

IBM Cúram Social Program Management



Handbuch 'Cúram Deployment for WebLogic Server'

Version 6.0.5

IBM Cúram Social Program Management



Handbuch 'Cúram Deployment for WebLogic Server'

Version 6.0.5

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen in „Bemerkungen“ auf Seite 27 gelesen werden.

Überarbeitung: Mai 2013

Diese Ausgabe bezieht sich auf IBM Cúram Social Program Management v6.0.5 und alle nachfolgenden Releases, sofern nicht anderweitig in neuen Ausgaben angegeben.

Licensed Materials - Property of IBM.

© Copyright IBM Corporation 2012, 2013.

© Cúram Software Limited. 2011. Alle Rechte vorbehalten.

Inhaltsverzeichnis

Abbildungsverzeichnis v

Tabellen vii

Kapitel 1. Einführung 1

1.1 Handbuch 'Curam Deployment'. 1

Kapitel 2. Erstellung von EAR-Dateien. . . 3

2.1 Einführung. 3

2.2 Enterprise ARhive-Datei 3

2.2.1 .ear-Datei für die Anwendung erstellen. . . 3

2.2.2 Technische Details. 3

2.2.3 Inhalt der .ear-Datei für die Anwendung . . 4

2.3 Onlinehilfeanwendung. 5

2.3.1 Datei CuramHelp.ear erstellen. 5

2.3.2 Inhalt der Datei CuramHelp.ear 5

2.4 Web-Service-Anwendung 6

2.4.1 EAR-Datei für Web-Services erstellen . . . 6

2.4.2 Technische Details. 6

2.4.3 Inhalt der EAR-Datei für Web-Services . . . 6

2.4.4 Web-Service-WSDL 7

2.5 Mehrere EAR-Dateien 7

Kapitel 3. Anwendungsserverkonfiguration 9

3.1 Einführung. 9

3.2 WebLogic Server-Konfiguration 9

3.3 Sicherheitskonfiguration 10

3.4 Zeitonenkonfiguration 11

3.5 WebLogic Server starten und stoppen 11

3.5.1 WebLogic Server starten 11

3.5.2 WebLogic Server stoppen 11

3.5.3 WebLogic Server erneut starten 12

3.5.4 Serverprotokollierung 12

Kapitel 4. Bereitstellung 13

4.1 Einführung 13

4.2 Bereitstellung 13

4.2.1 Anwendung bereitstellen 13

4.2.2 Benutzernamen SYSTEM ändern 13

4.2.3 Bereitstellung einer Anwendung zurücknehmen 14

4.3 JSPs vorkompilieren 14

4.4 Bereitgestellte Anwendung testen 15

Anhang. Manuelle WebLogic Server-Konfiguration 17

A.1 Einführung 17

A.2 WebLogic Server-Domäne erstellen 17

A.2.1 Domänenkonfiguration 17

A.3 Java Virtual Machine konfigurieren 18

A.4 Datenbanktreiber einrichten 18

A.5 WebLogic Administration Service starten . . . 19

A.6 Datenbank konfigurieren 19

A.7 Sicherheit einrichten 21

A.8 JMS einrichten 22

A.9 Warteschlange einrichten 23

A.10 Warteschlangen-Verbindungsfactory einrichten 24

A.11 Topic einrichten 25

A.12 Topic-Verbindungsfactory einrichten 26

A.13 Manuelle Bereitstellung ausführen 26

Bemerkungen. 27

Marken. 29

Abbildungsverzeichnis

1.	Beispiel für Datei deployment_packaging.xml	8	5.	Anwendungsbeispiel	12
2.	Beispiel für AppServer-Eigenschaften 10	6.	Anwendungsbeispiel	13
3.	Anwendungsbeispiel	7.	Anwendungsbeispiel	14
4.	Anwendungsbeispiel	8.	Anwendungsbeispiel	14

Tabellen

1. Speicherargumente	18	2. Providerspezifische Optionen	21
--------------------------------	----	---	----

Kapitel 1. Einführung

1.1 Handbuch 'Cúram Deployment'

Dieses Handbuch beschreibt die Schritte, die zur Erstellung einer IBM® Cúram Social Program Management-Anwendung für die Bereitstellung auf Oracle WebLogic Server erforderlich sind. Das Handbuch beschreibt außerdem die zur Verfügung gestellte Unterstützung für Konfiguration und Bereitstellung auf WebLogic Server und (wo erforderlich) die dafür nötigen manuellen Schritte.

Es wird vorausgesetzt, dass der Leser die IBM Cúram Social Program Management-Anwendungsentwicklungsumgebung kennt und weiß, wie eine Serveranwendung und ein Web-Client entwickelt und erstellt werden. Das Handbuch setzt außerdem voraus, dass WebLogic Server bereits installiert ist. Weitere Informationen zur Installation von WebLogic Server finden Sie im *Handbuch 'Cúram Third Party Tools Installation'*¹.

1. Weitere Informationen finden Sie in dem Installationshandbuch für die betreffende Plattform, z. B. Microsoft Windows oder UNIX.

Kapitel 2. Erstellung von EAR-Dateien

2.1 Einführung

Der wichtigste Schritt vor der Bereitstellung einer IBM Cúram Social Program Management-Anwendung ist das Packen der Anwendung in .ear-Dateien (Enterprise ARchive). Die Anwendung (Web-Client, Server) und die Web-Service-Anwendung werden in separate .ear-Dateien gepackt und Server Development Environment for Java (SDEJ) stellt Apache Ant-Ziele zur Verfügung, die diese Task ausführen.

Bevor die Ziele im folgenden Abschnitt ausgeführt werden, muss die Umgebungsvariable WLS_HOME zusammen mit den zuvor eingerichteten Umgebungsvariablen aus dem *Handbuch 'Cúram Third Party Tools Installation'* eingestellt werden.

Die Variable WLS_HOME muss auf das Verzeichnis server der WebLogic Server-Installation verweisen. Beispiel: d:\weblogic\server oder /opt/weblogic/server.

2.2 Enterprise ARchive-Datei

Die folgenden Abschnitte beschreiben den Prozess der Erstellung der .ear-Datei (Enterprise ARchive) für die Anwendung. Dies umfasst auch Informationen über das, was "in technischer Hinsicht" und mit dem Inhalt der EAR-Datei geschieht.

2.2.1 .ear-Datei für die Anwendung erstellen

Das folgende Ziel muss vom Stammverzeichnis des Projekts aus ausgeführt werden, um die .ear-Datei für WebLogic Server zu erstellen:

build weblogicEAR

Dieses Ziel erstellt eine installationsbereite .ear-Datei <SERVERMODELLNAME>.ear im Verzeichnis <SERVERVERZEICHNIS>/build/ear/WLS².

Dieses Ziel erstellt eine installationsbereite .ear-Datei CuramHelp.ear im Verzeichnis <SERVERVERZEICHNIS>/build/ear/WLS - vorausgesetzt, die Onlinehilfeanwendung wurde erstellt und es wurde ein Build dieser Anwendung ausgeführt. Weitere Informationen zum Build der Onlinehilfeanwendung finden Sie im *Handbuch 'Cúram Online Help System Development and Deployment'*. Informationen zum Inhalt der Datei CuramHelp.ear finden Sie in 2.3.2, „Inhalt der Datei CuramHelp.ear“, auf Seite 5.

Bevor Sie dieses Ziel ausführen, muss eine vollständig erstellte IBM Cúram Social Program Management-Anwendung zur Verfügung stehen. Weitere Informationen zum Erstellen einer IBM Cúram Social Program Management-Anwendung finden Sie im *Handbuch 'Cúram Server Developer'*.

Anmerkung: Die EAR-Datei kann nicht für die H2-Datenbank erstellt werden.³

2.2.2 Technische Details

Das Ziel **weblogicEAR** packt verschiedene zuvor generierte Java™-Dateien und Deployment-Deskriptoren und fasst sie zu einer EAR-Datei zusammen.

2. SERVERMODELLNAME und SERVERVERZEICHNIS sind Umgebungsvariablen, die den Namen des Modells im Projekt bzw. das Stammverzeichnis des Projekts angeben.

3. Weitere Informationen zur H2-Datenbank finden Sie im *Handbuch 'Cúram Third-Party Tools Installation for Windows'*.

Die Java-Dateien und Deployment-Deskriptoren werden während des Buildprozesses abhängig vom Vorhandensein von Business Process Object-Klassen - z. B. Methoden von *Fassaden*- oder Web-Service-Klassen - generiert und können von fernen Clients aufgerufen werden.

Standardmäßig werden alle Fernaufrufe an den Server von der Session-Bean `curam.util.invoke.EJBMethod` verarbeitet - und nicht über eine Session-Bean per öffentlich verfügbarer Schnittstelle. Diese Bean unterstützt Anwendungsfeatures, wie z. B. Autorisierung, Auditfunktionalität und Tracereinstellung. Bei Bedarf ist es auch möglich, eine Fassadenschnittstelle zu generieren⁴.

2.2.3 Inhalt der .ear-Datei für die Anwendung

Die .ear-Datei, die erstellt wird, hat folgende Struktur und folgenden Inhalt:

- **META-INF-Verzeichnis**

Das META-INF-Verzeichnis enthält Folgendes:

- `application.xml`

Diese Datei wird automatisch generiert und zeigt die Zuordnung zwischen EJB-Modulen und JAR-Dateien, die in der Anwendung enthalten sind.

- `MANIFEST.MF`

Diese Datei beschreibt den Inhalt der .ear-Datei.

- **Zentrale .jar-Dateien**

Zu den zentralen .jar-Dateien gehören⁵:

- `appinf.jar`

- `appinf_internal.jar`

- `coreinf.jar`

- `rules.jar`

- `jde_commons.jar`

- `log4j.jar`

- `commons-pool.jar`

- `commons-codec.jar`

- `commons-discovery.jar`

- `jdom.jar`

- `axis.jar`

- `castor.jar`

- `jaxrpc.jar`

- `saaj.jar`

- `java_cup.zip`

- `InfrastructureModule.jar`

- `InvalidationModule.jar`

- `DBtoJMS.war`

- `ClientModule.war`

- **.jar-Fassadendateien**

Diese Dateien sind nur vorhanden, wenn die Fassadengenerierung aktiviert wurde. Alle in der Anwendung definierten Fassaden werden zusammen in die .jar-Datei `FacadeModule.jar` gepackt. Die .jar-Datei enthält die Bean-Implementierungsklassen für die EJB-Module, die die Fassaden darstellen. Die .jar-Datei enthält die folgenden Dateien im META-INF-Verzeichnis:

4. Der optionale Buildparameter `-Denablefacade=true` schaltet die Generierung von Fassadencode ein.

5. Die Versionsnummern sind für die angegebenen .jar-Dateien nicht aufgeführt.

- `ejb-jar.xml`
Diese Datei wird automatisch generiert und enthält die Definition jedes EJB-Moduls, das in der `.jar`-Datei enthalten ist. Alle öffentlich verfügbaren Methoden und Details zu den für die EJB-Module verfügbaren Ressourcen sind aufgeführt.
- `weblogic-ejb-jar.xml`
Generierter WebLogic Server-spezifischer Deployment-Deskriptor
- `Manifest.mf`
Manifestdatei, die den Klassenpfad für die EJB angibt
- **Andere `.jar`-Dateien**
Die anderen `.jar`-Dateien enthalten den generierten und manuell erstellten Code aus der Anwendung. Dazu gehören `struct.jar`, `application.jar`, `workflow.jar`, `implementation.jar` und `properties.jar`. Die Datei `properties.jar` enthält die Datei `Bootstrap.properties`. Dies ist die Datei mit den maschinenspezifischen Konfigurationseigenschaften, mit denen eine erste Verbindung zur Datenbank angefordert wird.

2.3 Onlinehilfeanwendung

Der Build der Onlinehilfeanwendung von IBM Cúram Social Program Management erfolgt in einer separaten Datei `CuramHelp.ear`. Die Datei `CuramHelp.ear` wird automatisch generiert, wenn die EAR-Datei der Anwendung erstellt wird. Weitere Informationen zum Build der EAR-Datei der Anwendung finden Sie in 2.2.1, „`.ear`-Datei für die Anwendung erstellen“, auf Seite 3. Es besteht auch die Möglichkeit, die EAR-Datei der Onlinehilfeanwendung separat zu erstellen.

2.3.1 Datei `CuramHelp.ear` erstellen

Das folgende Ziel muss aus dem Stammverzeichnis des Projekts ausgeführt werden, um die Datei `CuramHelp.ear` für WebLogic Server zu erstellen:

build weblogicHelpEAR

Dieses Ziel erstellt eine installationsbereite Datei `CuramHelp.ear` im Verzeichnis `<SERVERVERZEICHNIS>/build/ear/WLS` - vorausgesetzt, die Onlinehilfeanwendung wurde erstellt und es wurde ein Build dieser Anwendung ausgeführt. Weitere Informationen zum Build der Onlinehilfeanwendung finden Sie in *Handbuch 'Cúram Online Help System Development and Deployment'*.

2.3.2 Inhalt der Datei `CuramHelp.ear`

Die Datei `CuramHelp.ear`, die erstellt wird, hat folgende Struktur und folgenden Inhalt:

- **META-INF-Verzeichnis**

Das META-INF-Verzeichnis enthält Folgendes:

- `application.xml`

Diese Datei wird automatisch generiert und zeigt die Zuordnung zwischen EJB-Modulen und JAR-Dateien, die in der Anwendung enthalten sind.

- `MANIFEST.MF`

Diese Datei beschreibt den Inhalt der `.ear`-Datei.

- **help.war-Dateien**

Eine Datei `help.war` ist eine Webanwendung, die Onlinehilfeanzeigen für die Anwendung enthält. Eine Datei `help.war` wird für jede unterstützte Ländereinstellung erstellt. Weitere Informationen zur Onlinehilfe finden Sie im *Handbuch 'Cúram Online Help System Development and Deployment'*.

2.4 Web-Service-Anwendung

Die automatische Generierung von mit WSDL⁶definierten Web-Services wird unterstützt. Anwendungsentwickler können so das Potenzial des IBM Cúram Social Program Management-Modells mit dem komfortablen Zugang über Web-Services verbinden, um flexibel wiederverwendbare Softwarekomponenten zu erstellen.

2.4.1 EAR-Datei für Web-Services erstellen

Das folgende Ziel muss aus dem Stammverzeichnis des Projekts ausgeführt werden, um die .ear-Datei für Web-Services zu erstellen:

```
build weblogicWebservices -Dprp.webipaddress="adresse" -Dprp.contextproviderurl="url" -Dprp.contextfactoryname="factory"
```

Dabei gilt Folgendes:

- 'adresse' ist die IP-Adresse, an der der Server, der die Web-Services hostet, empfangsbereit ist. Die Standardeinstellung ist `http://localhost:7001`.
- 'url' ist die URL des JNDI-Kontextproviders. Dies ist die Adresse des Servers, der die IBM Cúram Social Program Management-Komponenten hostet, die über die Web-Services zugänglich gemacht werden. Die Standardeinstellung⁷ ist `t3://localhost:7001`.
- 'factory' ist der Name der JNDI-Kontextfactory. Der Standardwert hierfür ist `weblogic.jndi.WLInitialContextFactory` und sollte nur selten geändert werden müssen.

Dieses Ziel erstellt eine installationsbereite .ear-Datei `<SERVERMODELLNAME>WebServices.ear` im Verzeichnis `<SERVERVERZEICHNIS>/build/ear/WLS`.

Anmerkung: Bevor Sie dieses Ziel ausführen, muss eine vollständig erstellte IBM Cúram Social Program Management-Anwendung zur Verfügung stehen, die bereitgestellt werden kann.

2.4.2 Technische Details

Das Ziel `weblogicWebServices` packt verschiedene zuvor generierte Java-Dateien und Deployment-Deskriptoren und fasst sie zu einer .ear-Datei zusammen.

Die Java-Dateien und Deployment-Deskriptoren werden während des Buildprozesses (vgl. *Handbuch 'Cúram Server Developer'*) auf der Basis der Web-Service-Stereotypen generiert, die im Modell definiert wurden. BPO-Klassen sollten Serverkomponenten mit dem Stereotyp 'Webservice' zugeordnet werden, damit diese Generierung stattfindet⁸. Jede Serverkomponente mit dem Stereotyp 'Webservice' wird so behandelt, als hätte sie auch den Stereotyp 'ejb'. Dies liegt daran, dass Web-Service-Schnittstellen als Wrapper für öffentlich zugängliche BPOs fungieren.

2.4.3 Inhalt der EAR-Datei für Web-Services

Die .ear-Datei für Web-Services, die erstellt wird, hat folgende Struktur und folgenden Inhalt:

- META-INF-Verzeichnis
 - application.xml
Die Datei beschreibt das Kernmodul für die Web-Service-Anwendung, nämlich die Datei `webservices.war`.
 - MANIFEST.MF

6. WSDL ist die Abkürzung von Web Service Definition Language.

7. Der Standardwert für den JNDI-Kontextprovider verwendet ein WebLogic Server-spezifisches Protokoll (t3) für den Zugriff auf die JNDI-Baumstruktur.

8. Weitere Informationen zum Zuordnen von BPOs zu Serverkomponenten finden Sie im *Handbuch 'Cúram Server Modelling'*

Die Manifestdatei, die den Inhalt der .ear-Datei beschreibt.

- **.war-Datei für Web-Services**

Diese Datei enthält unterstützende .jar-Dateien im Verzeichnis WEB-INF/lib, wie z. B.:

- coreinf.jar

Diese .jar-Datei enthält die Konvertierungsmethoden, die zur Unterstützung der Serialisierung der in der Schnittstelle verwendeten komplexen Typen genutzt werden.

- axis.jar

Diese .jar-Datei enthält die Apache Axis-Web-Service-Engine.

- appwebservices.jar

Diese .jar-Datei enthält die Wrapperklassen, die es den Axis-Web-Services ermöglichen, eine Verbindung zu der/den IBM Cúram Social Program Management-Session-Bean(s) herzustellen, und die Klassen für komplexe Typen, die in der Schnittstelle zu den Web-Services verwendet werden.

- server-config.wsdd

Die .wsdd-Datei befindet sich im Verzeichnis WEB-INF und enthält eine Web-Service-Engine-Konfiguration, die IBM Cúram Social Program Management-BPOs einzelnen Web-Services zuordnet.

2.4.4 Web-Service-WSDL

Ein IBM Cúram Social Program Management Axis-Web-Service macht die eigene WSDL bei seiner Bereitstellung zugänglich.

Beispiel: Angenommen, ein Service steht an folgender URL zur Verfügung:

<http://localhost:7001/CuramWS/services/MyTestService>

Dann finden Sie die WSDL-Beschreibung unter folgender URL:

<http://localhost:7001/CuramWS/services/MyTestService?wsdl>

Die URL

<http://localhost:7001/CuramWS/services>

gibt eine Webseite zurück, auf der alle bereitgestellten Axis-Web-Services mit einem Link zu ihren WSDL-Dateien aufgeführt sind.

Das allgemeine URL-Format der Positionen lautet:

<http://<Web-Server>:<Portnummer>/<Servermodellname>WS/services/<BPO-Name>>.

2.5 Mehrere EAR-Dateien

Beim Erstellen einer Anwendungs-EAR steht auch eine optionale Datei zur Verfügung, mit der die Clientkomponenten in verschiedene WAR- und EAR-Dateien aufgeteilt werden können und mit der die EAR-Konfiguration und die beinhalteten Module besser gesteuert werden kann. Dabei handelt es sich um die Datei `deployment_packaging.xml`, die sich in Ihrem Verzeichnis `SERVERVERZEICHNIS/project/config` befinden sollte.

Die Datei `deployment_packaging.xml` hat folgendes Format:

```

<deployment-config>
  <ear name="Curam"
    requireServer="true">
    <components>custom,sample,SamplePublicAccess,core</components>
    <context-root>/Curam</context-root>
  </ear>
  <ear name="CuramExternal">
    <components>SamplePublicAccessExternal</components>
    <context-root>/CuramExternal</context-root>
    <custom-web-xml>${client.dir}/custom_web_xml</custom-web-xml>
  </ear>
</deployment-config>

```

Abbildung 1. Beispiel für Datei *deployment_packaging.xml*

Jede Datei kann mehrere <ear>-Elemente haben und führt zur Erstellung einer EAR-Datei im Verzeichnis SERVERVERZEICHNIS/build/ear/WLS. Für jedes Element gibt es folgende Optionen:

- name
Diese Option steuert den Namen der EAR, die in dem Prozess erstellt wird.
- requireServer
Dieses optionale Attribut steuert, ob das Servermodul in die EAR-Datei aufgenommen wird. Gültige Einträge sind true oder false. Der Standardwert ist false. Beim Bereitstellen mehrerer EAR-Dateien für einen Anwendungsserver darf dieses Attribut für nur eine EAR-Datei auf true eingestellt werden, da nur ein IBM Cúram Social Program Management-Servermodul pro Cluster bereitgestellt werden sollte. Wenn requireServer für mehrere EAR-Dateien auf true eingestellt wird, müssen die anderen EAR-Dateien in einem anderen Cluster bereitgestellt werden, um Konflikte zu vermeiden.
- components
Diese Option steuert, welche Clientkomponenten in die EAR-Datei aufgenommen werden. Sie steuert auch die Reihenfolge für den erneuten Build des Clients, der erforderlich ist. Normalerweise ist das zentrale Verzeichnis nicht Teil der Reihenfolge, aber in diesem Fall ist es wichtig, dass es hinzugefügt wird, um anzugeben, ob es in eine bestimmte WAR-Datei aufgenommen werden soll. Die Einträge müssen hier der üblichen Reihenfolge entsprechen, die im *Handbuch 'Cúram Server Developer'* beschrieben ist, und sie sind durch Kommas zu trennen.
- context-root
Diese Option definiert das Kontextstammverzeichnis des WAR-Moduls im Deployment-Deskriptor application.xml. Die Einträge müssen mit einem Schrägstrich beginnen.
- custom-web-xml
Dieses optionale Element steuert, ob eine angepasste Datei web.xml die Standardversion in der WAR-Datei überschreiben soll. Die Einträge müssen jeweils einen Apache-Ant-Pfad zu dem Verzeichnis angeben, in dem die Datei web.xml enthalten ist.
Es ist möglich, in diesem Pfad Verweise auf Umgebungsvariablen zu verwenden. Beispiel:
{client.dir} kann verwendet werden, um auf das Web-Client-Verzeichnis zu verweisen, und
{SERVER_DIR} kann verwendet werden, um auf das Serververzeichnis zu verweisen.

Für jeden Web-Client (z. B. die WAR-Datei) ist eine separate Web-Client-Komponente erforderlich, die die Anpassungen enthält. Im Falle mehrerer Web-Clients enthält Ihre Umgebungsvariable CLIENT_COMPONENT_ORDER alle Ihre angepassten Komponenten. Aber es sind separate <ear>-Elemente erforderlich - eines für jede angepasste Webkomponente (und andere Komponenten, falls erforderlich).

Wie beim Standardziel muss eine vollständig erstellte Anwendung zur Verfügung stehen. Weitere Informationen zum Erstellen einer Anwendung finden Sie im *Handbuch 'Cúram Server Developer'*.

Kapitel 3. Anwendungsserverkonfiguration

3.1 Einführung

Dieses Kapitel setzt voraus, dass WebLogic Server bereits installiert ist. Im *Handbuch 'Cúram Third Party Tools Installation'*⁹ finden Sie ausführliche Informationen zur Installation.

Die Konfiguration von WebLogic Server ist auf allen Plattformen gleich und Cúram Server Development Environment for Java (SDEJ) stellt eine Reihe von Ant-Zielen zur Verfügung, die Sie bei der Konfiguration und dem Management der Installation unterstützen. „Manuelle WebLogic Server-Konfiguration“, auf Seite 17 beschreibt die manuellen Aktivitäten, die von den Konfigurationsscripts ausgeführt werden.

Das von SDEJ zur Verfügung gestellte Konfigurationsziel ist eine einfache Standardkonfiguration, die möglicherweise nicht für eine Produktionsumgebung geeignet ist.

Anmerkung: Das Ziel **configure** überschreibt jede vorhandene WebLogic Server-Domäne, die sich auf 'configure' bezieht.

3.2 WebLogic Server-Konfiguration

Die Konfiguration von WebLogic Server umfasst das Einrichten einer Datenquelle, einer Domäne und eines Servers und das Konfigurieren der JMS- und Sicherheitseinstellungen. Alle diese Tasks können mithilfe des Ziels **configure** ausgeführt werden, das durch Cúram SDEJ zur Verfügung gestellt wird.

Der Befehl **build configure** muss vom Verzeichnis <SERVERVERZEICHNIS> ausgeführt werden, um die automatische Konfiguration aufzurufen. Dieses Ziel setzt voraus, dass die Dateien `AppServer.properties` und `Bootstrap.properties` im Verzeichnis <SERVERVERZEICHNIS>/project/properties¹⁰ enthalten sind. Weitere Informationen zum Einrichten einer Datei `Bootstrap.properties` finden Sie im *Handbuch 'Cúram Server Developer'*. 3.2, „WebLogic Server-Konfiguration“ zeigt den erforderlichen Inhalt der Datei `AppServer.properties`.

9. Weitere Informationen finden Sie im Installationshandbuch für die betreffende Installation (z. B. Windows oder UNIX).

10. Es ist möglich, diese Standardposition für die Eigenschaftendatei bei Ausführung des Ziels **configure** durch Angabe von `-Dprop.file.location` zu überschreiben.

```

## APPLICATION SERVER PROPERTIES

## IT IS VERY IMPORTANT TO USE '/' FOR DIRECTORY PATHS. ##

# Property to indicate WebLogic Server is installed.
as.vendor=BEA

# The username and password for admin server.
# The password must be encrypted.
security.username=<z. B. weblogic>
security.password=<z. B. verschlüsseltes Kennwort>

# The name of the WebLogic Server Domain Name.
node.name=MYNODE

# The name of the server on which the application will be hosted.
curam.server.name=CuramServer
curam.server.port=7001

#####
## THE FOLLOWING PROPERTIES ARE FOR WEBLOGIC ONLY ##
#####

# Property to set JVM initial and maximum heap size
# when starting and stopping WebLogic Server.
curam.server.jvm.heap.size=1024

```

Abbildung 2. Beispiel für AppServer-Eigenschaften

Anmerkung:

1. Die Angabe 'security.username' in Datei AppServer.properties darf mit keinem Benutzer identisch sein, der als Anwendungsbenutzer besteht.
Durch die Implementierung von Mechanismen zur Gewährleistung sicherer Kennwörter in WebLogic Server 11g Release 1 wird sichergestellt, dass das Kennwort für den Weblogic-Anwendungsserver mindestens acht alphanumerische Zeichen lang ist und mindestens eine Ziffer oder ein Sonderzeichen enthält.
2. Das Ziel **configure** kann nicht ausgeführt werden, solange die H2-Datenbank verwendet wird.¹¹

3.3 Sicherheitskonfiguration

Authentifizierungsprovider sind die Standardauthentifizierungsmechanismen in WebLogic Server und die Anwendungssicherheit wird mit einem benutzerdefinierten Authentifizierungsprovider implementiert. Weitere Informationen hierzu finden Sie im Abschnitt *Default Configuration for Oracle WebLogic Server* im Handbuch 'Cúram Security'.

Die Anwendung und WebLogic Server unterstützen die Verwendung alternativer Authentifizierungsmechanismen (wie z. B. LDAP-Verzeichnisserver oder Single-Sign-on-Lösung). WebLogic Server stellt Authentifizierungsprovider zur Verfügung, die für die Zusammenarbeit mit LDAP-Verzeichnissen konfiguriert werden können. Für Single-Sign-on-Lösungen stellt der Fremdanbieter häufig einen benutzerdefinierten Authentifizierungsprovider her, der mit WebLogic Server zusammenarbeitet. Wenn ein alternativer Provider für die Authentifizierung verwendet werden muss, sollte der Cúram-Authentifizierungsprovider nur sicherstellen, dass Benutzer für Berechtigungszwecke gültig sind.

Um den Cúram-Provider nur für die Authentifizierung der Identität zu konfigurieren, muss die Eigenschaft `curam.security.check.identity.only` in der Datei AppServer.properties auf `true` gesetzt werden, bevor das Ziel **configure** ausgeführt wird. Die zusätzlichen Authentifizierungsprovider müssen nach Ausführung des Ziels `configure` manuell konfiguriert werden.

11. Weitere Informationen zur H2-Datenbank finden Sie im Handbuch 'Cúram Third-Party Tools Installation for Windows'.

Weitere Informationen hierzu finden Sie im Abschnitt *Identity Only Authentication* im Handbuch 'Cúram Security'.

Ferner steht eine optionale Eigenschaft zur Verfügung, die eine Protokollierung für den Cúram-Authentifizierungsprovider ermöglicht. Wenn die Eigenschaft `curam.security.login.trace` auf `true` eingestellt wird, bewirkt dies, dass Traceinformationen während des Authentifizierungsprozesses zur WebLogic Server-Protokolldatei hinzugefügt werden. Diese Eigenschaft muss in der Datei `AppServer.properties` eingestellt werden, bevor das Ziel `configure` ausgeführt wird.

3.4 Zeitzonekonfiguration

Wenn mehrere Servermaschinen verwendet werden, müssen sie alle zeitlich synchron getaktet sein und sich in derselben Zeitzone befinden, damit die "natürliche" Reihenfolge von Datum/Uhrzeiten in der Datenbank genau der tatsächlichen Reihenfolge der Ereignisse entspricht. Beispiel: Wenn Datum und Uhrzeit der Erstellung von Datenbanksatz *A* vor dem entsprechenden Datum/der entsprechenden Zeit von Datenbanksatz *B* liegen, ist sicher davon auszugehen, dass *A* vor *B* erstellt wurde, unabhängig davon, von welchem Server die einzelnen Datensätze erstellt wurden.

Die Zeitzone des/der Server(s) darf sich während der Lebenszeit der Anwendung nicht ändern. Grund hierfür ist, dass beim Speichern von Daten in der Datenbank immer von der Zeitzone des aktuellen Servers ausgegangen wird. Wenn sich die Zeitzone des Servers ändert, weichen alle vor der Änderung eingegebenen Datumsangaben um die Differenz zwischen alter und neuer Zeitzone ab.

3.5 WebLogic Server starten und stoppen

Bei Windows-Plattformen stellt Cúram SDEJ Ant-Ziele zur Verfügung, um das Starten und Stoppen von WebLogic Server zu unterstützen. Diese Ziele müssen vom Verzeichnis `<SERVERVERZEICHNIS>` ausgeführt werden und sie setzen für das Ziel `configure` voraus, dass die Datei `AppServer.properties` richtig konfiguriert wurde (siehe 3.2, „WebLogic Server-Konfiguration“, auf Seite 9). Außerdem müssen verschiedene zusätzliche Parameter angegeben werden. Diese werden im Folgenden beschrieben.

3.5.1 WebLogic Server starten

Das Ant-Ziel für das Starten eines WebLogic Server lautet:

```
build startserver
```

Dies erfordert die folgende Option:

- `-Dserver.name`

Der Name des zu startenden Servers.

Wichtig: Bevor Sie den Anwendungsserver zum ersten Mal starten, müssen Sie das Ziel `database` gefolgt vom Ziel `prepare.application.data` ausgeführt haben. Wenn Sie diese Reihenfolge nicht einhalten, werden beim ersten Anmelden wahrscheinlich Transaktionszeitlimits überschritten. D. h. die Anwendung kann nicht initialisiert werden und der Zugriff ist nicht möglich. Bei jeder neuen Ausführung des Ziels `database` (z. B. in einer Entwicklungsumgebung) muss auch das Ziel `prepare.application.data` erneut ausgeführt werden.

```
build startserver -Dserver.name=CuramServer
```

Abbildung 3. Anwendungsbeispiel

3.5.2 WebLogic Server stoppen

Das Ant-Ziel für das Stoppen eines WebLogic Server lautet:

```
build stopserver
```

Dies erfordert die folgende Option:

- `-Dserver.name`

Der Name des zu stoppenden Servers.

```
build stopserver -Dserver.name=CuramServer
```

Abbildung 4. Anwendungsbeispiel

3.5.3 WebLogic Server erneut starten

Das Ant-Ziel für das erneute Starten eines WebLogic Server lautet:

build restartserver

Dies erfordert die folgende Option:

- `-Dserver.name`

Der Name des erneut zu startenden Servers.

```
build restartserver -Dserver.name=CuramServer
```

Abbildung 5. Anwendungsbeispiel

Anmerkung: Wenn der Server beim Neustartversuch noch nicht gestartet ist, bewirkt der Abschnitt 'stop' im Ant-Ziel nicht, dass das Ziel fehlschlägt.

3.5.4 Serverprotokollierung

Die Server werden als Windows-Dienste gestartet und gestoppt und sie werden in der Liste dieser Dienste im Format `<Domänenname>_<Servername>` aufgeführt. Die Standardserverprotokollierung dieses Dienstes wird in der Datei `<bea.home>/user_projects/domains/<knoten.name>/servers/<server.name>/logs/<server.name>_Redirect.log` gespeichert. Dieses Protokoll wird beim Starten des Servers gelöscht.

Kapitel 4. Bereitstellung

4.1 Einführung

Der letzte Schritt nach dem Packen der .ear-Dateien für die IBM Cúram Social Program Management-Anwendung und -Web-Services ist deren Bereitstellung auf dem Anwendungsserver.

Standardmäßig werden die .ear-Dateien für die Anwendung und Web-Services auf dem Server in derselben Installation von WebLogic Server bereitgestellt.

4.2 Bereitstellung

Vor der Bereitstellung einer Anwendung starten Sie WebLogic Server (erneut), wie dies in 3.5, „WebLogic Server starten und stoppen“, auf Seite 11 beschrieben ist. Stellen Sie beim Bereitstellen der IBM Cúram Social Program Management-Anwendung sicher, dass die Datenbank richtig konfiguriert ist¹².

SDEJ stellt Ant-Ziele zur Verfügung, mit denen Anwendungen auf einem WebLogic Server bereitgestellt bzw. ihre Bereitstellung zurückgenommen wird. Wie bei **startserver/stopserver**-Zielen muss auch bei **installapp/uninstallapp**-Zielen die Datei AppServer.properties richtig konfiguriert werden (siehe 3.2, „WebLogic Server-Konfiguration“, auf Seite 9). Für die Ziele müssen außerdem eine Reihe von Optionen angegeben werden, die in den folgenden Abschnitten beschrieben werden.

4.2.1 Anwendung bereitstellen

Das Ant-Ziel zum Bereitstellen oder Installieren einer Anwendung (in Form einer .ear-Datei) lautet:

build installapp

Dies erfordert die folgenden Optionen:

- **-Dserver.name**
Der Name des Servers, auf dem die Anwendung installiert werden soll.
- **-Dear.file**
Der vollständig qualifizierte Name der .ear-Datei, die installiert werden soll.
- **-Dapplication.name**
Der Name zur Identifizierung der Anwendung, wenn sie installiert ist.

```
build installapp -Dserver.name=CuramServer  
-Dear.file=d:/ear/WLS/Curam.ear  
-Dapplication.name=Curam
```

Abbildung 6. Anwendungsbeispiel

Anmerkung: Die EAR-Datei, die das Servermodul enthält, muss bereitgestellt werden, bevor andere EAR-Dateien (nur Client) installiert werden.

4.2.2 Benutzernamen SYSTEM ändern

Es wird dringend empfohlen, diesen Benutzernamen zu ändern, nachdem Sie die Anwendung mithilfe der WebLogic Server-Administrationskonsole bereitgestellt haben. Die Eigenschaft Run As User muss von

¹². Wenn die Datenbank nicht die erforderlichen Informationen enthält, kann die Bereitstellung aufgrund eines Sicherheits-/ Prüfproblems fehlschlagen.

SYSTEM in den gewünschten Benutzer geändert werden. Das Kennwort dieses Benutzers ist unerheblich, da keine Authentifizierung, sondern nur eine Identitätszusicherung von WebLogic Server für JMS-Aufrufe durchgeführt wird.

Die Eigenschaft Run As Principal Name befindet sich auf der Registerkarte 'Configuration' für jedes der bereitgestellten MDB-EJB-Module (MDB = Message Driven Bean) und muss diesem Wert entsprechen. Dies macht es außerdem erforderlich, die Eigenschaft Run As User von *myrealmCuramAuthenticator* (Curam Authentication Provider) in einen anderen Benutzer zu ändern (im Abschnitt Security Realms der WebLogic Server-Administrationskonsole). Weitere Informationen zur Verwendung der Administrationskonsole finden Sie in der Dokumentation zu WebLogic Server.

Hinweis: Wenn der Benutzername geändert wird, muss der neue Benutzername in der Datenbanktabelle 'Users' enthalten sein und dieser Benutzer muss die Rolle 'SUPERROLE' haben.

Der Benutzer SYSTEM ist der Benutzer, unter dem JMS-Nachrichten ausgeführt werden.

4.2.3 Bereitstellung einer Anwendung zurücknehmen

Das Ant-Ziel, mit dem die Bereitstellung einer Anwendung zurückgenommen oder mit dem sie deinstalliert werden kann, lautet:

build uninstallapp

Dies erfordert die folgenden Optionen:

- `-Dserver.name`
Der Name des Servers, auf dem die Anwendung installiert ist.
- `-Dapplication.name`
Der Name der (bei der Installation konfigurierten) Anwendung, die deinstalliert werden soll.

```
build uninstallapp -Dserver.name=CuramServer  
-Dapplication.name=Curam
```

Abbildung 7. Anwendungsbeispiel

4.3 JSPs vorkompilieren

Während der Bereitstellung steht ein weiteres Ziel (**precompilejsp**) zur Verfügung. Es ermöglicht die Vorkompilierung der JSPs einer `.ear-Clientdatei` vor der Installation der `.ear-Datei`. Dies beschleunigt die Anzeige einer bestimmten Seite im Web-Browser, wenn zum ersten Mal auf sie zugegriffen wird.

Folgende Optionen stehen für das Ziel **precompilejsp** zur Verfügung:

- `-Dear.file`
Der vollständig qualifizierte Name der `.ear-Datei`, die vorkompiliert werden soll.

```
build precompilejsp -Dear.file=d:/Curam.ear
```

Abbildung 8. Anwendungsbeispiel

Dieses Ziel überschreibt die Datei `<Curam.ear>` mit einer Kopie, die die vorkompilierten JSPs enthält, und kann dann wie in 4.2, „Bereitstellung“, auf Seite 13 beschrieben bereitgestellt werden.

4.4 Bereitgestellte Anwendung testen

Wenn die Anwendung¹³ auf einem konfigurierten WebLogic Server installiert wurde, sollte die Anwendung gestartet und getestet werden.

Stellen Sie hierzu sicher, dass der relevante Server gestartet wurde,¹⁴ und öffnen Sie die folgende Seite in einem Web-Browser:

`https://<eine.maschine.com>:<port>/<Kontextstammverzeichnis>`

Dabei gilt Folgendes:

`<eine.maschine.com>` gibt den Hostnamen oder die IP-Adresse der Maschine an, auf der das WebLogic Server-System ausgeführt wird; `<port>` gibt den SSL-Port des Servers an, auf dem die Anwendung bereitgestellt wird;¹⁵ und `<Kontextstammverzeichnis>` gibt das Kontextstammverzeichnis des WAR-Moduls an (vgl. 2.5, „Mehrere EAR-Dateien“, auf Seite 7).

Bevor die Seite geöffnet werden kann, wird der Browser zunächst zu einer Anmeldeseite geleitet. Melden Sie sich mit einem gültigen Benutzernamen und Kennwort für die Anwendung an. Anschließend wird der Browser zu der gewünschten Seite geleitet.

Anmerkung: Der EAR-Dateiname `Curam.ear` für Option `-Dear.file` und der Anwendungsservername `Curam` für Option `-Dapplication.name` in den Beispielen dieses Kapitels dienen nur der Illustration. Die Werte können sich abhängig von Ihrer Anwendung und Bereitstellungsstrategie ändern.

13. Möglicherweise ist die Installation einer Web-Service-Anwendung erforderlich.

14. Es ist nicht erforderlich, den Server nach der Bereitstellung einer Anwendung erneut zu starten.

15. Der SSL-Port ist um eins größer als der `curam.server.port`, der in der Datei `AppServer.properties` angegeben ist.

Anhang. Manuelle WebLogic Server-Konfiguration

A.1 Einführung

Bei Bedarf kann die WebLogic Server-Installation manuell konfiguriert werden. Dies wird jedoch nicht empfohlen. In diesem Anhang werden nur zu Informationszwecken die zur Konfiguration und Bereitstellung auf WebLogic Server erforderlichen Schritte beschrieben.¹⁶

Dieser Anhang setzt voraus, dass WebLogic Server bereits installiert ist. Im *Handbuch 'Cúram Third Party Tools Installation'*¹⁷ finden Sie ausführliche Informationen zur Installation.

A.2 WebLogic Server-Domäne erstellen

Der Assistent zur Domänenkonfiguration ist ein Tool, das Sie bei der Konfiguration einer WebLogic Server-Domäne unterstützt.

Bei Windows-Plattformen kann der Assistent zur Domänenkonfiguration wie folgt aufgerufen werden:

Programme > Oracle WebLogic > WebLogic Server 11gR1 > Tools > Configuration Wizard

Bei UNIX-Plattformen kann der Assistent zur Domänenkonfiguration wie folgt aufgerufen werden:

```
<wls.home>/weblogic/common/bin/config.sh
```

Dabei ist <wls.home> das WebLogic-Installationsverzeichnis (z. B. /opt/wls).

A.2.1 Domänenkonfiguration Informationen zu diesem Vorgang

Wenn der Assistent für die Domänenkonfiguration geöffnet wurde, führen Sie die folgenden Schritte *genau wie angegeben* durch, um die Domäne zu konfigurieren:

Vorgehensweise

1. Wählen Sie **Create a new WebLogic Domain** aus und klicken Sie auf **Next**.
2. Wählen Sie **Generate a domain configured automatically to support the following products** aus. Stellen Sie sicher, dass das Kontrollkästchen **WebLogic Server** ausgewählt ist, und klicken Sie auf **Next**.
3. Geben Sie einen Wert für **Domain name** an.
Die Angabe **Domain name** (und alle Servernamen) können beliebig gewählt werden, solange die Datei `AppServer.properties` richtig aktualisiert wird. Übernehmen Sie die **Domain location**-Standardstellung und klicken Sie auf **Next**.
4. Konfigurieren Sie zu Administrationszwecken einen Benutzernamen und ein Kennwort. Dabei ist zu beachten, dass das Kennwort mindestens acht alphanumerische Zeichen lang sein und mindestens eine Ziffer oder ein Sonderzeichen enthalten muss; beispielsweise als Benutzername „weblogic“ und als Kennwort „weblogic1“. Bestätigen Sie das Kennwort und klicken Sie auf **Next**.
5. Wählen Sie **Startup mode** und **Java SDK** (bei Bedarf) und klicken Sie auf **Next**.

16. Die manuellen Aktivitäten beziehen sich nur auf die Windows-Plattform.

17. Weitere Informationen finden Sie im Installationshandbuch für die betreffende Installation (z. B. Windows oder UNIX).

6. Nehmen Sie keine Änderungen in der Anzeige **Select Optional Configuration** vor und klicken Sie auf **Next**.
7. Prüfen Sie die Konfigurationseinstellungen und klicken Sie auf **Create**, um die Domäne zu erstellen.
8. Der Konfigurationsassistent ist nun abgeschlossen. Klicken Sie auf **Done**, um den Konfigurationsassistenten zu beenden, und verlassen Sie das Installationsprogramm.

Zur Ausführung der IBM Cúram Social Program Management-Anwendung ist ein MaxPermSize-Wert von mindestens 256m erforderlich. Dieser muss als Parameter für die neu erstellte Domäne konfiguriert werden. Bearbeiten Sie die für Ihre Plattform relevante Datei (unter <WebLogic-Installationsverzeichnis>/<Domänenrepository>/<Domänenname>/bin/):

- a. setDomainEnv.cmd (unter Windows)
- b. setDomainEnv.sh (unter UNIX)

Wo MaxPermSize übergeben wird, stellen Sie den Wert auf 256m ein.

Anmerkung: Die Einstellung 256m wird hier beispielhaft verwendet. Der optimale Wert sollte durch Überwachung des Speichers ermittelt werden, der für Ihren Server verwendet wird.

A.3 Java Virtual Machine konfigurieren

Die IBM Cúram Social Program Management-Anwendung setzt voraus, dass zusätzliche Parameter an die JVM übergeben werden, die den Anwendungsserver ausführt. Stellen Sie folgende Systemumgebungsvariablen ein:

- USER_MEM_ARGS: Hier muss ein für Ihre Anwendung relevanter Wert eingestellt werden, z. B. -Xmx712m. Diese Umgebungsvariable wird vom Script app_runtimewls.xml festgelegt. Dabei werden die folgenden Eigenschaften verwendet, für die Standardwerte definiert werden können oder die in der Eigenschaftendatei AppServer.properties festgelegt werden können:

Tabelle 1. Speicherargumente

Eigenschaftsname	Standardwert	Beschreibung
curam.server.jvm.heap.size	1024	Gibt die Anfangs- und Maximalgröße des JVM-Heapspeichers an.
curam.server.jvm.permgen.size	-XX:PermSize=128m -XX:MaxPermSize=256m	Gibt die PermGen-Anfangs- und Maximalgröße an.

Wenn Sie die Umgebungsvariable USER_MEM_ARGS direkt überschreiben, müssen Sie auch Werte für den JVM-PermGen-Bereich angeben (siehe Tabelle 1).

- WLS_REDIRECT_LOG: Ausgabedatei (z. B. <WebLogic Server-Installationsverzeichnis>/<Domänenrepository>/<Domänenname>/logs/<server.name>_Redirect.log
- JAVA_OPTIONS: Diese Umgebungsvariable dient zwei Zwecken.
 1. Die Umgebungsvariable muss zum Einstellen des automatischen Modus (headless mode) verwendet werden. Die Eigenschaft für den automatischen Modus wird nur für UNIX benötigt und muss den Wert -Djava.awt.headless=true haben.
 2. Die Umgebungsvariable kann verwendet werden, um zusätzliche Parameter an WebLogic Server JVM beim Starten des Anwendungsservers zu übergeben.

A.4 Datenbanktreiber einrichten

Die Version der Oracle-Datenbanktreiberdatei ojdbc6.jar, die von WebLogic Server verwendet wird, ist nicht identisch mit der Version, die im Lieferumfang des Anwendungsprodukts enthalten ist und sich im Verzeichnis drivers der SDEJ-Installation befindet.

Gehen Sie wie folgt vor, um WebLogic Server so einzurichten, dass der Oracle **Database Driver** verwendet, der mit dem Anwendungsprodukt geliefert wird:

- Ersetzen Sie die Datei `ojdbc6.jar` im Ordner `WLS_HOME\lib` durch eine Datei, die im Lieferumfang des Anwendungsprodukts enthalten ist und die sich im Verzeichnis `drivers` der SDEJ-Installation befindet, z. B. `D:\Curam\SDEJ\drivers`.

Dabei verweist die Variable `WLS_HOME` auf das Verzeichnis `/server` der WebLogic Server-Installation, z. B. `d:\WLS\weblogic\server` oder `/opt/wls/weblogic/server`.

Anmerkung: Die Kopien von Oracle-Thin-Treibern, die mit WebLogic Server und anderen unterstützenden Dateien installiert werden, werden im Verzeichnis `WLS_HOME\ext\jdbc\` installiert. In diesem Ordner gibt es ein Unterverzeichnis für jedes DBMS. Wenn Sie Ihre Installation später auf die mit WebLogic Server installierte Version des Treibers zurücksetzen müssen, können Sie einfach die Datei von `WLS_HOME\ext\jdbc\oracle\11g\` in den Ordner `WLS_HOME\lib` kopieren.

A.5 WebLogic Administration Service starten

Vorgehensweise

1. Öffnen Sie eine Eingabeaufforderung und navigieren Sie zum Verzeichnis `<WebLogic-Installationsverzeichnis>/<Domänenrepository>/<Domänenname>` (z. B. `D:/bea/user_projects/domains/<Domänenname>`). Führen Sie den Befehl **startWeblogic** von diesem Verzeichnis aus. Wenn der Server richtig gestartet wurde, sollte die Zeile `<Server started in RUNNING mode>` in der Protokolldatei ausgegeben werden, die zuvor in A.3, „Java Virtual Machine konfigurieren“, auf Seite 18 oder über die Eingabeaufforderung konfiguriert wurde (falls nicht bereits zuvor geschehen).
2. Um die Administrationskonsole zu öffnen, ist folgende URL in einem Web-Browser zu öffnen:
`http://<IP-Adresse>:7001/console`
3. Melden Sie sich mit dem Benutzernamen und Kennwort an, das während der Installation konfiguriert wurde (z. B. „weblogic“/„weblogic1“).

A.6 Datenbank konfigurieren

Informationen zu diesem Vorgang

Öffnen Sie die Administrationskonsole, wie im vorherigen Abschnitt beschrieben wurde.

Vorgehensweise

1. Navigieren Sie zu `<Domänenname> > Services > JDBC > Data Sources`.
2. Klicken Sie auf die Schaltfläche **New**.
3. Nehmen Sie folgende Eingaben in den Feldern vor:
Name: „curamdb“
JNDI Name: „jdbc/curamdb“
Ändern Sie den Wert für **Database Type** in „Oracle“.
4. Klicken Sie auf die Schaltfläche **Next**.
5. Stellen Sie den Wert für **Database Driver** wie folgt ein: „Oracles Driver (Thin XA) for Instance connections; Versions:9.0.1,9.2.0,10,11“
6. Klicken Sie auf die Schaltfläche **Next**.
7. Übernehmen Sie den Standardwert für **Transaction Options** und klicken Sie auf die Schaltfläche **Next**.
8. Nehmen Sie in den Feldern folgende Einstellungen vor:
Database Name: Diese Einstellung hängt davon ab, wie Sie die Verbindung zur Oracle-Datenbank herstellen wollen (d. h. entweder über den Oracle-Servicenamen oder den Oracle-SID-Namen).
Stellen Sie diesen Wert auf den Wert von `curam.db.oracle.servicename` in `<SERVERVERZEICHNIS>/project/properties/Bootstrap.properties` ein, um die Verbindung zur Datenbank über den Oracle-Servicenamen herzustellen.

Wenn Sie die Verbindung zur Oracle-Datenbank über den SID-Namen herstellen wollen, stellen Sie diesen Wert auf den Wert von `curam.db.name` in `<SERVERVERZEICHNIS>/project/properties/Bootstrap.properties` ein (z. B. „yourhost“).

Hostname: Stellen Sie diesen Wert auf den Wert von `curam.db.servername` in `<SERVERVERZEICHNIS>/project/properties/Bootstrap.properties` ein (z. B. „gonzo.<hostname>“).

Port: Stellen Sie diesen Wert auf den Wert von `curam.db.serverport` in `<SERVERVERZEICHNIS>/project/properties/Bootstrap.properties` ein (z. B. „1521“).

Database Username: Stellen Sie diesen Wert auf den Wert von `curam.db.username` in `<SERVERVERZEICHNIS>/project/properties/Bootstrap.properties` ein (z. B. „curam“).

Password: Stellen Sie diesen Wert auf den Wert von `curam.db.password` in `<SERVERVERZEICHNIS>/project/properties/Bootstrap.properties` ein. Dabei ist zu beachten, dass das Kennwort in der `Bootstrap.properties`-Eigenschaftendatei verschlüsselt ist, Sie aber die nicht verschlüsselte Version dieses Kennworts hier einstellen müssen.

Confirm Password: Bestätigen Sie das eingegebene Kennwort.

9. Klicken Sie auf die Schaltfläche **Next**.
10. Wenn Sie hier die Verbindung über den Oracle-Servicenamen herstellen, ändern Sie den **URL**-Wert, wie unten angegeben, bevor Sie die Konfiguration testen:
`jdbc:oracle:thin:@//serverName:port/databaseServiceName`
serverName ist der Name des Servers, auf dem die Datenbank gehostet wird.
port ist die Nummer des Ports, an dem die Datenbank empfangsbereit ist.
databaseServiceName ist der Servicename der Datenbank.
Lassen Sie alle anderen Felder unverändert, es sei denn, eine bestimmte Änderung ist erforderlich.
Klicken Sie auf die Schaltfläche **Test Configuration**, um die Einstellungen zu testen.
11. Klicken Sie auf die Schaltfläche **Next**.
12. Prüfen Sie die Einstellungen und klicken Sie auf die Schaltfläche **Next**.
13. Wählen Sie **AdminServer** als Zielservers aus.

Anmerkung:

Bei manueller Konfiguration müssen Sie den Wert der Eigenschaft `curam.server.name` in der Datei `AppServer.properties` auf `AdminServer` einstellen.

14. Klicken Sie auf die Schaltfläche **Finish**.
Ein sinnvoller Lösungsansatz ist hier, den `AdminServer` jetzt erneut zu starten, um sicherzustellen, dass die Änderungen korrekt vorgenommen wurden. Gehen Sie hierzu wie folgt vor:
 - a. Navigieren Sie zu `<Domänenname> > Environment > Servers`.
 - b. Wählen Sie die Registerkarte **Control** aus, wählen Sie anschließend 'AdminServer' in der Serverliste aus und klicken Sie auf **Shutdown > When work completes**.¹⁸
 - c. Klicken Sie auf die Schaltfläche **Yes**, um den `AdminServer` herunterzufahren.

18. Es ist zwar sehr einfach, einen verwalteten Server mittels Strg+C abzubrechen, wenn dieser an einer Eingabeaufforderung ausgeführt wird, doch sollte diese Tastenkombination grundsätzlich nicht verwendet werden. Verwenden Sie immer die Administrationskonsole, um alle verwalteten Server wie oben beschrieben herunterzufahren. Bei Verwendung von Strg+C von einer Eingabeaufforderung verlangsamt sich eine Maschine nach einer Reihe von Neustarts ganz erheblich. Dies liegt daran, dass der Speicher nicht wie eigentlich erforderlich freigegeben wird. Abhilfe bietet hier nur ein Neustart der Maschine.

A.7 Sicherheit einrichten

Informationen zu diesem Vorgang

Kopieren Sie die Datei CuramMBean.jar aus dem Verzeichnis <SDEJ>/lib in das Verzeichnis <WebLogic Server-Installationsverzeichnis>/weblogic/server/lib/mbeantypes (<WebLogic Server-Installationsverzeichnis>/wlserver_10.3/server/lib/mbeantypes für 10.3.3).

Starten Sie den AdminServer erneut und starten Sie die Administrationskonsole, wie im vorherigen Abschnitt beschrieben wurde.

Vorgehensweise

1. Navigieren Sie zu <Domänenname> > **Security Realms**;
2. Klicken Sie auf **myrealm** in der Liste **Realms**.
3. Klicken Sie auf die Registerkarte **Providers**.
4. Klicken Sie auf die Registerkarte **Authentication**.
5. Klicken Sie auf die Schaltfläche **New**.
6. Nehmen Sie folgende Eingaben in den Feldern vor:
Name : „myrealmCuramAuthenticator“
Type : „CuramAuthenticator“
7. Klicken Sie auf die Schaltfläche **OK**.
8. Klicken Sie in der Liste von **Authentication Providers** auf das Kontrollkästchen **DefaultAuthenticator**.
9. Klicken Sie auf die Schaltfläche **Delete**.
10. Klicken Sie in der Liste **Authentication Providers** auf **myrealmCuramAuthenticator**.
11. Stellen Sie sicher, dass der Wert von **Control Flag** auf „REQUIRED“ eingestellt ist.
Wenn nicht, ändern Sie den Wert in „REQUIRED“.
12. Klicken Sie auf die Schaltfläche **Save**.
13. Wählen Sie die Registerkarte **Provider Specific** aus. Diese Registerkarte enthält Einstellungen für das Konfigurieren der Cúram-Sicherheit in WebLogic Server. Die Standardwerte sollten nur geändert werden, wenn Sie die Sicherheitskonfiguration ändern möchten. Tabelle 2 erläutert die Details zu den verschiedenen Optionen.
Sie müssen das verschlüsselte Kennwort für den Wert 'Admin Password' eingeben. Generieren Sie dieses Kennwort, indem Sie das mitgelieferte Ant-Verschlüsselungsziel ausführen; Beispiel: `ant digest -Dpassword=weblogic1`.
Wenn Sie Änderungen vornehmen, klicken Sie danach auf die Schaltfläche **Save**.
14. Klicken Sie auf die Schaltfläche **Save**. Stellen Sie sicher, dass keine Fehler vorliegen.

Tabelle 2. Providerspezifische Optionen

Feld	Beschreibung
Check Identity Only	Optional. Wenn dieses Feld aktiviert ist, führt der Authentifizierungsprovider nicht die üblichen Authentifizierungsprüfungen durch. Stattdessen stellt er nur sicher, dass der Benutzer in der Datenbanktabelle enthalten ist. Diese Option wird benötigt, wenn LDAP-Unterstützung erforderlich ist oder wenn ein alternativer Authentifizierungsmechanismus verwendet werden soll.
Admin Username	Erforderlich. Dies ist der Benutzername des WebLogic Server-Benutzers mit Administrationsrechten. Dieser Benutzer ist von der Cúram-Authentifizierung ausgenommen.

Tabelle 2. Providerspezifische Optionen (Forts.)

Feld	Beschreibung
Admin Password	Erforderlich. Dies ist das verschlüsselte Kennwort des WebLogic Server-Benutzers mit Administrationsrechten. Generieren Sie das verschlüsselte Kennwort, indem Sie das mitgelieferte Ant-Verschlüsselungsziel ausführen; Beispiel: <code>ant digest -Dpassword=weblogic1</code> .
Port	Erforderlich. Dies ist der Port der Maschine, auf der die Cúram-Anwendung ausgeführt wird. Der Standardwert ist 7003. In einer Clusterumgebung muss hierfür eine Liste mit Ports angegeben werden, die durch Kommas (,) getrennt sind, sodass mehrere Server unterstützt werden.
Login Trace	Optional. Dieses Feld sollte aktiviert werden, um ein Debugging für den Authentifizierungsprozess durchzuführen. Wenn das Feld aktiviert wird, bewirkt der Aufruf des Cúram-Authentifizierungsproviders, dass Traceinformationen zur WebLogic Server-Protokolldatei hinzugefügt werden.
Run As User	Erforderlich. Eine Beschreibung dieser Eigenschaft finden Sie im Abschnitt 4.2.2, „Benutzernamen SYSTEM ändern“, auf Seite 13. Der Standardwert ist SYSTEM.
Hostname	Erforderlich. Dies ist der Hostname der Maschine, auf der die Cúram-Anwendung ausgeführt wird. Der Standardwert ist localhost. In einer Clusterumgebung muss hierfür eine Liste mit Hostnamen angegeben werden, die durch Kommas (,) getrennt sind, sodass mehrere Server unterstützt werden.

Anmerkung: Beim Konfigurieren des Cúram Authenticator Provider in einer Clusterumgebung ist die Reihenfolge der Attribute hostname und port wichtig. Es besteht eine 1:1-Zuordnung zwischen den angegebenen Servern und Ports. Beispiel:

```
Port=7001,7003,7005
Hostname=host1,host2,host3
```

Hier führt host1 den WebLogic Server an Port 7001 aus und host3 führt den WebLogic Server an Port 7005 aus.

Sie müssen die SSL-Unterstützung aktivieren. Gehen Sie hierzu wie folgt vor:

- Navigieren Sie zu **<Domänenname> > Environment > Servers**.
- Wählen Sie den AdminServer aus der Liste der Server aus.
- Klicken Sie auf der Registerkarte **General** das Kontrollkästchen **SSL Listen Port Enabled** an.
- Klicken Sie auf die Schaltfläche **Save**.
- Starten Sie den Server erneut, damit die Änderungen wirksam werden.

A.8 JMS einrichten

Informationen zu diesem Vorgang

Die Cúram-Anwendung verwendet persistente Nachrichten. Ein JMS-Dateispeicher muss zum Speichern persistenter Nachrichten erstellt werden. Um ein Verzeichnis in dem Dateisystem zu erstellen, in dem der JMS-Dateispeicher angelegt werden soll (z. B. `<WebLogic Server-Installationsverzeichnis>/user_projects/domains/<Domänenname>/jms_file_store`), führen Sie die folgenden Schritte über die Administrationskonsole aus:

Vorgehensweise

1. Navigieren Sie zu **<Domänenname> > Services > Persistent Stores**.
2. Klicken Sie auf **New > Create FileStore** und stellen Sie folgende Eigenschaften ein:
Name: „CuramJMSFileStore“
Target: „AdminServer“
Directory: <das oben erstellte Verzeichnis>;
3. Klicken Sie auf die Schaltfläche **Finish** (OK bei WebLogic Server 10.3.3).
4. Navigieren Sie zu **<Domänenname> > Services > Messaging > JMS Servers**.
5. Klicken Sie auf die Schaltfläche **New**, um einen neuen JMS-Server zu konfigurieren, und stellen Sie folgende Eigenschaften ein:
Name: „CuramJMSServer“
Persistent Store: „CuramJMSFileStore“
6. Klicken Sie auf die Schaltfläche **Next** und wählen Sie den AdminServer als Zielsever aus.
7. Klicken Sie auf die Schaltfläche **Finish**, um den Vorgang abzuschließen.

A.9 Warteschlange einrichten

Vorgehensweise

1. Navigieren Sie zu **<Domänenname> > Services > Messaging > JMS Modules**.
2. Klicken Sie auf die Schaltfläche **New** und geben Sie die folgenden Details ein:
Name: „op-jms“
Descriptor File Name: „jms/op-jms.xml“
3. Klicken Sie auf die Schaltfläche **Next** und aktivieren Sie das Kontrollkästchen **AdminServer** als Zielsever.
4. Klicken Sie auf die Schaltfläche **Next**.
5. Klicken Sie auf das Kontrollkästchen **Would you like to add resources?** und klicken Sie auf die Schaltfläche **Finish**, um den Vorgang abzuschließen.
Es gibt vier reguläre Warteschlangen und zwei Fehlerwarteschlangen, die konfiguriert werden müssen. Die Fehlerwarteschlangen müssen zuerst konfiguriert werden. Die folgende Einrichtung muss wiederholt werden. Dabei ist **<Warteschlangennamenname>** durch jede der folgenden Warteschlangen zu ersetzen (in der angegebenen Reihenfolge): CuramDeadMessageQueue, DPEError, WorkflowError, DPE-nactment, WorkflowEnactment und WorkflowActivity.
6. Navigieren Sie zu **<Domänenname> > Services > Messaging > JMS Modules**.
7. Klicken Sie in der Liste **JMS Modules** auf **op-jms**.
8. Klicken Sie auf der Registerkarte **Configuration** auf die Schaltfläche **New**.
9. Wählen Sie **Type** „Quota“ aus und klicken Sie auf die Schaltfläche **Next**.
10. Geben Sie die folgenden Details ein:
Name: „<QueueName>.Quota“
Übernehmen Sie den Standardwert für **Bytes Maximum:** „9223372036854775807“
Übernehmen Sie den Standardwert für **Messages Maximum:** „9223372036854775807“
Übernehmen Sie den Standardwert für **Policy:** „FIFO“
Übernehmen Sie den Standardwert für **Shared:** „False“.
11. Klicken Sie auf die Schaltfläche **OK**.
12. Klicken Sie auf die Schaltfläche **New** und wählen Sie **Type** „Queue“ aus.
13. Klicken Sie auf die Schaltfläche **Next** und geben Sie die folgenden Details ein:
Name: „<Warteschlangennamenname>“
JNDI Name: „jms/<Warteschlangennamenname>“.

14. Klicken Sie auf die Schaltfläche **Next**.
15. Wenn die Dropdown-Liste **SubDeployments** leer ist, klicken Sie auf **Create a New SubDeployment** und geben Sie Folgendes für **SubDeployment Name** ein: „CuramJMSServer“
16. Klicken Sie auf die Schaltfläche **OK**.
17. Stellen Sie **SubDeployment** wie auf folgenden Wert ein: „CuramJMSServer“
18. Wählen Sie „CuramJMSServer“ als JMS-Zielservers aus.
19. Klicken Sie auf die Schaltfläche **Finish**.
20. Klicken Sie auf den gerade konfigurierten „<Warteschlangennamen>“.
21. Klicken Sie die Registerkarte **Thresholds and Quotas**.
22. Stellen Sie **Quota** auf folgenden Wert ein: „<Warteschlangenname>.Quota“.
23. Klicken Sie auf die Schaltfläche **Save**.
24. Wählen Sie die Registerkarte **Overrides** aus und stellen Sie **Delivery Mode Override** auf „Persistent“ ein. Klicken Sie auf **Save**.
25. Wählen Sie die Registerkarte **Delivery Failure** aus und stellen Sie **Redelivery Limit** auf „1“ ein. Stellen Sie **Error Destination** auf „none“ für „CuramDeadMessageQueue“, „CuramDeadMessageQueue“ für „DPError“ und „WorkflowError“, „DPError“ für „DPEnactment“ und „WorkflowError“ für „WorkflowEnactment“ und „WorkflowActivity“ ein.
26. Klicken Sie auf die Schaltfläche **Save**.

A.10 Warteschlangen-Verbindungsfactory einrichten

Informationen zu diesem Vorgang

Führen Sie die folgenden Schritte aus, um die XA-Warteschlangen-Verbindungsfactory zu konfigurieren:

Vorgehensweise

1. Navigieren Sie zu <Domänenname> > **Services** > **Messaging** > **JMS Modules**.
2. Klicken Sie in der Liste **JMS Modules** auf **op-jms**.
3. Klicken Sie in der Registerkarte **Configuration** auf die Schaltfläche **New**.
4. Wählen Sie **Type** „Connection Factory“ aus und klicken Sie auf **Next**.
5. Nehmen Sie in den Feldern folgende Einstellungen vor:
Name: „CuramQueueConnectionFactory“
JNDIName: „jms/CuramQueueConnectionFactory“
6. Klicken Sie auf die Schaltfläche **Next**.
7. Klicken Sie auf die Schaltfläche **Finish**.
8. Klicken Sie auf die gerade konfigurierte „CuramQueueConnectionFactory“.
9. Wählen Sie die Registerkarte **Configuration** und anschließend die untergeordnete Registerkarte **Transactions** aus. Stellen Sie sicher, dass **XAConnection Factory Enabled** ausgewählt ist. Klicken Sie auf die Schaltfläche **Save**.
Führen Sie die folgenden Schritte aus, um die Nicht-XA-Warteschlangen-Verbindungsfactory zu konfigurieren:
10. Navigieren Sie zu <Domänenname> > **Services** > **Messaging** > **JMS Modules**.
11. Klicken Sie in der Liste **JMS Modules** auf **op-jms**.
12. Klicken Sie in der Registerkarte **Configuration** auf die Schaltfläche **New**.
13. Wählen Sie **Type** „Connection Factory“ aus und klicken Sie auf **Next**.
14. Nehmen Sie in den Feldern folgende Einstellungen vor:
Name: „CuramQueueConnectionFactoryNonXA“
JNDI Name: „jms/CuramQueueConnectionFactoryNonXA“

15. Klicken Sie auf die Schaltfläche **Next**.
16. Klicken Sie auf die Schaltfläche **Advanced Targeting**.
17. Wenn die Dropdown-Liste **SubDeployments** leer ist, klicken Sie auf **Create a New SubDeployment** und geben Sie Folgendes für **SubDeployment Name** ein: „CuramJMSServer“.
18. Klicken Sie auf die Schaltfläche **OK**.
19. Stellen Sie **SubDeployment** wie folgt ein: „CuramJMSServer“
20. Wählen Sie „CuramJMSServer“ als JMS-Zielservers aus.
21. Klicken Sie auf die Schaltfläche **Finish**.
22. Klicken Sie auf die gerade konfigurierte „CuramQueueConnectionFactoryNonXA“.
23. Wählen Sie die Registerkarte **Configuration** und anschließend die untergeordnete Registerkarte **Transactions** aus. Stellen Sie sicher, dass **XAConnection Factory Enabled** nicht ausgewählt ist. Klicken Sie auf die Schaltfläche **Save**.

A.11 Topic einrichten

Informationen zu diesem Vorgang

Um das erneute Laden des Cache in der Cúram-Anwendung zu unterstützen, muss ein Topic wie folgt eingerichtet werden:

Vorgehensweise

1. Navigieren Sie zu <Domänenname> > **Services** > **Messaging** > **JMS Modules**.
2. Klicken Sie in der Liste **JMS Modules** auf **op-jms**.
3. Klicken Sie auf der Registerkarte **Configuration** auf die Schaltfläche **New**.
4. Wählen Sie **Type** „Quota“ aus und klicken Sie auf **Next**.
5. Geben Sie die folgenden Details ein:
 - Name:** „CuramCacheInvalidationTopic.Quota“
 - Bytes Maximum:** Übernehmen Sie den Standardwert „9223372036854775807“.
 - Messages Maximum:** Übernehmen Sie den Standardwert „9223372036854775807“.
 - Policy:** Übernehmen Sie den Standardwert „FIFO“.
 - Shared:** Übernehmen Sie den Standardwert „False“.
6. Klicken Sie auf die Schaltfläche **OK**.
7. Klicken Sie auf die Schaltfläche **New** und wählen Sie **Type** „Topic“ aus.
8. Klicken Sie auf **Next** und geben Sie die folgenden Details ein:
 - Name:** „CuramCacheInvalidationTopic“
 - JNDI Name:** „jms/CuramCacheInvalidationTopic“
9. Klicken Sie auf die Schaltfläche **Next**.
10. Wenn die Dropdown-Liste **SubDeployments** leer ist, klicken Sie auf **Create a New SubDeployment** und geben Sie Folgendes für **SubDeployment Name** ein: „CuramJMSServer“.
11. Klicken Sie auf die Schaltfläche **Finish**.
12. Stellen Sie **SubDeployment** wie auf folgenden Wert ein: „CuramJMSServer“
13. Wählen Sie „CuramJMSServer“ als JMS-Zielservers aus.
14. Klicken Sie auf die Schaltfläche **Finish**.
15. Klicken Sie auf den gerade konfigurierten Wert „CuramCacheInvalidationTopic“.
16. Klicken Sie auf die Registerkarte **Thresholds and Quotas**.
17. Stellen Sie **Quota** auf folgenden Wert ein: „CuramCacheInvalidationTopic.Quota“.
18. Klicken Sie auf die Schaltfläche **Save**.

19. Wählen Sie die Registerkarte **Delivery Failure** aus und stellen Sie **Redelivery Limit** auf „1“ ein. Prüfen Sie, ob **Error Destination** auf (none) eingestellt ist, und klicken Sie auf die Schaltfläche **Save**.

A.12 Topic-Verbindungsfactory einrichten

Vorgehensweise

1. Navigieren Sie zu <Domänenname> > **Services** > **Messaging** > **JMS Modules**.
2. Klicken Sie in der Liste **JMS Modules** auf **op-jms**.
3. Klicken Sie auf der Registerkarte **Configuration** auf die Schaltfläche **New**.
4. Wählen Sie **Type** „Connection Factory“ aus und klicken Sie auf **Next**.
5. Nehmen Sie in den Feldern folgende Einstellungen vor:
Name: „CuramTopicConnectionFactory“
JNDIName: „jms/CuramTopicConnectionFactory“
6. Klicken Sie auf die Schaltfläche **Next**.
7. Klicken Sie auf die Schaltfläche **Advanced Targeting**.
8. Wenn die Dropdown-Liste **SubDeployments** leer ist, klicken Sie auf **Create a New SubDeployment** und geben Sie Folgendes für **SubDeployment Name** ein: „CuramJMSServer“.
9. Klicken Sie auf die Schaltfläche **OK**.
10. Stellen Sie **SubDeployment** auf folgenden Wert ein: „CuramJMSServer“
11. Wählen Sie „CuramJMSServer“ als JMS-Zielserver aus.
12. Klicken Sie auf die Schaltfläche **Finish**.
13. Klicken Sie auf den gerade konfigurierten Wert „CuramTopicConnectionFactory“.
14. Wählen Sie die Registerkarte **Configuration** und anschließend die untergeordnete Registerkarte **Transactions** aus. Stellen Sie sicher, dass **XAConnection Factory Enabled** ausgewählt ist. Klicken Sie auf die Schaltfläche **Save**.

A.13 Manuelle Bereitstellung ausführen

Informationen zu diesem Vorgang

Es ist möglich, eine .ear-Datei über die Administrationskonsole manuell bereitzustellen. Hierfür muss der betreffende Server gestartet werden. Führen Sie in der Administrationskonsole die folgenden Schritte aus:

Vorgehensweise

1. Navigieren Sie zu <Domänenname> > **Deployments**.
2. Klicken Sie auf **Install**.
3. Navigieren Sie zu der Position der .ear-Datei.
Die Standardposition für die Server-.ear-Datei ist:
%SERVERVERZEICHNIS%/build/ear/WLS/Curam.ear
4. Wählen Sie die .ear-Datei aus der Liste aus und klicken Sie auf die Schaltfläche **Next**.
5. Übernehmen Sie die Standardwerte und klicken Sie auf die Schaltfläche **Next**.
6. Übernehmen Sie die Standardwerte und klicken Sie auf die Schaltfläche **Finish**.
7. Wählen Sie die gerade bereitgestellte Anwendung in der Liste **Deployments** aus und klicken Sie auf **Start** > **Servicing all requests**.
8. Klicken Sie auf die Schaltfläche **Yes**.
9. Testen Sie abschließend die Anwendungsbereitstellung.
Rufen Sie z. B. die URL der bereitgestellten Anwendung mit einem Web-Browser auf. Beispiel:
<https://localhost:7002/Curam>.

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden. Für die in diesem Handbuch beschriebenen Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing

IBM Europe, Middle East & Africa

Tour Descartes

2, avenue Gambetta

92066 Paris La Defense

France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann jederzeit ohne vorherige Bekanntgabe Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht. Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Corporation

Dept F6, Bldg 1

294 Route 100

Somers NY 10589-3216

U.S.A.

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

Die Lieferung des in diesem Dokument beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen.

IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können u. U. von den hier genannten Preisen abweichen.

Diese Veröffentlichung dient nur zu Planungszwecken. Die in dieser Veröffentlichung enthaltenen Informationen können geändert werden, bevor die beschriebenen Produkte verfügbar sind.

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

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM übernimmt keine Haftung für Schäden, die durch die Verwendung der Beispielprogramme entstehen.

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

© (Name Ihres Unternehmens) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corporation abgeleitet.

© Copyright IBM Corp. _enter the year or years_. Alle Rechte vorbehalten.

Wird dieses Buch als Softcopy (Book) angezeigt, erscheinen keine Fotografien oder Farbabbildungen.

Marken

IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der International Business Machines Corp. in den USA und/oder anderen Ländern. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Website "Copyright and trademark information" unter <http://www.ibm.com/legal/us/en/copytrade.shtml>.

Apache ist eine Marke der Apache Software Foundation.

Microsoft und Windows sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

UNIX ist in den USA und/oder anderen Ländern eine eingetragene Marke der Open Group.

Oracle, WebLogic Server, Java und alle auf Java basierenden Marken und Logos sind eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

Andere Namen sind möglicherweise Marken der jeweiligen Rechtsinhaber. Weitere Firmen-, Produkt- und Servicenamen können Marken oder Servicemarken anderer Unternehmen sein.



Gedruckt in Deutschland