogramm zeigt Statusmeldungen zur Prüfung des Namens des Hostsystems an.

- 4 Klicken Sie im Bildschirm **Kopiervorgang starten** (Start Copying Files) auf **Weiter** (Next).

Hinweis: Dieser Bildschirm wird während einer Wartungs- oder Upgrade-Installation nicht angezeigt.



Im Bildschirm mit dem Setup-Status wird angezeigt, dass Get-Services Dateien auf dem System installiert.



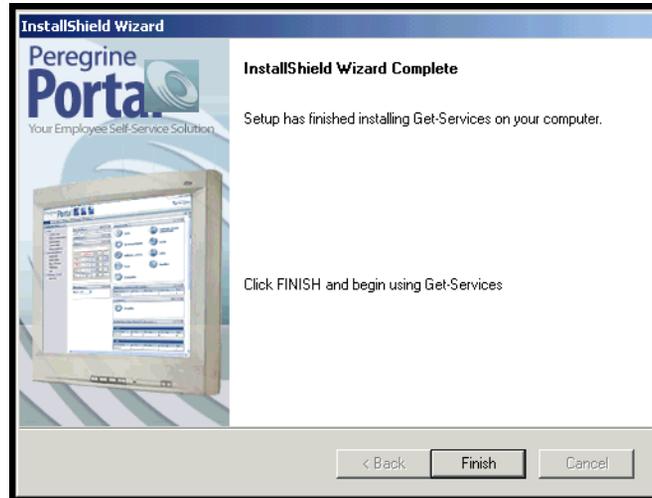
Eine Statusmeldung weist darauf hin, dass die Get-Services-Pakete auf dem Server bereitgestellt werden. Diese Phase kann einige Minuten in Anspruch nehmen.



Während diese Statusmeldung angezeigt wird, führt das Setup-Programm die folgenden Schritte durch:

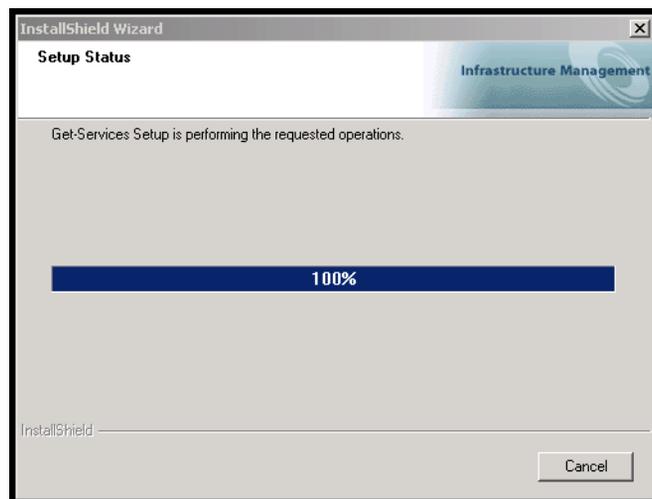
- Kopieren der Dokumente
- Erstellen und Starten der Dienste Apache und Peregrine Tomcat

- 5 Klicken Sie im Bildschirm über den Abschluss des InstallShield-Assistenten auf **Fertig stellen** (Finish).



Hinweis: Bei einer Wartungs- oder Upgrade-Installation wird darauf hingewiesen, dass die Wartung abgeschlossen ist.

Das Setup-Programm führt die letzten Installationsschritte durch.



- 6 Wenn der Bildschirm des InstallShield-Assistenten geschlossen wurde, rufen Sie die Windows-Dienste auf, um sicherzustellen, dass die Dienste Apache und Peregrine Tomcat erstellt wurden. Beide Dienste müssen über den Status **Gestartet** verfügen.

Wenn einer oder beide Dienste nicht gestartet wurden, finden Sie unter *Fehlerbehebung* Informationen zur Lösung möglicher Installationsprobleme.

Die Verfahren für eine typische Installation von Get-Services auf einem Windows-Server sind hiermit abgeschlossen.

Installationsoption „Benutzerdefiniert“

Im folgenden Abschnitt wird die Durchführung einer benutzerdefinierten Installation von Get-Services auf einem Server unter dem Windows-Betriebssystem erläutert.

Komponenten einer benutzerdefinierten Installation

Es folgt eine kurze Beschreibung der Komponenten, die bei einer benutzerdefinierten Installation von Get-Services installiert werden können:

Anwendungsoptionen

Get-Services-Komponente	Optionen
Webserver	<ul style="list-style-type: none"> ■ Apache 2.0.43 ■ IBM HTTP Server 1.3.19 ■ Microsoft IIS 5.0
Anwendungsserver	<ul style="list-style-type: none"> ■ Tomcat 4.1.12 ■ WebSphere 4.02 ■ WebLogic 6.1 SP3 ■ JRun 3.1
Java Development Kit	Java 2 SDK

Kommunikationsanschlüsse

Die bei einer benutzerdefinierten Installation von Get-Services verwendeten Kommunikationsanschlüsse hängen von den ausgewählten Anwendungskomponenten ab. Informationen zu den benötigten Kommunikationsanschlüssen finden Sie in der Dokumentation des

Webservers und des Anwendungsservers. Nach der Installation können Sie Get-Services zur Verwendung alternativer Kommunikationsanschlüsse konfigurieren, wenn die angegebenen Kommunikationsanschlüsse im lokalen Netzwerk bereits verwendet werden.

Standard-anschluss	Wird verwendet von	Alternativer Anschluss
8005	Verwaltung des Tomcat-Anwendungsservers	8015
8009	Worker-Datei des Tomcat-Anwendungsservers	8019
80	IIS-Webserver	8081

Get-Services auf Servern mit Oracle 9.2.0.1

Wenn Sie Get-Services auf einem Server mit Oracle 9.2.0.1 verwenden, kann ein Anschlusskonflikt bei den Kommunikationsanschlüssen 8009 und 8080 auftreten. Prüfen Sie in der Dokumentation des Webservers und des Anwendungsservers, ob diese Anschlüsse verwendet werden.

Wenn Sie Tomcat als Anwendungsserver einsetzen, tritt standardmäßig ein Anschlusskonflikt beim Anschluss 8009 auf. Es empfiehlt sich, Tomcat auf Servern mit Oracle 9.2.0.1 zur Verwendung eines anderen Kommunikationsanschlusses konfigurieren.

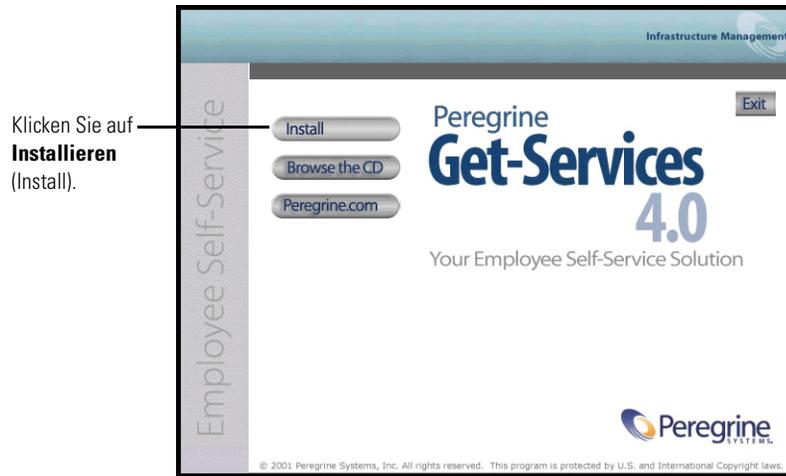
Verfahren für die benutzerdefinierte Installation

So führen Sie eine benutzerdefinierte Installation von Get-Services unter Windows durch:

- 1 Legen Sie die Get-Services-Installations-CD in das CD-ROM-Laufwerk des Computers ein. In der Regel wird das Installationsprogramm automatisch gestartet.

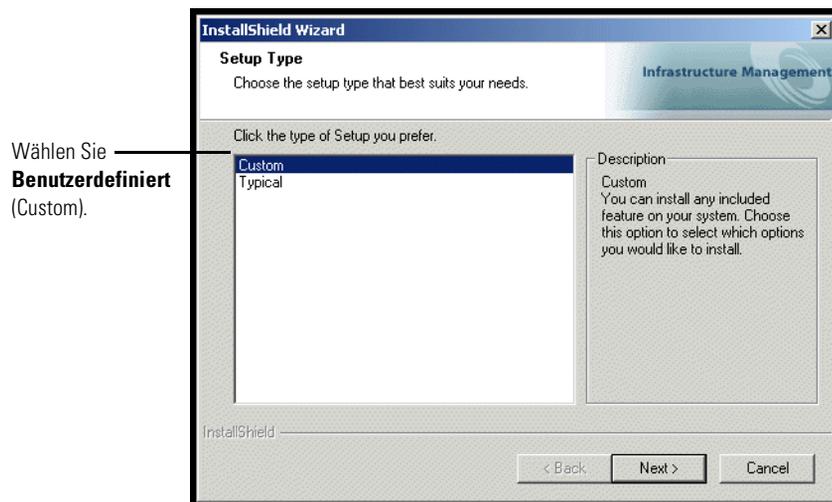
Wenn das Installationsprogramm nicht automatisch gestartet wird, verwenden Sie den Windows-Befehl **Start > Ausführen**, wechseln Sie zum CD-ROM-Laufwerk und öffnen Sie die Datei **Setup.exe**.

- 2 Klicken Sie im Hauptbildschirm des Installationsprogramms auf **Installieren** (Install).



In Statusmeldungen wird darauf hingewiesen, dass das Setup-Programm den InstallShield-Assistenten vorbereitet.

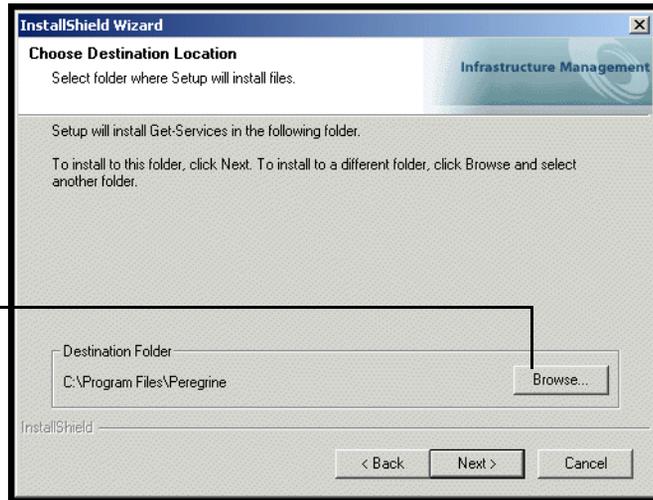
- 3 Wählen Sie im Bildschirm **Setup-Typ** (Setup Type) die Option **Benutzerdefiniert** (Custom) aus und klicken Sie dann auf **Weiter** (Next).



- 4 Wechseln Sie im Bildschirm **Zielpfad wählen** (Choose Destination Location) zu dem Ordner, in dem Get-Services installiert werden soll. Klicken Sie auf **Weiter** (Next), um fortzufahren.

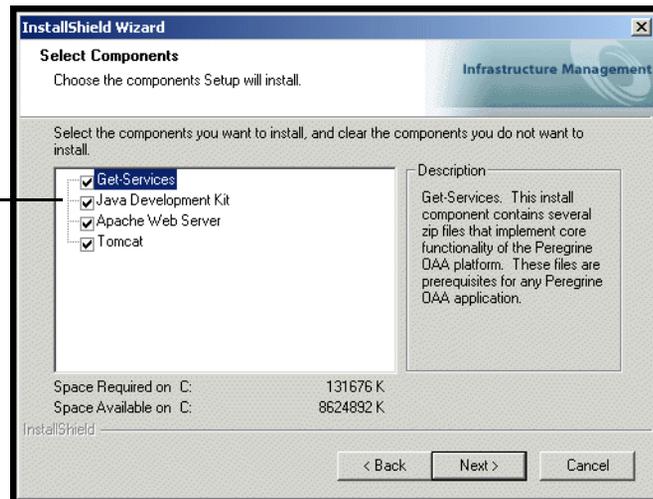
Der Standardpfad ist C:\Program Files\Peregrine.

Wählen Sie **Durchsuchen** (Browse), um einen anderen Pfad anzugeben.



- 5 Wählen Sie im Bildschirm **Komponenten auswählen** (Select Components) die auf dem Computer zu installierenden Komponenten aus und klicken Sie auf **Weiter** (Next).

Auswählen der zu installierenden Komponenten



Wichtig: Deaktivieren Sie die Kontrollkästchen neben den Komponenten, die Sie nicht installieren möchten.

Aktivieren bzw. deaktivieren Sie für eine Entwicklungsumgebung die Komponenten, die Sie manuell installieren oder an deren Stelle Sie alternative Software verwenden möchten. Deaktivieren Sie beispielsweise die Option **Apache-Webserver**, wenn Sie einen anderen Webserver verwenden möchten.

Wählen Sie für eine Produktionsumgebung die Komponenten aus, die auf diesem Computer ausgeführt werden sollen.

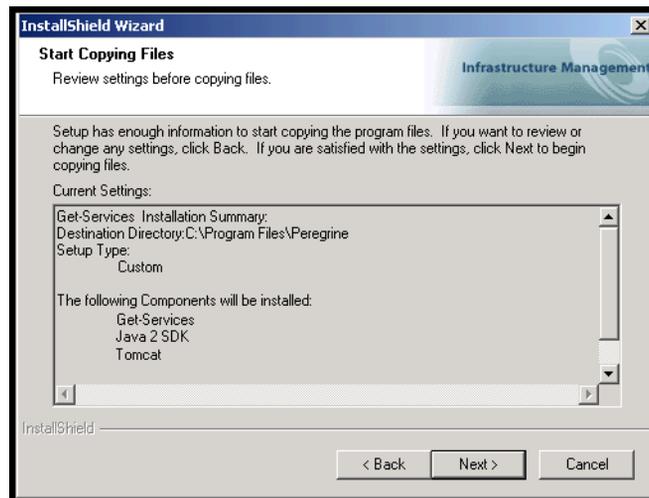
- **Get-Services.** Installiert die für Get-Services benötigten Programmdateien. Sie müssen die Get-Services-Dateien auf demselben Computer wie den primären Anwendungsserver installieren.
- **Apache-Webserver.** Get-Services benötigt einen Webserver, um Seiten bereitstellen zu können. Deaktivieren Sie dieses Kontrollkästchen, wenn der Apache-Webserver nicht installiert werden soll.
- **Java Development Kit.** Installiert das Java Development Kit (Java 2 SDK) von Sun Microsystems. Sie müssen diese Komponente auf allen Computern installieren, auf denen ein Anwendungsserver ausgeführt wird.
- **Tomcat.** Installiert den Tomcat-Anwendungsserver. Get-Services benötigt mindestens einen Anwendungsserver, um Java-Anwendungen verarbeiten zu können. Sie können zum Lastenausgleich mehrere Instanzen des Anwendungsservers installieren. Sie müssen diese Komponente auf demselben Computer wie die Get-Services-Dateien installieren.

Es werden Statusmeldungen zur Prüfung des Namens des Hostsystems angezeigt.

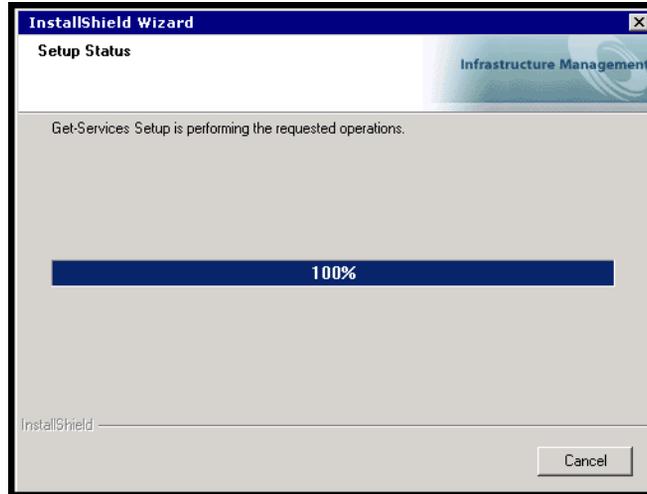
- Wählen Sie im Dialogfeld **Typ des Anwendungsservers auswählen** (Select Application Server Type) den Anwendungsserver aus, den Get-Services konfigurieren soll und klicken Sie dann auf **Weiter** (Next).



- Überprüfen Sie im Bildschirm **Kopiervorgang starten** (Start Copying Files) die Komponenten der benutzerdefinierten Installation. Klicken Sie auf **Zurück** (Back), wenn Sie Einstellungen überprüfen oder ändern möchten. Klicken Sie auf **Weiter** (Next), um mit der Installation fortzufahren.



Während des Installationsvorgangs wird der Bildschirm mit dem Setup-Status angezeigt.



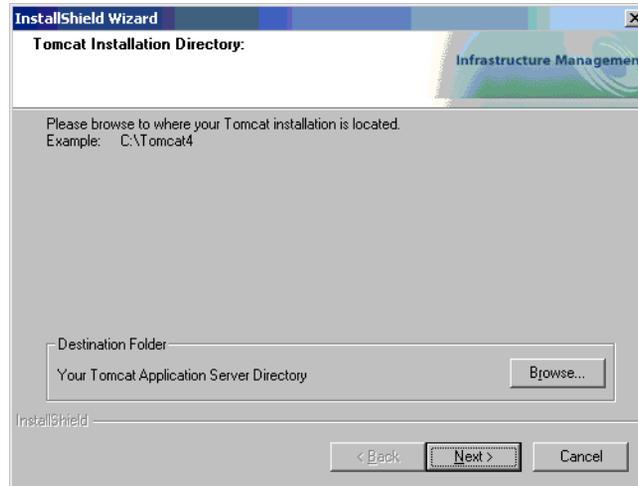
- 8 Wählen Sie **Ja (Yes)** im Bildschirm **CATALINA_OPTS einstellen** (Setting CATALINA_OPTS) aus, wenn Sie diesen Parameter für die Tomcat-Speichereinstellungen konfigurieren möchten, und klicken Sie dann auf **Weiter (Next)**.

Eine Statusmeldung weist darauf hin, dass die Get-Services-Pakete auf dem Server bereitgestellt werden.



Diese Phase kann einige Minuten in Anspruch nehmen.

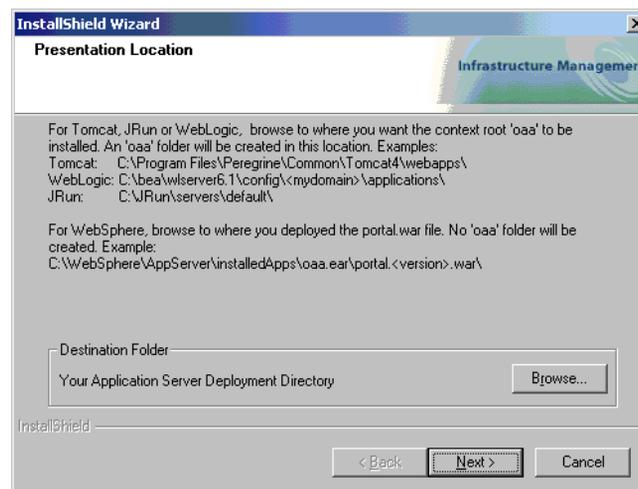
Wenn Sie Tomcat verwenden, wird das Fenster zur Angabe des Tomcat-Installationsverzeichnisses geöffnet.



- 9 Wählen Sie Ihr Installationsverzeichnis und klicken Sie auf Next (Weiter).

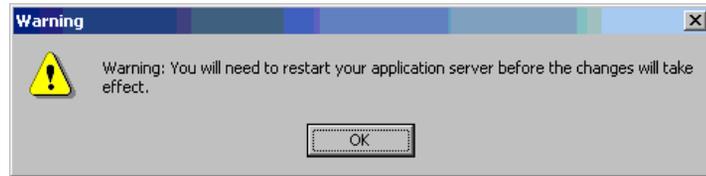
Hinweis: Wenn Sie Tomcat nicht verwenden, wird dieses Fenster nicht angezeigt. Wechseln Sie zu Schritt 10.

- 10 Wählen Sie im Dialogfeld Präsentationsverzeichnis (Presentation Location) ein Verzeichnis und klicken Sie dann auf Weiter (Next).



Beim Abschluss der Installation wird ein Dialogfeld mit dem Setup-Status angezeigt. Zu diesem Zeitpunkt werden die Dokumente kopiert.

Die folgende Warnmeldung wird ausgegeben:



- 11 Klicken Sie auf **OK**.
- 12 Klicken Sie im Bildschirm über den Abschluss des InstallShield-Assistenten auf **Fertig stellen** (Finish), um das Get-Services-Installationsprogramm zu beenden.



- 13 Wenn der Bildschirm des InstallShield-Assistenten geschlossen wurde, rufen Sie die Windows-Dienste auf, um sicherzustellen, dass der Webserver und die Anwendungsserver-Dienste gestartet wurden.

Wenn einer oder beide Dienste nicht gestartet wurden, finden Sie unter [Fehlerbehebung](#) Informationen zur Lösung möglicher Installationsprobleme.

Die benutzerdefinierte Installation von Get-Services auf einem Windows-Server ist hiermit abgeschlossen.

Deinstallieren von Get-Services

Führen Sie die folgenden Verfahren durch, um Get-Services von einem Windows-System zu deinstallieren:

- 1 Rufen Sie das Windows-Dienstprogramm **Software** auf.
- 2 Wählen Sie **Peregrine Portal 4.0** und klicken Sie auf **Ändern/Entfernen**. Eine Statusmeldung weist darauf hin, dass das Setup-Programm den InstallShield-Assistenten vorbereitet.
- 3 Wenn Get-Services-Dienste oder -Anwendungen ausgeführt werden, werden Sie aufgefordert, diese zu beenden. Klicken Sie auf **Weiter**, um fortzufahren.
- 4 In einer Meldung wird die folgende Frage angezeigt: **Möchten Sie gemeinsam genutzte Anwendungen und/oder Dienste wirklich anhalten?** Klicken Sie auf **Ja**, um fortzufahren.
Es werden Statusmeldungen angezeigt, die darauf hinweisen, dass die Dienste für Apache und Tomcat beendet werden.
- 5 Ein Dialogfeld zur Bestätigung der Installation wird geöffnet. Klicken Sie auf **Ja**, um Get-Services zu deinstallieren.

Wichtig: Erstellen Sie ein Backup aller Daten, die nicht gelöscht werden sollen, und setzen Sie den Vorgang anschließend fort.

- 6 Sind gemeinsam genutzte Dateien vorhanden, die bei der Deinstallation entfernt werden können, wird der Bildschirm für gemeinsam genutzte Dateien angezeigt.

Wenn WebSphere auf dem Computer installiert ist, fordert das Setup-Programm Sie auf, das Entfernen von sechs JAR-Dateien zu bestätigen. Klicken Sie auf **Nein** oder **Nein zu allen**, um diese JAR-Dateien beizubehalten.

Warnung: Entfernen Sie die gemeinsam genutzten JAR-Dateien nicht, denn sie werden für die Ausführung der Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere benötigt.

Wenn keine gemeinsam genutzten Dateien vorhanden sind, die entfernt werden können, weist eine Statusmeldung darauf hin, dass das Deinstallationsprogramm Dateien vom Computer entfernt.

- 7 Der Bildschirm über den Abschluss der Installationswartung wird geöffnet. Klicken Sie auf **Fertig stellen**, um die Deinstallation von Get-Services abzuschließen.

Testen der Installation

Gehen Sie folgendermaßen vor, um sicherzustellen, dass die Installation von Get-Services unter Windows ordnungsgemäß durchgeführt wurde.

So testen Sie die Get-Services-Installation:

- 1 Stellen Sie sicher, dass der Anwendungsserver und der Webserver gestartet wurden.
- 2 Öffnen Sie einen Web-Browser und geben Sie Folgendes im Adressfeld ein:

`http://<Servername>:<Anschluss>/oaa/admin.jsp`

Geben Sie für *<Servername>* den Namen des Servers ein, auf dem der Get-Services-Webserver installiert ist.

Geben Sie für *<Anschluss>* eine der folgenden Anschlussnummern ein:

Verwendeter Anwendungsserver	Anschlussnummer
WebSphere	9080
WebLogic	7001
JRun	80, kann im URL ausgelassen werden
Tomcat	80, kann im URL ausgelassen werden

Wurde die Konfiguration ordnungsgemäß durchgeführt, wird die Seite **Administratoranmeldung** geöffnet.

Ist dies nicht der Fall, finden Sie weitere Informationen unter *Fehlerbehebung*.

3 Installation unter AIX, Linux oder Solaris

KAPITEL

Dieses Kapitel behandelt folgende Themen:

- *Auswählen einer Installationsumgebung* auf Seite 90
- *Get-Services-Migration von früheren Versionen* auf Seite 93
- *Konfigurieren alternativer Anwendungsserver* auf Seite 96
- *Installationsoption „Typisch“* auf Seite 131
- *Installationsoption „Benutzerdefiniert“* auf Seite 138
- *Deinstallation – AIX, Linux oder Solaris* auf Seite 151

Auswählen einer Installationsumgebung

Sie können Get-Services in einer der folgenden Umgebungen installieren:

- Entwicklungsumgebung
- Produktionsumgebung

In der Entwicklungsumgebung von Get-Services können Sie die Funktionen des Produkts testen und die Installation ggf. anpassen, bevor Sie das Produkt in einer Produktionsumgebung bereitstellen. In einer Entwicklungsumgebung wird die gesamte für Get-Services benötigte Software auf einem Computersystem installiert.

Sie haben zwei Auswahlmöglichkeiten für die Entwicklungsumgebung:

- Typische Installation
 - Apache 2.0-Webserver
 - Get-Services auf Tomcat 4.1.12-Anwendungsserver
- Benutzerdefinierte Installation
 - Auswahl des Webservers
 - Auswahl des Anwendungsservers für Get-Services

Die Produktionsumgebung von Get-Services ist für eine optimale Serverleistung und Skalierbarkeit ausgelegt und ermöglicht die Bereitstellung aller Anpassungen. In einer Produktionsumgebung werden die einzelnen Komponenten von Get-Services auf verschiedenen Servern installiert, um eine besonders hohe Leistung zu gewährleisten.

Sie haben zwei Auswahlmöglichkeiten für die Produktionsumgebung:

- Typische Installation
 - Apache 2.0-Webserver
 - Get-Services auf mehreren Instanzen des Tomcat 4.1.12-Anwendungsservers
- Benutzerdefinierte Installation
 - Auswahl des Webservers
 - Auswahl des Anwendungsservers für Get-Services

Entwicklungsumgebung

In den folgenden Verfahren wird die Installation von Get-Services in einer Entwicklungsumgebung beschrieben.

So installieren Sie Get-Services in einer typischen Entwicklungsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank.
- Schritt 3** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Typisch**. Siehe *Installationsoption „Typisch“* auf Seite 131.
- Schritt 4** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer. Weitere Informationen hierzu finden Sie im Kapitel *Verwalten von ServiceCenter* in diesem Handbuch.

So installieren Sie Get-Services in einer benutzerdefinierten Produktionsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank.
- Schritt 3** Installieren Sie alternative Anwendungs- und Webserver.
- Schritt 4** Konfigurieren Sie den alternativen Anwendungsserver für Get-Services. Siehe *Konfigurieren alternativer Anwendungsserver* auf Seite 96.
- Schritt 5** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 138.
- Schritt 6** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer. Weitere Informationen hierzu finden Sie im Kapitel *Verwalten von ServiceCenter* in diesem Handbuch.

Produktionsumgebung

In den folgenden Verfahren wird die Installation von Get-Services in einer Produktionsumgebung beschrieben.

So installieren Sie Get-Services in einer typischen Produktionsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank auf einem separaten Server.
- Schritt 3** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Typisch**. Siehe *Installationsoption „Typisch“* auf Seite 131.
- Schritt 4** Konfigurieren Sie mehrere Instanzen von Tomcat für den Lastenausgleich auf dem Apache-Webserver. Weitere Informationen hierzu finden Sie im Kapitel *Lastenausgleich* in diesem Handbuch.
- Schritt 5** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer. Weitere Informationen hierzu finden Sie im Kapitel *Verwalten von ServiceCenter* in diesem Handbuch.

So installieren Sie Get-Services in einer benutzerdefinierten Produktionsumgebung:

- Schritt 1** Stellen Sie sicher, dass Sie über die gesamte benötigte Hard- und Software verfügen.
- Schritt 2** Installieren Sie die für Get-Services benötigte Backend-Datenbank.
- Schritt 3** Installieren Sie den alternativen Anwendungsserver und Webserver auf separaten Servern.
- Schritt 4** Konfigurieren Sie den alternativen Anwendungsserver für Get-Services. Siehe *Konfigurieren alternativer Anwendungsserver* auf Seite 96.
- Schritt 5** Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 138.

- Schritt 6** Konfigurieren Sie die Anwendungs- und Webserver für den Lastenausgleich. Weitere Informationen hierzu finden Sie im Kapitel *Lastenausgleich* in diesem Handbuch.
- Schritt 7** Konfigurieren Sie die Backend-Datenbank und erstellen Sie Get-Services-Benutzer. Weitere Informationen hierzu finden Sie im Kapitel *Verwalten von ServiceCenter* in diesem Handbuch.

Get-Services-Migration von früheren Versionen

Für die Migration von früheren Versionen von Get-It oder Get-Services in Get-Services 4.0 muss ein manueller Migrationsprozess für die Daten durchgeführt werden und die von Ihnen vorgenommenen Oberflächenanpassungen müssen neu erstellt werden. In den folgenden Schritten wird der Migrationsprozess beschrieben.

So führen Sie eine Migration früherer Versionen in Get-Services 4.0 durch:

- Schritt 1** Überprüfen Sie die in der früheren Version vorgenommenen Anpassungen und bestimmen Sie, welche Anpassungen in Get-Services 4.0 neu erstellt werden müssen. Siehe *Neuerstellen von Anpassungen in Get-Services 4.0* auf Seite 93.
- Schritt 2** Installieren Sie Get-Services 4.0 auf einem neuen System. Siehe *Auswählen einer Installationsumgebung* auf Seite 90.
- Schritt 3** Nehmen Sie die erforderlichen Änderungen an der Backend-Datenbank vor, die in Get-Services 4.0 migriert werden soll. Siehe *Konfigurieren einer vorhandenen Backend-Datenbank für Get-Services 4.0* auf Seite 96.

Neuerstellen von Anpassungen in Get-Services 4.0

Anpassungen, die in früheren Versionen implementiert wurden, können nicht direkt in Get-Services 4.0 migriert werden. Stattdessen müssen Sie die Änderungen mit den in Get-Services 4.0 verfügbaren Funktionen und Methoden neu erstellen.

In den folgenden Abschnitten wird beschrieben, wie die in früheren Versionen implementierten Anpassungen neu erstellt werden.

Keine Anpassungen

Wenn Sie keine Anpassungen für Get-Services erstellt haben, können Sie Get-Services 4.0 problemlos auf einem neuen System installieren und die Daten aus der vorhandenen Backend-Datenbank migrieren.

Benutzerdefinierte JSP-Dateien

In früheren Versionen mussten JSP-Dateien direkt geändert werden, um bestimmte Funktionen hinzuzufügen oder zu entfernen. In der folgenden Tabelle wird beschrieben, wie häufig an JSP-Dateien vorgenommene Änderungen neu erstellt werden.

JSP-Datei, Änderung	Neues Verfahren
Entfernen der Option zur Benutzerregistrierung von der Anmeldeseite	Aktivieren oder Deaktivieren der Option zur Benutzerregistrierung über die Seite Admin-Einstellungen
Entfernen der Option zur Kennwortänderung von der Anmeldeseite	Aktivieren oder Deaktivieren der Option zur Kennwortänderung über die Seite Admin-Einstellungen

Personalisierte Seiten

Get-Services 4.0 bietet eine Vielzahl weiterer Seiten, die Sie direkt über die Webschnittstelle personalisieren können. Wenn Sie in einer früheren Version personalisierte Seiten eingerichtet haben, müssen Sie diese personalisierten Seiten in Get-Services 4.0 mit DocExplorer neu erstellen.

Sie können die Personalisierung für folgende Aufgaben verwenden:

- Hinzufügen oder Entfernen von Feldern auf einer Seite
- Speichern personalisierter Suchergebnisse oder Details auf der Portalseite

Benutzerdefinierte Darstellungen, Formatvorlagen und Designs

Get-Services In 4.0 werden alle Bilder und Formatvorlagen für die Benutzeroberfläche in Designs zusammengefasst. Die Benutzer haben nicht mehr die Möglichkeit, einzelne Darstellungen und Formatvorlagen auszuwählen. Die neuen Designs umfassen Darstellungen (die wiederum Bilddateien, Rahmendefinitionen und Ebenendateien enthalten), CSS-Definitionen und XSL-Vorlagen.

Sie können benutzerdefinierte Designs aus einer früheren Version in Get-Services 4.0 kopieren, allerdings können dann aufgrund der neuen Bilder, CSS-Definitionen, Rahmendefinitionen und Ebenen Anzeigefehler auftreten. Daher sollten Sie alle benutzerdefinierten Designs auf der Basis der Get-Services 4.0-Version des klassischen Designs neu erstellen.

Alternative Anmeldeseiten und Authentifizierungsmethoden

Wenn Sie in einer früheren Version eine benutzerdefinierte Anmeldeseite oder eine alternative Authentifizierungsmethode verwendet haben, können Sie diese Anpassungen mit der aktualisierten Anweisung wieder verwenden oder neu erstellen. Informationen zu alternativen Sicherheitsmethoden finden Sie im *Get-Services-Verwaltungshandbuch*.

Mit einem früheren Tailoring Kit erstellte Anpassungen

Viele Anpassungen, für die in früheren Versionen ein Tailoring Kit benötigt wurde, können jetzt direkt über die Webschnittstelle von Get-Services erstellt werden. In der folgenden Tabelle wird beschrieben, wie bestimmte häufig mit einem Tailoring Kit vorgenommene Änderungen neu erstellt werden.

Tailoring Kit-Änderung	Neues Verfahren
Hinzufügen oder Entfernen von Feldern in einem Formular.	Hinzufügen oder Entfernen von Feldern über Personalisierung.
Hinzufügen neuer Sprachen oder Ländereinstellungen zur Get-Services-Oberfläche.	Direktes Erstellen und Bearbeiten von Dateien mit sprachspezifischen Zeichenfolgen. Sie haben auch die Möglichkeit, offiziell unterstützte Sprachpakete von Peregrine Systems zu erwerben.
Vornehmen von Änderungen am allgemeinen Paket, am Portalpaket oder am Peregrine Studio-Paket.	Diese Pakete können nicht mehr mit Tailoring-Verfahren geändert werden, die meisten Schnittstelleneinstellungen können jetzt jedoch über die Seite Admin-Einstellungen angepasst werden.
Vornehmen von Änderungen an Schemas oder serverseitigen ECMA-Skripten.	Überprüfen der neuen Funktionen und Entscheiden, ob die benutzerdefinierten Skripten und Schemas noch benötigt werden. Ist dies der Fall, müssen Sie sie in der aktuellen Version des Tailoring Kit von Get-Services neu erstellen.

Konfigurieren einer vorhandenen Backend-Datenbank für Get-Services 4.0

In der folgenden Tabelle werden die für die Datenmigration verfügbaren Optionen aufgeführt.

Migration von Get-Services 2.3 in Get-Services 4.0

Backend-Version	Erforderliche Migration
ServiceCenter 3.0	Aktualisieren auf ServiceCenter 4.x oder 5.0.x
ServiceCenter 4.x	Anwenden von Entladefdateien aus Get-Services 4.0 auf vorhandenes ServiceCenter 4.x
ServiceCenter 5.0.x	Anwenden von Entladefdateien aus Get-Services 4.0 auf vorhandenes ServiceCenter 5,0.x

Konfigurieren alternativer Anwendungsserver

Sie müssen einen Java-aktivierten Anwendungsserver zur Unterstützung der Peregrine-Webanwendungen installieren. Peregrine OAA unterstützt die folgenden alternativen Anwendungsserver:

- *Tomcat 4.1.12*
- *WebSphere 4.0.2*
- *WebLogic 6.1 SP3 oder SP4*
- *JRun 3.1*

Wenn Sie Get-Services mit der Installationsoption **Typisch** installieren, wird Tomcat 4.1.12 installiert und mit einem Apache 2.0-Webserver verbunden. Sie können Tomcat 4.1.12 jedoch auch bei Auswahl der Installationsoption **Benutzerdefiniert** verwenden.

Wichtig: Wenn Sie einen anderen Anwendungsserver als Tomcat 4.1.12 verwenden möchten, müssen Sie die Anwendungs- und Webserver konfigurieren, *bevor* Sie das Get-Services-Installationsprogramm ausführen.

In den folgenden Abschnitten finden Sie Anweisungen zur Konfiguration alternativer Anwendungsserver für Get-Services.

Tomcat 4.1.12

Wenn Sie bei der Installation von Get-Services den Apache-Webserver und den Tomcat-Anwendungsserver auswählen, konfiguriert das Get-Services-Installationsprogramm automatisch eine Verbindung zwischen Apache und Tomcat. Bei einer typischen Installation werden darüber hinaus die Get-Services-JSP-Dateien automatisch vom Installationsprogramm vorkompiliert. Wenn Sie eine benutzerdefinierte Installation durchführen, können Sie die Get-Services-JSP-Dateien beim Bereitstellen einer Produktionsumgebung manuell vorkompilieren.

Vorkompilieren von JSP-Dateien für eine Tomcat-Produktionsumgebung

Wenn Sie Tomcat 4.1.12 in einer Produktionsumgebung oder auf einem Mehrprozessorsystem verwenden möchten, müssen Sie die für Get-Services bereitgestellten JSP-Dateien vorkompilieren. Dieser zusätzliche Konfigurationsschritt ist aufgrund eines bekannten Tomcat-Fehlers erforderlich. Unter dem folgenden URL finden Sie detaillierte Informationen zu diesem Problem:

http://nagoya.apache.org/bugzilla/show_bug.cgi?id=14077

So kompilieren Sie JSP-Dateien für Tomcat 4.1.12 vor:

- 1 Halten Sie den Tomcat-Anwendungsserver an.
- 2 Öffnen Sie eine Eingabeaufforderung.
- 3 Wechseln Sie zum folgenden Verzeichnis: `usr/local/peregrine/samples`.
- 4 Kopieren Sie die folgenden Dateien an die angegebenen Orte:

Zu kopierende Datei	Zielverzeichnis
<code>excludes.txt</code>	<code><Tomcat>/webapps</code>
<code>builds.xml</code>	<code><Tomcat>/webapps</code>
<code>precompile.sh</code>	<code><Tomcat>/bin</code>

Geben Sie für `<Tomcat>` den Pfad zur Tomcat-Installation ein. Beispiel:
`/usr/local/peregrine/common/Tomcat4`

- 5 Geben Sie den folgenden Befehl ein:

Betriebssystem	Erforderlicher Befehl
UNIX ksh	export JASPER_HOME=<Tomcat-Verzeichnis>
UNIX csh	setenv JASPER_HOME <Tomcat-Verzeichnis>

Geben Sie für <Tomcat-Verzeichnis> den absoluten Pfad zur Tomcat-Installation ein.

- 6 Wechseln Sie zum Tomcat-Ordner bin.

Betriebssystem	Erforderlicher Befehl
UNIX, Solaris, AIX	cd \$JASPER_HOME/bin/samples/

- 7 Führen Sie die Batchdatei `precompile` aus.

Betriebssystem	Erforderlicher Befehl
UNIX, Solaris, AIX	<code>./precompile.sh <Name der Webanwendung> <Name der Tomcat-Instanz></code>

Geben Sie für <Name der Webanwendung> den Namen des Get-Services-Bereitstellungsordners ein. Wenn Sie den Standardordner `oaa` verwenden, müssen Sie diesen Namen nicht angeben.

Geben Sie für <Name der Tomcat-Instanz> den Namen der installierten Instanz von Tomcat an. Wenn Sie die Standardinstanz `Standalone` verwenden, müssen Sie diesen Namen nicht angeben.

Die Batchdatei zeigt den Status der Konvertierung an. Wenn die Konvertierung abgeschlossen ist, wird wieder die Eingabeaufforderung angezeigt.

- 8 Starten Sie den Tomcat-Anwendungsserver.

WebSphere 4.0.2

Führen Sie die folgenden Verfahren durch, um WebSphere zur Ausführung von Get-Services unter AIX, Linux oder Solaris zu konfigurieren.

So konfigurieren Sie WebSphere 4.0.2:

- Schritt 1** Installieren Sie WebSphere 4.0.2. Ihre Version von WebSphere 4.0.2 umfasst den IBM HTTP Server. *Installieren von WebSphere 4.0.2* auf Seite 99.
- Schritt 2** Stellen Sie die WAR-Datei des Portals für WebSphere bereit, so dass die für Get-Services benötigte Ordnerstruktur erstellt wird. Siehe *Bereitstellen der WAR-Datei des Portals für WebSphere* auf Seite 99.
- Schritt 3** Stellen Sie die JVM Java Heap-Größe für jede Instanz von WebSphere ein, die Get-Services ausführt. Siehe *Einstellen der Java Heap-Größe* auf Seite 102.
- Schritt 4** Erstellen Sie auf dem Webserver das virtuelle Verzeichnis, das für Get-Services verwendet werden soll. Siehe *Konfigurieren eines virtuellen Verzeichnisses für IBM HTTP Server* auf Seite 104.
- Schritt 5** Starten Sie das Get-Services-Installationsprogramm.

Wenn Sie einen WebSphere Portal Server oder einen WebSphere Translation Server einrichten möchten, finden Sie entsprechende Informationen unter *Installieren von WebSphere Portal Server* auf Seite 105 oder *Konfigurieren des WebSphere-Übersetzungsservers für Get-Services* auf Seite 115.

Installieren von WebSphere 4.0.2

Erwerben und installieren Sie IBM WebSphere 4.0.2. Ihre Version von WebSphere 4.0.2 umfasst den IBM HTTP Server.

Bereitstellen der WAR-Datei des Portals für WebSphere

Die WAR-Datei des Portals erstellt die Ordnerstruktur, die zur Bereitstellung von Get-Services auf dem Anwendungsserver benötigt wird. Wenn Sie diese Datei für WebSphere bereitgestellt haben, können Sie das Get-Services-Installationsprogramm ausführen.

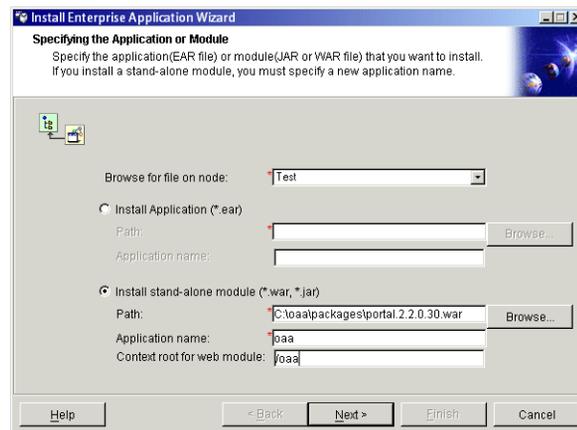
So stellen Sie die WAR-Datei des Portals für WebSphere bereit:

- 1 Stellen Sie sicher, dass der WebSphere Admin Server gestartet ist.
- 2 Öffnen Sie die Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere (/WebSphere/AppServer/bin/adminclient.sh).

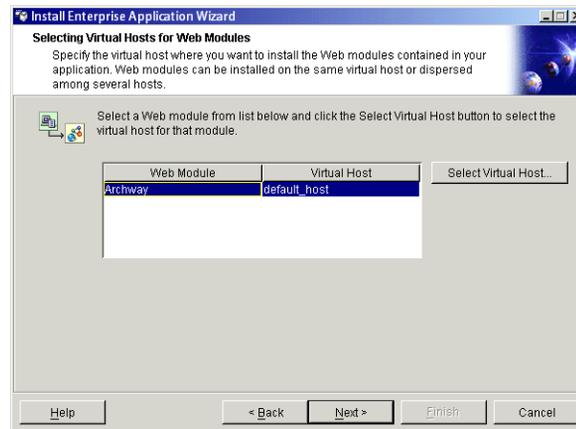
- 3 Klicken Sie im Menü auf der linken Seite der Konsole mit der rechten Maustaste auf **Enterprise Applications** (Unternehmensanwendungen) und wählen Sie **Install Enterprise Application** (Unternehmensanwendung installieren).
- 4 Führen Sie im angezeigten Bildschirm die folgenden Schritte durch:
 - a Wählen Sie **Install stand-alone module** (Standalone-Modul installieren).
 - b Geben Sie im Feld **Path** (Pfad) den Pfad zur Datei **Portal<Versionsnummer>.war** an. Standardmäßig lautet dieser Pfad **<CD-ROM-Laufwerk>/portal<Versionsnummer>.war**.
Wählen Sie für **<Versionsnummer>** die neueste verfügbare Version aus (4.0.0.44 oder höher).
 - c Geben Sie im Feld **Application Name** (Anwendungsname) **oaa** ein.
 - d Geben Sie im Feld **Context Root** (Kontextstamm) den Namen des für Get-Services zu verwendenden virtuellen Webserver-Verzeichnisses an.
Beispiel: **/oaa**.

Wichtig: Sie müssen ein virtuelles Webserver-Verzeichnis erstellen, das dem unter **Context Root** (Kontextstamm) eingegebenen Verzeichnis entspricht.

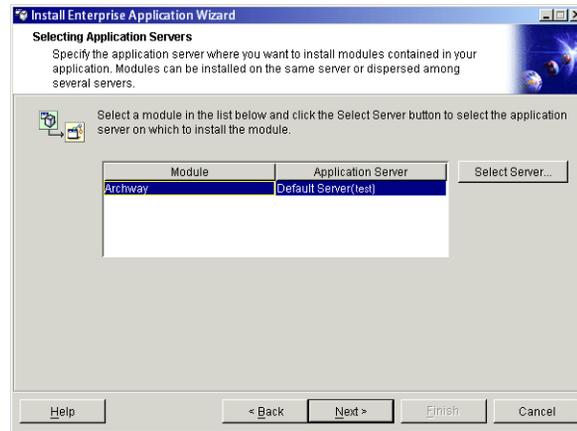
Im folgenden Bildschirm wird das ausgefüllte Formular angezeigt.



- 5 Klicken Sie auf **Next** (Weiter).
- 6 Klicken Sie in den folgenden Dialogfeldern auf **Next** (Weiter). Diese Dialogfelder werden nicht verwendet.
 - Mapping Users to Roles (Zuordnen von Benutzern zu Rollen)
 - Mapping EJB Run As Roles to Users (Zuordnen von EJB-Run-As-Rollen zu Benutzern)
 - Binding Enterprise Beans to JNDI Names (Binden von Unternehmens-Beans an JNDI-Namen)
 - Mapping EJB References to Enterprise Beans (Zuordnen von EJB-Referenzen zu Unternehmens-Beans)
 - Mapping Resource References to Resources (Zuordnen von Ressourcenreferenzen zu Referenzen)
 - Specifying the Default Datasource (Angabe der standardmäßigen Datenquelle)
 - Specifying Data Sources for Individual CMP Beans (Angabe der Datenquellen für einzelne CMP-Beans)
- 7 Wählen Sie im Dialogfeld **Selecting Virtual Hosts for Web Modules** (Auswählen virtueller Hosts für Webmodule) die zu verwendende Serverinstanz von WebSphere aus und klicken Sie dann auf **Next** (Weiter).



- 8 Wählen Sie im Dialogfeld **Selecting Application Servers** (Auswählen der Anwendungsserver) die zu verwendende Serverinstanz von WebSphere aus und klicken Sie dann auf **Next** (Weiter).



- 9 Klicken Sie im angezeigten Dialogfeld auf **Finish** (Fertig stellen).

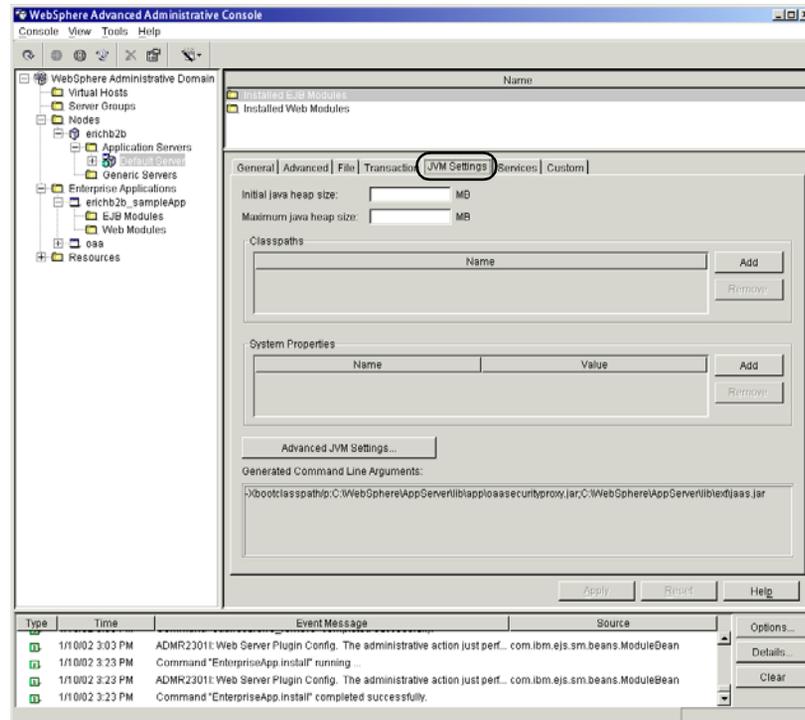
Einstellen der Java Heap-Größe

Sie können die Größe des für die Instanzen des Anwendungsservers verfügbaren Speichers konfigurieren. In den folgenden Anweisungen wird davon ausgegangen, dass Sie nur eine Instanz von WebSphere verwenden. Wenn Sie zum Lastenausgleich mit mehreren Instanzen von WebSphere arbeiten, müssen Sie die Heap-Größe entsprechend anpassen.

So stellen Sie die Java Heap-Größe ein:

- 1 Stellen Sie sicher, dass der WebSphere Admin Server gestartet ist.
- 2 Öffnen Sie die Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere: **Start > Programme > IBM WebSphere > Application Server (Anwendungsserver) > Administrator's Console (Verwaltungskonsole)**.
- 3 Klicken Sie auf **Nodes (Knoten) > <Systemname> > Application Servers (Anwendungsserver) > <Name des Anwendungsservers>**.

Die Seite für Servereinstellungen wird geöffnet.



- 4 Klicken Sie auf das Register **JVM Settings** (JVM-Einstellungen).
- 5 Nehmen Sie die folgenden JVM-Einstellungen vor:
 - a **Initial java heap size** (Anfängliche Java Heap-Größe). Geben Sie 60 ein.
 - b **Maximum java heap size** (Maximale Java Heap-Größe). Geben Sie den gewünschten Wert für den Heap-Speicher ein. Dieser Wert sollte zwischen 225 MB und 512 MB liegen.

Hinweis: Stellen Sie sicher, dass der Höchstwert für die Heap-Größe kleiner ist als der für die Anwendungsserver verfügbare freie Arbeitsspeicher. Andernfalls lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt. Für die meisten Systeme reicht die Einstellung von 256 MB aus.

Konfigurieren eines virtuellen Verzeichnisses für IBM HTTP Server

Sie müssen auf dem Webserver ein virtuelles Verzeichnis für Get-Services konfigurieren. In den folgenden Anweisungen wird davon ausgegangen, dass Sie den in WebSphere integrierten Webserver verwenden, d. h. IBM HTTP Server. Wenn Sie einen anderen Webserver verwenden, finden Sie in der Dokumentation dieses Webserver Informationen zum Erstellen eines virtuellen Verzeichnisses.

So konfigurieren Sie IBM HTTP Server für Get-Services:

- 1 Halten Sie den IBM HTTP Server an.
- 2 Öffnen Sie die Datei `httpd.conf` in einem Texteditor. Standardmäßig befindet sich diese Datei unter:

```
<Stammverzeichnis>/usr/HTTPServer/conf
```

- 3 Fügen Sie die folgende Zeile am Ende der Datei ein:

```
Alias /oaa/  
"<Stammverzeichnis>/WebSphere/AppServer/installedApps/oaa.ear/portal.  
<version>.war/"
```

Geben Sie für `<Stammverzeichnis>` das Stammverzeichnis des Systems ein.

Geben Sie für `<Version>` die Versionsnummer der installierten WAR-Datei ein.

Wichtig: Der hier angegebene Name des virtuellen Verzeichnisses muss dem in WebSphere unter **Context Root** (Kontextstamm) festgelegten Verzeichnis entsprechen.

- 4 Speichern Sie die Datei.
- 5 Starten Sie den IBM HTTP Server.

Ausführen des Get-Services-Installationsprogramms

Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 138.

Wenn Sie einen WebSphere Portal Server oder einen WebSphere Translation Server einrichten möchten, finden Sie entsprechende Informationen unter *Installieren von WebSphere Portal Server* auf Seite 105 oder *Konfigurieren des WebSphere-Übersetzungsservers für Get-Services* auf Seite 115.

Installieren von WebSphere Portal Server

Sie können Get-Services so einrichten, dass es auf einem WebSphere Portal Server in einer der folgenden zwei Konfigurationen angezeigt wird:

- Alle Komponenten von Get-Services und WebSphere werden auf einem einzigen System ausgeführt. Siehe *Empfohlene Konfiguration für WebSphere Portal Server* auf Seite 105.
- Die Komponenten von Get-Services werden auf einem System ausgeführt und die Komponenten von WebSphere auf einem anderen. Siehe *Alternative Konfiguration für WebSphere Portal Server* auf Seite 107.

Wichtig: Für beide Konfigurationen müssen Sie zunächst WebSphere Portal Server installieren. Weitere Informationen finden Sie in der Dokumentation zu WebSphere Portal Server.

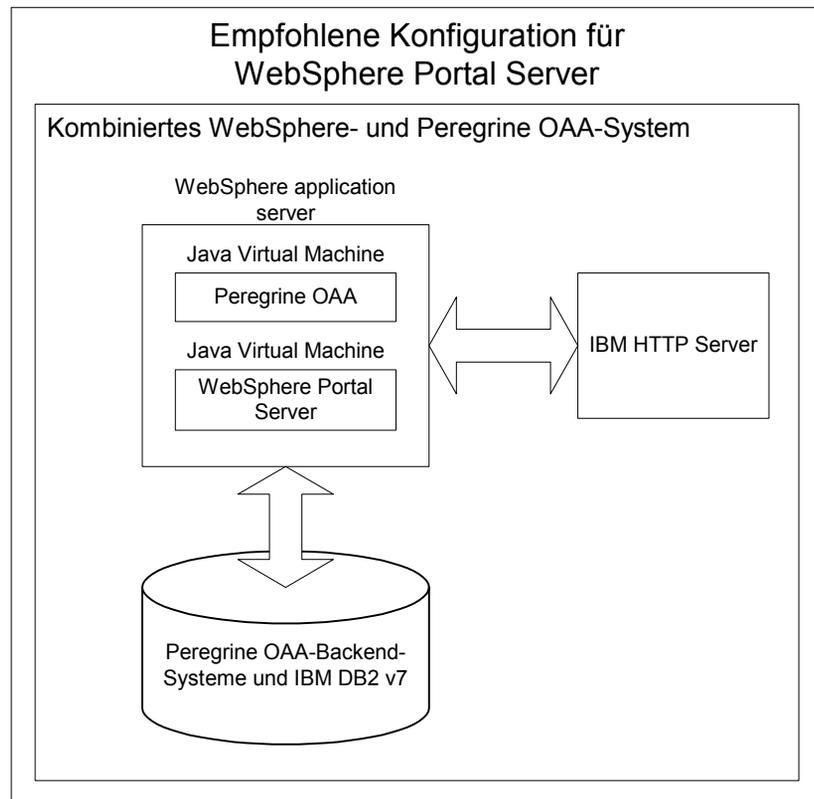
Empfohlene Konfiguration für WebSphere Portal Server

Führen Sie die folgenden Schritte durch, um Get-Services gemäß der für WebSphere Portal Server empfohlenen Konfiguration einzurichten:

- Schritt 1** Prüfen Sie die Installationsanforderungen für WebSphere Portal Server. Siehe *Installationsanforderungen für WebSphere Portal Server* auf Seite 109.
- Schritt 2** Erstellen Sie eine Get-Services-WAR-Datei mit den Portalkomponenten, die in WebSphere Portal Server angezeigt werden sollen. Siehe *Erstellen einer Get-Services-WAR-Datei* auf Seite 109.
- Schritt 3** Melden Sie sich am Get-Services-Server an und halten Sie den WebSphere-Anwendungsserver an. Siehe *Anhalten des WebSphere-Anwendungsservers* auf Seite 110.
- Schritt 4** Bearbeiten Sie die Datei `archway.xml`, um die HTTP-Authentifizierungsmethode von Basisauthentifizierung in alternative Authentifizierung zu ändern. Siehe *Ändern der Datei „archway.xml“* auf Seite 110.
- Schritt 5** Ändern Sie die Datei `web.xml`, um das AuthController-Servlet zu aktivieren. Siehe *Ändern der Datei „web.xml“* auf Seite 111.
- Schritt 6** Setzen Sie in der Datei `ibm-web-ext.xmi` den Parameter `fileServingEnabled`. Siehe *Ändern der Datei „ibm-web-ext.xmi“* auf Seite 112.

- Schritt 7** Starten Sie den WebSphere-Anwendungsserver. Siehe *Starten des WebSphere-Anwendungsservers* auf Seite 112.
- Schritt 8** Stellen Sie die Get-Services-WAR-Datei für WebSphere Portal Server bereit. Siehe *Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server* auf Seite 113.
- Schritt 9** Erstellen Sie in WebSphere Portal Server Orte und Seiten für die Anzeige von Get-Services-Portlets. Siehe *Konfigurieren von Orten und Seiten in WebSphere Portal Server* auf Seite 113.
- Schritt 10** Aktivieren Sie die Bearbeitungsrechte für Get-Services-Portlets. Siehe *Aktivieren der Bearbeitungsrechte für Get-Services-Portlets* auf Seite 114.

Wenn Sie diese Schritte durchgeführt haben, weist die Installation die folgende Konfiguration auf:



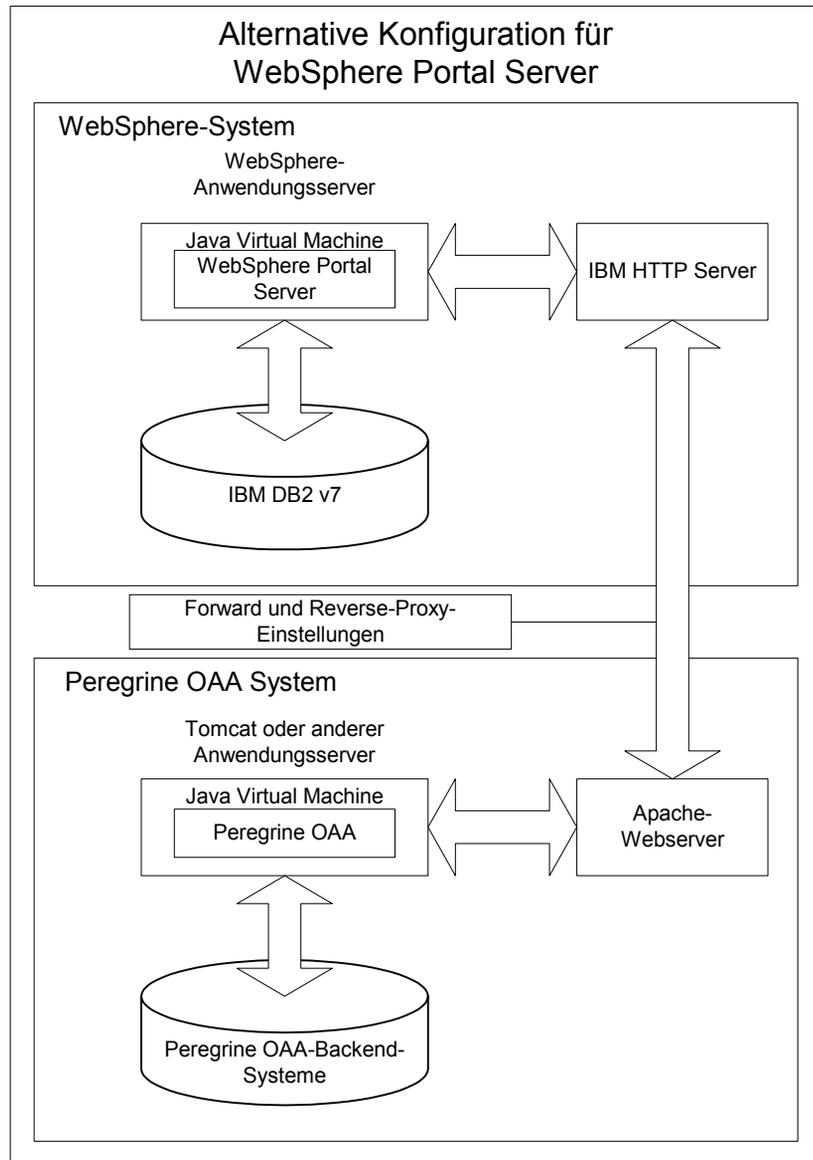
Alternative Konfiguration für WebSphere Portal Server

Führen Sie die folgenden Schritte durch, um Get-Services gemäß der alternativen Konfiguration für WebSphere Portal Server einzurichten:

- Schritt 1** Prüfen Sie die Installationsanforderungen für WebSphere Portal Server. Siehe *Installationsanforderungen für WebSphere Portal Server* auf Seite 109.
- Schritt 2** Erstellen Sie eine Get-Services-WAR-Datei mit den Portalkomponenten, die in WebSphere Portal Server angezeigt werden sollen. Siehe *Erstellen einer Get-Services-WAR-Datei* auf Seite 109.
- Schritt 3** Melden Sie sich am Get-Services-Server an und halten Sie den WebSphere-Anwendungsserver an. Siehe *Anhalten des WebSphere-Anwendungsservers* auf Seite 110.
- Schritt 4** Bearbeiten Sie die Datei `archway.xml`, um die HTTP-Authentifizierungsmethode von Basisauthentifizierung in alternative Authentifizierung zu ändern. Siehe *Ändern der Datei „archway.xml“* auf Seite 110.
- Schritt 5** Ändern Sie die Datei `web.xml`, um das AuthController-Servlet zu aktivieren. Siehe *Ändern der Datei „web.xml“* auf Seite 111.
- Schritt 6** Setzen Sie in der Datei `ibm-web-ext.xmi` den Parameter `fileServingEnabled`. Siehe *Ändern der Datei „ibm-web-ext.xmi“* auf Seite 112.
- Schritt 7** Ändern Sie `setDomain.js`, so dass die Funktion `SetDomain` aufgerufen wird. Siehe *Ändern der Datei „setDomain.js“* auf Seite 112.
- Schritt 8** Starten Sie den WebSphere-Anwendungsserver. Siehe *Starten des WebSphere-Anwendungsservers* auf Seite 112.
- Schritt 9** Stellen Sie die Get-Services-WAR-Datei für WebSphere Portal Server bereit. Siehe *Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server* auf Seite 113.
- Schritt 10** Erstellen Sie in WebSphere Portal Server Orte und Seiten für die Anzeige von Get-Services-Portlets. Siehe *Konfigurieren von Orten und Seiten in WebSphere Portal Server* auf Seite 113.
- Schritt 11** Aktivieren Sie die Bearbeitungsrechte für Get-Services-Portlets. Siehe *Aktivieren der Bearbeitungsrechte für Get-Services-Portlets* auf Seite 114.

Schritt 12 Ändern Sie die Datei `httpd.conf` von IBM HTTP Server, um Forward- und Reverse-Proxy-URLs hinzuzufügen. Siehe *Ändern der Datei „httpd.conf“ für IBM HTTP Server* auf Seite 115.

Wenn Sie diese Schritte durchgeführt haben, weist die Installation die folgende Konfiguration auf:



Installationsanforderungen für WebSphere Portal Server

Für die empfohlene Konfiguration von WebSphere Portal Server müssen die folgenden Komponenten auf demselben Server installiert sein:

- WebSphere-Anwendungsserver 4.0.2
- IBM HTTP Server 1.3.19
- IBM DB2 v7-Datenbankservers
- WebSphere Portal Server
- Eine benutzerdefinierte Installation von Get-Services, bei der WebSphere als Anwendungsserver ausgewählt ist

Für die alternative Konfiguration von WebSphere Portal Server müssen die folgenden Komponenten auf mindestens zwei Servern installiert sein:

- Server 1
 - WebSphere-Anwendungsserver 4.0.2
 - IBM HTTP Server 1.3.19
 - IBM DB2 v7-Datenbankservers
 - WebSphere Portal Server
- Server 2
 - Get-Services-kompatibler Anwendungsserver
 - Webserver
 - Backend-Datenbank für Get-Services
 - Eine Installation von Get-Services

Erstellen einer Get-Services-WAR-Datei

Um Get-Services in WebSphere Portal Server anzuzeigen, müssen Sie die Portalkomponenten von Get-Services zunächst als WAR-Datei exportieren. Anschließend können Sie diese WAR-Datei in WebSphere Portal Server importieren und die Portalkomponenten auswählen, die als WebSphere Portal Server-Portlets angezeigt werden sollen.

So erstellen Sie eine Get-Services-WAR-Datei:

- 1 Melden Sie sich bei der Get-Services-Verwaltungsseite an (`admin.jsp`).
- 2 Klicken Sie auf **Integration des IBM-Portals WebSphere**.

- 3 Geben Sie die folgenden Konfigurationsdaten ein:
 - a **Ausgangspfad.** Geben Sie den vollständigen Pfad zur Datei `WebSphere.war` im Ordner `package` von Get-Services an. Standardmäßig handelt es sich hierbei um den folgenden Ordner:


```
<WebSphere>/oaa/packages
```
 - b **Zielpfad.** Geben Sie den vollständigen Pfad und den Dateinamen an, der für die erstellte Get-Services-WAR-Datei verwendet werden soll.
 - c **Standard-URL.** Geben Sie den vollständigen URL des Get-Services-Bereitstellungsverzeichnis ein. Standardmäßig handelt es sich hierbei um den folgenden URL:


```
http://<Server>:<Anschluss>/oaa/servlet/basicauth
```
- 4 Klicken Sie auf **WAR-Datei generieren.**
Get-Services erstellt eine neue WAR-Datei mit dem in Schritt 3 unter **Zielpfad** angegebenen Pfad und Namen.

Anhalten des WebSphere-Anwendungsservers

Um mit der Konfiguration fortzufahren, müssen Sie sich beim Get-Services-Server anmelden und den WebSphere-Anwendungsserver anhalten.

So halten Sie den WebSphere-Anwendungsserver an:

- 1 Melden Sie sich beim Get-Services-Server an.
- 2 Fahren Sie den WebSphere-Anwendungsserver herunter.

Ändern der Datei „archway.xml“

Um die Anmeldung über WebSphere Portal Server zu ermöglichen, müssen Sie Get-Services zur Verwendung einer alternativen HTTP-Authentifizierungsmethode konfigurieren.

So ändern Sie die Datei „archway.xml“:

- 1 Öffnen Sie die Datei `archway.xml` in einem Texteditor. Sie finden die Datei unter:


```
<Anwendungsserver>/webapps/oaa/WEB-INF/default.
```
- 2 Bearbeiten Sie die folgende Zeile:


```
<httpauthclass ...>HttpBasicAuthenticationManager</httpauthclass>
```

- 3 Ändern Sie den Wert `HttpBasicAuthenticationManager` in `HttpAlternateAuthenticationManager`.
- 4 Speichern Sie die Datei.

Ändern der Datei „web.xml“

Sie müssen das `AuthController`-Servlet zur Bereitstellung eines Proxy für die HTTP-Basisauthentifizierung aktivieren.

So ändern Sie die Datei „web.xml“:

- 1 Öffnen Sie die Datei `web.xml` in einem Texteditor. Diese Datei befindet sich unter:

```
<Anwendungsserver>\webapps\oaa\WEB-INF.
```

- 2 Suchen Sie die folgende Zeile:

```
<!-- Uncomment to add support for http basic authentication proxy
```

- 3 Verschieben Sie das Tag für das Kommentar-Ende (`-->`) vom Ende der Servlet-Definition zum Kommentar am Anfang der Servlet-Definition.

Die neue Servlet-Definition sollte wie folgt lauten:

```
<!-- Uncomment to add support for http basic authentication proxy-->
<!--
  <servlet>
    <servlet-name>AuthController</servlet-name>
    <display-name>AuthController</display-name>
    <description>Ein Controller (Decorator)-Servlet, mit dem der
    konfigurierbare Autorisierungsschutz für jede beliebige Ressource
    aktiviert werden kann.</description>

    <servlet-class>com.peregrine.oaa.archway.AuthControllerServlet</servl
    et-class>
    <load-on-startup>2</load-on-startup>
  </servlet>

  <servlet-mapping>
    <servlet-name>AuthController</servlet-name>
    <url-pattern>/servlet/basicauth/*</url-pattern>
  </servlet-mapping>
  <servlet-mapping>
    <servlet-name>AuthController</servlet-name>
    <url-pattern>/servlet/auth/*</url-pattern>
  </servlet-mapping>
```

- 4 Speichern Sie die Datei.

Ändern der Datei „ibm-web-ext.xmi“

Sie müssen den Parameter `fileServingEnabled` auf `true` setzen, um statische Inhalte verarbeiten zu können.

So ändern Sie die Datei „ibm-web-ext.xmi“:

- 1 Öffnen Sie die Datei `ibm-web-ext.xmi` in einem Texteditor. Der Standardpfad lautet:

```
c:\WebSphere\AppServer\installedApps\getit.ear\getit.war\WEB-INF
```

- 2 Setzen Sie den Parameter `fileServingEnabled` auf `true`.

```
fileServingEnabled="true"
```

- 3 Speichern Sie die Datei.

Ändern der Datei „setDomain.js“

Um die alternative Konfiguration von WebSphere Portal Server verwenden zu können, müssen Sie die Funktion `setDomain` aktivieren.

Hinweis: Wenn Sie WebSphere Portal Server in der empfohlenen Konfiguration einrichten, können Sie diese Anweisungen überspringen.

So ändern Sie die Datei „setDomain.js“:

- 1 Melden Sie sich beim Get-Services-Server an.
- 2 Fahren Sie den Anwendungsserver herunter.
- 3 Öffnen Sie die Datei `setDomain.js` in einem Texteditor. Diese Datei befindet sich unter:

```
<Anwendungsserver>/webapps/oa/js.
```

- 4 Fügen Sie die folgende Zeile am Ende der Datei ein:

```
setDomain();
```

- 5 Speichern Sie die Datei.

Starten des WebSphere-Anwendungsservers

Um mit der Konfiguration fortzufahren, müssen Sie den WebSphere-Anwendungsserver neu starten.

Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server

Nach dem Bereitstellen der Get-Services-WAR-Datei für WebSphere Portal Server können Sie die anzuzeigenden Portlets, die Einstellungen für die Anzeige und die Zugriffsrechte für jedes Portlet konfigurieren.

Ausführliche Anweisungen finden Sie in der Dokumentation zu WebSphere Portal Server.

So stellen Sie die Get-Services-WAR-Datei bereit:

- 1 Melden Sie sich als `wpsadmin` oder als ein anderer Benutzer mit Verwaltungsrechten bei WebSphere Portal an.
- 2 Wählen Sie im Menü **Places** (Orte) den Befehl **Portal Administration** (Portalverwaltung) aus.
- 3 Klicken Sie auf **Portlets** > **Install Portlets** (Portlets installieren).
- 4 Klicken Sie auf **Browse** (Durchsuchen) und navigieren Sie zu dem Zielpfad, den Sie beim Erstellen der Get-Services-WAR-Datei eingegeben haben.
- 5 Klicken Sie auf **Next** (Weiter), um die Get-Services-WAR-Datei zu laden. WebSphere Portal Server zeigt eine Liste der zu installierenden Portlets an.
- 6 Klicken Sie auf **Install** (Installieren).

WebSphere Portal Server installiert die Portlets und zeigt die folgende Meldung an: **Portlets successfully installed** (Portlets wurden erfolgreich installiert).

Konfigurieren von Orten und Seiten in WebSphere Portal Server

Sie können Get-Services-Portlets an allen Orten und auf allen Seiten bereitstellen, die den folgenden Anforderungen entsprechen.

Orte

Die Orte in WebSphere Portal Server müssen die folgende Eigenschaft aufweisen:

- HTML muss unterstützt werden

Seiten

Die Seiten in WebSphere Portal Server müssen die folgenden Eigenschaften aufweisen:

- HTML muss unterstützt werden

- Die folgende Einstellung muss für die Seite ausgewählt werden: **Allow all portlets that a user can access** (Alle Portlets zulassen, auf die ein Benutzer zugreifen kann)
- Der Benutzergruppe **All authenticated users** (Alle authentifizierten Benutzer) muss die minimale Berechtigung zum Bearbeiten aller auf einer Seite angezeigten Get-Services-Portlets zugewiesen werden

Aktivieren der Bearbeitungsrechte für Get-Services-Portlets

Benutzer von WebSphere Portal Server benötigen Bearbeitungsrechte für die Get-Services-Portlets, um sie anpassen und ihrer Portalseite hinzufügen zu können.

So aktivieren Sie Bearbeitungsrechte für Get-Services-Portlets:

- 1 Melden Sie sich als **wpsadmin** oder als ein anderer Benutzer mit Verwaltungsrechten bei WebSphere Portal an.
- 2 Wählen Sie im Menü **Places** (Orte) den Befehl **Portal Administration** (Portalverwaltung) aus.
- 3 Klicken Sie auf **Security** (Sicherheit) > **Access Control List** (Zugriffskontrollliste).
- 4 Wählen Sie in der Dropdown-Liste für die Option **Special groups** (Spezielle Gruppen) den Eintrag **All authenticated users** (Alle authentifizierten Benutzer).
- 5 Wählen Sie in der Dropdown-Liste **Select the objects for the permissions** (Auswählen der Objekte für die Berechtigungen) den Eintrag **portlet applications** (Portlet-Anwendungen).
- 6 Wählen Sie die Option **Search on** (Suchen nach) und geben Sie dann im Feld **Name contains** (Name enthält) **Peregrine** ein.
- 7 Klicken Sie auf **Go** (Los).
WebSphere Portal Server zeigt eine Liste der Portlets an, deren Name die Zeichenfolge **Peregrine** enthält.
- 8 Klicken Sie in der Spalte **Edit** (Bearbeiten) am Ende der Tabelle auf **Select All** (Alle auswählen).
- 9 Klicken Sie auf **Speichern**.
Die Benutzer können Get-Services-Portlets jetzt über WebSphere Portal Server anzeigen und bearbeiten.

Ändern der Datei „httpd.conf“ für IBM HTTP Server

Um die alternative Konfiguration von WebSphere Portal Server verwenden zu können, müssen Sie die von IBM HTTP Server verwendete Datei `httpd.conf` ändern, so dass der Remote-Instanz von Get-Services die Forward- und Reverse-Proxy-URLs hinzugefügt werden.

Hinweis: Wenn Sie WebSphere Portal Server in der empfohlenen Konfiguration einrichten, können Sie diese Anweisungen überspringen.

So ändern Sie die Datei „httpd.conf“ für IBM HTTP Server:

- 1 Melden Sie sich beim Get-Services-Server an.
- 2 Halten Sie den IBM HTTP Server an.
- 3 Öffnen Sie die Datei `httpd.conf` in einem Texteditor. Diese Datei befindet sich unter:

```
<Stammverzeichnis>/usr/HTTPServer/conf
```

- 4 Fügen Sie die folgenden Zeilen am Ende der Datei ein:

```
ProxyPass /<aaa-Stammverzeichnis>/ http://<Server>:<Anschluss>/
<aaa-Stammverzeichnis>/servlet/basicauth/
ProxyPassReverse /<aaa-Stammverzeichnis>/
http://<Server>:<Anschluss>/
<aaa-Stammverzeichnis>/servlet/basicauth/
```

Geben Sie für `<OAA-Stammverzeichnis>` den Namen des von IBM HTTP Server verwendeten virtuellen OAA-Verzeichnisses ein. Standardmäßig lautet der Name dieses virtuellen Verzeichnisses `aaa`.

Geben Sie für `<Server>:<Anschluss>` den Servernamen und die Nummer des Kommunikationsanschlusses der Get-Services-Installation ein.

- 5 Speichern Sie die Datei.

Konfigurieren des WebSphere-Übersetzungsservers für Get-Services

Sie können Get-Services zur Verwendung eines WebSphere Translation Server konfigurieren, so dass Sie in Echtzeit auf Übersetzungen angezeigter Textdaten zugreifen können.

So konfigurieren Sie WebSphere Translation Server für Get-Services:

- Schritt 1** Kopieren Sie die Datei `wts.jar` in den Get-Services-Bereitstellungsordner. Siehe *Kopieren von „wts.jar“ in den Get-Services-Bereitstellungsordner* auf Seite 116.
- Schritt 2** Konfigurieren Sie Get-Services zur Verwendung des WebSphere Translation Server. Siehe *Konfigurieren des WebSphere-Übersetzungsservers für Get-Services* auf Seite 115.

Kopieren von „wts.jar“ in den Get-Services-Bereitstellungsordner

Nachfolgend wird beschrieben, an welchem Speicherort sich die Datei `wts.jar` befindet und in welchen Ordner sie kopiert werden muss.

So kopieren Sie „wts.jar“ in den Get-Services-Bereitstellungsordner:

- 1 Fahren Sie den Anwendungsserver herunter.
- 2 Wechseln Sie zum Installationsordner des WebSphere Translation Server.
- 3 Kopieren Sie in diesem Ordner die Datei `wts.jar`.
- 4 Fügen Sie die Datei `wts.jar` in den Get-Services-Bereitstellungsordner ein. Dieser Ordner befindet sich unter:
<Anwendungsserver-Installation>/WEB-INF/lib
- 5 Starten Sie den Anwendungsserver neu.

Konfigurieren von Get-Services zur Verwendung des WebSphere Translation Server

In den folgenden Anweisungen wird beschrieben, wie Get-Services zur Verwendung des WebSphere Translation Server konfiguriert wird.

So konfigurieren Sie Get-Services zur Verwendung des WebSphere Translation Server:

- 1 Melden Sie sich bei der Get-Services-Verwaltungsseite an (`admin.jsp`).
- 2 Klicken Sie auf das Register **Einstellungen** > **Allgemein**.

Die Seite Admin-Einstellungen wird geöffnet.

Portal		MobileServer		AssetCenter		Allgemein		Service Desk		Portal-DB		Designs		Webanwendung	
Protokollieren		ServiceCenter		Get-Resources		XSL		E-Mail							
Allgemeines Backend:								Adapterzielname zur Unterstützung allgemeiner Benutzervorgänge.							
portalDB															
Liste der Ziel-Allianzen:								Gibt eine Liste mit Ziel-Allianzen an, die von den Webanwendungen in diesem Paket verwendet werden. Die Einträge sind durch ein Semikolon getrennt.							
weblication@mail															
Admin-Name:								Anmeldungsname für den Administrator. Diese Anmeldung ermöglicht den Zugriff auf die administrativen Funktionen ohne Authentifizierung gegen die Backendadapter.							
Admin															
Admin-Kennwort:								Admin-Kennwort							
Übersetzung															
Editorklasse des Übersetzungsservers:								Die Java-Editorklasse, die eine eigene, dem Übersetzungsserver zugeordnete Klasse generiert.							
com.peregrine.util.WTSLanguageTranslatorFactory															
Sprache, aus der übersetzt wird:								Die Sprache, aus der übersetzt werden soll oder die Standardsprache, in der die Texte derzeit angezeigt werden sollen.							
Englisch															
IP-Adresse des Übersetzungsservers:								Die IP-Adresse des Übersetzungsservers. Je nach den für den Übersetzungsserver definierten Anforderungen kann diese Adresse u.U. eine Portnummer enthalten. Klicken Sie für den Standardwert hier!							
10.3.128.181:1097															
Anwendungspfad:								Verzeichnisstandort für die Webanwendungen des Peregrine-Portals							
WEB-INF/apps/															
Ereigniswarteschlange:								Geben Sie den Namen des Adapters ein, der von der Warteschlangen-Engine des Peregrine-Portals verwendet werden soll. Beispiel:							
portalDB								<p>Um die ServiceCenter-Datenbank zu verwenden, geben Sie "sc"</p> <ul style="list-style-type: none"> • ein. Um die AssetCenter-Datenbank zu verwenden, geben Sie "ac" 							
								ein.							

3 Geben Sie die folgenden Konfigurationseinstellungen ein:

- a **Editorklasse des Übersetzungsservers:** Geben Sie die Java-Editorklasse für den Übersetzungsserver ein. Die standardmäßige Java-Editorklasse ist:
com.peregrine.util.WTSLanguageTranslatorFactory
- b **Sprache, aus der übersetzt wird:** Geben Sie die Ausgangssprache für die Übersetzung ein. Dies ist standardmäßig **Deutsch**.
- c **IP-Adresse des Übersetzungsservers:** Geben Sie die IP-Adresse und den Kommunikationsanschluss des Übersetzungsservers ein. Beispiel:
10.3.128.181:1097.
- d **Anwendungspfad:** Geben Sie den relativen Pfad zum Anwendungsverzeichnis des Anwendungsservers ein. Der Standardwert ist:
WEB-INF/apps/
- e **Ereigniswarteschlange:** Geben Sie den Namen des Adapters ein, der von der Warteschlangen-Engine verwendet werden soll. Der Standardwert ist:
PortalDB

4 Klicken Sie auf **Speichern**.

Die Systemsteuerung wird geöffnet.

5 Klicken Sie auf **Server zurücksetzen**.

Übersetzen angezeigter Textdaten mit einem Übersetzungsserver

Wenn Sie Get-Services-Daten in verschiedenen Sprachen speichern möchten, können Sie Get-Services so konfigurieren, dass Daten an einen Übersetzungsserver gesendet und dort in Echtzeit übersetzt werden. Über diese Schnittstelle werden nur Textdaten übersetzt, die aus der Backend-Datenbank abgerufen oder manuell in Formulare eingegeben wurden. Wenn Sie eine übersetzte Benutzeroberfläche benötigen, können Sie ein Get-Services-Sprachpaket direkt bei Peregrine Systems erwerben.

So übersetzen Sie angezeigte Textdaten mit einem Übersetzungsserver:

- 1 Aktivieren Sie den Übersetzungsserver über die Seite **Verwaltung > Einstellungen** (siehe *Konfigurieren des WebSphere-Übersetzungsservers für Get-Services* auf Seite 115.).

Die Schaltfläche **Übersetzen** wird oben rechts auf der Symbolleiste angezeigt.

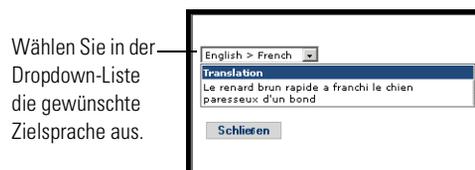


- 2 Klicken Sie auf die (aus der Backend-Datenbank abgerufenen oder in ein Formular eingegebenen) Textdaten, die übersetzt werden sollen.



- 3 Klicken Sie auf die Schaltfläche **Übersetzen**.

Das Fenster **Übersetzung** wird geöffnet.



- 4 Wählen Sie in der Dropdown-Liste die Zielsprache aus, in die die ausgewählten Textdaten übersetzt werden sollen.

Die Übersetzung der ausgewählten Textdaten wird im Feld **Übersetzung** angezeigt.

WebLogic 6.1 SP3 oder SP4

Mit den folgenden Verfahren wird WebLogic zur Ausführung von Get-Services unter UNIX konfiguriert.

So konfigurieren Sie WebLogic 6.1 SP3 oder SP4:

- Schritt 1** Halten Sie WebLogic und den Webserver an. Siehe *Anhalten von WebLogic* auf Seite 119.
- Schritt 2** Bearbeiten Sie die Datei `startWebLogic.cmd`, um das Systemkennwort, die Speichereinstellungen und den Startmodus festzulegen. Siehe *Bearbeiten von „startWebLogic.cmd“* auf Seite 119.
- Schritt 3** Starten Sie das Get-Services-Installationsprogramm. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 138
- Schritt 4** Erstellen Sie auf dem Webserver ein virtuelles Verzeichnis für Get-Services. Siehe *Erstellen eines virtuellen Verzeichnisses für Get-Services* auf Seite 120.
- Schritt 5** Starten Sie WebLogic und den Webserver neu. Siehe *Neustarten der Server* auf Seite 121.

Anhalten von WebLogic

Bevor Sie mit der Konfiguration von WebLogic beginnen, müssen Sie sowohl Weblogic als auch den Webserver anhalten.

Bearbeiten von „startWebLogic.cmd“

So bearbeiten Sie „startWebLogic.cmd“:

- 1 Öffnen Sie die Datei `startWebLogic.cmd` in einem Texteditor. Standardmäßig befindet sich diese Datei unter:
`/bea/wlserver6.1/config/<Eigene Domäne>/`

2 Wechseln Sie zu folgendem Abschnitt des Skripts:

```

echo *****
echo * To start WebLogic Server, use the password      *
echo * assigned to the system user. The system        *
echo * username and password must also be used to    *
echo * access the WebLogic Server console from a web  *
echo * browser.                                       *
echo *****
@rem Set WLS_PW equal to your system password for no password
prompt server startup.
set WLS_PW=password

```

- 3 Ersetzen Sie in der letzten Zeile das Wort `password` durch Ihr Systemkennwort für WebLogic.
- 4 Suchen Sie in der Datei nach der Parametereinstellung `-mx`. Ändern Sie diese Einstellung auf einen Wert zwischen 225 MB und 512 MB.

Hinweis: Stellen Sie sicher, dass der Höchstwert für die Heap-Größe kleiner ist als der für die Anwendungsserver verfügbare freie Arbeitsspeicher. Andernfalls lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt. Für die meisten Systeme reicht die Einstellung von 256 MB aus.

- 5 Setzen Sie die Variable `STARTMODE` auf `STARTMODE=false`.
Wenn Sie WebLogic nach der Installation erstmals starten, müssen Sie den Entwicklungsmodus auswählen, damit es die bereitgestellten Webanwendungen erkennt.
- 6 Speichern Sie die Datei.

Ausführen des Get-Services-Installationsprogramms

Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 138.

Erstellen eines virtuellen Verzeichnisses für Get-Services

Um Get-Services ausführen zu können, müssen Sie auf dem Webserver ein virtuelles Verzeichnis erstellen, das eine Zuordnung zum WebLogic-Bereitstellungsordner festlegt. Bei einer typischen Installation wird ein virtuelles Verzeichnis namens `oaa` erstellt; Sie können dem virtuellen Verzeichnis jedoch auch einen anderen Namen zuweisen.

Anforderungen für das virtuelle Verzeichnis für Get-Services

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	< <i>oaa</i> >
Zuordnung zu physischem Pfad	<WebLogic>/applications/ <i>oaa</i>
Verzeichniszugriffsrechte	Schreiben, Skripts ausführen, Ausführen

Geben Sie für <*oaa*> den Namen des virtuellen Verzeichnisses ein, das für Get-Services verwendet werden soll. Den hier gewählten Namen müssen Sie ebenfalls in der Konfiguration des Anwendungsservers verwenden.

Geben Sie für <*WebLogic*> den Pfad zur WebLogic-Installation ein. Der Standardpfad lautet:

`/bea/wlserver6.1/config/<Eigene Domäne>`

Neustarten der Server

Starten Sie WebLogic und den Webserver neu, damit die neuen Einstellungen wirksam werden.

JRun 3.1

Mit den folgenden Verfahren wird JRun zur Ausführung von Get-Services unter UNIX konfiguriert.

So konfigurieren Sie JRun 3.1:

- Schritt 1** Installieren Sie eine Java-Laufzeitumgebung. Siehe *Installieren einer Java-Laufzeitumgebung* auf Seite 122.
- Schritt 2** Laden Sie JRun von der Macromedia-Website herunter und installieren Sie das Programm im Stammverzeichnis auf der Festplatte. Siehe *Installieren von JRun* auf Seite 122.
- Schritt 3** Wenden Sie das aktuelle JRun-Update an. Siehe *Anwenden des aktuellen JRun-Updates* auf Seite 123.
- Schritt 4** Stellen Sie die WAR-Datei des Portals für JRun bereit, so dass die für Get-Services benötigte Ordnerstruktur erstellt wird. Siehe *Bereitstellen der WAR-Datei des Portals für JRun* auf Seite 123.

- Schritt 5** Starten Sie das Get-Services-Installationsprogramm. *Ausführen des Installationsprogramms* auf Seite 126.
- Schritt 6** Verschieben Sie `js.jar` in den Ordner `ext` des Java Development Kit. Siehe *Verschieben von „js.jar“ in das Java Development Kit* auf Seite 126.
- Schritt 7** Führen Sie den JRun Connector Wizard (JRun-Verbindungsassistent) aus, um eine Verbindung zwischen JRun und dem Webserver herzustellen. Siehe *Ausführen des JRun Connector Wizard (JRun-Verbindungsassistent)* auf Seite 126.
- Schritt 8** Konfigurieren Sie die JRun Java-Einstellungen. Siehe *Konfigurieren der Java-Einstellungen* auf Seite 127.
- Schritt 9** Legen Sie Umgebungsvariablen fest, um Bibliothekspfade für die Backend-Datenbanken anzugeben. Siehe *Festlegen von Umgebungsvariablen für Bibliothekspfade* auf Seite 130.
- Schritt 10** Erstellen Sie auf dem Webserver ein virtuelles Verzeichnis für Get-Services. Siehe *Erstellen eines virtuellen Verzeichnisses für Get-Services* auf Seite 131.
- Schritt 11** Starten Sie JRun und den Webserver neu. Siehe *Neustarten von JRun* auf Seite 131.

Installieren einer Java-Laufzeitumgebung

Das Get-Services-Installationsprogramm umfasst Java 2 SDK Standard Edition v1.3.1_05; Sie können jedoch auch JRE 1.3.1 verwenden, wenn diese Version bereits installiert ist. Siehe *Komponenten einer benutzerdefinierten Installation* auf Seite 139.

Installieren von JRun

Vor der Installation von Get-Services müssen Sie JRun in Ihrem Stammverzeichnis installieren.

So installieren Sie JRun:

- 1 Rufen Sie den folgenden URL auf:
`http://www.macromedia.com/software/jrun/`
- 2 Klicken Sie auf den Link **JRun 3.1 Available for Purchase** (JRun 3.1 verfügbar).
- 3 Folgen Sie den angezeigten Installationsanweisungen.

Anwenden des aktuellen JRun-Updates

Bevor Sie Get-Services installieren, müssen Sie das aktuelle Update für JRun 3.1 anwenden.

So installieren Sie das aktuelle JRun-Update:

- 1 Rufen Sie den folgenden URL auf:
http://www.macromedia.com/support/jrun/updates/3/updates_31.html
- 2 Klicken Sie auf den Link für die JRun-Edition (Enterprise, Advanced oder Professional) und das Betriebssystem des Servers.
- 3 Folgen Sie den angezeigten Installationsanweisungen.

Bereitstellen der WAR-Datei des Portals für JRun

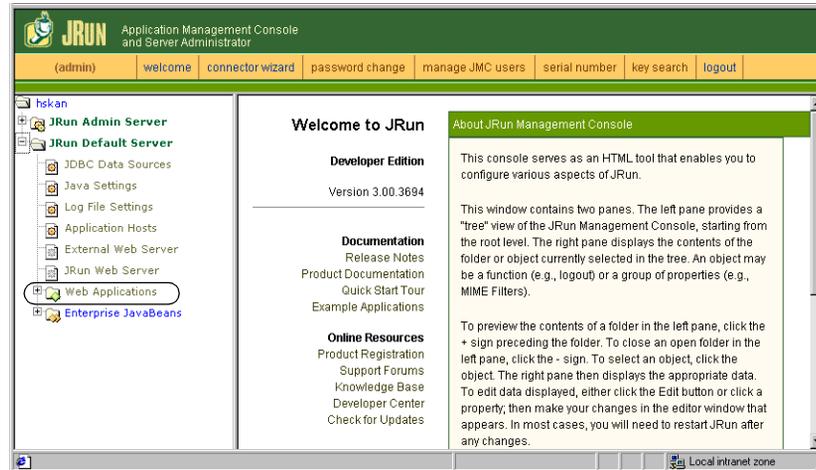
Die WAR-Datei des Portals erstellt die Ordnerstruktur, die zur Bereitstellung von Get-Services auf dem Anwendungsserver benötigt wird. Wenn Sie diese Datei für WebLogic bereitgestellt haben, können Sie das Get-Services-Installationsprogramm ausführen.

So stellen Sie die Get-Services-WAR-Datei des Portals für JRun bereit:

- 1 Öffnen Sie die JRun Management Console (JRun-Managementkonsole) und melden Sie sich an.



- Wählen Sie JRun Default Server (JRun-Standardserver) > Web Applications (Webanwendungen).



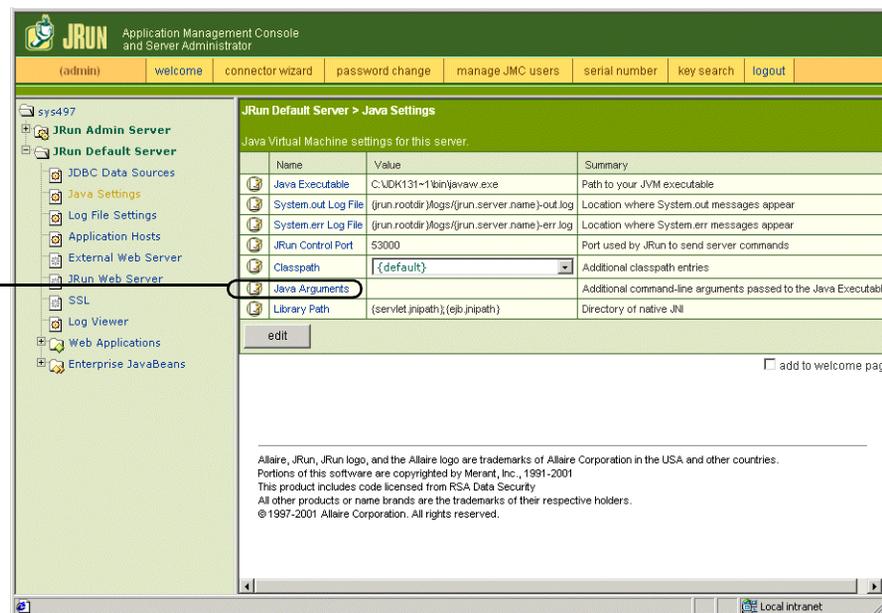
Die Seite **Edit / Create / Deploy and Remove Applications** (Anwendungen bearbeiten/erstellen/bereitstellen und entfernen) wird geöffnet.



- Klicken Sie auf den Link **Deploy an Application** (Anwendung bereitstellen).
- Machen Sie in den Feldern auf der angezeigten Seite folgende Angaben:

- Servlet War File or Directory (Servlet-WAR-Datei oder -Verzeichnis):
Wechseln Sie zu <CD-ROM-Laufwerk>/portal<Versionsnummer>.war.
Wählen Sie für <Versionsnummer> die aktuellste Version aus.
Wählen Sie diese Datei aus und klicken Sie auf **Accept** (Annehmen).
 - JRun Server Name (Name des JRun-Servers):
Wählen Sie **JRun Default Server** (JRun-Standardserver).
 - Application Name (Anwendungsname):
Geben Sie **oaa** ein.
 - Application URL (Anwendungs-URL):
Geben Sie **/oaa** ein.
 - Application Deploy Directory (Bereitstellungsverzeichnis der Anwendung):
Dieses Verzeichnis wird von JRun erstellt. Notieren Sie diesen Pfad.
(Sie benötigen diese Angabe weiter unten in diesem Verfahren.)
- 5 Klicken Sie auf **deploy** (Bereitstellen).
Es wird eine Meldung angezeigt, die darauf hinweist, dass OAA bereitgestellt wurde.
- 6 Klicken Sie auf der Seite **Java Settings** (Java-Einstellungen) auf **Java Arguments** (Java-Argumente).

Klicken Sie auf **Java Arguments** (Java-Argumente).



- 7 Geben Sie gemäß dem Formatbeispiel im angezeigten Bearbeitungsfenster einen `-Xmx`-Wert ein, um eine Obergrenze für den Heap-Speicher des Systems festzulegen. Dieser Wert sollte zwischen 225 MB und 512 MB liegen.

Hinweis: Stellen Sie sicher, dass der Höchstwert für die Heap-Größe kleiner ist als der für die Anwendungsserver verfügbare freie Arbeitsspeicher. Andernfalls lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt. Für die meisten Systeme reicht die Einstellung von 256 MB aus. Für Anwendungen, die dauerhaft verwendet werden, müssen Sie ggf. einen höheren Wert angeben.

Ausführen des Installationsprogramms

Starten Sie das Get-Services-Installationsprogramm und wählen Sie die Option **Benutzerdefiniert**. Siehe *Installationsoption „Benutzerdefiniert“* auf Seite 138.

Verschieben von „js.jar“ in das Java Development Kit

Um JRun ausführen zu können, muss eine aktuelle Version von `js.jar` im Java Development Kit verfügbar sein.

So verschieben Sie „js.jar“ in das Java Development Kit:

- 1 Halten Sie JRun an.
- 2 Suchen Sie `js.jar`. Standardmäßig befindet sich diese Datei unter:
`<JRun>/servers/default/oa/WEB-INF/lib`
- 3 Schneiden Sie die Datei aus und fügen Sie sie in den folgenden Ordner ein:
`/usr/local/peregrine/common/jdk1.3/jre/lib/ext`
- 4 Starten Sie JRun neu.

Ausführen des JRun Connector Wizard (JRun-Verbindungsassistent)

Der JRun Connector Wizard (JRun-Verbindungsassistent) stellt eine Verbindung zwischen JRun und dem Webserver her.

So führen Sie den JRun Connector Wizard (JRun-Verbindungsassistent) aus:

- 1 Melden Sie sich bei der JRun Management Console (JRun-Managementkonsole) an.
- 2 Klicken Sie auf **Connector Wizard** (Verbindungsassistent).
- 3 Wählen Sie unter **JRun Server Name** (Name des JRun-Servers) den Eintrag **JRun Default Server** (JRun-Standardserver).
- 4 Wählen Sie in der Dropdown-Liste den von Ihnen verwendeten Webserver aus.
- 5 Wenn der Webserver eine andere IP-Adresse als der JRun-Server verwendet, geben Sie unter **JRun Server IP Address** (IP-Adresse des JRun-Servers) die IP-Adresse des JRun-Servers ein.
- 6 Stellen Sie sicher, dass der Verbindungsanschluss des JRun-Servers nicht in Konflikt mit einem anderen auf diesem Server verwendeten Kommunikationsanschluss steht.
- 7 Geben Sie den Pfad zum Skriptverzeichnis ein.
- 8 Klicken Sie auf **Done** (Fertig).

Konfigurieren der Java-Einstellungen

Nach der Installation von Get-Services müssen Sie die Java-Einstellungen konfigurieren, mit denen JRun die Webanwendung ausführt.

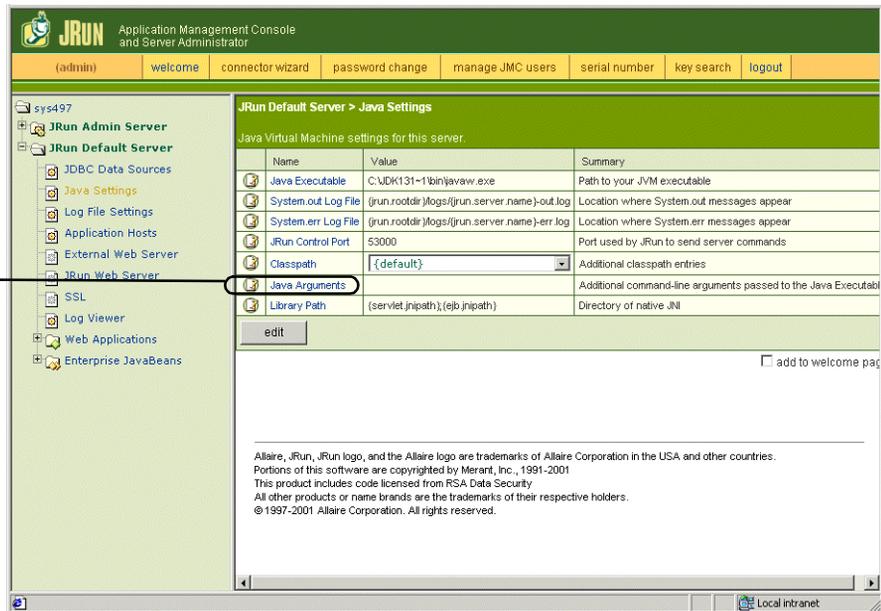
So konfigurieren Sie die Java-Einstellungen:

- 1 Melden Sie sich bei der JRun Management Console (JRun-Managementkonsole) an.
- 2 Klicken Sie auf **JRun Default Server** (JRun-Standardserver) > **Java Settings** (Java-Einstellungen).

Die Seite **Java Settings** (Java-Einstellungen) wird geöffnet.

3 Klicken Sie auf **Java Arguments** (Java-Argumente).

Klicken Sie auf
Java Arguments
(Java-Argumente).



Das Bearbeitungsfenster wird geöffnet.

- 4 Geben Sie einen `-Xmx`-Wert ein, um eine Obergrenze für den Heap-Speicher des Systems festzulegen. Dieser Wert sollte zwischen 225 MB und 512 MB liegen.

Hinweis: Stellen Sie sicher, dass der Höchstwert für die Heap-Größe kleiner ist als der für die Anwendungsserver verfügbare freie Arbeitsspeicher. Andernfalls lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt. Für die meisten Systeme reicht die Einstellung von 256 MB aus. Für Anwendungen, die dauerhaft verwendet werden, müssen Sie ggf. einen höheren Wert angeben.

5 Klicken Sie auf der Seite Java Settings (Java-Einstellungen) auf Classpath (Klassenpfad).



Das Bearbeitungsfenster wird geöffnet.



6 Geben Sie die folgenden Klassenpfade ein:

- Ordner ext des Java Development Kit. Beispiel:
/usr/peregrine/Common/jdk1.3.1_05/jre/lib/ext

- 7 Klicken Sie auf der Seite **Java Settings** (Java-Einstellungen) auf **Java Executable** (Ausführbare Java-Datei).
- 8 Stellen Sie sicher, dass der Pfad des Java Development Kit dem unter **Classpath** (Klassenpfad) angegebenen Pfad entspricht. Beispiel:
/usr/peregrine/Common/jdk1.3.1_05/bin/javaw.exe
- 9 Klicken Sie auf **update** (Aktualisieren).

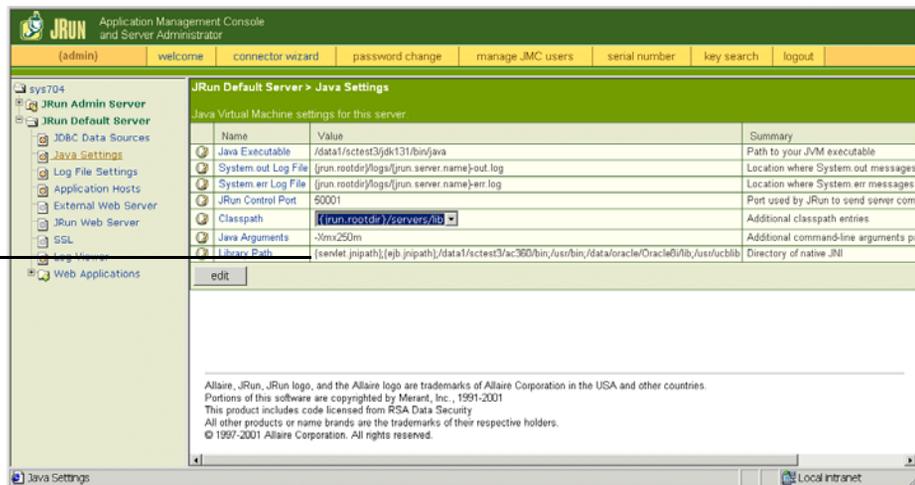
Festlegen von Umgebungsvariablen für Bibliothekspfade

Auf UNIX-basierten Systemen müssen Sie die Bibliothekspfade für die Backend-Datenbanken festlegen.

- 1 Öffnen Sie die JRun Management Console (JRun-Managementkonsole) und melden Sie sich an.
- 2 Wählen Sie im Menü auf der linken Seite **JRun Default Server** (**JRun-Standardserver**) > **Java Settings** (**Java-Einstellungen**).
- 3 Klicken Sie auf **Library Path** (Bibliothekspfad).
- 4 Fügen Sie bei Bedarf die folgenden Bibliothekspfade hinzu:
 - Wenn Sie Get-Services unter Solaris ausführen, fügen Sie Folgendes hinzu:
 - /usr/bin
 - /usr/ucblib
- 5 Klicken Sie auf **update** (Aktualisieren).

In JRun werden die aktualisierten Bibliothekspfade angezeigt.

Hier werden die aktualisierten Pfade angezeigt.



- 6 Melden Sie sich bei der Management Console (Managementkonsole) ab.

Erstellen eines virtuellen Verzeichnisses für Get-Services

Um Get-Services ausführen zu können, müssen Sie auf dem Webserver ein virtuelles Verzeichnis erstellen, das eine Zuordnung zum JRun-Bereitstellungsordner festlegt. Bei einer typischen Installation wird ein virtuelles Verzeichnis namens `oaa` erstellt; Sie können dem virtuellen Verzeichnis jedoch auch einen anderen Namen zuweisen.

Anforderungen für das virtuelle Verzeichnis für Get-Services

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	<code><oaa></code>
Zuordnung zu physischem Pfad	<code><JRun>/oaa</code>
Verzeichniszugriffsrechte	Schreiben, Skripts ausführen, Ausführen

Geben Sie für `<oaa>` den Namen des virtuellen Verzeichnisses ein, das für Get-Services verwendet werden soll. Den hier gewählten Namen müssen Sie ebenfalls in der Konfiguration des Anwendungsservers verwenden.

Geben Sie für `<JRun>` den Pfad zur JRun-Installation ein. Der empfohlene Installationspfad ist:

```
/JRun/servers/default
```

Neustarten von JRun

Starten Sie JRun und den Webserver neu, damit die neuen Einstellungen wirksam werden.

Installationsoption „Typisch“

Bei einer Get-Services-Installation mit der Option **Typisch** werden die am häufigsten verwendeten Komponenten des Produkts installiert und die Anwendungsdateien und -daten werden in den standardmäßigen Zielverzeichnissen gespeichert. Die meisten Benutzer verwenden die typische Installation.

Komponenten der typischen Installation

Es folgt eine kurze Beschreibung der Komponenten, die bei einer typischen Installation von Get-Services automatisch installiert werden:

Anwendungs- und Dateipfade

Get-Services-Komponente	Standardmäßiges Installationsverzeichnis
Apache-Webserver	/usr/local/peregrine/common/apache2
Tomcat-Anwendungsserver	/usr/local/peregrine/common/tomcat4
Java Development Kit	/usr/local/peregrine/common/jdk1.3.1
OAA-Plattform und Get-Services	/usr/local/peregrine/oa

Kommunikationsanschlüsse

Get-Services verwendet bei einer typischen Installation die in der folgenden Tabelle aufgeführten Kommunikationsanschlüsse. Nach der Installation können Sie Get-Services zur Verwendung alternativer Kommunikationsanschlüsse konfigurieren, wenn die angegebenen Kommunikationsanschlüsse im lokalen Netzwerk bereits verwendet werden.

Standard-anschluss	Wird verwendet von	Alternativer Anschluss
80	Apache-Webserver	8081
8005	Verwaltung des Tomcat-Anwendungsservers	8015
8009	Worker-Datei des Tomcat-Anwendungsservers	8019
8011	Worker-Datei des Tomcat-Anwendungsservers für Lastenausgleich (optional).	8021
8013	Worker-Datei des Tomcat-Anwendungsservers für Lastenausgleich (optional).	8023
8015	Worker-Datei des Tomcat-Anwendungsservers für Lastenausgleich (optional).	8025

Hinweis: Wenn Sie Einstellungen dieser Komponenten ändern oder andere Komponenten installieren möchten, verwenden Sie die Installationsoption **Benutzerdefiniert** (Custom) für Get-Services.

Die Installation ist hiermit abgeschlossen. Sie müssen jetzt die Verbindung des Systems zur Backend-Datenbank konfigurieren, wenn Sie diese Aufgabe noch nicht durchgeführt haben. Verwenden Sie hierzu die Seite **Admin-Einstellungen** des Verwaltungsmoduls.

Verfahren für die typische Installation

In diesem Abschnitt wird das Verfahren für die Installation von Get-Services mit einem Tomcat-Anwendungsserver und einem Apache-Webserver unter einem AIX-, Linux- oder Solaris-Betriebssystem erläutert.

So führen Sie eine typische Installation von Get-Services unter UNIX durch:

- 1 Melden Sie sich beim Server an.

Wichtig: Sie müssen sich mit einem Konto anmelden, das über Root-Privilegien verfügt.

- 2 Legen Sie die Get-Services-Installations-CD in das CD-ROM-Laufwerk des Computers ein. In der Regel wird das Installationsprogramm automatisch gestartet.

Brechen Sie den automatischen Startvorgang ab und laden Sie das CD-ROM-Laufwerk. Beispiel:

```
mount /cdrom
```

Wechseln Sie zum CD-ROM-Laufwerk. Beispiel:

```
cd /cdrom
```

Geben Sie das Installationskript für Ihr Betriebssystem ein:

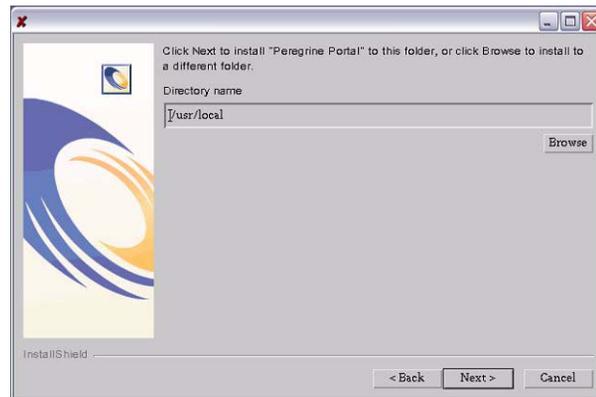
Betriebssystem	Auszuführendes Shell-Skript
AIX 5.1	./setupaix
Red Hat Linux 7.3	./setuplinux
Solaris 2.7	./setupsolaris
Solaris 2,8	./setupsolaris

Die Willkommen-Seite des Installationsprogramms wird geöffnet.



- 3 Klicken Sie auf **Weiter** (Next), um zur nächsten Seite des Assistenten zu wechseln.

Die Seite für das Installationsverzeichnis wird geöffnet.

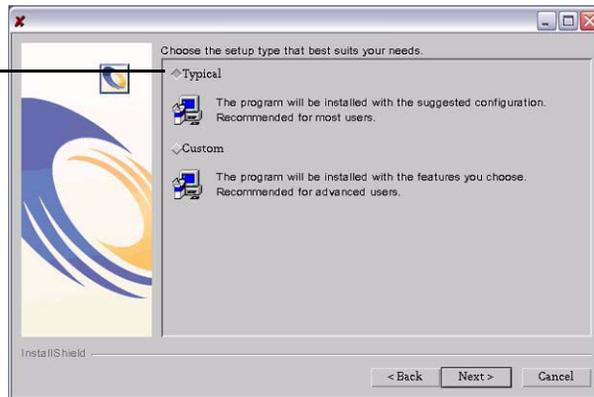


- 4 Klicken Sie auf **Durchsuchen** (Browse), um das standardmäßige Installationsverzeichnis (/usr/local) zu ändern.

- 5 Klicken Sie auf **Weiter** (Next), um zur nächsten Seite des Assistenten zu wechseln.

Die Seite für den Setup-Typ wird geöffnet.

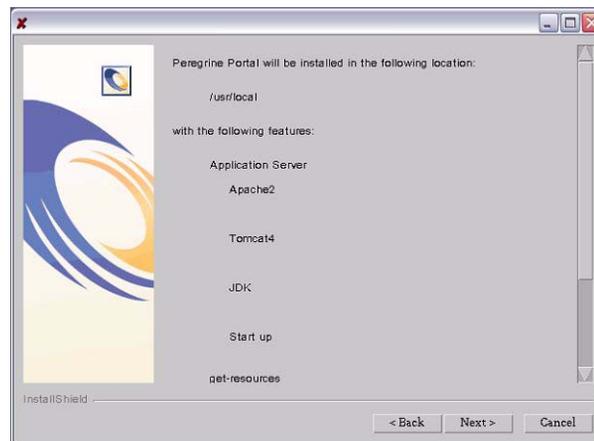
Klicken Sie auf
Typisch.



- 6 Wählen Sie **Typisch** (Typical).

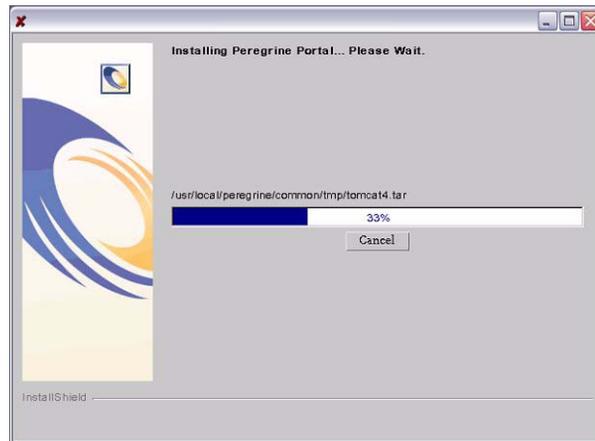
- 7 Klicken Sie auf **Weiter** (Next), um zur nächsten Seite des Assistenten zu wechseln.

Die Seite für die Überprüfung der Komponenten wird geöffnet.

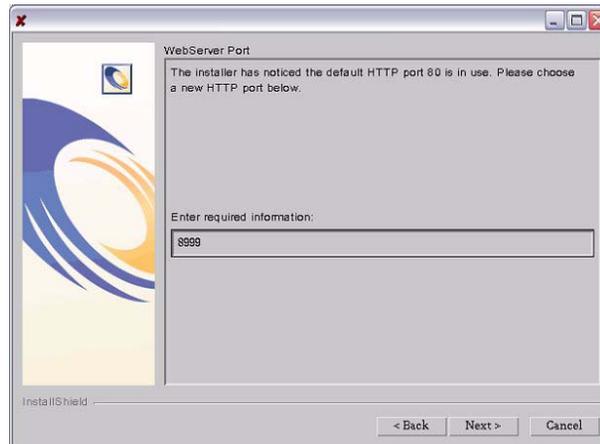


- 8 Klicken Sie auf **Weiter** (Next), um mit der Installation der Get-Services-Komponenten zu beginnen.

Auf der daraufhin geöffneten Seite wird der Installationsfortschritt angezeigt.



Wenn die Installation abgeschlossen wurde, prüft das Installationsprogramm die Verfügbarkeit des Anschlusses 80 für den Apache-Webserver. Wenn das Installationsprogramm einen Anschlusskonflikt bei Anschluss 80 ermittelt, wird die Seite für den Webserver-Anschluss angezeigt.



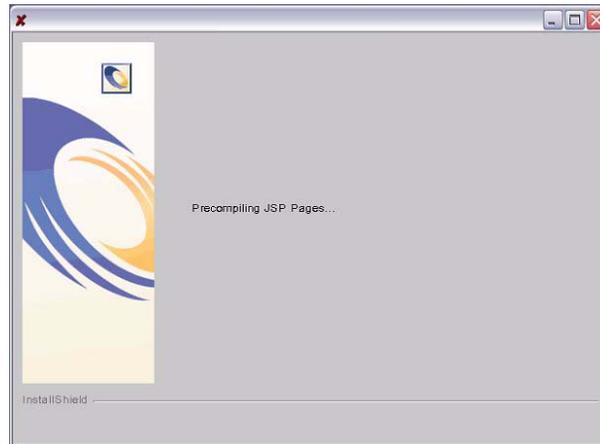
- 9 Geben Sie, falls erforderlich, den neuen Kommunikationsanschluss des Webservers ein.

- 10 Klicken Sie auf **Weiter** (Next), um mit der Bereitstellung der Get-Services-Komponenten zu beginnen.

Die Seite für das Bereitstellungsdienstprogramm von Get-Services wird geöffnet.

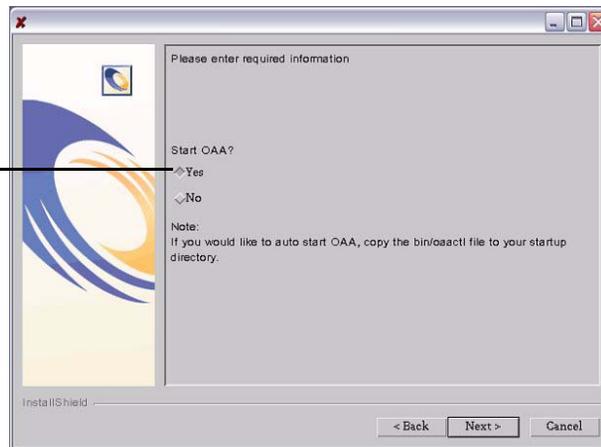


Während der Bereitstellung werden die zur Verwendung durch Tomcat bestimmten Java-Serverseiten vom Installationsprogramm vorkompiliert.



Wenn das Installationsprogramm die Bereitstellung von Get-Services abgeschlossen hat, wird die Seite zum Starten von OAA angezeigt.

Wählen Sie **Ja** (Yes), um Get-Services sofort zu starten.



- 11 Klicken Sie auf **Ja** (Yes), um Get-Services sofort zu starten, oder klicken Sie auf **Nein** (No), um Get-Services nach Abschluss der Installation manuell zu starten.

Wenn Get-Services bei jedem Start des Servers gestartet werden soll, kopieren Sie die Datei `oaactl` in das Startverzeichnis. Standardmäßig befindet sich diese Datei unter:

```
/usr/local/peregrine/bin/
```

Die Installation von Get-Services ist hiermit abgeschlossen.

Installationsoption „Benutzerdefiniert“

Im folgenden Abschnitt wird beschrieben, wie eine benutzerdefinierte Installation von Get-Services auf einem Unix-Server durchgeführt wird. Unter anderem finden Sie hier eine Übersicht über die für Entwicklungs- und Produktionsumgebungen erforderlichen Schritte.

Komponenten einer benutzerdefinierten Installation

Es folgt eine kurze Beschreibung der Komponenten, die bei einer benutzerdefinierten Installation von Get-Services installiert werden können:

Anwendungsoptionen

Get-Services-Komponente Optionen

Webserver	<ul style="list-style-type: none"> ■ Apache 2.0.43 ■ IBM HTTP Server 1.3.19 ■ Microsoft IIS 5.0
Anwendungsserver	<ul style="list-style-type: none"> ■ Tomcat 4.1.12 ■ WebSphere 4.02 ■ WebLogic 6.1 SP3 ■ JRun 3.1
Java Development Kit	Java 2 SDK

Kommunikationsanschlüsse

Bei einer benutzerdefinierten Installation von Get-Services hängen die verwendeten Kommunikationsanschlüsse von den ausgewählten Anwendungskomponenten ab. Informationen zu den benötigten Kommunikationsanschlüssen finden Sie in der Dokumentation des Webserver und des Anwendungsservers. Nach der Installation können Sie Get-Services zur Verwendung alternativer Kommunikationsanschlüsse konfigurieren, wenn die angegebenen Kommunikationsanschlüsse im lokalen Netzwerk bereits verwendet werden.

Get-Services auf Servern mit Oracle 9.2.0.1

Wenn Sie Get-Services auf einem Server mit Oracle 9.2.0.1 verwenden, kann ein Anschlusskonflikt bei den Kommunikationsanschlüssen 8009 und 8080 auftreten. Prüfen Sie in der Dokumentation des Webserver und des Anwendungsservers, ob diese Anschlüsse verwendet werden.

Wenn Sie Tomcat als Anwendungsserver einsetzen, tritt standardmäßig ein Anschlusskonflikt beim Anschluss 8009 auf. Es empfiehlt sich, Tomcat auf Servern mit Oracle 9.2.0.1 zur Verwendung eines anderen Kommunikationsanschlusses konfigurieren.

Verfahren für die benutzerdefinierte Installation

So führen Sie eine benutzerdefinierte Installation von Get-Services unter UNIX durch:

- 1 Melden Sie sich beim Server an.

Wichtig: Sie müssen sich mit einem Konto anmelden, das über Root-Privilegien verfügt.

- 2 Legen Sie die Get-Services-Installations-CD in das CD-ROM-Laufwerk des Computers ein. In der Regel wird das Installationsprogramm automatisch gestartet.

Wenn das Installationsprogramm nicht automatisch gestartet wird, laden Sie das CD-ROM-Laufwerk. Beispiel:

```
mount /cdrom
```

Wechseln Sie zum CD-ROM-Laufwerk. Beispiel:

```
cd /cdrom
```

Geben Sie das Installationskript für Ihr Betriebssystem ein:

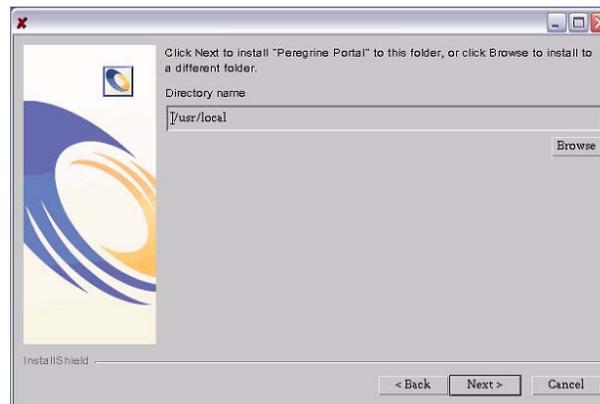
Betriebssystem	Auszuführendes Shell-Skript
AIX 5.1	./setupaix
Red Hat Linux 7.3	./setuplinux
Solaris 2.7	./setupsolaris
Solaris 2,8	./setupsolaris

Die Willkommen-Seite des Installationsprogramms wird geöffnet.



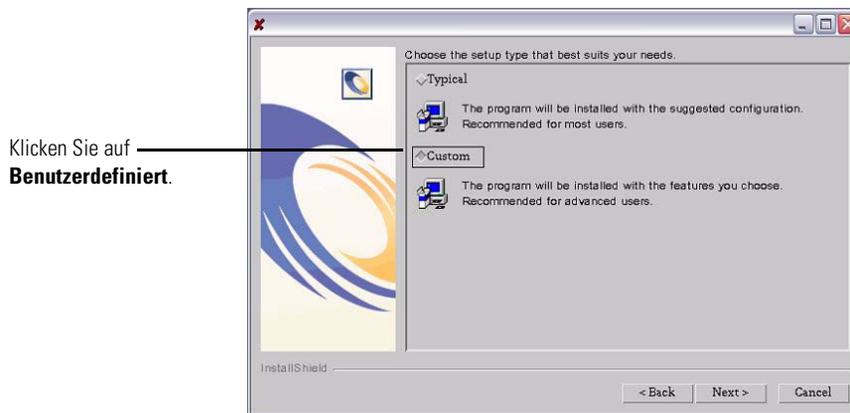
- 3 Klicken Sie auf **Weiter** (Next), um zur nächsten Seite des Assistenten zu wechseln.

Die Seite für das Installationsverzeichnis wird geöffnet.



- 4 Klicken Sie auf **Durchsuchen** (Browse), um das standardmäßige Installationsverzeichnis (`/usr/local`) zu ändern.
- 5 Klicken Sie auf **Weiter** (Next), um zur nächsten Seite des Assistenten zu wechseln.

Die Seite für den Setup-Typ wird geöffnet.



6 Wählen Sie **Benutzerdefiniert** (Custom).

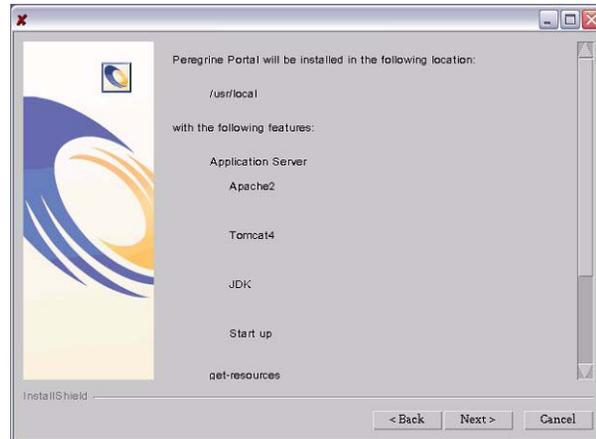
7 Klicken Sie auf **Weiter** (Next), um zur nächsten Seite des Assistenten zu wechseln.

Die Seite für die Auswahl des J2EE-Anwendungsservers wird geöffnet.



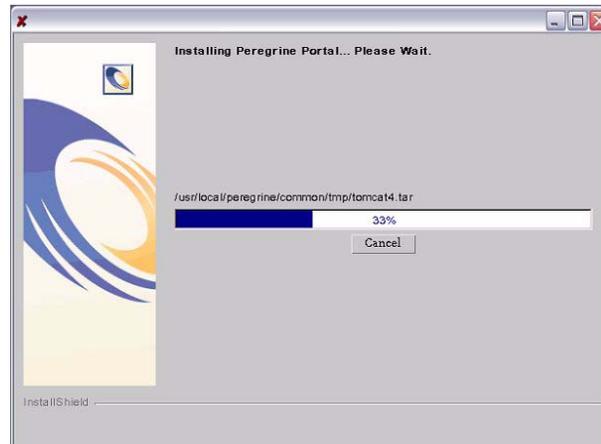
8 Klicken Sie auf **WebSphere**, um einen WebSphere-Anwendungsserver zu konfigurieren, oder klicken Sie auf **Anderer** (Other), um einen anderen Anwendungsserver zu konfigurieren. Klicken Sie auf **Weiter** (Next), um zur nächsten Seite des Assistenten zu wechseln.

Die Seite zum Überprüfen der Komponenten wird geöffnet.



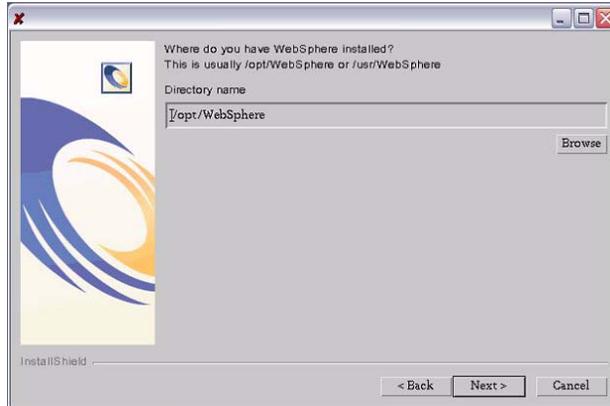
- 9 Klicken Sie auf **Weiter** (Next), um mit der Installation der Get-Services-Komponenten zu beginnen.

Auf der daraufhin geöffneten Seite wird der Installationsfortschritt angezeigt.



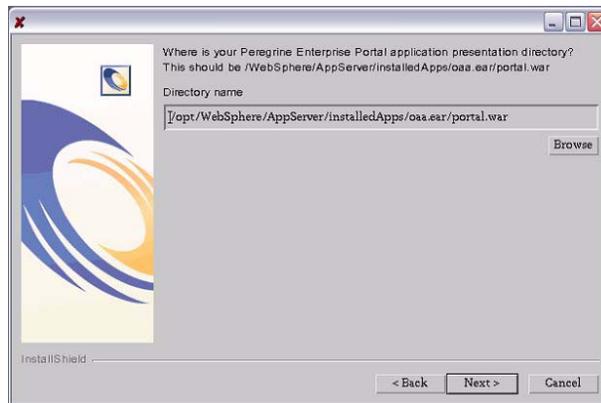
Wenn Sie einen WebSphere-Anwendungsserver konfigurieren, werden die Bildschirme in Schritt a bis Schritt e angezeigt.

Die Seite für das Installationsverzeichnis des WebSphere-Anwendungsservers wird geöffnet.



- a Klicken Sie auf **Durchsuchen** (Browse), um das Verzeichnis anzugeben, in dem der WebSphere-Anwendungsserver installiert ist. Klicken Sie auf **Weiter** (Next), um fortzufahren.

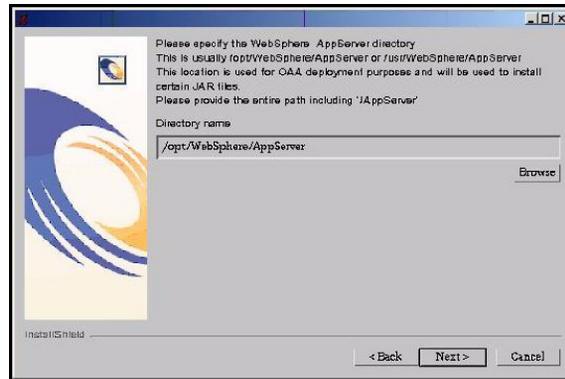
Die Seite für das Präsentationsverzeichnis wird geöffnet.



- b Klicken Sie auf **Durchsuchen** (Browse), um das Verzeichnis anzugeben, in dem die Datei `portal.war` bereitgestellt wird. Klicken Sie auf **Weiter** (Next), um fortzufahren.

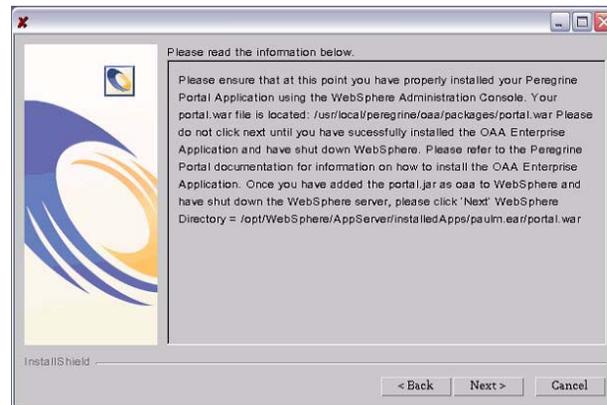
Dieses Verzeichnis wurde beim Bereitstellen der Get-Services-Datei `portal.war` als Unternehmensanwendung automatisch von WebSphere erstellt. Weitere Informationen zum Bereitstellen einer WAR-Datei finden Sie unter [WebSphere 4.0.2](#) auf Seite 99.

Die Seite für das Installationsverzeichnis des WebSphere-JDK wird geöffnet.



- c Klicken Sie auf **Durchsuchen** (Browse), um das Verzeichnis anzugeben, in dem das von WebSphere verwendete Java Development Kit installiert ist. Klicken Sie auf **Weiter** (Next), um fortzufahren.

Die Seite zur Bestätigung der Bereitstellung wird geöffnet.

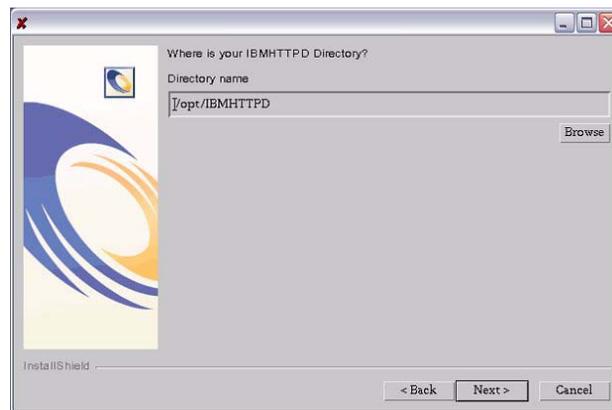


- d Wenn Sie die Datei **portal.war** in WebSphere bereitgestellt haben, klicken Sie auf **Weiter** (Next), um fortzufahren.

Die Seite für das Bereitstellungsdienstprogramm von Get-Services wird geöffnet.



Die Seite für das Verzeichnis von IBM HTTP Server wird geöffnet.

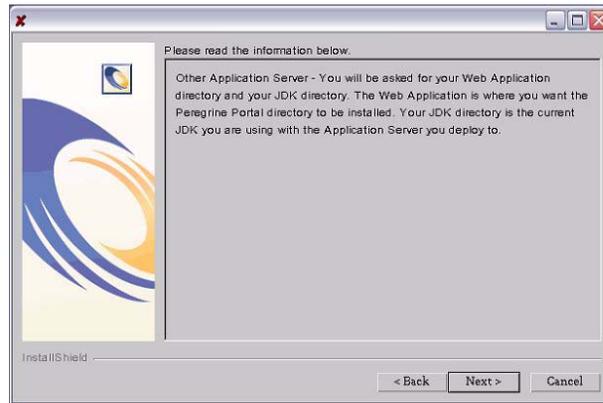


- e Klicken Sie auf **Durchsuchen** (Browse), um das Verzeichnis anzugeben, in dem der IBM HTTP Server installiert ist. Klicken Sie auf **Weiter** (Next), um fortzufahren.

Das Get-Services-Installationsprogramm konfiguriert automatisch ein virtuelles Webserver-Verzeichnis namens `oaa`. Wenn Sie ein anderes virtuelles Webserver-Verzeichnis festlegen möchten, finden Sie unter [WebSphere 4.0.2](#) auf Seite 99 Informationen über die hierbei zu erfüllenden Anforderungen.

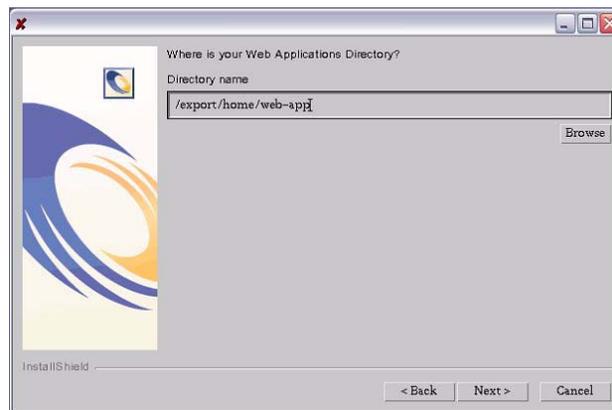
Wenn Sie einen anderen Anwendungsserver konfigurieren, werden die Bildschirme in Schritt f bis Schritt h angezeigt.

Die Seite mit Informationen zu dem anderen Anwendungsserver wird geöffnet.



- f** Klicken Sie auf **Weiter** (Next), um den Anwendungsserver zu konfigurieren.

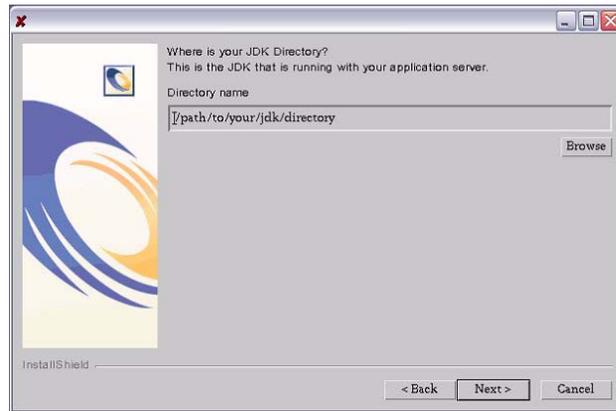
Die Seite zur Angabe des Verzeichnisses für die Webanwendungen wird geöffnet.



- g** Klicken Sie auf **Durchsuchen** (Browse), um das Verzeichnis anzugeben, in dem die Datei `portal.war` bereitgestellt wird. Klicken Sie auf **Weiter** (Next), um fortzufahren.

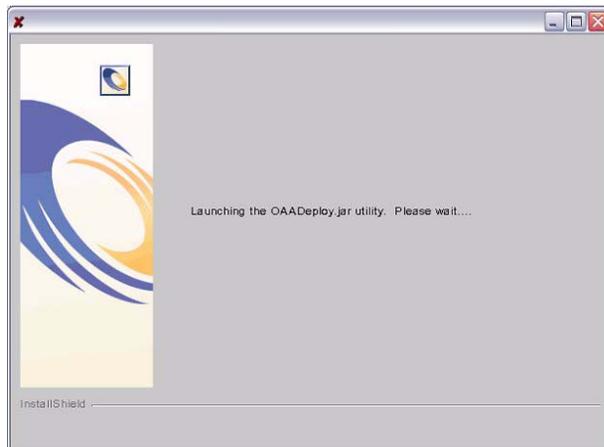
Weitere Informationen zum Bereitstellen einer WAR-Datei für Anwendungsserver finden Sie unter *Konfigurieren alternativer Anwendungsserver* auf Seite 96.

Die Seite für das Installationsverzeichnis des WebSphere-JDK wird geöffnet.



- h** Klicken Sie auf **Durchsuchen** (Browse), um das Verzeichnis anzugeben, in dem das vom Anwendungsserver verwendete Java Development Kit installiert ist. Klicken Sie auf **Weiter** (Next), um fortzufahren.

Die Seite für das Bereitstellungsdienstprogramm von Get-Services wird geöffnet.



- 10** Klicken Sie auf **Fertig stellen** (Finish), um das Installationsprogramm zu beenden.

- 11 Legen Sie die Umgebungsvariablen der Bibliothekspfade fest, die für das Betriebssystem und den Anwendungsserver benötigt werden.

Betriebssystem	Umgebungsvariable	Hinzuzufügende Pfadwerte
AIX	LIBPATH	<ul style="list-style-type: none"> ■ /<Anwendungsserver>/WEB-INF/lib/AIX ■ /<Anwendungsserver>/WEB-INF/lib/AIX/ServiceCenter4
Red Hat Linux 7.3	LD_LIBRARY_PATH	<ul style="list-style-type: none"> ■ /<Anwendungsserver>/WEB-INF/lib/Linux ■ /<Anwendungsserver>/WEB-INF/lib/Linux/ServiceCenter4
Solaris 2.7 Solaris 2.8	LD_LIBRARY_PATH	<ul style="list-style-type: none"> ■ /<Anwendungsserver>/WEB-INF/lib/SunOS ■ /<Anwendungsserver>/WEB-INF/lib/SunOS/ServiceCenter4 ■ /usr/ucblib

Geben Sie für <Anwendungsserver> den Pfad zum Kontextstamm des Anwendungsservers an (einschließlich des Verzeichnisses, in dem die oaa-Anwendung installiert ist). Der folgenden Tabelle können Sie den Kontextstamm des von Ihnen verwendeten Anwendungsservers entnehmen.

Anwendungsserver	Kontextstamm
WebSphere	/WebSphere/AppServer/installedApps/<Anwendung>.ear/portal.4.0.0.<x>/oaa
WebLogic	/bea/wlserver6.1/config/<Eigene Domäne>/applications/oaa
JRun	/JRun/servers/default/oaa

Geben Sie für <Anwendung> den für Get-Services erstellten Ordner ein.
Geben Sie für <x> die Versionsnummer der WAR-Datei Ihres Portals ein.
Geben Sie für <Eigene Domäne> die erstellte WebLogic-Domäne ein.

Konfigurieren der WebSphere-Umgebung unter Linux und Solaris

Wenn Sie mit einem Linux- oder Solaris-Server arbeiten, können Sie die WebSphere-Umgebung mit der Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere konfigurieren.

So konfigurieren Sie die WebSphere-Umgebung unter Linux und Solaris:

- 1 Stellen Sie sicher, dass der WebSphere Admin Server gestartet ist.
- 2 Öffnen Sie die Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere:
/WebSphere/AppServer/bin/adminclient.sh
- 3 Klicken Sie auf der linken Seite der Konsole auf **Nodes** (Knoten) > <Servername> > **Application Servers** (Anwendungsserver).
- 4 Klicken Sie auf den Anwendungsserver, auf dem Get-Services installiert ist.
- 5 Klicken Sie auf das Register **General** (Allgemein).
- 6 Klicken Sie auf **Environment** (Umgebung).
- 7 Klicken Sie auf **Add** (Hinzufügen).
- 8 Geben Sie die folgenden Daten zur Umgebung ein:
 - a **Name**. Geben Sie LD_LIBRARY_PATH ein.
 - b **Value** (Wert). Geben Sie die Pfadwerte für das Betriebssystem ein. Diese Werte setzen sich aus dem Kontextstamm und dem relativen Pfad zu den LIB-Verzeichnissen zusammen.
- 9 Klicken Sie auf **OK**.
- 10 Klicken Sie auf **Apply** (Übernehmen).
- 11 Schließen Sie die Advanced Administrative Console (Erweiterte Verwaltungskonsole) von WebSphere.

Konfigurieren des WebSphere-Skripts „startupServer.sh“ unter AIX

Wenn Sie mit einem AIX-Server arbeiten, müssen Sie die WebSphere-Umgebung durch Bearbeiten des Skripts startupServer.sh konfigurieren.

So konfigurieren Sie die WebSphere-Umgebung unter AIX:

- 1 Öffnen Sie `startupServer.sh` in einem Texteditor.
- 2 Fügen Sie einen Eintrag für `LIBPATH` hinzu und setzen Sie ihn auf die Pfadwerte für AIX.

Beispiel:

```
#!/bin/sh
LIBPATH=/usr/lib:/WebSphere/AppServer/installedApps/oa.ear/porta1.4.0.0.55.war/WEB-INF/lib/AIX:/WebSphere/AppServer/installedApps/answer.ear/porta1.4.0.0.55.war/WEB-INF/lib/AIX/ServiceCenter4
export LIBPATH
```

- 3 Speichern Sie die Datei.

Deinstallation – AIX, Linux oder Solaris

Verwenden Sie die folgenden Anweisungen, um Get-Services zu deinstallieren.

So deinstallieren Sie Get-Services unter AIX, Linux oder Solaris:

- 1 Öffnen Sie eine Eingabeaufforderung.
- 2 Wechseln Sie zum folgenden Verzeichnis:
<Stammverzeichnis>/usr/peregrine/_uninst
- 3 Geben Sie den folgenden Befehl ein, um Get-Services zu deinstallieren:
./uninstall.bin
- 4 Folgen Sie den Anweisungen auf dem Bildschirm, um die Deinstallation durchzuführen.

4 Lastenausgleich

KAPITEL

Dieses Kapitel behandelt folgende Themen:

- *Lastenausgleich für Anwendungsserver* auf Seite 154
- *Erstellen mehrerer Tomcat-Instanzen für Apache* auf Seite 156
- *Erstellen mehrerer Tomcat-Instanzen für IIS* auf Seite 165

Lastenausgleich für Anwendungsserver

Ein Server, auf dem eine Webanwendung wie Get-Services oder Get-Resources von Peregrine ausgeführt wird, beansprucht etwa 256 MB Speicher je Anwendungsserver-Instanz. Die maximale Heap-Größe für die JVM sollte den für die Anwendungsserver verfügbaren freien Arbeitsspeicher nicht übersteigen. Andernfalls lagern die JVM-Prozesse Daten auf die Festplatte aus, was zu einer deutlich geringeren Systemleistung führt.

Im Unterschied zu anderen Adaptern stellen die Adapter für AssetCenter und ServiceCenter eine eigene Verbindung zur Backend-Datenbank her. Aus diesem Grund wird auf dem AssetCenter-Datenbankserver genauso viel Speicher benötigt wie bei einer Verbindung mit einem einzelnen Client. Der Speicherbedarf des ServiceCenter-Servers entspricht ebenfalls dem eines einzelnen ServiceCenter-Clientprozesses.

Beachten Sie, dass die Speicherverwendung mit einer zunehmenden Anzahl von Sitzungen nicht wesentlich zunimmt, da eine Architektur verwendet wird, bei der alle von derselben Anwendungsserver-Instanz verarbeiteten Sitzungen die verfügbaren Ressourcen und Datenbankverbindungen gemeinsam nutzen. Für sitzungsspezifische Daten wird nur sehr wenig Speicher verwendet und dieser Speicher wird wieder freigegeben, wenn die Benutzer sich abmelden bzw. wenn ihre Sitzungen ablaufen. Beachten Sie, dass Serversitzungen nur dann ablaufen, wenn der Browser geschlossen wird oder der Benutzer zu einer anderen Domäne navigiert.

Da die Adapter für AssetCenter und ServiceCenter jeweils eine eigene Verbindung zur Backend-Datenbank herstellen, kann durch Hinzufügen zusätzlicher Anwendungsserver-Instanzen ein gleichzeitiger Zugriff auf die Backend-Datenbank ermöglicht werden.

Der Bedarf an zusätzlichen Anwendungsserver-Instanzen und JVMs hängt von drei Variablen ab:

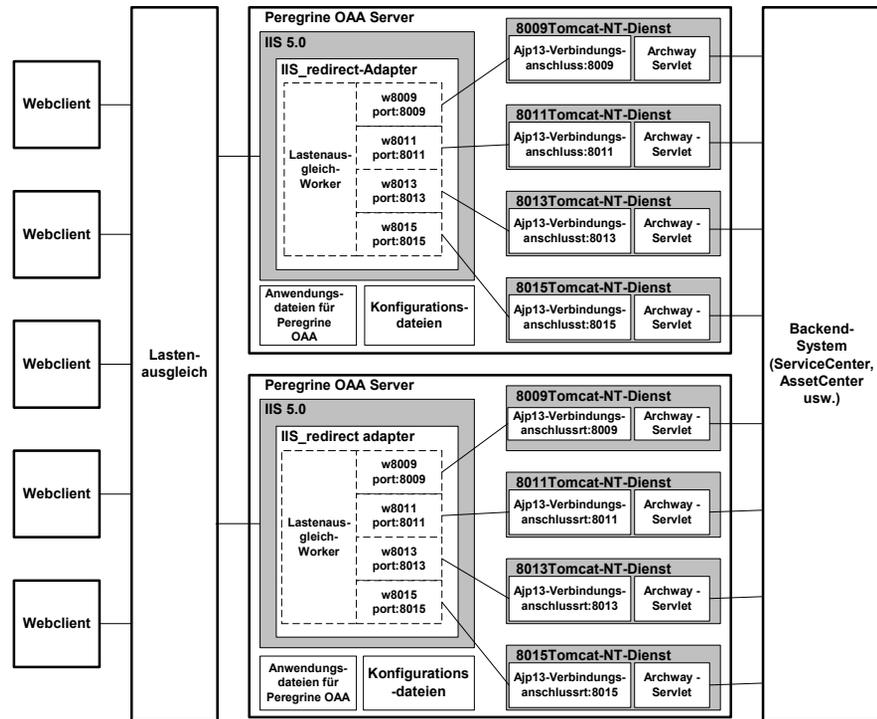
- Anzahl gleichzeitiger Benutzer.
- Prozessorleistung des Computers, auf dem der Get-Services-Webserver ausgeführt wird.
- Anzahl der Prozessoren des Computers.

Die Anforderungen an Software und Hardware können je nach Umgebung stark variieren, ein optimaler Backend-Durchsatz für ServiceCenter und AssetCenter wird jedoch in jedem Fall dann erzielt, wenn die maximale Anzahl von Anwendungsserver-Instanzen verfügbar ist, die der Server verarbeiten kann, ohne dass eine mangelnde Prozessorleistung, eine verstärkte Datenauslagerung oder eine Zunahme von Kontextänderungen zu einem Leistungsabfall führt.

Außer bei besonders großen Systemen ist die Cache-Synchronisation mit SMP-Servern (Symmetric MultiProcessing) in der Regel kein wichtiger Faktor zur Leistungsoptimierung.

Einfache Prozessoren (z. B. Pentium 450) bieten in der Regel eine akzeptable Lastenverarbeitung für etwa 100 gleichzeitige Sitzungen eines einzelnen Anwendungsserver-Prozesses. Zwei Pentium 1000 Prozessoren mit 2 Gigabyte Arbeitsspeicher (gebräuchliche Konfiguration für Datenzentren) können üblicherweise mindestens 400 gleichzeitige Sitzungen mit mehreren Anwendungsserver-Instanzen verarbeiten. Wenn Adapter eingesetzt werden, die Pooling unterstützen (z. B. JDBCAdapter oder BizDocAdapter), können mehr als 400 gleichzeitige Benutzersitzungen verarbeitet werden.

Das folgende Diagramm veranschaulicht die beim Arbeiten mit mehreren JVMs verwendete Architektur:



Erstellen mehrerer Tomcat-Instanzen für Apache

Beim Einrichten mehrerer Tomcat-Instanzen werden die Instanzen als Dienste installiert. Dies ist zwar nicht erforderlich, jedoch verbessert es die Leistung, vereinfacht die Verwaltung der Instanzen und bietet zusätzliche Funktionen. So kann ein Dienst neu gestartet werden, wenn er ausfällt oder wenn ein Neustart des Computer, auf dem die Instanzen installiert sind, durchgeführt wird.

Informationen zu Systemen mit IIS finden Sie unter *Erstellen mehrerer Tomcat-Instanzen für IIS* auf Seite 165.

So erstellen Sie mehrere Tomcat-Instanzen für Apache:

- Schritt 1** Erstellen Sie Kopien des Tomcat-Verzeichnisses. Siehe *Kopieren des Tomcat-Verzeichnisses* auf Seite 157.
- Schritt 2** Löschen Sie das Verzeichnis `\webapps\oaa` aus den neu kopierten Tomcat-Instanzen. Siehe *Kopieren des Tomcat-Verzeichnisses* auf Seite 157.
- Schritt 3** Bearbeiten Sie die Datei `workers.properties` der ersten bzw. Hauptinstanz von Tomcat, um die Werte für die weiteren Instanzen von Tomcat einzustellen. Siehe *Bearbeiten der Datei „workers.properties“* auf Seite 158.
- Schritt 4** Bearbeiten Sie die Datei `mod_jk.conf-auto` der ersten bzw. Hauptinstanz von Tomcat, um eine Verbindung zwischen Tomcat und Apache herzustellen. Siehe *Bearbeiten der Datei „mod_jk.conf-auto“* auf Seite 160.
- Schritt 5** Bearbeiten Sie die Datei `httpd.conf`, um die für Apache verfügbaren Tomcat-Worker festzulegen. Siehe *Bearbeiten der Datei „httpd.conf“* auf Seite 161.
- Schritt 6** Bearbeiten Sie die Dateien `server.xml` für jede Tomcat-Instanz. Siehe *Bearbeiten der Dateien „server.xml“ für Apache* auf Seite 161.
- Schritt 7** Bearbeiten Sie die Dateien `jk2.properties` für jede Tomcat-Instanz. Siehe *Bearbeiten der Dateien „jk2.properties“ für Apache* auf Seite 163.
- Schritt 8** Installieren Sie mehrere Tomcat-Instanzen mit Hilfe von `installservice.bat` als Dienst. Diese Datei befindet sich im Verzeichnis `Tomcat\bin`. Siehe *Installieren von Tomcat-Instanzen als Dienste für Apache* auf Seite 163.
- Schritt 9** Testen Sie die Konfiguration. Siehe *Testen des Lastenausgleichs für Apache* auf Seite 164.

Kopieren des Tomcat-Verzeichnisses

Sie müssen einen eigenen Ordner für jede zum Lastenausgleich einzusetzende Tomcat-Instanz erstellen.

So kopieren Sie das Tomcat-Verzeichnis:

- 1 Öffnen Sie den Windows-Explorer und kopieren Sie den Tomcat-Installationsordner. Der Standardpfad lautet:
`C:\Program Files\Peregrine\Common\Tomcat4`

- 2 Fügen Sie eine Kopie unter demselben Stammpfad ein. Der Standardpfad lautet:

C:\Program Files\Peregrine\Common

- 3 Benennen Sie den neuen Ordner um, so dass er einen eindeutigen Namen aufweist.

Hinweis: Geben Sie im Ordnernamen die von der betreffenden Tomcat-Instanz zu verwendende Anschlussnummer an. Beispiel: Wenn für vier Tomcat-Instanzen die Anschlüsse 8009, 8011, 8013 und 8015 verwendet werden sollen, können Sie vier Kopien des Tomcat-Ordners erstellen und ihnen die folgenden Namen zuweisen: \Tomcat4_8009, \Tomcat4_8011, \Tomcat4_8013 und \Tomcat4_8015.

- 4 Löschen Sie das Unterverzeichnis `\webapps\oaa` aus den neu kopierten Tomcat-Instanzen.

Die zusätzlichen Instanzen verwenden dasselbe Dokument-Stammverzeichnis wie die erste bzw. primäre Tomcat-Instanz.

- 5 Wiederholen Sie Schritt 1 bis Schritt 4 für jede zu verwendende Tomcat-Instanz.

Bearbeiten der Datei „workers.properties“

Für jeden Server, auf dem Tomcat-Instanzen installiert sind, gibt es nur eine Datei `workers.properties`. Tomcat installiert die Datei `workers.properties` im Verzeichnis `conf` der primären Instanz von Tomcat. Diese Datei wird auch von allen anderen Tomcat-Instanzen auf dem betreffenden Server verwendet.

Die Datei `workers.properties` gibt die Worker-Threads an, die der Webserver-Connector zur Kommunikation mit den Instanzen von Tomcat erstellt. Jede Tomcat-Instanz muss einen eigenen Kommunikationsanschluss verwenden. Als Host muss der Server angegeben werden, auf dem die Tomcat-Instanzen ausgeführt werden. (Wenn die Tomcat-Instanzen auf demselben Server ausgeführt werden wie Apache, muss `localhost` als Host angegeben werden.)

Mit `cachesize` wird die maximale Anzahl gleichzeitiger Benutzersitzungen angegeben, die Apache an die Tomcat-Instanz leiten soll.

Lbfactor ist eine Zahl größer oder gleich 1, anhand derer Apache den Lastenausgleich für die Worker vornimmt. Wenn alle Worker auf Servern desselben Leistungsniveaus ausgeführt werden, sollten die für **lbfactor** angegebenen Zahlen identisch sein. Der Lastenausgleich-Worker in Apache weist Workern, die einen niedrigeren Wert für **lbfactor** aufweisen, weniger Benutzersitzungen zu.

So bearbeiten Sie die Datei „workers.properties“:

- 1 Öffnen Sie die Datei `workers.properties` in einem Texteditor.
Diese Datei befindet sich im Verzeichnis `/conf` der Tomcat-Installation.
- 2 Bearbeiten Sie die Datei gemäß den folgenden Anweisungen. Die Pfade für `workers.tomcat_home` und `workers.java.home` sind die Verzeichnisse der Tomcat- und Java-SDK-Installationen.

Beispiel:

```
workers.tomcat_home="c:\Program Files\Peregrine\common\Tomcat4"
workers.java.home="c:\Program Files\Peregrine\common\jdk1.3.1_05"
ps=\
worker.list=loadbalancer, w8009, w8011, w8013, w8015
worker.loadbalancer.type=lb
worker.loadbalancer.balanced_workers=w8009, w8011, w8013, w8015
```

Hinweis: Sie können den Workern auch andere Namen zuweisen, allerdings müssen Sie die gewählte Namensgebungskonvention während des gesamten Verfahrens beibehalten.

- 3 Fügen Sie die folgenden Zeilen für jede installierte (und in Schritt 2 anhand der jeweiligen Anschlussnummer angegebene) Tomcat-Instanz ein:

```
worker.w8009.port=8009
worker.w8009.host=localhost
worker.w8009.type=ajp13
worker.w8009.cachesize=40
worker.w8009.lbfactor=10
```

- 4 Kommentieren Sie die folgenden Zeilen aus. Diese Standard-Worker werden nicht verwendet.

```
worker.ajp12.port=8007
worker.ajp12.host=localhost
worker.ajp12.type=ajp12
worker.ajp12.lbfactor=1
```

```
worker.ajp13.port=8009
worker.ajp13.host=localhost
worker.ajp13.type=ajp13
worker.ajp13.lbfactor=1
worker.ajp13.cachesize=10
```

- 5 Speichern Sie die Datei.

Bearbeiten der Datei „mod_jk.conf-auto“

Anhand der Datei `mod_jk.conf-auto` wird festgelegt, wo in Apache die Worker-Dateien verfügbar sind. Diese Datei wird von allen Tomcat-Instanzen auf dem Server gemeinsam genutzt. Nach dem Bereitstellen der erforderlichen Get-Services-Dateien müssen Sie dieses Verfahren durchführen. Anderenfalls werden Bereitstellungspunkte, Dateipfade und Verzeichnisse von Get-Services nicht in die Datei `mod_jk.conf-auto` aufgenommen, so dass Sie diese manuell hinzufügen müssen.

So bearbeiten Sie die Datei „mod_jk.conf-auto“:

- 1 Erstellen Sie eine Kopie der Datei `mod_jk.conf-auto` und benennen Sie diese Kopie in `mod_jk.conf-local` um.

Die Datei `mod_jk.conf-auto` befindet sich im Tomcat-Verzeichnis `conf`.

- 2 Öffnen Sie die Datei `mod_jk.conf-local` in einem Texteditor.
- 3 Ändern Sie den Eintrag `JkWorkersFile`, so dass er auf die Datei `worker.properties` der primären Tomcat-Instanz verweist.

Beispiel:

```
JkWorkersFile "C:\Apache\Tomcat\conf\worker.properties"
```

- 4 Ändern Sie alle `JkMount`-Einträge, so dass dort `loadbalancer` anstelle des Standard-Workers `ajp12` verwendet wird.

Syntax: `JkMount<Datei(en) oder Verzeichnis> <Workername>`

Beispiel:

```
JkMount/aaa/servlet/* loadbalancer
JkMount/aaa/*.jsp loadbalancer
```

- 5 Speichern Sie die Datei.

Bearbeiten der Datei „httpd.conf“

Die Datei `httpd.conf` muss die Angabe `mod_jk.conf-local` enthalten.

So bearbeiten Sie die Datei „httpd.conf“:

- 1 Öffnen Sie die Datei `httpd.conf` in einem Texteditor.
- 2 Fügen Sie die folgende Zeile hinzu:

```
include "<Tomcat>/conf/mod_jk.conf-local"
```

Geben Sie für `<Tomcat>` den Pfad zur Tomcat-Installation ein. Der Standardpfad lautet:

```
C:\Program Files\Peregrine\Common\Tomcat4
```

- 3 Speichern Sie die Datei.

Bearbeiten der Dateien „server.xml“ für Apache

Die Datei `server.xml` enthält die Daten, die Tomcat benötigt, um eine Verbindung zum Webserver herzustellen und die Webanwendungsdateien der Peregrine OAA-Plattform zu finden

So bearbeiten Sie die Dateien „server.xml“:

- 1 Jede Tomcat-Instanz verfügt über eine im Verzeichnis `\conf` abgelegte Datei `server.xml`. Öffnen Sie diese Datei in einem Texteditor.
- 2 Setzen Sie das Anschlussnummer-Attribut des Elements `<Server>` auf einen eindeutigen Wert, der nicht in Konflikt mit anderen Anschlussnummern von Tomcat steht. Es wird empfohlen, beim Konfigurieren der Tomcat-Instanzen die Anschlussnummern 8005-8008 zu verwenden.

Beispiel:

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

- 3 Ersetzen Sie die vom Coyote-Connector verwendete Anschlussnummer durch eine eindeutige Nummer, so dass kein Konflikt auftritt. Wenn Sie vier Tomcat-Instanzen konfigurieren, werden die Werte 8009, 8011, 8013 und 8015 empfohlen.

Beispiel:

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8009" minProcessors="5" maxProcessors="75" enableLookups="true"
redirectPort="8443" acceptCount="10" debug="0"
connectionTimeout="20000" useURIVValidationHack="false"
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler" />
```

- 4 Geben Sie im Attribut `jvmRoute` des Elements `<Engine>` den Servernamen und den Kommunikationsanschluss für jede Tomcat-Instanz an.

Beispiel:

```
<Engine jvmRoute="localhost:8009" name="Standalone"
defaultHost="localhost" debug="0">
```

Die Anschlussnummern müssen gemäß der in der gesamten Konfiguration verwendeten Konvention angegeben werden (8009, 8011 usw.). Diese Einträge müssen den Tomcat-ID-Einträgen entsprechen, die Sie der Datei `workers2.properties` hinzugefügt haben.

- 5 Geben Sie im Attribut `appBase` des Elements `<Host>` den absoluten Pfad zum Verzeichnis `webapps` der primären Tomcat-Instanz an.

Beispiel:

```
<Host name="localhost" debug="0"
appBase="C:\Apache\Tomcat_w8009\webapps" unpackWARs="true"
autoDeploy="true">
```

- 6 Kommentieren Sie den Tag für den folgenden `<Connector>` mit dem Anschluss 8080 aus: `className="org.apache.coyote.tomcat4.CoyoteConnector"`.

Tomcat verwendet diesen Anschluss, um für direkte HTTP-Anforderungen mit einem Browser zu kommunizieren. Da die statischen Daten von Apache bereitgestellt werden, muss Tomcat nicht mit diesem Connector verbunden sein. Darüber hinaus wird auf diese Weise verhindert, dass Benutzer direkt auf Tomcat-Instanzen zugreifen.

Beispiel:

```
<!--
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8080" minProcessors="5" maxProcessors="75" enableLookups="true"
redirectPort="8443" acceptCount="10" debug="0"
connectionTimeout="20000" useURIValidationHack="false" />
-->
```

- 7 Ändern Sie den OAA-Kontext, so dass er nicht neu geladen werden kann. Auf diese Weise wird verhindert, dass Tomcat das Servlet neu lädt, ohne den Dienst erneut zu starten. Somit wird die Leistung verbessert und bei einer Aktualisierung bleibt der von den Tomcat-Instanzen bereitgestellte JSP-Code in der Regel synchron. Alle anderen Kontexte sollten auf `reload=false` gesetzt werden.

Beispiel:

```
<Context path="/oaa"
docBase="oaa"
crossContext="false"
debug="0"
reloadable="false" >
</Context>
```

- 8 Speichern Sie die Datei.
- 9 Wiederholen Sie Schritt 2 bis Schritt 7 für jede erstellte Kopie der `server.xml`.

Bearbeiten der Dateien „`jk2.properties`“ für Apache

Sie müssen die Datei `jk2.properties` jeder Tomcat-Instanz ändern. Mit dieser Datei wird der Kommunikationsanschluss für `jk2` eingestellt.

So bearbeiten Sie die Dateien „`jk2.properties`“:

- 1 Öffnen Sie die Datei `jk2.properties` für eine Tomcat-Instanz in einem Texteditor.
Diese Datei befindet sich im Tomcat-Verzeichnis `conf`.
- 2 Fügen Sie eine Zeile für den Anschluss `channelSocket` ein. Die Anschlussnummer muss der in der Datei `workers.properties` für diese Tomcat-Instanz festgelegten Anschlussnummer entsprechen.

Beispiel:

```
channelSocket.port=8009
```

- 3 Speichern Sie die Datei.
- 4 Wiederholen Sie Schritt 1 bis Schritt 3 für jede Tomcat-Instanz.

Installieren von Tomcat-Instanzen als Dienste für Apache

Wenn Sie die Tomcat-Dateien bearbeitet haben, können Sie jede Tomcat-Instanz mit Hilfe von `installservice.bat` als Windows-Dienst installieren.

So installieren Sie Tomcat-Instanzen als Dienste für Apache:

- 1 Öffnen Sie eine DOS-Eingabeaufforderung und wechseln Sie zum Tomcat-Verzeichnis bin.
- 2 Geben Sie für jede zu erstellende Instanz von Tomcat den folgenden Befehl ein:

```
installservice <Dienstname> <Tomcat_Basis> <jvm_dll_Pfad>
```

Hierbei gilt: *<Dienstname>* ist der Name, den Sie dem Tomcat-Dienst zuweisen, *<Tomcat_Basis>* ist das Installationsverzeichnis der Instanz, für die Sie den Dienst erstellen, und *<jvm_dll_Pfad>* ist das Installationsverzeichnis von Java SDK.

Wenn Sie bereits die Umgebungsvariablen CATALINA_HOME und JAVA_HOME eingestellt haben, sind der zweite und dritte Parameter optional.

Beispiel:

```
installservice Tomcat8009 C:\Apache\Tomcat8009 C:\jdk130_05
```

- 3 Wiederholen Sie Schritt 1 bis Schritt 2 für jeden zu erstellenden Tomcat-Dienst.

Testen des Lastenausgleichs für Apache

Wenn Sie zusätzliche Tomcat-Instanzen erstellt haben, können Sie anhand der folgenden Schritte testen, ob der Lastenausgleich durchgeführt wird.

So testen Sie den Lastenausgleich:

- 1 Starten Sie die Dienste für alle Tomcat-Instanzen.
Wenn Sie Tomcat als Dienst installiert haben, können Sie die einzelnen Instanzen über das Dialogfeld **Dienste** der Windows-Systemsteuerung starten.
- 2 Öffnen Sie einen Browser und melden Sie sich bei Get-Services an.
- 3 Führen Sie eine Aktion in Get-Services durch. Starten Sie beispielsweise eine Suche.
- 4 Melden Sie sich bei Get-Services ab.
- 5 Schließen Sie den Browser, um den Verbindungscache zu löschen.
- 6 Wiederholen Sie Schritt 1 bis Schritt 5 einmal für jede Tomcat-Instanz.
Wenn Sie z. B. 4 Tomcat-Instanzen erstellt haben, müssen Sie sich insgesamt 4-mal anmelden und abmelden.

Für den Lastenausgleich wird ein Round-Robin-Algorithmus eingesetzt, d. h., wenn der Lastenausgleich ordnungsgemäß ausgeführt wird, wird für jeden Anmeldeversuch eine andere Tomcat-Instanz verwendet.

- 7 Laden Sie die Datei `archway.log` herunter.

Sie können die Datei `archway.log` von der Seite **Administration > Serverprotokoll** herunterladen.

- 8 Öffnen Sie die Datei `archway.log` in einem Texteditor.

- 9 Überprüfen Sie, ob in den Verbindungsdetails eine unterschiedliche Tomcat-Instanz für jede Verbindung aufgeführt wird.

Wenn für jede Verbindung eine andere Tomcat-Instanz angegeben wird, wird der Lastenausgleich ordnungsgemäß durchgeführt.

Wird hingegen für jede Verbindung die gleiche Tomcat-Instanz angegeben, erfolgt kein Lastenausgleich im System und der Fehler muss behoben werden.

Erstellen mehrerer Tomcat-Instanzen für IIS

Beim Einrichten mehrerer Tomcat-Instanzen werden die Instanzen als Dienste installiert. Dies ist zwar nicht erforderlich, jedoch verbessert es die Leistung, vereinfacht die Verwaltung der Instanzen und bietet zusätzliche Funktionen. So kann ein Dienst neu gestartet werden, wenn er ausfällt oder wenn ein Neustart des Computer, auf dem die Instanzen installiert sind, durchgeführt wird.

So erstellen Sie mehrere Tomcat-Instanzen für IIS:

- Schritt 1** Erstellen Sie Kopien des Tomcat-Verzeichnisses. Siehe *Kopieren des Tomcat-Verzeichnisses* auf Seite 166.
- Schritt 2** Löschen Sie das Verzeichnis `\webapps\oaa` aus den neu kopierten Tomcat-Instanzen. Siehe *Kopieren des Tomcat-Verzeichnisses* auf Seite 157.
- Schritt 3** Konfigurieren Sie das ISAPI-Plug-In für IIS. Siehe *Konfigurieren des ISAPI-Plug-Ins für IIS* auf Seite 167.
- Schritt 4** Erstellen und konfigurieren Sie ein virtuelles Verzeichnis `jakarta` in IIS. Siehe *Erstellen und Konfigurieren eines virtuellen Verzeichnisses „jakarta“ in IIS* auf Seite 168.

- Schritt 5** Konfigurieren Sie IIS zur Verwendung von `isapi_redirector2.dll` als ISAPI-Filter. Siehe *Konfigurieren von „isapi_redirector2.dll“ als ISAPI-Filter* auf Seite 168.
- Schritt 6** Erstellen und konfigurieren Sie ein virtuelles Verzeichnis `oaa` in IIS. Siehe *Erstellen und Konfigurieren eines virtuellen Verzeichnisses „oaa“ in IIS* auf Seite 169.
- Schritt 7** Bearbeiten Sie die Datei `workers2.properties` der ersten bzw. Hauptinstanz von Tomcat, um die Werte für die weiteren Instanzen von Tomcat einzustellen. Siehe *Bearbeiten der Datei „workers2.properties“ für IIS* auf Seite 170.
- Schritt 8** Bearbeiten Sie die Dateien `server.xml` für jede Tomcat-Instanz. Siehe *Bearbeiten der Dateien „server.xml“ für IIS* auf Seite 171.
- Schritt 9** Bearbeiten Sie die Dateien `jk2.properties` für jede Tomcat-Instanz. Siehe *Bearbeiten der Dateien „jk2.properties“ für IIS* auf Seite 173.
- Schritt 10** Installieren Sie mehrere Tomcat-Instanzen mit Hilfe von `installservice.bat` als Dienst. Die Datei befindet sich im Verzeichnis `Tomcat\bin`. Siehe *Installieren von Tomcat-Instanzen als Dienste für IIS* auf Seite 174.
- Schritt 11** Testen Sie die Konfiguration. Siehe *Testen des Lastenausgleichs für IIS* auf Seite 174.

Kopieren des Tomcat-Verzeichnisses

Sie müssen einen eigenen Ordner für jede zum Lastenausgleich einzusetzende Tomcat-Instanz erstellen.

So kopieren Sie das Tomcat-Verzeichnis:

- 1 Öffnen Sie den Windows-Explorer und kopieren Sie den Tomcat-Installationsordner. Der Standardpfad lautet:
`C:\Program Files\Peregrine\Common\Tomcat4`
- 2 Fügen Sie eine Kopie unter demselben Stammpfad ein. Der Standardpfad lautet:
`C:\Program Files\Peregrine\Common`

- 3 Benennen Sie den neuen Ordner um, so dass er einen eindeutigen Namen aufweist.

Hinweis: Geben Sie im Ordnernamen die von der betreffenden Tomcat-Instanz zu verwendende Anschlussnummer an. Beispiel: Wenn für vier Tomcat-Instanzen die Anschlüsse 8009, 8011, 8013 und 8015 verwendet werden sollen, können Sie vier Kopien des Tomcat-Ordners erstellen und ihnen die folgenden Namen zuweisen: `\Tomcat4_8009`, `\Tomcat4_8011`, `\Tomcat4_8013` und `\Tomcat4_8015`.

- 4 Löschen Sie das Unterverzeichnis `\webapps\oaa` aus den neu kopierten Tomcat-Instanzen.
Die zusätzlichen Instanzen verwenden dasselbe Dokument-Stammverzeichnis wie die erste bzw. primäre Tomcat-Instanz.
- 5 Wiederholen Sie Schritt 1 bis Schritt 4 für jede zu verwendende Tomcat-Instanz.

Konfigurieren des ISAPI-Plug-Ins für IIS

Das Get-Services-Installationsprogramm legt im folgenden Ordner automatisch eine Kopie des ISAPI-Plug-Ins für IIS ab:

```
c:\Program Files\Peregrine\Common\Tomcat4\bin
```

Führen Sie die folgenden Verfahren durch, um das Plug-In für Ihre Intranetumgebung zu konfigurieren.

So konfigurieren Sie das ISAPI-Plug-In für IIS:

- 1 Öffnen Sie die Datei `jk2.reg` in einem Texteditor. Der Standardpfad lautet:
`C:\Program Files\Peregrine\Common\Tomcat4\conf`
- 2 Stellen Sie sicher, dass in den Werten `serverRoot` und `workersFile` der richtige Installationspfad von Tomcat angegeben wird. Standardmäßig lauten diese Werte:

```
"ServerRoot"="C:\\Program Files\\Peregrine\\Common\\Tomcat4"
"workersFile"="C:\\Program Files\\Peregrine\\Common\\Tomcat4\\conf\\workers2.properties"
```
- 3 Speichern und schließen Sie die Datei `jk2.reg`.
- 4 Doppelklicken Sie im Windows-Explorer auf die Datei `jk2.reg`.
Windows übernimmt die Einstellungen in die Windows-Registrierung.

Erstellen und Konfigurieren eines virtuellen Verzeichnisses „jakarta“ in IIS

Um das ISAPI-Plug-In für IIS ausführen zu können, wird ein bestimmtes virtuelles IIS-Verzeichnis benötigt. Verwenden Sie die folgenden Richtlinien zum Erstellen des virtuellen IIS-Verzeichnisses. Spezifische Anweisungen zu IIS finden Sie in der Windows-Hilfe.

Anforderungen für das virtuelle Verzeichnis „jakarta“

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	jakarta
Zuordnung zu physischem Pfad	<Tomcat>\bin\isapi_redirector2.dll
Verzeichniszugriffsrechte	Lesen, Skripts ausführen, Ausführen

Geben Sie für <Tomcat> den Pfad zur Tomcat-Installation ein. Der Standardpfad lautet:

C:\Program Files\Peregrine\Common\Tomcat4

Konfigurieren von „isapi_redirector2.dll“ als ISAPI-Filter

Um eine Verbindung zwischen Tomcat und IIS herzustellen, müssen Sie isapi_redirector2.dll als ISAPI-Filter installieren.

So installieren Sie „isapi_redirect2.dll“ als ISAPI-Filter:

- 1 Öffnen Sie in **Windows-Systemsteuerung > Verwaltung** die Managementkonsole für Internetdienste.
- 2 Klicken Sie mit der rechten Maustaste auf den Knoten **Standardwebsite** und klicken Sie dann auf **Eigenschaften**.
- 3 Klicken Sie auf das Register **ISAPI-Filter**.
- 4 Klicken Sie auf **Hinzufügen**.
- 5 Geben Sie die nachfolgenden Daten ein:
 - a **Filtername:** jakarta. Der Filtername muss dem in der Registrierungsdatei `jk2.reg` festgelegten Namen entsprechen. Der standardmäßige Name des Filters ist **jakarta**.

- b Ausführbare Datei:** isapi_redirector2.dll. Der Standardpfad lautet:
 C:\Program Files\Peregrine\Common\Tomcat4\bin\isapi_redirector2.dll
- 6 Klicken Sie auf OK.
 - 7 Klicken Sie in der Managementkonsole für Internetdienste mit der rechten Maustaste auf den Knoten **Standardwebsite** und wählen Sie dann erneut **Eigenschaften > ISAPI-Filter** aus.
 Neben dem ISAPI-Filter in IIS weist ein grüner Statuspfeil darauf hin, dass dieser ausgeführt wird.
 - 8 Schließen Sie die Managementkonsole für Internetdienste.
Hinweis: Damit die Änderungen wirksam werden, müssen Sie den IIS-Dienst anhalten und neu starten. Sie müssen außerdem Tomcat neu starten.

Erstellen und Konfigurieren eines virtuellen Verzeichnisses „oaa“ in IIS

Um Get-Services aus IIS ausführen zu können, müssen Sie ein virtuelles Verzeichnis erstellen, das eine Zuordnung zum Tomcat-Bereitstellungsordner festlegt.

Anforderungen für das virtuelle Verzeichnis „oaa“

Anforderung	Einstellung
Erstellen eines virtuellen Verzeichnisses	<oaa>
Zuordnung zu physischem Pfad	<Tomcat>\webapps\oaa
Verzeichniszugriffsrechte	Lesen, Skripts ausführen

Geben Sie für <oaa> den Namen des virtuellen Verzeichnisses ein, das für Get-Services verwendet werden soll. Der empfohlene Name für das virtuelle Verzeichnis lautet **oaa**. Wenn Sie einen anderen Namen verwenden möchten, müssen Sie diesen an den folgenden Stellen eingeben:

- Benennen Sie den Ordner <Tomcat>\webapps\oaa in <Tomcat>\webapps\<Neuer Name> um
- Benennen Sie die Zuordnungen [uri] in workers2.properties von oaa in den neuen Namen des virtuellen Verzeichnisses um.

- Benennen Sie alle Kontexteinträge vom Typ `oaa` in `mod_jk2.conf` von `oaa` in den neuen Namen des virtuellen Verzeichnisses um.
- Benennen Sie den Pfad `<Context>` und die Attribute `docBase` in `server.xml` von `oaa` in den neuen Namen des virtuellen Verzeichnisses um.

Wichtig: Der von Ihnen gewählte Name für das virtuelle Verzeichnis wird Bestandteil des URL, den Benutzer zum Herstellen einer Verbindung mit Get-Services eingeben. Beispiel:
`http://Servername/<Neuer Name>/login.jsp`

Geben Sie für `<Tomcat>` den Pfad zur Tomcat-Installation ein. Der Standardpfad lautet:

`C:\Program Files\Peregrine\Common\Tomcat4`

Bearbeiten der Datei „workers2.properties“ für IIS

Für jeden Server, auf dem Tomcat-Instanzen installiert sind, gibt es nur eine Datei `workers2.properties`. Tomcat installiert die Datei `workers.properties` im Verzeichnis `conf` der primären Instanz von Tomcat. Diese Datei wird auch von allen anderen Tomcat-Instanzen auf dem betreffenden Server verwendet.

Die Datei `workers2.properties` gibt die Worker-Threads an, die der Webserver-Connector zur Kommunikation mit den Instanzen von Tomcat erstellt. Jede Tomcat-Instanz muss einen eigenen Kommunikationsanschluss verwenden. Als Host muss der Server angegeben werden, auf dem die Tomcat-Instanzen ausgeführt werden. (Wenn die Tomcat-Instanzen auf demselben Server ausgeführt werden wie Apache, muss `localhost` als Host angegeben werden.)

So bearbeiten Sie die Datei „worker2.properties“:

- 1 Öffnen Sie die Datei `workers2.properties` in einem Texteditor. (Diese Datei befindet sich im Verzeichnis `conf` der Tomcat-Installation.)
- 2 Erstellen Sie einen Eintrag `channel.socket` für jede Tomcat-Instanz (wird auch als Worker bezeichnet).

Beispiel:

```
[channel.socket:<Server>:<Anschluss>]
info=Beschreibung der Tomcat-Instanz
debug=0
tomcatId=<Server>:<Anschluss>
lb_factor=1
disabled=0
```

Geben Sie für *<Server>* den Namen des Servers ein, auf dem sich die Tomcat-Instanz befindet.

Geben Sie für *<Anschluss>* den Kommunikationsanschluss für die Tomcat-Instanz an.

lb_factor ist eine Zahl größer oder gleich 1, anhand derer IIS den Lastenausgleich für die Worker vornimmt. Wenn alle Worker auf Servern desselben Leistungsniveaus ausgeführt werden, müssen Sie alle *lb_factor*-Einstellungen auf denselben Wert setzen (in der Regel auf 1). Wenn einem Worker weniger Sitzungen zugewiesen werden sollen, verwenden Sie eine niedrigere *lb_factor*-Einstellung als für die anderen Worker.

- 3 Stellen Sie sicher, dass in den *uri*-Einstellungen das richtige virtuelle IIS-Verzeichnis angegeben wird. Standardmäßig lautet der Name dieses virtuellen Verzeichnisses *oaa*.

Wenn Sie ein anderes virtuelles Verzeichnis als *oaa* für das Ausführen von Get-Services festgelegt haben, müssen Sie die hier aufgeführten *uri*-Werte ändern.

Beispiel:

```
[uri:/oaa/servlet/*]
info=Präfixzuordnung

[uri:/oaa/*.jsp]
info=Erweiterungszuordnung
```

- 4 Speichern Sie die Datei.

Bearbeiten der Dateien „server.xml“ für IIS

Für jede gleichzeitig auszuführende Tomcat-Instanz wird eine eigene Datei *server.xml* benötigt. Diese Datei enthält die Daten, die Tomcat benötigt, um eine Verbindung zum Webserver herzustellen und die Webanwendungsdateien der Peregrine OAA-Plattform zu finden.

Hinweis: Erstellen Sie eine Sicherungskopie der Datei *server.xml*, bevor Sie sie bearbeiten.

So bearbeiten Sie die Dateien „server.xml“:

- 1 Jede Tomcat-Instanz verfügt über eine Datei `server.xml`. Öffnen Sie diese Datei in einem Texteditor.
- 2 Setzen Sie das Anschlussnummer-Attribut des Elements `<Server>` auf einen eindeutigen Wert, der nicht in Konflikt mit anderen Anschlussnummern von Tomcat steht.

Es wird empfohlen, beim Konfigurieren der Tomcat-Instanzen die Anschlussnummern 8005-8008 zu verwenden.

Beispiel:

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

- 3 Setzen Sie das Anschlussnummer-Attribut des Coyote-Connector-Elements `<Connector>` auf einen eindeutigen Wert, der nicht in Konflikt mit anderen Anschlussnummern von Tomcat steht.

Es wird empfohlen, beim Konfigurieren des Coyote-Connectors die Anschlussnummern 8009, 8011, 8013 und 8015 zu verwenden.

Beispiel:

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8009" minProcessors="5" maxProcessors="75" enableLookups="true"
redirectPort="8443" acceptCount="10" debug="0"
connectionTimeout="20000" useURIVValidationHack="false"
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler" />
```

- 4 Erstellen Sie im Element `<Context>` einen Eintrag für die erste bzw. Hauptinstanz von Tomcat und das Bereitstellungsverzeichnis von Get-Services.

Fügen Sie den Eintrag direkt oberhalb des Eintrags `examples` des Elements `Context` ein.

Beispiel:

```
<Context path="/oaa"
docBase="<Erste Tomcat-Installation>/webapps/oaa"
crossContext="false"
debug="0"
reloadable="false" >
</Context>
```

Geben Sie für `<Erste Tomcat-Installation>` im Attribut `docBase` den absoluten Pfad der ersten bzw. Hauptinstanz von Tomcat an.

- 5 Geben Sie im Element `<Engine>` den Servernamen und den Kommunikationsanschluss für jede Tomcat-Instanz an.
Geben Sie im Attribut `jvmRoute` die Serverinformationen an.

Beispiel:

```
<Engine jvmRoute="localhost:8009" name="Standalone"
defaultHost="localhost" debug="0">
```

- 6 Geben Sie im Element `<Host>` das von der ersten bzw. Hauptinstanz von Tomcat verwendete Verzeichnis **webapps** an.
Geben Sie im Attribut `appBase` die Serverinformationen an.

Beispiel:

```
<Host name="localhost" debug="0" appBase="<Erste
Tomcat-Installation>/webapps" unpackWARs="true" autoDeploy="true">
```

Geben Sie für `<Erste Tomcat-Installation>` im Attribut `appBase` den absoluten Pfad der ersten bzw. Hauptinstanz von Tomcat an.

- 7 Speichern Sie die Datei `server.xml`.
- 8 Wiederholen Sie Schritt 2 bis Schritt 7 für jede erstellte Kopie der Datei `server.xml`.

Bearbeiten der Dateien „jk2.properties“ für IIS

Sie müssen die Datei `jk2.properties` jeder Tomcat-Instanz ändern. Mit dieser Datei wird der Kommunikationsanschluss für `jk2` eingestellt.

So bearbeiten Sie die Dateien „jk2.properties“:

- 1 Öffnen Sie die Datei `jk2.properties` für eine Tomcat-Instanz in einem Texteditor.
- 2 Fügen Sie eine Zeile für den Anschluss `channelSocket` ein. Die Anschlussnummer muss der in der Datei `workers2.properties` für diese Tomcat-Instanz festgelegten Anschlussnummer entsprechen.

Beispiel:

```
channelSocket.port=8009
```

- 3 Speichern Sie die Datei.
- 4 Wiederholen Sie Schritt 1 bis Schritt 3 für jede Tomcat-Instanz.

Installieren von Tomcat-Instanzen als Dienste für IIS

Wenn Sie die Tomcat-Dateien bearbeitet haben, können Sie jede Tomcat-Instanz mit Hilfe von `installservice.bat` als Windows-Dienst installieren.

So installieren Sie Tomcat-Instanzen als Dienste für IIS:

- 1 Öffnen Sie eine DOS-Eingabeaufforderung und wechseln Sie zum Tomcat-Verzeichnis `bin`.
- 2 Geben Sie für jede zu erstellende Instanz von Tomcat den folgenden Befehl ein:

```
installservice <Dienstname> <Tomcat_Basis> <jvm_dll_Pfad>
```

Hierbei gilt: `<Dienstname>` ist der Name, den Sie dem Tomcat-Dienst zuweisen, `<Tomcat_Basis>` ist das Installationsverzeichnis der Instanz, für die Sie den Dienst erstellen, und `<jvm_dll_Pfad>` ist das Installationsverzeichnis von Java SDK.

Wenn Sie bereits die Umgebungsvariablen `CATALINA_HOME` und `JAVA_HOME` eingestellt haben, sind der zweite und dritte Parameter optional.

Beispiel:

```
installservice Tomcat8009 C:\Program Files\Peregrine\Common\Tomcat4  
C:\Program Files\Peregrine\Common\jdk1.3.1_05\jre\bin\server\jvm.dll
```

- 3 Wiederholen Sie Schritt 1 bis Schritt 2 für jeden zu erstellenden Tomcat-Dienst.

Testen des Lastenausgleichs für IIS

Wenn Sie zusätzliche Tomcat-Instanzen erstellt haben, können Sie anhand der folgenden Schritte testen, ob der Lastenausgleich durchgeführt wird.

So testen Sie den Lastenausgleich:

- 1 Starten Sie die Dienste für alle Tomcat-Instanzen.
Wenn Sie Tomcat als Dienst installiert haben, können Sie die einzelnen Instanzen über das Dialogfeld **Dienste** der Windows-Systemsteuerung starten.
- 2 Öffnen Sie einen Browser und melden Sie sich bei Get-Services an.

- 3 Führen Sie eine Aktion in Get-Services durch. Starten Sie beispielsweise eine Suche.
- 4 Melden Sie sich bei Get-Services ab.
- 5 Schließen Sie den Browser, um den Verbindungscache zu löschen.
- 6 Wiederholen Sie Schritt 1 bis Schritt 5 einmal für jede Tomcat-Instanz.
Wenn Sie z. B. 4 Tomcat-Instanzen erstellt haben, müssen Sie sich insgesamt 4-mal anmelden und abmelden.
Für den Lastenausgleich wird ein Round-Robin-Algorithmus eingesetzt, d. h., wenn der Lastenausgleich ordnungsgemäß ausgeführt wird, wird für jeden Anmeldeversuch eine andere Tomcat-Instanz verwendet.
- 7 Laden Sie die Datei `archway.log` herunter.
Sie können die Datei `archway.log` von der Seite **Administration > Serverprotokoll** herunterladen.
- 8 Öffnen Sie die Datei `archway.log` in einem Texteditor.
- 9 Überprüfen Sie, ob in den Verbindungsdetails eine unterschiedliche Tomcat-Instanz für jede Verbindung aufgeführt wird.
Wenn für jede Verbindung eine andere Tomcat-Instanz angegeben wird, wird der Lastenausgleich ordnungsgemäß durchgeführt.
Wird hingegen für jede Verbindung die gleiche Tomcat-Instanz angegeben, erfolgt kein Lastenausgleich im System und der Fehler muss behoben werden.

5 | Verwalten von ServiceCenter

KAPITEL

Get-Services wird zur Verwendung von ServiceCenter als Backend-System konfiguriert. Mit ServiceCenter greift Get-Services auf ServiceCenter Incident Management zu. Im Verwaltungsmodul von Get-Services können Sie auch die Komponente Service Management hinzufügen. In ServiceCenter 5.0 ist außerdem Change Management verfügbar.

Wichtig: Um die Installation für Get-Services fortzusetzen, muss ein Benutzer mit Verwaltungsrechten die in diesem Kapitel beschriebenen Verfahren im ServiceCenter-Backend-System durchführen. Detaillierte Informationen hierzu finden Sie im *ServiceCenter-Verwaltungshandbuch*.

Dieses Kapitel umfasst folgende Abschnitte:

- *Konfigurieren von ServiceCenter für Get-Services* auf Seite 178
- *Aktualisieren von ServiceCenter* auf Seite 178
- *Anwenden von Entladefdateien auf ServiceCenter* auf Seite 179

Konfigurieren von ServiceCenter für Get-Services

Über ServiceCenter kann Get-Services auf die Module Incident Management und Service Management von ServiceCenter 4.x und ServiceCenter 5.x zugreifen. Außerdem kann Get-Services auf das Modul Change Management von ServiceCenter 5.x zugreifen.

Die Authentifizierung der Benutzer erfolgt anhand von ServiceCenter-Bearbeiterdatensätzen und speziellen Berechtigungswörtern für Get-Services. Weitere Informationen zur Benutzerauthentifizierung finden Sie im Kapitel „Sicherheit“ des *Get-Services-Verwaltungshandbuch*.

Aktualisieren von ServiceCenter

Sie müssen ServiceCenter 4.x aktualisieren, um die Get-Services 4.0-Funktion zum Öffnen zu einer Anfrage gehöriger Incidents verwenden zu können. Diese Änderung erleichtert das Übernehmen von Anfragedaten in Incidents. Das Verfahren muss für ServiceCenter 5.x nicht durchgeführt werden.

So aktualisieren Sie ServiceCenter 4.x mit den neuen Get-Services-Funktionen:

- 1 Klicken Sie im Register **Dienstprogramme** von ServiceCenter auf **Werkzeuge** und dann auf **Links**, um das Dialogfeld **Link** zu öffnen.

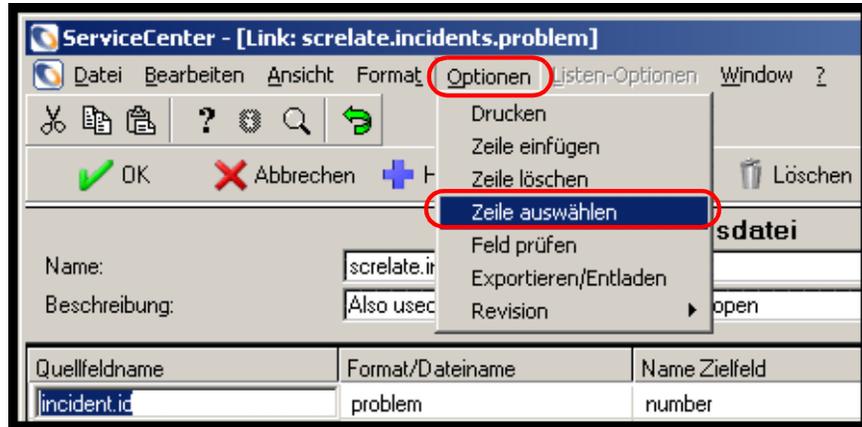
Geben Sie den Namen der Verknüpfungsdatei im Textfeld **Name** ein.



- 2 Geben Sie `screlate.incidents.problem` im Textfeld **Name** ein und klicken Sie anschließend auf **Suchen** (Search).

- 3 Markieren Sie **incident.id** und wählen Sie dann **Optionen > Zeile auswählen** (Select Line).

Markieren Sie **incident.id** in der Spalte **Quellfeldname**.



- 4 Ändern Sie in der Zeile **description** den Text in der Spalte **Zielfeld (Zielfeld/Übertragen an)** von **Action**, **action in action**.

The screenshot shows the ServiceCenter application window with the title bar 'ServiceCenter - [Link: screlate.incidents.problem]'. The 'description' row in the table is selected. The 'Zielfeld (Zielfeld/Übertragung an)' column is highlighted with a red circle.

Feld (Von/Quelle):	Format/Datei (Nach/Ziel):	Feld (Nach/Ziel):
incident.id	problem	number
Kommentar:		
Abfrage:		
QBE-Format:		
Ausdrücke:		
Quellfeld (Einfügen in/Übertragen von)		Zielfeld (Zielfeld/Übertragung an)
description		action

Klicken Sie auf **Speichern (Save)**.

Anwenden von Entladedateien auf ServiceCenter

Der nächste Schritt in der Get-Services-Installation besteht in der Installation neuer Formate, Anzeigeoptionen und RAD-Codes, die in Entladedateien enthalten sind. Welche Entladedateien Sie anwenden können hängt vom Verwendungszweck von Get-Services ab. Auf ServiceCenter 4.x und ServiceCenter 5.x werden unterschiedliche Entladedateien angewendet. (Siehe Tabellen in den folgenden Abschnitten.)

Bei dem in den folgenden Tabellen referenzierten Verzeichnis `\oaa` handelt es sich um ein Unterverzeichnis, in dem die Peregrine OAA-Serverdateien installiert werden. Beispiel: Wenn Sie Tomcat 4.1.12 als Peregrine OAA-Server installiert haben, ist `\oaa` ein Unterverzeichnis des Verzeichnisses `\Tomcat4\webapps` auf dem Computer, auf dem Sie Tomcat 4.1.12 installiert haben.

Warnung: Durch dieses Verfahren werden bestehende Dateien bei der Anwendung der neuen Entladedateien überschrieben. Wenn Sie das System mit Tailoring angepasst haben, gehen die von Ihnen vorgenommenen Änderungen verloren.

ServiceCenter 4.x-Entladedateien mit Get-Services 4.0

Der Pfad für die Entladedateien lautet `oaa\WEB-INF\etc\Version4\`.

Datei	Enthaltene Informationen
<code>axcessm.unl</code>	rad: axces.sm
<code>callgmt.unl</code>	eventmap: e service management eventregister: esmin
<code>callupdate.unl</code>	eventmap: e service management
<code>epmx.unl</code>	capability: getit.answers getit.service contacts: Hartke Tossi eventmap: e problem close e problem open e problem update eventregister: epmc epmo epmu operator: Hartke Tossi

Datei	Enthaltene Informationen
gs22upd.unl	rad: apm.bg.edit apm.get.display.format
openrelated.unl	eventregister: epmosmu eventmap: e problem open smu rad: axces.apm.epmosmu cc.open.related.incident

ServiceCenter 5.0-Entladefile mit Get-Services 4.0

Der Pfad für die Entladefile lautet oaa\WEB-INF\etc\Version5\.

Datei	Enthaltene Informationen
axcesm.unl	rad: axces.sm
callmgmt.unl	eventmap: e service management eventregister: esmin
callupdate.unl	eventmap: e service management
epmx.unl	capability: getit.answers getit.service contacts: Hartke Tossi eventmap: e problem close e problem open e problem update eventregister: epmc epmo epmu operator: Hartke Tossi

Datei	Enthaltene Informationen
gs22upd.unl	rad: apm.bg.edit apm.get.display.format
openrelated.unl	eventregister: eposmu eventmap: e problem open smu rad: axces.apm.eposmu cc.open.related.incident

ServiceCenter 5.0-Entladedateien mit Get-Services 4.0 Change Management

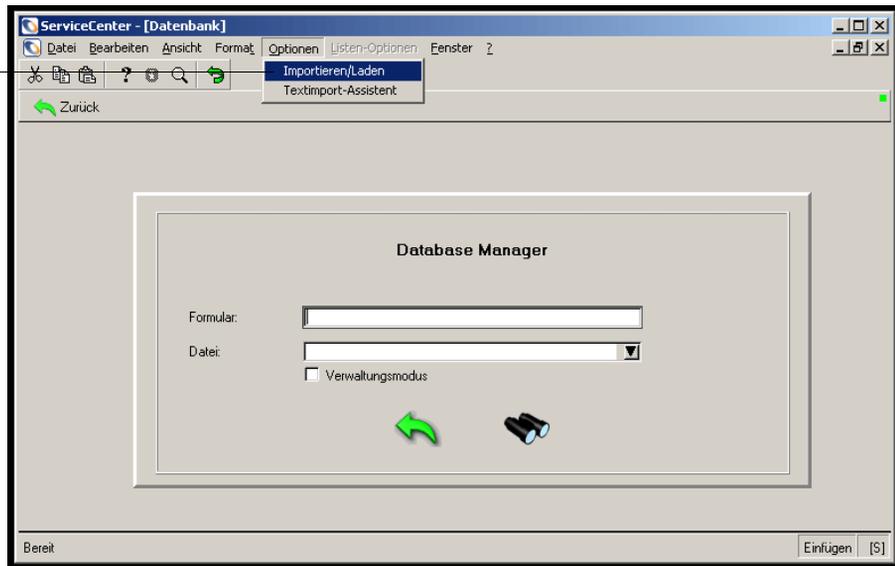
Wenn Sie Get-Services 4.0 mit Change Management einsetzen möchten, müssen Sie die folgenden Entladedateien anwenden. Der Pfad für die Entladedateien lautet oaa\WEB-INF\etc\Version5\.

Datei	Enthaltene Informationen
sc5_cmreopenall.unl	rad: cm3.unlock axces.cm3 cm3.lock cm3r.main cm3t.main process: cm.update.save eventmap: cm3r
sc5_cmsstructure.unl	eventmap: cm3t

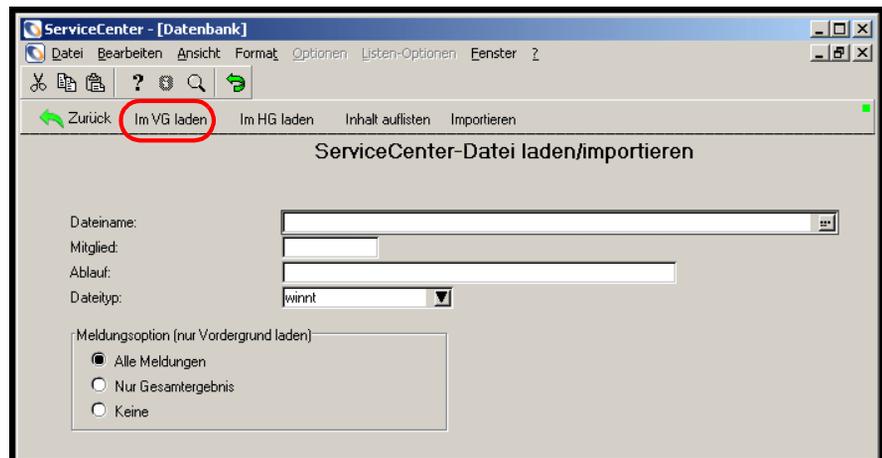
So wenden Sie Entladedateien auf ServiceCenter an:

- 1 Wechseln Sie in ServiceCenter zum Register **Werkzeuge** und wählen Sie **Database Manager**. Das Dialogfeld **Database Manager** wird geöffnet.

Wählen Sie im Menü **Optionen** den Befehl **Importieren/Laden** (Import/Load).



- 2 Wählen Sie im Menü **Optionen** den Befehl **Importieren/Laden** (Import/Load), um das Dialogfeld **ServiceCenter-Datei laden/importieren** zu öffnen.



- 3 Geben Sie im Textfeld **Dateiname** den Pfad der für Ihre Integration geeigneten Entladefdatei ein. Informationen hierzu entnehmen Sie den vorausgegangenen Tabellen.
- 4 Klicken Sie auf **Im VG laden** (Load FG) und achten Sie auf etwaige Fehler.
- 5 Wiederholen Sie Schritt 3 und Schritt 4 für alle für Ihre Integration erforderlichen Entladefdateien.

6 Konfigurieren des Adapters

KAPITEL

Die letzten Konfigurationseinstellungen, die zum Abschluss der Installation von Get-Services erforderlich sind, werden im Verwaltungsmodul von Peregrine Portal vorgenommen. Die durchzuführenden Aufgaben umfassen die Konfiguration der Verbindungsparameter, die Einstellung der Anmeldeinformationen für das Backend-System sowie die Überprüfung der Adapterverbindungen für Get-Services.

Eine ausführliche Beschreibung weiterer Funktionen, die zur Durchführung von Verwaltungs- und Wartungsaufgaben mit Hilfe des Peregrine Portal-Verwaltungsmoduls erforderlich sind, finden Sie im *Get-Services-Verwaltungshandbuch*.

Hinweis: Der Name des Moduls Incident Management lautet in ServiceCenter-Versionen vor Version 4.x Problem Management. Für einige Incident Management-Parameter wird *Problem*-Terminologie verwendet, da sie *Problem*-Tabellen in ServiceCenter zugeordnet sind.

Dieses Kapitel behandelt folgende Themen:

- *Zugreifen auf das Peregrine Portal-Verwaltungsmodul* auf Seite 186
- *Konfigurieren von Verbindungen zum Backend-System* auf Seite 190.

Zugreifen auf das Peregrine Portal-Verwaltungsmodul

Über die Seite **Administratoranmeldung** kann auf das Peregrine Portal-Verwaltungsmodul zugegriffen werden. In diesem Modul werden die Einstellungen für das Peregrine-System vorgenommen.

Hinweis: Nach dem Installieren und Einrichten von Get-Services muss sich ein Verwalter als ServiceCenter-Benutzer mit **getit.admin**-Rechten anmelden, um auf das Verwaltungsmodul zugreifen und die Get-Services-Integration mit ServiceCenter verwalten zu können. Eine Liste mit Berechtigungswörtern für den Zugriff finden Sie im *Get-Services-Verwaltungshandbuch*.

Als Standardverwalter **Admin** können Sie auf das Verwaltungsmodul zugreifen, ohne mit einem Backend-System verbunden zu sein. Nachdem Sie Ihren Benutzernamen im Register **Allgemein** konfiguriert haben, können Sie auch über das Navigationsmenü auf das Verwaltungsmodul zugreifen.

Wichtig: Wenn Sie Parameter über das Verwaltungsmodul ändern, wird im Verzeichnis `\<Anwendungsserver>\webapps\oaa\WEB-INF` eine Datei mit dem Namen `local.xml` erstellt, um diese Parameter zu speichern. Erstellen Sie vor einer Neuinstallation von Get-Services eine Kopie dieser Datei und speichern Sie diese außerhalb der Installationsumgebung von Get-Services. Andernfalls gehen Ihre Parameterwerte bei der Neuinstallation verloren.

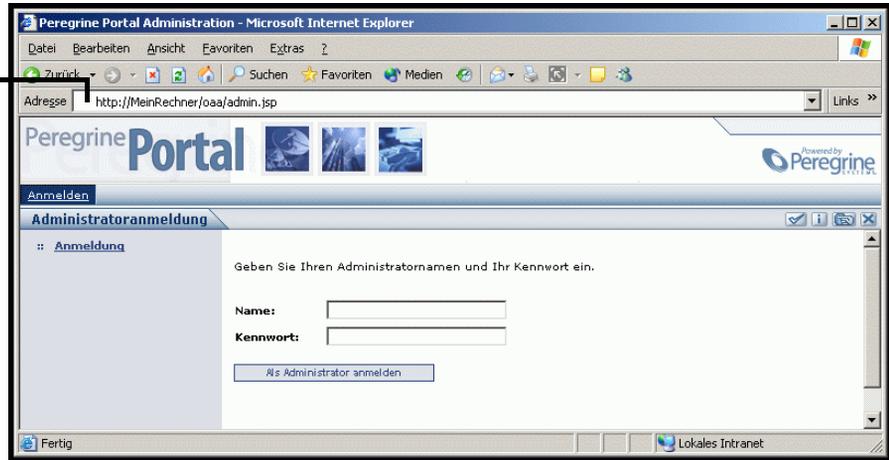
So greifen Sie auf die Seite „Administratoranmeldung“ von Peregrine Portal zu:

- 1 Stellen Sie sicher, dass der Anwendungsserver (z. B. Tomcat) ausgeführt wird.
- 2 Geben Sie Folgendes im Adressfeld Ihres Webbrowsers ein:
`<Hostname>/oaa/admin.jsp`

- Drücken Sie die Eingabetaste, um die Seite **Administratoranmeldung** zu öffnen.

Geben Sie den Hostnamen ein, um eine Verbindung zum lokalen Server herzustellen.

Der Standardverwaltername ist **Admin**.



- Geben Sie **Admin** im Feld Name ein.
Bei der ersten Anmeldung muss kein Kennwort eingegeben werden.
- Klicken Sie auf **Als Administrator anmelden**, um die Seite **Systemsteuerung** zu öffnen.

Legen Sie im Verwaltungsmodul die Get-Services-Einstellungen fest.



Aktivitätenmenü

Das Aktivitätenmenü am linken Seitenrand bietet Zugriff auf verschiedene Verwaltungsdienstprogramme für Get-Services. Sie werden im weiteren Verlauf dieses Installationsverfahrens auf einige dieser Dienstprogramme zugreifen. Die Dienstprogramme und ihre Funktionen werden im *Get-Services-Verwaltungshandbuch* ausführlich beschrieben.

Verwenden der Systemsteuerung

Über die Seite **Systemsteuerung** können Sie den Status der Verbindungen zu den Datenbanken prüfen, auf die Sie über Get-Services und die Webanwendungen zugreifen. Darüber hinaus können Sie die Verbindung zwischen dem Archway-Servlet und den Adaptern zu den Backend-Systemen zurücksetzen.

Hinweis: Beim ersten Zugriff auf die Seite **Systemsteuerung** weisen alle Ziele den Status *Getrennt* auf. Dies ändert sich, wenn Sie die Ziele mit dem weiter unten beschriebenen Verfahren festgelegt haben.

Wenn Sie im Verwaltungsmodul von Peregrine Portal Konfigurationsänderungen vorgenommen haben, wechselt das System wieder zur Seite **Systemsteuerung** und blendet oben auf der Seite eine Informationsmeldung ein.

So setzen Sie die Verbindung zwischen dem Archway-Servlet und dem Backend-System zurück:

- 1 Klicken Sie auf **Server zurücksetzen**.

Die folgende Meldung zeigt an, dass die Verbindungen zurückgesetzt wurden.

Archway-Servlet und zugehörige Adapterverbindungen erfolgreich zurückgesetzt.

- 2 Überprüfen Sie die vorgenommenen Änderungen in der Tabelle **Verbindungsstatus**.

Verwenden der Seite Einstellungen

Klicken Sie im Aktivitätenmenü auf **Einstellungen**, um die aktuellen Parametereinstellungen zu öffnen. Die Seite **Einstellungen** ist in Register unterteilt. Welche Register angezeigt werden, hängt von den installierten Webanwendungen und den verwendeten Adaptern ab. Das Register **Allgemein** steht bei allen Installationen zur Verfügung.

So öffnen Sie die Seite „Einstellungen“:

- 1 Klicken Sie im Aktivitätenmenü des Verwaltungsmoduls auf **Einstellungen**. Die Seite **Einstellungen**, über die Sie auf die Register für die Konfiguration der Get-Services-Einstellungen zugreifen können, wird geöffnet. Standardmäßig wird das Register **Allgemein** angezeigt.



- 2 Klicken Sie zum Ändern von Einstellungsparametern auf das entsprechende Register.

Wenn Sie den standardmäßigen Parameterwert einer Einstellung ändern und diesen speichern, wird in der zugehörigen Beschreibungsspalte ein Link hinzugefügt, in dem der Standardwert in Klammern aufgeführt wird, so dass Sie diesen bei Bedarf wiederherstellen können.

So stellen Sie die Standardeinstellungen wieder her:

- 1 Wechseln Sie zum gewünschten Einstellungsparameter.
- 2 Wählen Sie in der Beschreibungsspalte neben dem geänderten Parameter den Link **Klicken Sie für den Standardwert hier** aus.

Anonymes Kennwort:	Kennwort des anonymen Benutzers für den Mail-Adapter
Host für ausgehende E-Mails:	Vollständiger Name bzw. IP-Adresse des Rechners, auf dem der Server für ausgehende E-Mails gehostet wird. Klicken Sie für den Standardwert hier: [mailhost]
Benutzer-ID für ausgehende E-Mails:	Benutzer-ID für den Zugriff auf den Server für ausgehende E-Mails.

- 3 Um die Standardeinstellung wiederherzustellen, wählen Sie den Link **Klicken Sie für den Standardwert hier: [mailhost]** und klicken Sie anschließend auf **Speichern**.

Der Parameter wird wieder auf den Standardwert zurückgesetzt.

Anonymes Kennwort:	Kennwort des anonymen Benutzers für den Mail-Adapter
Host für ausgehende E-Mails:	Vollständiger Name bzw. IP-Adresse des Rechners, auf dem der Server für ausgehende E-Mails gehostet wird. Klicken Sie für den Standardwert hier: [mailhost]
Benutzer-ID für ausgehende E-Mails:	Benutzer-ID für den Zugriff auf den Server für ausgehende E-Mails.

Konfigurieren von Verbindungen zum Backend-System

Standardmäßig wird Get-Services so installiert, dass die Verbindung mit ServiceCenter über einen speziellen Adapter (SCAdapter) hergestellt wird.

Einstellen des ServiceCenter-Adapters

In diesem Abschnitt wird beschrieben, wie die Einstellungen des Backend-Systems konfiguriert und überprüft werden und wie die Verbindung zur ServiceCenter-Datenbank hergestellt wird.

So stellen Sie den ServiceCenter-Adapter ein:

- 1 Klicken Sie auf der Verwaltungsseite auf **Einstellungen**.
- 2 Klicken Sie oben auf der Seite auf das Register **ServiceCenter**, um die Seite mit den ServiceCenter-Einstellungen zu öffnen.

Portal	MobileServer	AssetCenter	Allgemein	Service Desk	Portal-DB	Designs
Webanwendung	Protokollieren	ServiceCenter	Get-Resources	XSL	E-Mail	
Host:	Hostname des ServiceCenter-Servers					
localhost						
Port:	Port-Nummer des ServiceCenter-Servers					
12670						
Protokolldatei:	Zugriffspfad zu der von der ServiceCenter-Client-Verbindung verwendeten ServiceCenter-Protokolldatei.					
Admin-Benutzer:	Administrationsbenutzer, auf den das Peregrine-Portal beim Durchführen von Aufgaben, z.B. Benutzerauthentifizierung und Registrierung in ServiceCenter, zurückgreift.					
falcon						
Admin-Kennwort:	Kennwort des Admin-Benutzers für ServiceCenter					
Anonymer Benutzer:	Anonymer Benutzername, der verwendet wird, wenn ein unbekannter Benutzer versucht, mit ServiceCenter zu kommunizieren.					
falcon						
Anonymes Kennwort:	Kennwort des anonymen Benutzers für ServiceCenter					
Standardfunktionen:	Liste der Standard-Zugriffsrechte, die allen Benutzern ungeachtet ihres jeweiligen Profils zugeteilt werden. Die Einträge sind durch Semikolon getrennt. Die Zugriffsrechte werden den Zieladaptern folgendermaßen zugeordnet: portalDB(getit.portal)					
Adapter:	Voller Klassenpfad für diesem Ziel zugeordnete Adapter.					
com.peregrine.oaa.adapter.sc.SCAdapter						

- 3 Aktualisieren Sie bei Bedarf die folgenden Felder:

Feldname	Standardwert	Wertbeschreibung
Host	localhost	Hostname des ServiceCenter-Servers.
Port	12670	Anschlussnummer des ServiceCenter-Servers.
Protokolldatei	(keine Angabe)	Der Pfad zum SC-Protokoll, das von der ServiceCenter-Clientverbindung verwendet wird.

Feldname	Standardwert	Wertbeschreibung
Admin-Benutzer	falcon	Anmeldename für Verwalter, der beim Durchführen von Aufgaben wie Benutzerauthentifizierung und -registrierung für Peregrine Portal verwendet wird.
Admin-Kennwort	(keine Angabe)	Verwalterkennwort.
Anonymer Benutzer	falcon	Der Name, der verwendet wird, wenn ein unbekannter Benutzer versucht, eine Verbindung mit ServiceCenter herzustellen.
Anonymes Kennwort	(keine Angabe)	Anonymes Benutzerkennwort.
Adapter	com.peregrine.oaa.adapter.sc.SCAdapter	Voller Klassenpfad für diesem Ziel zugeordnete Adapter.

- Nachdem Sie die gewünschten Änderungen vorgenommen haben, klicken Sie auf **Speichern**, um zur Seite **Systemsteuerung** zurückzukehren, und klicken Sie anschließend auf **Server zurücksetzen**.

Beheben von Fehlern bei der ServiceCenter-Datenbankverbindung

Treten beim Herstellen einer Verbindung zwischen Peregrine OAA und der ServiceCenter-Datenbank Probleme auf, überprüfen Sie die folgenden Einstellungen.

So beheben Sie Fehler bei der ServiceCenter-Datenbankverbindung:

- Überprüfen Sie auf der Seite **Systemsteuerung** des Verwaltungsmoduls den Status der Datenbankverbindung.
 - Weist **sc** den Status **Getrennt** auf, vergewissern Sie sich, dass der ServiceCenter-Dienst ausgeführt wird und die ServiceCenter-Konsole gestartet wurde.
- Gehen Sie im Register **ServiceCenter** der Seite **Einstellungen** folgendermaßen vor:
 - Überprüfen Sie, ob die Parameter in den Feldern **Host** und **Port** korrekt sind.

- b Stellen Sie sicher, dass für Get-Services der richtige Admin-Benutzername und das richtige Admin-Kennwort festgelegt wurden. Diese Werte müssen mit den Werten übereinstimmen, die für die Verwalteranmeldung beim ServiceCenter-Backend-System verwendet werden.
- 3 Vergewissern Sie sich, dass eine vollständige Clientverbindung mit ServiceCenter besteht, indem Sie einen Client starten, der auf den Anschluss verweist, der im Register **ServiceCenter** auf der Seite **Einstellungen** aufgeführt wird.

Einstellen des Portal-DB-Adapters

Damit Peregrine Portal die Einstellungen und Anpassungen der Portaloberfläche speichern kann, muss eine Verbindung über einen Datenbankadapter bestehen. Benutzer können erst dann Personalisierungsänderungen an der Startseite von Peregrine Portal vornehmen bzw. solche Änderungen anzeigen, wenn ein Datenbankadapter für die Portalseite festgelegt wurde.

So legen Sie den Portal-DB-Adapter fest:

- 1 Klicken Sie auf der Seite **Admin-Einstellungen** auf das Register **Portal-DB**, um die Seite mit den Einstellungen für die Portal-DB zu öffnen.

Portal	MobileServer	AssetCenter	Allgemein	Service Desk	Portal-DB	Designs
Webanwendung	Protokollieren	ServiceCenter	Get-Resources	XSL	E-Mail	
Standardfunktionen:			Liste der Standard-Zugriffsrechte, die allen Benutzern ungeachtet ihres jeweiligen Profils zugeteilt werden. Die Einträge sind durch Semikolon getrennt. Die Zugriffsrechte werden den Zieladaptern folgendermaßen zugeordnet: portalDB(getit.portal)			
portalDB(getit.portal)						
Aliasname für:			Gibt die Zielkonfiguration an, für die das Ziel als Aliasname dient. Klicken Sie für den Standardwert hier: <input type="checkbox"/>			
ac						

In der folgenden Tabelle werden die Parameter aufgeführt, durch die der Portal-DB-Adapter für die Portalseite festgelegt wird.

Feldname	Standardwert	Wertbeschreibung
Standardfunktionen	getit.portal	Eine Liste der standardmäßigen Zugriffsberechtigungen für alle Benutzer, unabhängig von ihren Benutzerrollen. Beim standardmäßigen Berechtigungswort der Portal-DB kann es sich um ein beliebiges OAA-Berechtigungswort handeln. Peregrine empfiehlt die Verwendung der Standardeinstellung, um eine optimale Sicherheit zu gewährleisten.
Aliasname für	sc	Der Name des Datenbankadapters, der zum Speichern von Einstellungen und Änderungen an der Peregrine Portal-Seite verwendet werden soll. Verwenden Sie den für Ihre Anwendung erforderlichen Adapter.

- 2 Geben Sie **sc** im Feld **Aliasname für** ein, um den Datenbankadapter für ServiceCenter einzustellen.
- 3 Klicken Sie auf **Speichern**, um zur Seite **Systemsteuerung** zurückzukehren, und klicken Sie anschließend auf **Server zurücksetzen**.

Einstellen des Datenbankadapters für die Webanwendung

Um Einstellungen und Anpassungen an der Oberfläche der Webanwendung speichern zu können, benötigt Get-Services eine Verbindung über einen Datenbankadapter. Benutzer können erst dann Personalisierungsänderungen an der Webanwendung vornehmen, wenn ein Datenbankadapter für die Webanwendung festgelegt wurde.

So stellen Sie den Datenbankadapter für die Webanwendungen ein:

- 1 Klicken Sie auf der Seite **Admin-Einstellungen** auf das Register **Webanwendung**, um die Seite mit den Einstellungen für die Webanwendungen zu öffnen.

Portal	MobileServer	AssetCenter	Allgemein	Service Desk	Portal-DB	Designs
Webanwendung	Protokollieren	ServiceCenter	Get-Resources	XSL	E-Mail	
Standardfunktionen:			Liste der Standard-Zugriffsrechte, die allen Benutzern ungeachtet ihres jeweiligen Profils zugeteilt werden. Die Einträge sind durch Semikolon getrennt. Die Zugriffsrechte werden den Zieladaptern folgendermaßen zugeordnet: portalDB(getit.portal)			
<input type="text" value="weblication(oaa.bva)"/>						
Aliasname für:			Gibt die Zielkonfiguration an, für die das Ziel als Aliasname dient. Klicken Sie für den Standardwert hier: []			
<input type="text" value="sc"/>						

- 2 Geben Sie **sc** im Feld **Aliasname für** ein und klicken Sie anschließend auf die Schaltfläche **Speichern**, um zur Seite **Systemsteuerung** zurückzukehren.
- 3 Klicken Sie auf **Speichern**, um zur Seite **Systemsteuerung** zurückzukehren, und klicken Sie anschließend auf **Server zurücksetzen**.

Einstellen der Service Desk-Parameter

In diesem Abschnitt werden die speziellen Get-Services-Parameter erläutert. Diese Einstellungen werden im Register **Service Desk** auf der Seite **Admin-Einstellungen** konfiguriert.

Bei Incident Management handelt es sich um das Modul, das standardmäßig für Incident-(Problem)-Tickets verwendet wird, die in Get-Services mit dem ServiceCenter-Adapter geöffnet werden. Sollen Endbenutzer auch ServiceCenter-Anfrage-Tickets erstellen, müssen Sie das Modul Service Management aktivieren und die entsprechenden Get-Services-Einstellungen konfigurieren.

So aktivieren Sie Service Management für Get-Services:

- 1 Klicken Sie im Verwaltungsmodul von Peregrine Portal auf **Einstellungen** und anschließend auf das Register **Service Desk**.
- 2 Wählen Sie **Ja** für den Parameter **Service Management aktivieren** aus, wenn die über Get-Services erstellten Tickets im Modul Service Management der ServiceCenter-Installation geöffnet werden sollen.



- 3 Klicken Sie auf **Speichern**, um zur Seite **Systemsteuerung** zurückzukehren.
- 4 Klicken Sie auf **Server zurücksetzen**, um Ihre Änderungen zu speichern.

Service Management

Wenn Sie das Modul Service Management aktivieren, können Benutzer sowohl Incident- als auch Anfrage-Tickets öffnen und anzeigen. Sie können dann die Service Management-Parameter nach Bedarf ändern.

So richten Sie Service Management für Get-Services ein:

- 1 Klicken Sie im Verwaltungsmodul von Peregrine Portal auf **Einstellungen**, dann auf das Register **Service Desk** und blättern Sie anschließend zum Abschnitt **ServiceCenter-Einstellungen für Service Management**.

ServiceCenter-Einstellungen für Service Management	
Service Management aktivieren: <input type="radio"/> Ja <input checked="" type="radio"/> Nein	Aktivieren Sie Service Management, wenn die über Services erstellten Tickets im Modul Service Management der ServiceCenter-Installation geöffnet werden sollen. Klicken Sie für den Standardwert hier: [Nein]
Standardkategorie für Service Management: example	Geben Sie die bei der Erstellung von Anfrage-Tickets zu verwendende Standardkategorie ein. Sie wird nur verwendet, wenn Service Management aktiviert ist. Klicken Sie für den Standardwert hier: [tbd]
Standard-Unterkategorie für Service Management: an	Geben Sie die bei der Erstellung von Anfrage-Tickets zu verwendende Standard-Unterkategorie ein. Sie wird nur verwendet, wenn Service Management aktiviert ist. Klicken Sie für den Standardwert hier: [tbd]
Standard-Produkttyp für Service Management: an	Geben Sie den bei der Erstellung von Anfrage-Tickets zu verwendenden Standard-Produkttyp ein. Er wird nur verwendet, wenn Service Management aktiviert ist. Klicken Sie für den Standardwert hier: [tbd]
Standard-Problemtyp für Service Management: routers	Geben Sie den bei der Erstellung von Anfrage-Tickets zu verwendenden Standard-Problemtyp ein. Er wird nur verwendet, wenn Service Management aktiviert ist. Klicken Sie für den Standardwert hier: [tbd]
Standardgewichtung für Ticket: Niedrig	Wählen Sie die bei der Erstellung von Tickets zu verwendende Standardgewichtung.
Standard-Standortkategorie für Service Management: A	Geben Sie die bei der Erstellung von Anfrage-Tickets zu verwendende Standard-Standortkategorie ein. Sie wird nur verwendet, wenn Service Management aktiviert ist.
Standard-Zuweisungsgruppe für Service Management: DEFAULT	Standard-Zuweisungsgruppe dient dem Weiterleiten von Tickets. Sie wird nur verwendet, wenn Service Management aktiviert ist.

- 2 Aktualisieren Sie bei Bedarf die folgenden Felder, um die ServiceCenter-Einstellungen für Service Management für Get-Services festzulegen.

Feldname	Standardwert	Wertbeschreibung
Service Management aktivieren	Nein	Wird dieser Wert auf Ja gesetzt, werden in Get-Services erstellte Tickets im Modul Service Management Ihrer ServiceCenter-Installation geöffnet.
Standardkategorie für Service Management	example	Die ServiceCenter-Kategorie, die standardmäßig zum Erstellen von Anfrage-Tickets verwendet wird. Dieser Parameter ist nur relevant, wenn Service Management aktiviert ist.
Standard-Unterkategorie für Service Management	tbd	Die ServiceCenter-Unterkategorie, die standardmäßig zum Erstellen von Anfrage-Tickets verwendet wird. Dieser Parameter ist nur relevant, wenn Service Management aktiviert ist.
Standard-Produkttyp für Service Management	tbd	Der Produkttyp, der standardmäßig zum Erstellen von Anfrage-Tickets verwendet wird. Dieser Parameter ist nur relevant, wenn Service Management aktiviert ist.
Standard-Problemtyp für Service Management	tbd	Gibt die standardmäßige ServiceCenter-Kategorie für Problem-Tickets an.
Standardgewichtung für Ticket	Niedrig	Die Standardgewichtung zum Erstellen von Tickets über die Dropdown-Liste. Dieser Parameter ist nur relevant, wenn Service Management aktiviert ist.
Standard-Standortkategorie für Service Management	A	Die Standortkategorie, die standardmäßig zum Erstellen von Anfrage-Tickets verwendet wird. Dieser Parameter ist nur relevant, wenn Service Management aktiviert ist.

Feldname	Standardwert	Wertbeschreibung
Standard-Zuweisungsgruppe für Service Management	Default	Wird zum Weiterleiten von Tickets verwendet, wenn Service Management aktiviert ist.

- 3 Klicken Sie auf **Speichern**, um zur Seite **Systemsteuerung** zurückzukehren.
- 4 Klicken Sie auf **Server zurücksetzen**, um Ihre Änderungen zu speichern.

Incident Management

Die folgenden Parameter legen die Standardeinstellungen fest, die von Incident Management verwendet werden, wenn ein Benutzer ein Get-Services-Ticket öffnet. Diese Einstellungen befinden sich im Register **Service Desk** auf der Seite **Admin-Einstellungen**.

So richten Sie Incident Management für Get-Services ein:

- 1 Klicken Sie im Verwaltungsmodul von Peregrine Portal auf **Einstellungen**, dann auf das Register **Service Desk** und blättern Sie anschließend zum Abschnitt **ServiceCenter-Einstellungen für Incident Management**.

ServiceCenter-Einstellungen für Incident Management	
Standardkategorie für Ticket: example 	Geben Sie die beim Einfügen eines neuen Tickets zu verwendende Standardkategorie ein. Wählen Sie mittels Suche über das Lupensymbol eine GÜLTIGE Kategorie aus.
Standard-Unterkategorie für Incident Management: <input type="text" value="tbd"/>	Geben Sie die bei der Erstellung von Incident-Tickets zu verwendende Standard-Unterkategorie ein.
Standard-Produkttyp für Incident Management: <input type="text" value="tbd"/>	Geben Sie den bei der Erstellung von Incident-Tickets zu verwendenden Standard-Produkttyp ein.
Standard-Problemtyp für Incident Management: <input type="text" value="tbd"/>	Geben Sie den bei der Erstellung von Incident-Tickets zu verwendenden Standard-Problemtyp ein.
Standardgewichtung für Ticket: <input type="text" value="2 - Dringend"/>	Wählen Sie die bei der Erstellung von Tickets zu verwendende Standardgewichtung. Klicken Sie für den Standardwert hier: [3 - Normal]
Standardpriorität für Ticket: <input type="text" value="Priorität 2"/>	Wählen Sie die bei der Erstellung von Tickets zu verwendende Standardpriorität. Klicken Sie für den Standardwert hier: [Priorität 3]
Standard-Standortkategorie für Incident Management: <input type="text" value="A"/>	Geben Sie die bei der Erstellung von Incident-Tickets zu verwendende Standard-Standortkategorie ein.
Liste der Ziel-Aliasnamen: <input type="text" value="sc"/>	Gibt eine Liste mit Ziel-Aliasnamen an, die von den Webanwendungen in diesem Paket verwendet werden. Die Einträge sind durch ein Semikolon getrennt.

In der folgenden Tabelle werden die Parameter beschrieben, die die ServiceCenter-Einstellungen für Incident Management für Get-Services festlegen.

Feldname	Standardwert	Wertbeschreibung
Standardkategorie für Ticket	example	Die Kategorie, die standardmäßig zum Erstellen neuer Incident-Tickets verwendet wird.
Standard-Unterkategorie für Incident Management	tbd	Die Unterkategorie, die standardmäßig zum Erstellen von Incident-Tickets verwendet wird.
Standard-Produkttyp für Incident Management	tbd	Der Produkttyp, der standardmäßig zum Erstellen von Incident-Tickets verwendet wird.
Standard-Problemtyp für Incident Management	tbd	Der Problemtyp, der standardmäßig zum Erstellen von Incident-Tickets verwendet wird.
Standardgewichtung für Ticket	3 - Normal	Die Gewichtung, die standardmäßig zum Erstellen von Incident-Tickets verwendet wird.
Standardpriorität für Ticket	Priorität 3	Die Standardpriorität zum Erstellen von Tickets.
Standard-Standortkategorie für Incident Management	A	Die Standortkategorie, die standardmäßig zum Erstellen von Incident-Tickets verwendet wird.
Liste der Ziel-Aliasnamen	sc	Eine Liste mit durch Semikolon getrennten Ziel-Aliasnamen, die von der Webanwendung verwendet wird.

- 2 Ändern Sie die Parameter bei Bedarf.
- 3 Klicken Sie auf **Speichern**, um zur Seite **Systemsteuerung** zurückzukehren.
- 4 Klicken Sie auf **Server zurücksetzen**, um Ihre Änderungen zu speichern.

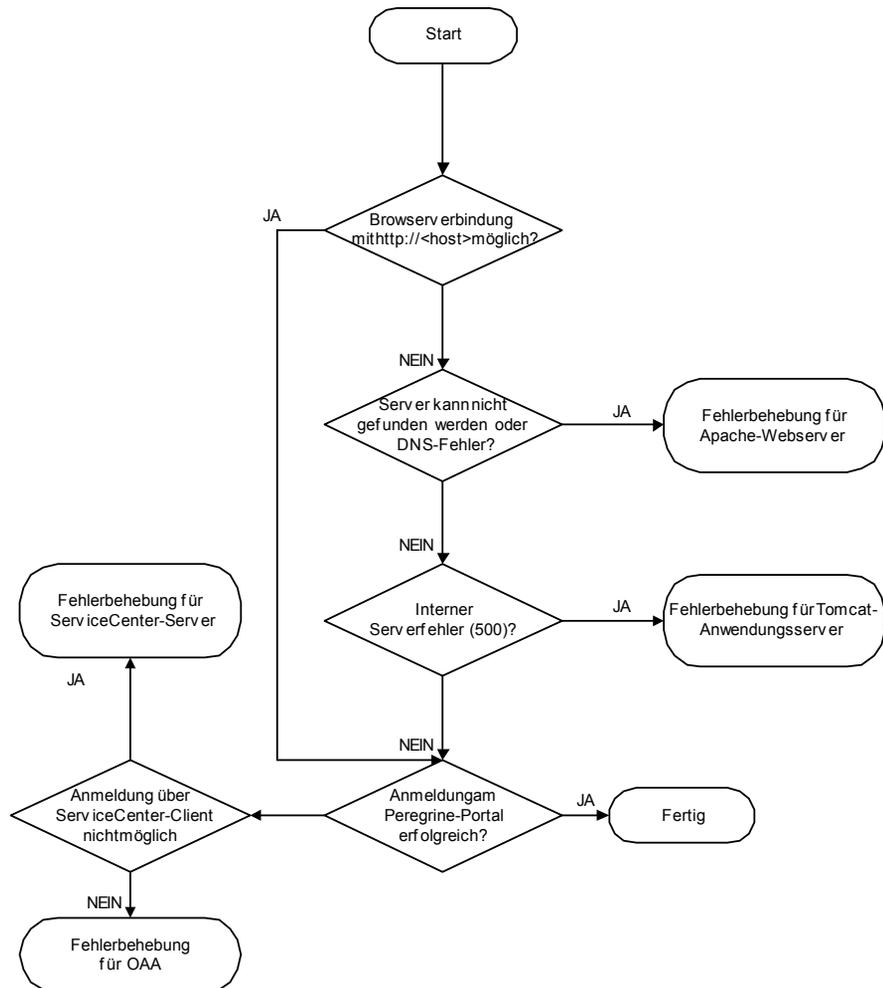
7 Fehlerbehebung

KAPITEL

Dieses Kapitel behandelt folgende Themen:

- *Beheben von Fehlern beim Apache-Webserver für Windows* auf Seite 203
- *Beheben von Fehlern beim Apache-Webserver für Unix* auf Seite 206
- *Beheben von Fehlern beim Tomcat-Dienst* auf Seite 207
- *Beheben von OAA-Fehlern* auf Seite 210

Beheben von Fehlern beim ServiceCenter-Server auf Seite 211 Im folgenden Fehlerbehebungsdiagramm werden die Bereiche aufgezeigt, die beim Lösen von Installationsproblemen zu berücksichtigen sind.



Beheben von Fehlern beim Apache-Webserver für Windows

Folgen Sie bei Problemen mit dem Apache-Webserver für Windows den folgenden Anweisungen.

Der Webserver antwortet nicht

Gehen Sie folgendermaßen vor, wenn der Webserver nicht antwortet:

- Schritt 1** Stellen Sie sicher, dass die Netzwerkverbindungen aktiviert sind.
- Schritt 2** Vergewissern Sie sich, dass das Programm `apache.exe` ausgeführt wird.
- Schritt 3** Starten Sie den Apache-Dienst neu.
- Schritt 4** Stellen Sie sicher, dass der von Apache verwendete Anschluss nicht von einem anderen Netzwerkdienst belegt ist (Apache verwendet standardmäßig Anschluss 80).

So stellen Sie sicher, dass die Netzwerkverbindungen aktiviert sind:

- 1 Klicken Sie auf **Start**.
- 2 Zeigen Sie auf **Einstellungen**.
- 3 Klicken Sie auf **Netzwerk- und DFÜ-Verbindungen**.
- 4 Klicken Sie auf **LAN-Verbindung**.
- 5 Überprüfen Sie im Dialogfeld, ob im Abschnitt **Verbindung** für den Status **Verbindung hergestellt** angezeigt wird.

So vergewissern Sie sich, dass das Programm „apache.exe“ ausgeführt wird:

- 1 Drücken Sie **Strg+Alt+Entf**.
- 2 Klicken Sie auf **Task-Manager**.
- 3 Überprüfen Sie, ob im Register **Prozesse** das Programm `Apache.exe` in der Spalte **Name** aufgeführt wird.

So starten Sie den Apache-Dienst neu:

- 1 Klicken Sie auf **Start**.
- 2 Klicken Sie auf **Programme**.
- 3 Klicken Sie auf **Verwaltung**.
- 4 Klicken Sie auf **Dienste**.

- 5 Markieren Sie den Apache-Dienst in der Liste und starten Sie ihn neu.

So stellen Sie sicher, dass der von Apache verwendete Anschluss nicht von einem anderen Netzwerkdienst belegt ist (Apache verwendet standardmäßig Anschluss 80):

- 1 Halten Sie den Apache-Dienst an.
 - a Klicken Sie auf **Start**.
 - b Klicken Sie auf **Programme**.
 - c Klicken Sie auf **Verwaltung**.
 - d Klicken Sie auf **Dienste**.
 - e Markieren Sie den Apache-Dienst in der Liste und halten Sie ihn an.
- 2 Klicken Sie auf **Start**.
- 3 Klicken Sie auf **Ausführen**.
- 4 Geben Sie `cmd` ein und klicken Sie auf **OK**.
- 5 Geben Sie im Befehlszeilenfenster `netstat -a` ein und drücken Sie die Eingabetaste.
- 6 Vergewissern Sie sich, dass der folgende Eintrag nicht vorhanden ist:
`Proto=TCP, Local Address=<Host>:http`.

Hinweis: Auf diese Weise ist sichergestellt, dass kein anderer Dienst den http-Anschluss 80 verwendet, wenn der Apache-Dienst nicht ausgeführt wird.
- 7 Beheben Sie etwaige Konflikte, indem Sie den Standardanschluss von Apache ändern (siehe Dokumentation zur Anpassung) bzw. den anderen Dienst deaktivieren oder ihm einen anderen Anschluss zuweisen.

Benutzer können nicht auf den Webserver zugreifen, obwohl er ausgeführt wird und die Netzwerk- und Internetverbindungen aktiviert sind

Gehen Sie folgendermaßen vor, wenn Benutzer nicht auf den Webserver zugreifen können:

- Schritt 1** Vergewissern Sie sich, dass der WINS-Server installiert ist.
- Schritt 2** Stellen Sie sicher, dass der DNS-Server installiert ist.
- Schritt 3** Überprüfen Sie die Apache-Protokolldateien auf weitere Fehler.

So vergewissern Sie sich, dass der WINS-Server installiert ist:

- 1 Klicken Sie auf **Start**.
- 2 Zeigen Sie auf **Einstellungen**.
- 3 Klicken Sie auf **Systemsteuerung**.
- 4 Klicken Sie auf **Software**.
- 5 Klicken Sie auf **Windows-Komponenten hinzufügen/entfernen**.
- 6 Klicken Sie auf **Netzwerkdienste**.
- 7 Klicken Sie auf **Details**.
- 8 Vergewissern Sie sich, dass das Kontrollkästchen für den WINS-Server aktiviert ist und dieser ordnungsgemäß im Netzwerk konfiguriert ist. Stellen Sie außerdem seine ordnungsgemäße Ausführung sicher.

So stellen Sie sicher, dass der DNS-Server installiert ist:

- 1 Klicken Sie auf **Start**.
- 2 Zeigen Sie auf **Einstellungen**.
- 3 Klicken Sie auf **Systemsteuerung**.
- 4 Klicken Sie auf **Software**.
- 5 Klicken Sie auf **Windows-Komponenten hinzufügen/entfernen**.
- 6 Klicken Sie auf **Netzwerkdienste**.
- 7 Klicken Sie auf **Details**.
- 8 Überprüfen Sie, ob DNS installiert ist und die DNS-Server mit dem Netzwerk verbunden sind und ordnungsgemäß funktionieren.

So überprüfen Sie die Apache-Protokolldateien auf weitere Fehler:

- ▶ Öffnen Sie die Apache-Protokolldateien in einem Texteditor.
Die Dateien befinden sich standardmäßig im Verzeichnis `c:\Program Files\Peregrine\Common\Apache2\logs`.

Beheben von Fehlern beim Apache-Webserver für Unix

Folgen Sie bei Problemen mit dem Apache-Webserver für Unix den folgenden Anweisungen.

Der Webserver antwortet nicht

Wenn der Apache-Webserver nicht antwortet, überprüfen Sie die Netzwerkinstallation.

So überprüfen Sie die Netzwerkinstallation:

- 1 Stellen Sie sicher, dass der von Apache verwendete Anschluss nicht von einem anderen Netzwerkdienst belegt ist.

Hinweis: Apache verwendet standardmäßig Anschluss 80. Sie können diese Einstellung in der Direktive **Port** in der Datei `httpd.conf` ändern. Listen Sie, nachdem Sie den Apache-Webserver heruntergefahren haben, mit dem Befehl `netstat` alle Anschlüsse auf, die verwendet werden.

```
$ /etc/init.d/oaactl stop
$ netstat -a | grep 80
```

- 2 Überprüfen Sie, ob die IP-Adresse und der Hostname des Servers ordnungsgemäß konfiguriert sind. Ist diese der Fall, werden folgende Aktionen durchgeführt:

- Der Ping-Befehl erhält eine Antwort vom Server.
- `nslookup hostname` zeigt die korrekte Zuordnung zwischen Hostname und IP-Adresse an.
- `telnet hostname 80` stellt eine Verbindung zum Server her.

```
$ /usr/sbin/ping hostname -n 5
```

```
$ telnet hostname 80
Trying...
Connected to hostname
Escape character is '^['.
```

Überprüfen der Apache-Protokolldateien auf weitere Fehler

Sehen Sie bei Problemen mit dem Apache-Webserver die Protokolldateien ein.

So überprüfen Sie die Apache-Protokolldateien auf weitere Fehler:

- ▶ Öffnen Sie die Apache-Protokolldateien in einem Texteditor.
Die Apache-Protokolldateien befinden sich standardmäßig unter:
<Basisinstallationsverzeichnis>/peregrine/common/apache2/logs.

Beheben von Fehlern beim Tomcat-Dienst

Bevor Sie Fehler beim Tomcat-Dienst beheben können, müssen Sie mit den Verfahren zum Starten und Anhalten von Tomcat unter Ihrem Betriebssystem vertraut sein. Außerdem muss Ihnen der Speicherort der Tomcat-Protokolldateien bekannt sein.

So starten/stoppen Sie Tomcat unter Windows:

- 1 Klicken Sie auf **Start**.
- 2 Klicken Sie auf **Programme**.
- 3 Klicken Sie auf **Verwaltung**.
- 4 Klicken Sie auf **Dienste**.
- 5 Markieren Sie den Dienst PeregrineTomcat in der Liste und starten Sie ihn (neu) bzw. halten Sie ihn an.

So starten/stoppen Sie Tomcat unter UNIX:

- ▶ `$ /etc/init.d/oaactl <start/stop/restart>`
In der folgenden Tabelle werden die Standard-Speicherorte der Tomcat-Protokolldateien aufgeführt.

Betriebssystem	Standard-Speicherort der Tomcat-Protokolldateien
Windows	C:\Program Files\Peregrine\Common\Tomcat4\logs
Unix	/<Installiertes Basisverzeichnis>/peregrine/common/tomcat4/logs

Überprüfen von Tomcat-Anschlusskonflikten

In der folgenden Tabelle werden die standardmäßig von Tomcat verwendeten Anschlüsse aufgeführt.

Anschlussnummer	Tomcat-Dienst
8005	Tomcat Administration
8009	Tomcat AJP13 Worker Port

In der Tomcat-Protokolldatei `stderr.log` wird in der folgenden Zeile darauf hingewiesen, dass der AJP13-Anschluss verwendet wird:

```
[INFO] ChannelSocket - -JK2: ajp13 listening on tcp port 8009
```

So überprüfen Sie Tomcat-Anschlusskonflikte:

- 1 Fahren Sie Tomcat herunter.
- 2 Listen Sie mit dem Befehl `netstat -a` die Anschlüsse auf, die verwendet werden, um Konflikte zu vermeiden.
- 3 Nehmen Sie die erforderlichen Änderungen an der Tomcat-Anschlusskonfiguration vor bzw. deaktivieren Sie den anderen Dienst oder weisen Sie ihm einen anderen Anschluss zu.

Ermitteln von Tomcat-Fehlern

Stellen Sie sicher, dass Sie mit Dateien arbeiten, die nur die erforderlichen Daten enthalten.

So stellen Sie eine ordnungsgemäße Umgebung für die Fehlerbehebung sicher:

- 1 Fahren Sie den Apache- und Tomcat-Dienst herunter.
- 2 Entfernen Sie alle Protokolldateien.
- 3 Starten Sie Apache und Tomcat neu.
- 4 Stellen Sie über einen Browser eine Verbindung zum Webserver her.
- 5 Halten Sie die Dienste an.

Datei „mod_jk.log“

In dieser Datei werden Informationen zu deaktivierten TCP-Verbindungen zwischen dem Apache-Webserver und Tomcat protokolliert.

Diese Datei enthält nur Einträge, wenn Fehler aufgetreten sind. Sie enthält Hinweise zu Verbindungsfehlern, wenn der AJP13-Anschluss mit einem anderen Dienst in Konflikt steht oder der Tomcat-Connector `mod_jk` nicht ordnungsgemäß konfiguriert ist.

Datei „stdout.log“

Die Standardausgabe dieser Protokolldatei lautet wie folgt:

```
Bootstrap: Create Catalina server
Bootstrap: Starting service
Starting service Tomcat-Standalone
Apache Tomcat/4.1.12
Instantiating Archway Servlet...
2002-12-10 12:22:13,079 INFO [main] - Using application preferences in
/C:/Program Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/local.xml
2002-12-10 12:22:13,119 INFO [main] - Using default preferences in /C:/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/archway.xml
2002-12-10 12:22:13,200 INFO [main] - Using default preferences in /C:/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/common.xml
2002-12-10 12:22:13,240 INFO [main] - Using default preferences in /C:/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/logging.xml
2002-12-10 12:22:13,270 INFO [main] - Using default preferences in /C:/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/themes.xml
2002-12-10 12:22:13,280 INFO [main] - Using default preferences in /C:/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/xsl.xml
Bootstrap: Service started
```

Überprüfen Sie im Falle eines Fehlers folgende Punkte:

- Wurde das Archway-Servlet instanziiert?
- Ist der Speicherort der Webanwendungen korrekt?
- Wurde der Bootstrap-Dienst erfolgreich gestartet?

Datei „stderr.log“

Die Standardausgabe dieser Protokolldatei lautet wie folgt:

```
Created catalinaLoader in: C:\Program Files\Peregrine\Common\Tomcat4\server\lib
[INFO] Registry - -Loading registry information
[INFO] Registry - -Creating new Registry instance
[INFO] Registry - -Creating MBeanServer
[INFO] ChannelSocket - -JK2: ajp13 listening on tcp port 8009
[INFO] JkMain - -Jk running ID=0 time=0/120 config=C:\Program
Files\Peregrine\Common\Tomcat4\conf\jk2.properties
```

Überprüfen Sie im Falle eines Fehlers folgende Punkte:

- Wurde **catalinaLoader** erstellt und verweist auf den korrekten Speicherort?
- ChannelSocket - JK2: Konnte **ajp13** eine Verbindung an der korrekten Anschlussnummer herstellen?
- Verwendet **JkMain** das Recht **jk2.properties**?

Datei „localhost_log.<Datum>.txt“

Diese Datei enthält keine Java-Fehler. In dieser Datei werden die Aktivitäten des Anwendungs-Managers beim Bereitstellen von Peregrine OAA-Webanwendungen protokolliert.

Beheben von OAA-Fehlern

Überprüfen Sie bei Problemen mit der Peregrine OAA-Webanwendung den Backend-Server der Anwendung und sehen Sie die OAA-Protokolldateien ein.

OAA-Backend-Konfiguration

Stellen Sie sicher, dass die Peregrine OAA-Anwendung mit dem richtigen Backend-Server verbunden ist und ordnungsgemäß funktioniert.

So überprüfen Sie die Backend-Konfiguration:

- 1 Geben Sie im Adressfeld Ihres Browsers <http://Hostname/oaadmin/admin.jsp> ein.
- 2 Melden Sie sich als **Admin** ohne Kennwort an (dies hat sich nach der Installation nicht geändert).
- 3 Überprüfen Sie im Verwaltungsmodul den Verbindungsstatus der aufgeführten Adapter.

- 4 Klicken Sie auf das Ziel des Backend-Servers, beispielsweise `sc`.
- 5 Vergewissern Sie sich, dass der Host und Anschluss für den Backend-Server korrekt sind.

OAA-Protokolldateien

In der nachfolgenden Tabelle werden die Standard-Speicherorte der Peregrine OAA-Protokolldateien aufgeführt.

Betriebssystem	Standard-Speicherort der Peregrine OAA-Protokolldateien
Windows	C:\Program Files\Peregrine\Common\Tomcat4\bin\archway.log
Unix	/ <code><Installiertes Basisverzeichnis></code> /peregrine/common/tomcat4/archway.log

Stellen Sie sicher, dass in den Protokolldateien folgende Angaben enthalten sind:

- Eine Liste der installierten OAA-Komponenten einschließlich Versionsnummer.
- Eine vollständige Liste der registrierten Pakete.
- Die Anweisung **Archway initialization complete**.

Enthält die Datei Java-Ausnahmen vom Typ `ClassNotFoundException` überprüfen Sie, ob alle erforderlichen jar-Dateien gefunden wurden.

Beheben von Fehlern beim ServiceCenter-Server

Gehen Sie im Falle von Problemen mit dem ServiceCenter-Server folgendermaßen vor:

- Schritt 1** Überprüfen Sie den Autorisierungscode und die Anschlusseinstellung von ServiceCenter.
- Schritt 2** Überprüfen Sie das ServiceCenter-Protokoll.

Bevor Sie mit der Fehlerbehebung beginnen, müssen Sie sich mit dem Starten und Anhalten des ServiceCenter-Servers unter Ihrem Betriebssystem vertraut machen. Außerdem muss Ihnen der Speicherort der ServiceCenter-Protokolldateien bekannt sein.

So starten/stoppen Sie ServiceCenter unter Windows:

- 1 Klicken Sie auf **Start**.
- 2 Klicken Sie auf **Programme**.
- 3 Klicken Sie auf **Verwaltung**.
- 4 Klicken Sie auf **Dienste**.
- 5 Markieren Sie den Dienst PeregrineServiceCenter in der Liste und starten Sie ihn (neu) bzw. halten Sie ihn an.

So starten/stoppen Sie ServiceCenter unter UNIX:

- `$ /etc/init.d/oaactl <start/stop/restart>`

In der folgenden Tabelle werden die Standard-Speicherorte der ServiceCenter-Protokolldateien aufgeführt.

Betriebssystem	Standard-Speicherort der ServiceCenter-Protokolldateien
Windows	C:\Program Files\Peregrine\ServiceCenter\sc.log
Unix	/<Installiertes Basisverzeichnis>/peregrine/servicecenter/sc.log

Überprüfen des Autorisierungscode und der Anschlusseinstellung von ServiceCenter

In der folgenden Tabelle werden die Speicherorte der ServiceCenter-Einstellungsdatei aufgeführt.

Betriebssystem	Speicherort der ServiceCenter-Einstellungsdatei
Windows	C:\Program Files\Peregrine\ServiceCenter\RUN\sc.ini
Unix	/<Installiertes Basisverzeichnis>/peregrine/servicecenter/RUN//sc.ini

So überprüfen Sie den Autorisierungscode und die Anschlusseinstellung von ServiceCenter:

- 1 Stellen Sie sicher, dass der über das Tag **auth:** eingestellte Autorisierungscode korrekt ist.
- 2 Vergewissern Sie sich, dass die Anschlusseinstellung für **system:** mit der Einstellung für den OAA-Backend-Server übereinstimmt.

Überprüfen des ServiceCenter-Protokolls

So überprüfen Sie das ServiceCenter-Protokoll:

- 1 Überprüfen Sie die Protokolldatei auf Fehler hinsichtlich des Ablaufdatums des Autorisierungscode.
- 2 Überprüfen Sie die Protokolldatei auf Fehler bei Ressourcenanhängen.
- 3 Weitere Informationen zur Fehlerbehebung finden Sie ggf. im *ServiceCenter-Verwaltungshandbuch*.

Index

A

- Aktivieren von Service Management 196
- Alternative Authentifizierung
 - UNIX 95
 - Windows 23
- Anwendungsserver
 - JRun unter Unix 121
 - JRun unter Windows 62
 - Tomcat unter Unix 97
 - Tomcat unter Windows 27
 - WebLogic unter Unix 119
 - WebLogic unter Windows 56
 - WebSphere unter Unix 99
 - WebSphere unter Windows 35

B

- Benutzerdefinierte Designs
 - UNIX 94
 - Windows 22
- Benutzerdefinierte Installation
 - Unix 140
 - Windows 78
- Benutzerrechte
 - getit.admin 186
- Buch
 - Aufbau 10

D

- Datei `local.xml` 186
- Dateien, Entladen von ServiceCenter 179
- Deinstallation 87

Deinstallieren

- Unix 151
- Dokumentation, zugehörig 8

E

- Entladefordern 179
- Entwicklungsumgebung
 - Unix, Übersicht über die Schritte 91
 - Windows, Übersicht über die Schritte 19

F

- Fehlerbehebung
 - Apache-Webserver für Unix 206
 - Apache-Webserver für Windows 203
 - OAA 210
 - ServiceCenter-Server 211
 - Tomcat 207
- Festlegen 130

G

- getit.admin-Benutzerrechte 186
- Get-Services
 - Übersicht 7

H

- Handbuch
 - Zielgruppe 7

I

- Incident Management-Parameter
 - ServiceCenter 198

J

JRun

- Konfiguration unter Unix 121
- Konfiguration unter Windows 62

JSP-Datei, Änderungen

- UNIX 94
- Windows 22

K

Konfigurieren von Backend-Datenbanken

- Migrationsprozess, Unix 96
- Migrationsprozess, Windows 24

Kundendienst 11

M

Mehrere Instanzen von Tomcat, installieren 163

Mehrere Tomcat-Instanzen, installieren 174

Migrationsprozess

- Alternative Authentifizierung, Unix 95
- Alternative Authentifizierung, Windows 23

Benutzerdefinierte Designs, Unix 94

Benutzerdefinierte Designs, Windows 22

JSP-Datei, Änderungen, Unix 94

JSP-Datei, Änderungen, Windows 22

Konfigurieren von

- Backend-Datenbanken, Unix 96

Konfigurieren von

- Backend-Datenbanken, Windows 24

Personalisierte Seiten, Unix 94

Personalisierte Seiten, Windows 22

Tailoring Kit-Änderungen, Unix 95

Tailoring Kit-Änderungen, Windows 23

UNIX 93

Windows 21

P

Parameter

- Incident Management-Einstellungen 198
- Service Management-Einstellungen 196, 198

Peregrine Systems-Kundendienst 11

Personalisierte Seiten

- UNIX 94
- Windows 22

Produktionsumgebung

- Unix, Übersicht über die Schritte 92
- Windows, Übersicht über die Schritte 20

R

Regeln, typografisch 9

S

SCAdapter

- Konfiguration 195–199

Server.xml, Dateien bearbeiten 161, 172

Server.xml-Dateien, bearbeiten 32

Service Desk

- Aktivieren von Service Management 196
- Einstellen von Parametern für
Get-Services 196, 198

Service Management

- Aktivieren für Get-Services 196
- aktiviert 197
- Parameter, ServiceCenter 196, 198

ServiceCenter

- Aktualisieren von Version 4.x 178

Systemsteuerung 188

T

Tailoring Kit-Änderungen

- UNIX 95
- Windows 23

Technischer Kundendienst 11

Terminologie 8

Tomcat, Installieren mehrerer Instanzen 163

Tomcat, mehrere Instanzen installieren 174

Tomcat-Anwendungsserver

- Unix 97
- Windows 27

Typische Installation

- Unix 131
- Windows 72

Typografische Regeln 9

V

Verfahren

- Anwenden von Entladefdateien
auf ServiceCenter 183

Verwaltungsmodul

- Seite **Admin-Einstellungen** 195
- Seite **Einstellungen** 189
- Systemsteuerung 188

W

WebLogic

- Konfiguration unter Unix 119
- Konfiguration unter Windows 56

WebSphere

- Konfiguration unter Unix 99
- Konfiguration unter Windows 35

Workers.properties, Datei bearbeiten 158

Z

Zugehörige Dokumentation 8

Zurücksetzen des Servers 188

