Tivoli Application Dependency Discovery Manager
Version 7.3

Administratorhandbuch

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen” auf Seite 281 gelesen werden.

Impressum

Diese Ausgabe bezieht sich auf Version 7, Release 3 von IBM® Tivoli Application Dependency Discovery Manager (Produktnummer 5724-N55) und alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer Neuausgabe geändert wird.

© Copyright International Business Machines Corporation 2006, 2020.

Inhaltsverzeichnis

Tabellen

Zu dieser Veröffentlichung

Ziel dieser PDF-Dokumentversion ist es, die Referenzinformationen aus dem Information Center in druckbarem Format bereitzustellen.

In diesem Information Center verwendete Konventionen

In der IBM Tivoli Application Dependency Discovery Manager (TADDM)-Dokumentation gelten bestimmte Konventionen. Sie werden zur Hervorhebung der betriebssystemabhängigen Variablen und Pfade, des Verzeichnisses `COLLATION_HOME` und der Speicherposition der Datei `collation.properties` verwendet, auf die in der gesamten TADDM-Dokumentation sowie in den Nachrichten verwiesen wird.

Betriebssystemabhängige Variablen und Pfade

In diesem Information Center gelten für Umgebungsvariablen und die Schreibweise von Verzeichnissen die UNIX-Konventionen.

Wenn Sie die Windows-Befehlszeile verwenden, ersetzen Sie bei Umgebungsvariablen $ *Variable* durch %*Variable*% und ersetzen Sie in Verzeichnispfaden jeden Schrägstrich (/) durch einen umgekehrten Schrägstrich (\).

Wenn Sie die Bash-Shell auf einem Windows-System verwenden, können Sie die UNIX-Konventionen verwenden.

Verzeichnis `COLLATION_HOME`

Das TADDM-Stammverzeichnis wird auch als das Verzeichnis `COLLATION_HOME` bezeichnet.

Auf Betriebssystemen wie AIX oder Linux® ist das Standardinstallationsverzeichnis für TADDM das Verzeichnis `/opt/IBM/taddm`. Deshalb ist das Verzeichnis `$COLLATION_HOME` in diesem Fall `/opt/IBM/taddm/dist`.

Auf Windows-Betriebssystemen ist das Standardinstallationsverzeichnis für TADDM das Verzeichnis `c:\IBM\taddm`. Deshalb ist das Verzeichnis `%COLLATION_HOME%` in diesem Fall `c:\IBM\taddm\dist`.

Speicherposition der Datei `collation.properties`

Die Datei `collation.properties` enthält Eigenschaften zum TADDM-Server, einschließlich Kommentaren zu jeder Eigenschaft. Sie ist im Verzeichnis `$COLLATION_HOME/etc` zu finden.

Begriffsbestimmungen und Definitionen

In diesem Abschnitt finden Sie Begriffe und Definitionen zu den wichtigsten Konzepten von IBM Tivoli Application Dependency Discovery Manager (TADDM).

Zugriffsobjektgruppe
Eine Objektgruppe, mit der der Zugriff auf Konfigurationselemente und auf Berechtigungen zum Ändern von Konfigurationselementen gesteuert wird. Sie können Zugriffsobjektgruppen nur erstellen, wenn die Sicherheit auf Datenebene aktiviert ist.

Asynchrone Erkennung
Bei TADDM die Ausführung eines Erkennungsscripts auf einem Zielsystem, um Systeme zu erkennen, auf die vom TADDM-Server nicht direkt zugegriffen werden kann. Da diese Erkennung manuell und getrennt von einer Standarderkennung mit Berechtigungsnachweis ausgeführt wird, wird sie als "asynchron" bezeichnet.

Geschäftsanwendung
Eine Sammlung von Komponenten, mit denen Geschäftsfunktionen bereitgestellt werden, die Sie intern, extern oder gemeinsam mit anderen Geschäftsanwendungen verwenden können.

© Copyright IBM Corp. 2006, 2020

KE
Siehe *Konfigurationselement*.

Datenerfassung
Bei TADDM eine Gruppe von Konfigurationselementen.

Konfigurationselement (KE)
Eine Komponente der IT-Infrastruktur, die vom Konfigurationsmanagement gesteuert wird und daher dem formalen Änderungsmanagement unterliegt. Jedes Konfigurationselement in der TADDM-Datenbank ist mit einem persistenten Objekt- und Änderungsprotokoll verbunden. Beispiele für Konfigurationselemente sind das Betriebssystem, die Schnittstelle der Ebene 2 (L2) und die Größe des Datenbankpufferpools.

Erkennung mit Berechtigungsnachweis
TADDM-Sensor-Scannen, das detaillierte Informationen zu den folgenden Elementen erkennt:

- Jedes Betriebssystem in der Laufzeitumgebung. Dieses Scannen wird auch als Erkennung der Ebene 2 bezeichnet und erfordert Berechtigungsnachweise für das Betriebssystem.
- Die Anwendungsinfrastruktur, die implementierten Softwarekomponenten, physischen Server, Netzeinheiten, virtuellen Systeme und Hostdaten, die in der Laufzeitumgebung verwendet werden. Dieses Scannen wird auch als Erkennung der Ebene 3 bezeichnet und erfordert Berechtigungsnachweise sowohl für das Betriebssystem als auch für die Anwendung.

Erkennung ohne Berechtigungsnachweis
Scannen mit TADDM-Sensoren, bei dem Sie Basisinformationen zu den aktiven Computersystemen in der Laufzeitumgebung erkennen können. Dieses Scannen wird auch als Erkennung der Ebene 1 bezeichnet und erfordert keine Berechtigungsnachweise.

Datenmanagementportal
Die webbasierte TADDM-Benutzeroberfläche zum Anzeigen und Bearbeiten der Daten in einer TADDM-Datenbank. Diese Benutzerschnittstelle kann für Domänenserverimplementierungen, für Synchronisationsserverimplementierungen und für Speicherserver in Streaming-Serverimplementierungen verwendet werden. Die Funktionen der Benutzerschnittstelle sind bei allen Implementierungen sehr ähnlich. Bei der Synchronisationsserverimplementierung stehen jedoch einige Zusatzfunktionen zum Hinzufügen und Synchronisieren von Domänen zur Verfügung.

Arbeitsthread erkennen
Bei TADDM ein Thread, der Sensoren ausführt.

Discovery Management Console
Die TADDM-Client-Benutzeroberfläche zum Verwalten von Erkennungen. Diese Konsole wird auch als Produktkonsole (Product Console) bezeichnet. Sie kann in einer Domänenserverimplementierung und für Erkennungsserver in einer Streaming-Serverimplementierung verwendet werden. Die Funktion der Konsole ist in beiden Implementierungen identisch.

Erkennungsserver
Ein TADDM-Server, der Sensoren in einer Streaming-Server-Implementierung ausführt, aber keine eigene Datenbank besitzt.

Domäne
Bei TADDM eine logische Untergruppe der Infrastruktur eines Unternehmens oder einer anderen Organisation. Domänen können organisatorische, funktionale oder geografische Grenzen umreißen.

Domänenserver
Ein TADDM-Server, der Sensoren in einer Domänenserver-Implementierung ausführt und eine eigene Datenbank besitzt.

Domänenserverimplementierung
Eine TADDM-Implementierung mit nur einem Domänenserver. Eine Domänenserverimplementierung kann Teil einer Synchronisationsserverimplementierung sein.

Bei einer Domänenserverimplementierung muss die folgende Eigenschaft des TADDM-Servers auf den folgenden Wert gesetzt werden:

```
com.collation.cmdbmode=domain
```

Kontextbezogen aufrufen
Die Möglichkeit des nahtlosen Übergangs von einer Tivoli-Produktbenutzeroberfläche zu einer anderen Tivoli-Produktbenutzeroberfläche (in einer anderen Konsole oder in derselben Konsole bzw. Portalschnittstelle), wobei nur eine einmalige Anmeldung erforderlich ist und die Zielbenutzeroberfläche so angezeigt wird, dass die Benutzer an der entsprechenden Stelle mit ihrer Task fortfahren können.

Erkennung der Ebene 1
Scannen mit TADDM-Sensoren, bei dem Sie Basisinformationen zu den aktiven Computersystemen in der Laufzeitumgebung erkennen können. Dieser Scanvorgang wird auch als Erkennung ohne Berechtigungsnachweise bezeichnet, da keine Berechtigungsnachweise erforderlich sind. Dabei wird der Stack-Scan-Sensor und der IBM® Tivoli® Monitoring Scope-Sensor verwendet. Die Erkennung der Ebene 1 ist sehr oberflächlich. Sie erfasst lediglich den Hostnamen, das Betriebssystem, die IP-Adresse, den vollständig qualifizierten Domänennamen und die MAC-Adresse (MAC - Media Access Control) jeder erkannten Schnittstelle. Die Erkennung der MAC-Adresse ist außerdem auf Linux on System z®- und Windows-Systeme beschränkt. Bei der Erkennung der Ebene 1 werden keine Teilnetze erkannt. Für alle erkannten IP-Schnittstellen, die zu keinem bei der Erkennung der Ebene 2 oder Ebene 3 erkannten vorhandenen Teilnetz gehören, werden neue Teilnetze auf Basis des Werts der Eigenschaft `com.collation.IpNetworkAssignmentAgent.defaultNetmask` in der Datei `collation.properties` erstellt.

Erkennung der Ebene 2
Scannen mit TADDM-Sensoren, bei dem detaillierte Informationen zu den einzelnen Betriebssystemen in der Laufzeitumgebung erkannt werden. Dieser Scanvorgang wird auch als Erkennung mit Berechtigungsnachweis bezeichnet und erfordert Berechtigungsnachweise für das Betriebssystem. Bei der Erkennung der Ebene 2 werden Anwendungsnamen und die Betriebssystemnamen und Portnummern erfasst, die den einzelnen aktiven Anwendungen zugeordnet sind. Wenn eine Anwendung eine TCP/IP-Verbindung zu einer anderen Anwendung eingerichtet hat, wird diese Information als eine Abhängigkeit erfasst.

Erkennung der Ebene 3
Scannen mit TADDM-Sensoren, bei dem detaillierte Informationen zur Anwendungsinfrastruktur, zu implementierten Softwarekomponenten, physischen Servern, Netzeinheiten, virtuellen Systemen und Hostdaten ermittelt werden, die in der Laufzeitumgebung verwendet werden. Dieser Scanvorgang wird auch als Erkennung mit Berechtigungsnachweis bezeichnet und erfordert Berechtigungsnachweise sowohl für das Betriebssystem als auch für die Anwendung.

Mandantenfähigkeit
In TADDM die Verwendung einer TADDM-Installation durch einen Service-Provider oder IT-Anbieter zur Erkennung mehrerer Kundenumgebungen. Außerdem kann der Service-Provider oder IT-Anbieter die Daten aus allen Kundenumgebungen sehen, in den einzelnen Kundenumgebungen können jedoch nur die Daten in der Benutzeroberfläche oder in Berichten in dieser Kundenumgebung angezeigt werden, die für den jeweiligen Kunden spezifisch sind.

Produktkonsole
Siehe *Erkennungsverwaltungskonsole*.

Scriptbasierte Erkennung
Bei TADDM in einer Erkennung mit Berechtigungsnachweis die Verwendung derselben Sensorscripts, die von Sensoren bei der Unterstützung asynchroner Erkennung bereitgestellt werden.

SE
Siehe *Server-Entsprechung* (server equivalent).

Server-Entsprechung (server equivalent; SE)
Eine repräsentative Einheit der IT-Infrastruktur, definiert als ein Computersystem (mit Standardkonfiguration, -betriebssystem, -netzschnittstelle und -speicherschnittstelle) mit installierter Serversoftware (wie z. B. eine Datenbank, ein Web-Server oder ein Anwendungsserver). Das Konzept einer Server-Entsprechung schließt auch das Netz, den Speicher und andere Subsysteme ein, die Services für das optimale Funktionieren des Servers bereitstellen. Eine Server-Entsprechung ist vom Betriebssystem abhängig:

Betriebssystem	Ungefähre Anzahl der KE
Windows	500
AIX	1000
Linux	1000
HP-UX	500
Netzeinheiten	1000

Speicherserver
Ein TADDM-Server, der Erkennungsdaten verarbeitet, die von Erkennungsservern empfangen werden, und diese in der TADDM-Datenbank speichert. Der primäre Speicherserver koordiniert einerseits die Erkennungsserver und alle weiteren Speicherserver und dient andererseits als Speicherserver. Alle Speicherserver, die nicht primär sind, werden als sekundäre Speicherserver bezeichnet.

Streaming-Server-Implementierung
Eine TADDM-Implementierung mit einem primären Speicherserver und mindestens einem Erkennungsserver. Diese Art der Implementierung kann auch einen oder mehrere optionale sekundäre Speicherserver einschließen. Der primäre Speicherserver und die sekundären Speicherserver nutzen gemeinsam eine Datenbank. Die Erkennungsserver haben keine Datenbank.

Bei dieser Art der Implementierung fließen Erkennungsdaten parallel von mehreren Erkennungsservern zur TADDM-Datenbank.

Bei einer Streaming-Server-Implementierung muss die folgende TADDM-Server-Eigenschaft auf einen der folgenden Werte gesetzt werden:

- `com.collation.taddm.mode=DiscoveryServer`
- `com.collation.taddm.mode=StorageServer`

Für alle Server außer dem primären Speicherserver müssen die folgenden Eigenschaften (für den Hostnamen und die Portnummer des primären Speicherservers) ebenfalls gesetzt werden:

- `com.collation.PrimaryStorageServer.host`
- `com.collation.PrimaryStorageServer.port`

Wird die Eigenschaft 'com.collation.taddm.mode' gesetzt, darf die Eigenschaft 'com.collation.cmdbmode' nicht gesetzt bzw. muss auskommentiert werden.

Synchronisationsserver
Ein TADDM-Server, der Erkennungsdaten aller Domänenserver im Unternehmen synchronisiert und der seine eigene Datenbank besitzt. Dieser Server erkennt Daten nicht direkt.

Synchronisationsserverimplementierung
Eine TADDM-Implementierung mit einem Synchronisationsserver- und zwei oder mehr Domänenserver-Implementierungen, von denen jede ihre eigene lokale Datenbank hat.

Bei dieser Art von Implementierung kopiert der Synchronisationsserver Erkennungsdaten von mehreren Domänenservern domänenweise und in einem stapelorientierten Synchronisationsprozess.

Bei einer Synchronisationsserverimplementierung muss die folgende TADDM-Server-Eigenschaft auf den folgenden Wert gesetzt werden:

```
com.collation.cmdbmode=enterprise
```

Dieser Fehlertyp ist veraltet. Verwenden Sie daher bei einer neuen TADDM-Implementierung, bei der mehr als ein Server benötigt wird, die Streaming-Server-Implementierung. Ein Synchronisationsserver kann konvertiert werden, damit er künftig bei einer Streaming-Serverimplementierung als primärer Speicherserver fungiert.

TADDM-Datenbank
Bei TADDM die Datenbank, in der Konfigurationsdaten, Abhängigkeiten und Änderungsprotokoll gespeichert werden.

Jeder TADDM-Server, außer den Erkennungsservern und den sekundären Speicherservern, besitzt seine eigene Datenbank. Erkennungsserver haben keine Datenbank. Speicherserver nutzen die Datenbank gemeinsam mit dem primären Speicherserver.

TADDM-Server
Ein allgemeiner Begriff, mit dem folgende Begriffe bezeichnet werden können:

- Domänenserver in einer Domänenserverimplementierung
- Synchronisationsserver in einer Synchronisationsserverimplementierung
- Erkennungsserver in einer Streaming-Server-Implementierung
- Speicherserver (einschließlich des primären Speicherservers) in einer Streaming-Server-Implementierung

Zielsystem
Im TADDM-Erkennungsprozess das zu erkennende System.

Auslastungserkennung
Scannen mit TADDM-Sensoren, bei dem die Auslastungsinformationen für das Hostsystem erfasst werden. Für eine Auslastungserkennung sind Betriebssystem-Berechtigungsnachweise erforderlich.

Kapitel 1. Verwaltung

TADDM-Übersicht

IBM Tivoli Application Dependency Discovery Manager (TADDM) ist ein Konfigurationsmanagement-Tool, das IT-Betriebsmitarbeitern hilft, die Verfügbarkeit von Anwendungen in Anwendungsumgebungen sicherzustellen und zu verbessern. TADDM stellt Ihnen die Einzeldaten von Konfigurationselementen mithilfe einer automatisierten, agentenfreien Erkennung der Ressourcen und ihrer Anwendungsabhängigkeiten zur Verfügung und enthält eine Erkennungsbibliothekentechnologie zur leichteren Nutzung von Daten aus anderen Quellen.

TADDM stellt dem Betriebspersonal eine Top-down-Ansicht der Anwendungen zur Verfügung, in der die Struktur, der Status, die Konfiguration und das Änderungsprotokoll der geschäftskritischen Anwendungen leicht ersichtlich sind. Bei Leistungs- und Verfügbarkeitsproblemen können die Mitarbeiter die Probleme über diese Anzeige unmittelbar eingrenzen; somit ist eine effektivere Planung von Anwendungsänderungen ohne Ausfälle möglich. Die TADDM-Datenbank, eine Configuration Management Database, wird erstellt und verwaltet, ohne dass eine Modellierung einer angepassten Infrastruktur erforderlich ist. TADDM stellt außerdem vollständige schichtenübergreifende Abhängigkeitszuordnungen, topologische Ansichten, Änderungsverfolgung, Ereignisweitergabe und detaillierte Berichte und Analysen bereit.

TADDM ist von der Erkennung von Informationen abhängig, die unter Verwendung von Sensoren ausgeführt wird, die als Teil des TADDM-Produkts implementiert werden. Die Daten, die sich aus dem Erkennungsprozess ergeben, werden für die Erstellung von schichtenübergreifenden Abhängigkeitszuordnungen verwendet, die die physischen und logischen Topologien miteinander verbinden. Dieses hierarchische Verzeichnis stellt Ihre gesamte Laufzeitumgebung dar.

Die folgenden Schritte stellen eine allgemeine Zusammenfassung der Aktivitäten von TADDM dar:

1. Sensoren bestimmen und erfassen die Identität, Attribute und Einstellungen jeder Anwendung, jedes Systems und jeder Netzkomponente.
2. Die Konfigurationsdaten, Abhängigkeiten und das Änderungsprotokoll werden in der TADDM-Datenbank gespeichert, die Topologien auf dem TADDM-Server. Wenn Konfigurationselemente erkannt werden, werden sie aus den folgenden Quellen in der TADDM-Datenbank gespeichert:
   - Sensoren
   - Erkennungsbibliotheksdateien, die auch als IdML-Bücher bezeichnet werden (IdML - Identity Markup Language) und von externen Management-Software-Systemen erstellt werden
   - APIs
3. Die erkannten Daten werden als schichtenübergreifende Laufzeitanwendungstopologien in der TADDM-Benutzerschnittstelle angezeigt. Die Topologie wird durch nachfolgende Erkennungsprozesse aktualisiert. Darüber hinaus führt TADDM das Änderungsprotokoll der Infrastrukturkonfiguration und der darin bestehenden Abhängigkeiten.
4. Mit TADDM werden Berichte und zusätzliche topologische Ansichten der in der TADDM-Datenbank gespeicherten Informationen generiert.

Von TADDM erkannte Entitäten

In Tabelle 1 auf Seite 2 werden die Entitäten, die TADDM in Ihrer Umgebung erkennt, aufgeführt und beschrieben.

© Copyright IBM Corp. 2006, 2020

Tabelle 1. Erkannte Entitäten mit Beschreibungen	
Element	**Beschreibung**
Netzschicht	Die folgenden Einheiten werden mit den MIB2-Parameterwerten (RFC 1213) für jede Einheit erkannt: • Router • Switches • Lastausgleichsfunktionen • Firewalls • Generische IP-Schnittstellen
Systemschicht	Die folgenden Einheiten werden in der Systemschicht erkannt: • Server-Hosts und -Datenträger • Host-IP-Schnittstellen • Datenbankserver • Lastausgleichsfunktionen oder Cluster
Anwendungsschicht	Die folgenden Komponenten werden in der Anwendungsschicht erkannt. Für jede Komponente (generische Prozesse ausgenommen) werden außerdem die Versionsnummer, Konfigurationsdateien und Eigenschaften, Hostinformationen und herstellerspezifische Erweiterungen erkannt. • Angepasste Server nach selbst entworfenen Vorlagen • Java EE-Anwendungsserver und -Konfigurationen • Java EE- und Java SE-Komponenten und -Module • Web-Serverkomponenten • Webmodule, Konfigurationsdateien und Installationsverzeichnisse • Generische JVM-Prozesse • Datenbanken
Infrastrukturservices	Die Systeminfrastrukturservices für die Anwendungsumgebung und die Abhängigkeitsstruktur zwischen diesen Servicekomponenten und den Anwendungskomponenten werden erkannt. Folgende Komponenten gehören zum Infrastrukturservice: • DNS- und NFS-Services • LDAP
Beziehungsstruktur	Zusätzlich zur Komponentenerkennung wird die physische und logische Konnektivität der Netz-, System- und Anwendungsschichten auf der folgenden Unterstützungsstufe in jeder der Schichten erkannt: • Schicht 3 - IP-Konnektivität • Schicht 2 - Konnektivität • Laufzeitabhängigkeiten von Anwendungskomponenten • Abhängigkeiten von Infrastrukturservices

Konfigurationen und gegenseitige Abhängigkeiten werden bei den folgenden Entitäten erkannt:

- Anwendungskomponenten wie z. B. Web-Server, Anwendungsserver und Datenbanken

- Systemkomponenten wie z. B. Hosts, Betriebssysteme, Lastausgleichsfunktionen und Datenbankserver
- Netzkomponenten wie z. B. Router, Switches und Firewalls
- Infrastrukturservices wie z. B. DNS- und LDAP-Services

Anmerkung: Durch die Verwendung virtueller IP-Adressen oder mehrerer Netzschnittstellencontroller dokumentiert TADDM möglicherweise falsche Ergebnisse. Berücksichtigen Sie bei der Planung einer Erkennung die Netzinfrastruktur.

Übersicht über den Erkennungsprozess

Die Erkennung ist ein auf mehreren Ebenen ablaufender Prozess, bei dem Konfigurationsdaten zur gesamten Anwendungsinfrastruktur erfasst werden. Dabei werden implementierte Softwarekomponenten, physische Server, Netzeinheiten, virtuelle Systeme und Hostdaten ermittelt, die in der Laufzeitumgebung verwendet werden. Die Erkennung wird unter Verwendung von Sensoren ausgeführt, die als Teil des TADDM-Produkts implementiert werden.

Die Aufgabe des Sensors besteht in der Erkennung von Konfigurationselementen (CIs, Configuration Items), der Erstellung von Modellobjekten und der Persistenzdefinition der Modellobjekte in der TADDM-Datenbank. Die Sensoren verwenden Protokolle, die sich speziell auf die Ressourcen beziehen, die für die Erkennung entwickelt wurden. Im Folgenden finden Sie Beispiele für diese Protokolle:

- Cisco Discovery Protocol (CDP)
- Java™ Management Extensions (JMX)
- Secure Shell (SSH)
- Simple Network Management Protocol (SNMP)
- Structured Query Language (SQL)

Wenn möglich, wird zwischen dem TADDM-Server und den Zielsystemen eine sichere Verbindung verwendet.

TADDM führt keine Erkennungen über IPv6-Netze aus, IPv6-Attribute werden jedoch bei Erkennungen in IPv4-Netzen erkannt.

Sensoren

TADDM bietet verschiedene spezialisierte Sensoren, mit denen die Erkennung nahezu aller Komponenten im typischen Rechenzentrum, in der gesamten Anwendungssoftware, auf dem Host und in den Netzschichten ermöglicht wird. Sie können auch angepasste Sensoren für eindeutige Komponenten entwickeln. Sensoren befinden sich auf dem TADDM-Server und erfassen Konfigurationsattribute und -abhängigkeiten.

Sensoren sind nicht intrusiv, das heißt, sie werden auf dem TADDM-Server ausgeführt, nicht auf der Client-Workstation. Durch die Verwendung von TADDM können also ohne den Aufwand einer lokalen Installation und Verwaltung eines Agenten auf den einzelnen Client-Workstations, die erkannt werden sollen, erkennungsbezogene Informationen zusammengestellt werden.

Da Sensoren sichere Netzverbindungen, verschlüsselte Zugriffsberechtigungsnachweise und hosteigene Dienstprogramme verwenden, sind sie sicher und bieten dieselbe Datenübernahmestärke, die bei der Verwendung von Software auf der Client-Workstation möglich ist.

Ein Sensor verfügt über die drei folgenden konfigurierbaren Aspekte:

Bereich
Der Erkennungsbereich ist in der Regel ein gültiger IP-Bereich, ein Teilnetz oder eine bestimmte Adresse. Er legt die Grenzen für die Erkennung fest.

Zugriffsliste
Die Zugriffsliste ist eine Zusammenstellung von Berechtigungsnachweisen, z. B. der Benutzernamen, Kennwörter und SNMP-Community-Zeichenfolgen (SNMP - Simple Network Management Protocol), die der Sensor beim Zugriff auf die Konfigurationselemente in der Anwendungsinfrastruktur verwendet. Sie müssen die Zugriffsliste für die Konfigurationselemente konfigurieren, die erkannt werden sollen.

Zeitplan
Die Erkennung kann bei Bedarf, nach einem Zeitplan oder nach bestimmten, extern ausgelösten Ereignissen ausgeführt werden. Der Zeitplan gibt an, ob Sensoren bei Bedarf oder nach einem Zeitplan ausgeführt werden.

Erkennung von Konfigurationselementen durch einen Sensor

In diesen Schritten wird erläutert, wie ein Sensor Konfigurationselemente in Ihrer Umgebung erkennt.

1. Zur Identifizierung der aktiven IP-Einheiten im angegebenen Bereich versucht der Sensor, an mehreren Ports (beispielsweise 22, 23 und 135) eine Verbindung herzustellen und so eine Antwort zu erhalten. Eine beliebige Antwort genügt, um dem Sensor zu melden, dass die Einheit vorhanden ist.
2. Der Sensor versucht an mehreren Ports (beispielsweise 22 und 135) die Herstellung einer Verbindung, um die Technologie zu bestimmen, die für die Erkennung des Hosts verwendet werden soll.
3. Wenn ein Port, der das Secure Shell-Protokoll (SSH) nutzt, offen ist, versucht der Sensor, eine SSH-Verbindung mit Berechtigungsnachweisen aus der Zugriffsliste herzustellen. Der Sensor probiert der Reihe nach Zugriffslisteneinträge vom Typ **Computersystem** oder **Windows-Computersystem** aus, bis ein Eintrag erfolgreich ist oder der Sensor das Ende der Zugriffsliste erreicht, ohne dass er erfolgreich war.
4. Wenn ein WMI-Port (WMI = Windows Management Instrumentation) offen ist, wird eine SSH-Verbindung mit einem Gateway-Computersystem hergestellt (vorausgesetzt, es wird ein Computersystem für das Zielsystem gefunden). Der Sensor probiert der Reihe nach Zugriffslisteneinträge vom Typ **Windows-Computersystem** aus, bis ein Eintrag erfolgreich ist oder der Sensor das Ende der Zugriffsliste erreicht, ohne dass er erfolgreich war.
5. Wenn keine Sitzung aufgebaut werden kann, wird ein SNMP-Sensor ausgeführt. Wenn eine Sitzung aufgebaut wurde, wird ein Computersystemsensor ausgeführt.
6. Ein Computersystemsensor versucht, die Art des installierten Betriebssystems zu ermitteln.
7. TADDM führt einen Sensor aus, der speziell auf des jeweilige Betriebssystem abgestimmt ist, um so eine genauere Erkennung des Betriebssystems vornehmen zu können.
8. Während der umfassenden Erkennung des Betriebssystems, die auf bestimmten Kriterien basiert (Anschlussnummer, Prozessname usw.), werden von TADDM softwarespezifische Sensoren für die Erkennung von Anwendungsdetails gestartet.

Starten eines Anwendungssensors

In diesem Abschnitt wird beschrieben, wie ein Anwendungssensor gestartet wird.

Der GenericServerSensor führt folgende Befehle aus:

Auf den Betriebssystemen Linux, Solaris, AIX und Linux on zSeries

- **`lsof -nP -i`** zum Abruf der Portinformationen
- **`ps axww`** zum Abruf der Befehlszeileninformationen

Auf Windows-Betriebssystemen

- **`netstat.exe -nao`** zum Abruf der Portinformationen
- **`wmic process list`** zum Abruf der Befehlszeileninformationen

Über die Prozess-ID (PID) wird die Ausgabe zusammengeführt. Anschließend arbeitet die Vorlagenabgleichkomponente mit den zusammengeführten Daten. Wenn die Protokollierungsstufe in der Datei `collation.properties` auf DEBUG gesetzt ist, wird die Ausgabe dieser Befehle in die folgenden Protokolle gestellt:

- `GenericServerSensor.log`
- `DiscoverManager.log`

Die zusammengeführten Daten müssen den Kriterien entsprechen, die in der Sensorvorlage definiert sind. Die folgende Beispielschablonendefinition für den DB2-Sensor enthält die Schablonenkriterien für den Start eines Sensors.

Führen Sie folgenden Befehl aus (dabei ist die Umleitung in eine Datei hilfreich) und ersetzen Sie *<Benutzername>* und *<Kennwort>* durch einen gültigen Benutzernamen und das zugehörige Kennwort (beispielsweise `...dist/sdk/bin/api.sh -u administrator -p collation find --depth=5 AppServerTemplate`):

```
...dist/sdk/bin/api.sh -u <Benutzername> -p <Kennwort> find --depth=5 AppServerTemplate
```

Mit dem vorhergehenden Befehl wird eine XML-Ausgabe erstellt, die als Vorlagendefinition fungiert. Wenn der Wert für das Element `<order>` in der Schablonendefinition unter 0 liegt, gilt die Schablone für einen Sensor. Wenn der Wert für das Element `<order>` in der Schablonendefinition größer als 0 ist, gilt die Schablone für einen eigenen Server. Da der Abgleich beim niedrigsten Wert für das Element `<order>` beginnt, haben Sensoren eine höhere Abgleichspriorität als eigene Server.

Die folgende Beispielschablonendefinition gilt für den DB2-Sensor. Beachten Sie die beiden `<operand1>`-Elemente, von denen das eine den Wert `db2tcpcm` und das andere den Wert `db2agent` hat. Der Wert für das Element `<boolExp>` gibt an, ob beide oder nur einer der `<operand1>`-Werte vorhanden sein müssen. Der Wert 1 des Elements `<boolExp>` steht für den logischen Operator OR, was bedeutet, dass nur einer der `<operand1>`-Werte vorhanden sein muss. Der Wert `0` des Elements `<boolExp>` steht für den logischen Operator AND, was bedeutet, dass beide `<operand1>`-Werte vorhanden sein müssen.

```
<Template array="18" guid="C1A992327AFF33409C41D5C71046DBB9"
lastModified="1177555771479"
xsi:type="coll:com.collation.platform.model.discovery.template.AppServerTemplate">
   <displayName>DB2</displayName>
   <name>DB2</name>
   <type>DatabaseServer</type>
   <internal>true</internal>
   <filterSet guid="B599AED918F436C99FDA0E8EDA578F02"
   lastModified="1177555771475"
   parent="C1A992327AFF33409C41D5C71046DBB9"
   xsi:type="coll:com.collation.platform.model.discovery.template.FilterSet">
      <displayName>DB2</displayName>
      <filterList array="1"
      guid="BBE4D351653B37E38BFFD2DEBD532EE8"
      lastModified="1177555771476"
      parent="B599AED918F436C99FDA0E8EDA578F02"
      xsi:type="coll:com.collation.platform.model.discovery.template.Filter">
         <displayName>unknown</displayName>
         <operand1>db2tcpcm</operand1>
         <operator>contains</operator>
         <part>Program Name</part>
      </filterList>
      <filterList array="2"
      guid="63816C902B0A317F8C3B24C7A1EEBC17"
      lastModified="1177555771471"
      parent="B599AED918F436C99FDA0E8EDA578F02"
      xsi:type="coll:com.collation.platform.model.discovery.template.Filter">
         <displayName>unknown</displayName>
         <operand1>db2agent</operand1>
         <operator>contains</operator>
         <part>Program Name</part>
      </filterList>
      <boolExp>1</boolExp>
   </filterSet>
   <index>0</index>
   <order>-10</order>
   <enabled>true</enabled>
   <action>1</action>
   <source>0</source>
   <seedClass>com.collation.discover.seed.app.db.db2.Db2Seed</seedClass>
</Template>
```

Erkennungsebenen
TADDM stellt vier Erkennungsebenen bereit: Erkennung der Ebene 1, Erkennung der Ebene 2, Erkennung der Ebene 3 und Auslastungserkennung.

Erkennung der Ebene 1
Scannen mit TADDM-Sensoren, bei dem Sie Basisinformationen zu den aktiven Computersystemen in der Laufzeitumgebung erkennen können. Dieses Scannen wird auch als Erkennung *ohne Berechtigungsnachweise* bezeichnet, da keine Berechtigungsnachweise erforderlich sind. Dafür wird der Stack-Scan-Sensor und der IBM Tivoli Monitoring Scope-Sensor verwendet.

Die Erkennung der Ebene 1 ist sehr oberflächlich. Sie erfasst lediglich den Hostnamen, das Betriebssystem, die IP-Adresse, den vollständig qualifizierten Domänennamen und die MAC-Adresse (MAC - Media Access Control) jeder erkannten Schnittstelle. Die MAC-Adressenerkennung ist außerdem ausschließlich unter Linux on zSeries- und Windows-Systemen verfügbar.

Bei der Erkennung der Ebene 1 werden keine Teilnetze erkannt. Für alle erkannten IP-Schnittstellen, die nicht einem bereits vorhandenen, während einer Erkennung der Ebene 2 oder 3 erkannten Teilnetz angehören, werden basierend auf dem Wert der Eigenschaft 'com.collation.IpNetworkAssignmentAgent.defaultNetmask' in der Datei `collation.properties` neue Teilnetze erstellt.

Erkennung der Ebene 2
Scannen mit TADDM-Sensoren, bei dem detaillierte Informationen zu den einzelnen Betriebssystemen in der Laufzeitumgebung erkannt werden. Dieses Scannen wird auch als Erkennung *mit Berechtigungsnachweisen* bezeichnet, da Betriebssystem-Berechtigungsnachweise erforderlich sind.

Bei der Erkennung der Ebene 2 werden Anwendungsnamen und die Betriebssystemnamen und Portnummern erfasst, die den einzelnen aktiven Anwendungen zugeordnet sind. Wenn eine Anwendung eine TCP/IP-Verbindung zu einer anderen Anwendung eingerichtet hat, wird diese Information als eine Abhängigkeit erfasst.

Erkennung der Ebene 3
Scannen mit TADDM-Sensoren, bei dem detaillierte Informationen zur Anwendungsinfrastruktur, zu implementierten Softwarekomponenten, physischen Servern, Netzeinheiten, virtuellen Systemen und Hostdaten ermittelt werden, die in der Laufzeitumgebung verwendet werden. Dieses Scannen wird auch als Erkennung *mit Berechtigungsnachweisen* bezeichnet, da sowohl Betriebssystem-Berechtigungsnachweise als auch Anwendungsberechtigungsnachweise erforderlich sind.

Auslastungserkennung
Scannen mit TADDM-Sensoren, bei dem die Auslastungsinformationen für das Hostsystem erfasst werden. Für eine Auslastungserkennung sind Betriebssystem-Berechtigungsnachweise erforderlich.

Erkennungen der Ebenen 2 und 3 erfassen detailliertere Informationen als Erkennungen der Ebene 1. Wenn während einer Erkennung der Ebene 2 oder 3 erstellte Objekte mit zuvor in einer Erkennung der Ebene 1 erstellten Objekten übereinstimmen, werden die in der Erkennung der Ebene 1 erstellten Objekte durch die neu erstellten Objekte ersetzt, wodurch sich wiederum die GUIDs (Globally Unique Identifiers - global eindeutige IDs) für die Objekte ändern. Grundsätzlich sollten daher Daten der Ebene 1 nicht für den kombinierten Einsatz mit anderen Produkten verwendet werden.

Erkennungsprofile
Um eine Erkennung auszuführen, müssen Sie ein Erkennungsprofil angeben, das eine Reihe von Erkennungsoptionen definiert. Mithilfe von Erkennungsprofilen können Sie einzelne Sensoren konfigurieren, mehrere Konfigurationen desselben Sensors verwalten, die passende Konfiguration auf Grundlage einer Reihe von Kriterien auswählen und mehrere Konfigurationen unterschiedlicher Sensoren verwalten, die auf eine einzelne Erkennungsausführung angewendet werden sollen.

Durch Auswahl des entsprechenden Erkennungsprofils können Sie die Erkennungstiefe oder die Erkennungsebene bestimmen.

Standardmäßig stellt TADDM vier Erkennungsprofile bereit. Drei Profile sind für die drei Erkennungsebenen bestimmt, die Sie auswählen können (Ebene 1, Ebene 2 oder Ebene 3), je nachdem, ob Sie eine Erkennung ohne Berechtigungsnachweise oder mit Berechtigungsnachweisen ausführen möchten. Das verbleibende Profil ist für eine Auslastungserkennung bestimmt.

Wenn kein Profil angegeben wird, wird standardmäßig das Erkennungsprofil der Ebene 3 verwendet. Sie können das Standardprofil jedoch in der Discovery Management Console ändern.

Weitere Informationen zu Erkennungsprofilen finden Sie unter *A Flexible Approach to Discovery* auf dem TADDM-Wiki: https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/Tivoli%20Application%20Dependency%20Discovery%20Manager/page/A%20Flexible%20Approach%20to%20Discovery.

Sensoren aktivieren und inaktivieren

Sie können einen Sensor auch dann global inaktivieren, wenn er von einem Profil aktiviert wurde. Sie können auch einen Sensor global aktivieren und ermöglichen, dass die Einstellung im Profil funktioniert.

Wenn beispielsweise ein Sensor global aktiviert ist und im Profil aktiviert ist, wird der Sensor ausgeführt. Wenn der Sensor global aktiviert ist, im Profil jedoch inaktiviert ist, wird er nicht ausgeführt, wenn das entsprechende Profil für die Ausführung einer Erkennung ausgewählt wurde.

Damit die globale Aktivierung und Inaktivierung bei Sensoren funktioniert, die ein `osgi`-Verzeichnis (`$COLLATION_HOME/osgi/plugins`) haben, müssen Sie **`AgentConfigurations`** im Verzeichnis `osgi` ändern.

Suchen Sie beispielsweise beim Db2Sensor die folgenden Verzeichnisse:

- `$COLLATION_HOME/osgi/plugins/com.ibm.cdb.discover.sensor.app.db.db2_`*`x.x.x`*`/Db2Sensor.xml`
- `$COLLATION_HOME/osgi/plugins/com.ibm.cdb.discover.sensor.app.db.db2windows_`*`x.x.x`*`/Db2WindowsSensor.xml`

wobei *x.x.x* die Sensor-Plug-in-Version ist, zum Beispiel `7.3`.

Bei der Bearbeitung der XML-Dateien müssen Sie 'enabled' auf `true` setzen, damit der Sensor aktiviert wird. Zur Inaktivierung des Sensors müssen Sie 'enabled' auf `false` setzen.

Bei Sensoren, die das Verzeichnis `osgi/plugins` nicht verwenden, werden die Konfigurationsdaten in der XML-Datei mit der Sensorkonfiguration gespeichert, die sich im Verzeichnis `etc/discover-sensors` befindet.

Asynchrone und scriptbasierte Erkennung

Bei einer asynchronen und scriptbasierten Erkennung führen Sensoren keine Einzelbefehle aus, sondern stellen ein Erkennungsscript bereit, das für das Zielsystem ausgeführt wird.

Nicht alle Sensoren unterstützen die asynchrone und die scriptbasierte Erkennung. Nur Sensoren, die ein Erkennungsscript bereitstellen, können diese Erkennungstypen unterstützen.

Informationen zu den Sensoren, die eine asynchrone und scriptbasierte Erkennung unterstützen, finden Sie im Abschnitt *Sensoren, die scriptbasierte und asynchrone Erkennung unterstützen* in der TADDM-*Sensorreferenz*.

Einige Unterschiede zu einer nicht-scriptbasierten Erkennung

Die asynchrone Erkennung und die scriptbasierte Erkennung unterscheiden sich in den folgenden wichtigen Punkten von einer nicht-scriptbasierten Erkennung:

- Im Vergleich zu den Erkennungsergebnissen einer nicht-scriptbasierten Erkennung der Ebene 2 oder der Ebene 3 sind die Erkennungsergebnisse einer asynchronen oder einer scriptbasierten Erkennung möglicherweise nicht vollständig. Die meisten Sensoren erkennen bei einer nicht-scriptbasierten Erkennung eine größere Anzahl an Modellobjekten, Attributen und Beziehungen als bei einer asynchronen oder einer scriptbasierten Erkennung.
- Bei der asynchronen oder der scriptbasierten Erkennung werden Anwendungssensoren für ein bestimmtes Zielsystem nur einmal gestartet. Wenn die Anwendung jedoch an mehreren Ports empfangsbereit ist, wird jede Anwendungsinstanz erkannt.
  Bei einer nicht-scriptbasierten Erkennung wird für jede Anwendungsinstanz ein Anwendungssensor gestartet.

Asynchrone Erkennung

Sie können die asynchrone Erkennung ausführen, um Systeme zu erkennen, auf die vom TADDM-Server nicht direkt zugegriffen werden kann. Dazu gehören Systeme mit Standortschutz (z. B. Systeme, die nicht über das Netz zugänglich sind), Systeme ohne Secure Shell (SSH) und Systeme mit schutzwürdigen Informationen, für die keine Berechtigungsnachweise erhältlich sind.

Bei der asynchronen Erkennung führen Benutzer ein Erkennungsscript auf einem Zielsystem aus. Das Erkennungsscript enthält ein Hauptscript und mehrere Sensorscripts. Jedes Sensorscript stellt eine Erkennungsfunktion bereit, die einer Funktion ähnlich ist, die der Sensor ausführt, wenn er bei einer Standarderkennung ausgeführt wird.

Die Ausgabe des Erkennungsscripts wird in Form einer Archivdatei erstellt, die das Ergebnis der Erkennung enthält. Diese Datei muss auf den TADDM-Server kopiert werden. Bei einer TADDM-Erkennung verarbeiten die TADDM-Sensoren die Erkennungsergebnisse aus dieser Archivdatei (anstatt Befehle auszuführen).

Da diese Erkennung manuell und getrennt von einer Standarderkennung, für die Berechtigungsnachweise erforderlich sind, ausgeführt wird, wird sie als "asynchron" bezeichnet.

Zur Ausführung einer asynchronen Erkennung ist der Sensor für asynchrone Erkennung erforderlich. Weitere Informationen finden Sie in der TADDM-*Sensorreferenz*.

Informationen zur Konfiguration von Sensoren für die Ausführung asynchroner Erkennungen finden Sie im Abschnitt „Umgebung für eine asynchrone Erkennung konfigurieren” auf Seite 100.

Scriptbasierte Erkennung

Bei einer scriptbasierten Erkennung können Sie ein Erkennungsscript in einer Standarderkennung verwenden, für die Berechtigungsnachweise erforderlich sind. Bei diesem Erkennungstyp werden dieselben Sensorscripts verwendet wie bei der asynchronen Erkennung.

Bei einer scriptbasierten Erkennung führt ein Sensor keine einzelnen Befehle aus. Stattdessen wird das Sensorscript auf dem Zielsystem ausgeführt. Anwendungsspezifische Berechtigungsnachweise sind möglicherweise nicht erforderlich.

Um beispielsweise bei einer Standarderkennung die Anwendung IBM WebSphere zu erkennen, müssen Sie einen Zugriffslisteneintrag mit Berechtigungsnachweisen für die Anwendung WebSphere erstellen, wenn die Sicherheitsfunktion aktiviert ist. Bei der scriptbasierten Erkennung ist der WebSphere-Zugriffslisteneintrag jedoch nicht erforderlich. Bei der scriptbasierten Erkennung sind auch keine anwendungsspezifischen Protokolle wie Java Management Extensions (JMX) erforderlich, wodurch die Anwendungserkennung durch IBM Tivoli Monitoring erweitert werden kann.

Informationen zur Konfiguration von Sensoren für die Ausführung scriptbasierter Erkennungen finden Sie im Abschnitt „Konfiguration für die scriptbasierte Erkennung” auf Seite 104.

Gleichzeitige Erkennung

Sie können mehrere Erkennungen gleichzeitig ausführen (*gleichzeitige Erkennung*). Beispiel: Da eine große Erkennung möglicherweise mehrere Stunden dauert, können Sie die Ausführung kleinerer Erkennungen starten, bevor die große Erkennung abgeschlossen ist. Bevor Sie gleichzeitige Erkennungen ausführen, müssen Sie sie ordnungsgemäß konfigurieren.

Für gleichzeitig ausgeführte Erkennungen können sogar unterschiedliche Erkennungsprofile verwendet werden.

Um gleichzeitige Erkennungen zu verwalten, verwenden Sie die Discovery Management Console oder das Script `api.sh`. Weitere Informationen zur Verwendung des Scripts `api.sh` finden Sie im Abschnitt *Command-line interface API* im Handbuch *TADDM SDK Developer's Guide*.

Es können gleichzeitige Erkennungen auf demselben Zielsystem ausgeführt werden. Wenn mindestens zwei Erkennungen teilweise dieselben IP-Adressen überwachen, werden alle Erkennungen unabhängig voneinander ausgeführt.

Wenn es während einer aktiven Erkennung zu einer Kennwortänderung kommt und eine weitere Erkennung gleichzeitig gestartet wird, verwenden die Sensoren in dieser zweiten, gleichzeitig ausgeführten Erkennung sofort die neuen Berechtigungsnachweise, weil angenommen wird, dass diese Sensoren erst nach der Kennwortänderung gestartet wurden.

TADDM unterstützt keine gleichzeitige Erkennung mit einer profilbasierten Zugriffsliste.

Wenn Änderungen an der angepassten Servervorlage vorgenommen werden, während eine Erkennung ausgeführt wird, verwendet eine gestartete gleichzeitige Erkennung weiterhin die Version der angepass-

ten Servervorlage. Die nächste separate und nicht gleichzeitige Erkennung, die gestartet wird, verwendet die neue Version der angepassten Servervorlage.

Angezeigten vollständig qualifizierten Domänennamen bestimmen
Sie können eine bevorzugte Methode zur Bestimmung des vollständig qualifizierten Domänennamens für jedes einzelne erkannte System konfigurieren.

Bei einer Erkennung der Ebene 1 ist der vollständig qualifizierte Domänenname das Ergebnis einer umgekehrten Auflösung der IP-Adresse. Bei dieser Adressauflösung wird die Auflösungsbibliothek verwendet, die vom Betriebssystem bereitgestellt wird, sowie alle Konfigurationen, die dort zur Verfügung gestellt werden. Wenn beispielsweise auf Betriebssystemebene die Datei 'hosts' dem Domain Name System (DNS) vorgezogen wird, werden die Informationen in der Datei 'hosts' zuerst berücksichtigt.

Bei einer Erkennung der Ebene 2 führt TADDM mithilfe der Auflösungsbibliothek, die vom Betriebssystem bereitgestellt wird, eine umgekehrte Adressauflösung aller erkannten IP-Adressen aus. Die Betriebssystemkonfiguration bestimmt auch hier, woher die Informationen für die umgekehrte Adressauflösung kommen. Wenn das DNS nicht konfiguriert ist oder unerwünschte vollständig qualifizierte Domänennamen zurückgibt, können Sie es mithilfe der Datei 'hosts' außer Kraft setzen.

Nachdem die erkannten IP-Adressen aufgelöst wurden, wird versucht, einen vollständig qualifizierten Domänennamen mit dem Computersystem abzugleichen. Es gibt eine Reihe verschiedener Möglichkeiten, um einen vollständig qualifizierten Domänennamen abzurufen, und die einzelnen Methoden werden in einer vordefinierten Reihenfolge ausprobiert, bis ein gültiger, vollständig qualifizierter Domänenname gefunden wird. Sie können die Reihenfolge ändern, sodass Ihre bevorzugte Methode eine höhere Priorität hat. Die folgenden Methoden stehen zur Verfügung:

Methode 1
TADDM wählt den vollständig qualifizierten Domänennamen einer IP-Schnittstelle aus, bei dem der Abschnitt für den Host mit dem Hostnamen des erkannten Systems übereinstimmt. Wenn es mehrere Übereinstimmungen gibt, hängt die Auswahl des vollständig qualifizierten Domänennamens von der Priorität des Domänennamens ab, die in der Eigenschaft `com.collation.platform.os.FqdnPriorities` definiert ist. Diese Eigenschaft listet die Domänennamen nach Priorität auf. Um für die Domänen Prioritäten zu vergeben, geben Sie die Namen der Domänen als eine durch Kommas getrennte Liste in einer Zeile ein:

```
com.collation.platform.os.FqdnPriorities=domain1.company.com,
domain2.company.com,domain3.company.com
```

Der vollständig qualifizierte Domänenname mit der höchsten Priorität für seine Domäne wird als vollständig qualifizierter Domänenname zurückgegeben. Bei dieser Methode werden Informationen verwendet, die zu vollständig qualifizierten Domänennamen von Schnittstellen und Computersystemnamen ermittelt werden.

Wenn die Prioritäten nicht definiert sind, geht TADDM alle IP-Schnittstellen durch. TADDM überprüft, ob der vollständig qualifizierte Domänenname, der einer bestimmten IP-Schnittstelle zugeordnet ist, mit dem Namen des Computersystems identisch ist oder ob der Abschnitt für den Hostnamen mit dem Namen des Computersystems identisch ist. Der vollständig qualifizierte Domänenname, der die Kriterien als Erstes erfüllt, wird als vollständig qualifizierter Domänenname zurückgegeben.

Ein Computersystem namens "myname" verfügt beispielsweise über zwei Schnittstellen mit den folgenden vollständig qualifizierten Domänennamen:

- interface #1 myname.domain1.com
- interface #2 myname.domain2.com

Wenn die Eigenschaft `com.collation.platform.os.FqdnPriorities` nicht definiert ist, wird die erste Übereinstimmung als vollständig qualifizierter Domänenname zurückgegeben. Bei beiden Namen stimmt der Abschnitt für den Host mit dem Hostnamen des erkannten Systems überein, der zurückgegebene, vollständig qualifizierte Domänenname ist jedoch "myname.domain1.com". Um Prioritäten für die Auswahl des Namens zu vergeben, verwenden Sie die Eigenschaft `com.collati-`

on.platform.os.FqdnPriorities. Beispiel: Der Eintrag com.collation.platform.os.FqdnPriorities enthält folgende Informationen:

```
com.collation.platform.os.FqdnPriorities=domain2.com,domain1.com
```

In diesem Fall wird als vollständig qualifizierter Domänenname "myname.domain2.com" zurückgegeben, da dieser Name eine höhere Priorität hat.

Methode 2
Die Eigenschaft com.collation.platform.os.command.fqdn gibt einen externen Befehl auf dem TADDM-Server an, der für die umgekehrte Adressauflösung verwendet wird. Die folgenden Beispiele zeigen die Verwendung dieser Eigenschaft (geben Sie die Eigenschaft in einer Zeile ein):

```
com.collation.platform.os.command.fqdn=nslookup $1
| grep Name | awk '{print $2}'
com.collation.platform.os.command.fqdn.AIX=nslookup $1
| grep Name | awk '{print $2}'
com.collation.platform.os.command.fqdn.Linux=nslookup $1
| grep Name | awk '{print $2}'
com.collation.platform.os.command.fqdn.SunOS=nslookup $1
| grep Name | awk '{print $2}'
com.collation.platform.os.command.fqdn.Windows=nslookup $1
```

Methode 3
Die Eigenschaft com.collation.platform.os.command.hostOfHostname gibt einen externen Befehl auf dem Zielsystem an, das für die Angabe des vollständig qualifizierten Domänennamens verwendet wird. Diese Eigenschaft kann dem Betriebssystemtyp durch das Anhängen von “.AIX”, “.Linux”,”.SunOS” oder “.Windows“ zugeordnet werden. Das folgende Beispiel zeigt, wie diese Eigenschaft auf einem Linux-System verwendet wird. Geben Sie die Eigenschaft in einer Zeile ein:

```
com.collation.platform.os.command.hostOfHostname.Linux=host `hostname`
| awk {'print $1'}
```

Methode 4
Der vollständig qualifizierte Domänenname der primären Schnittstelle wird verwendet. Die primäre IP-Schnittstelle wird als niedrigster IP-Wert angegeben, wobei die IP-Werte aufsteigend sortiert sind.

Methode 5
Die IP-Adresse der primären Schnittstelle wird verwendet.

Methode 6
Der Name des Computersystems wird verwendet.

Methode 7
Setzen der Sitzungskontext-IP.

Methode 8
Setzen des FQDN für CS als FQDN für die Sitzungs-IP.

Sie können die Reihenfolge definieren, in der diese Methoden ausprobiert werden, indem Sie die Eigenschaft com.collation.platform.os.fqdnSearchOrder festlegen. Der Wert dieser Eigenschaft ist eine durch Kommas getrennte Liste mit den Nummern der Methoden. Der Standardwert ist 1,2,3,4,5,6,7,8. In diesem Fall versucht TADDM zuerst, Methode 1 zu verwenden. Wenn diese keinen gültigen vollständig qualifizierten Domänennamen zurückgibt, wird Methode 2 ausprobiert usw., bis ein gültiger vollständig qualifizierter Domänenname gefunden wird, dann wird der Vorgang gestoppt. Ein gültiger vollständig qualifizierter Domänenname ist ein vollständig qualifizierter Domänenname, der den Regeln entspricht, die in RFC 1035 angegeben sind.

Diese Lösung ist auch auf Computersysteme anwendbar, bei denen die Erkennung mithilfe von SNMP-Sensoren ausgeführt wird. Sie können definieren, welche Lösungen eine höhere Priorität haben und deshalb verwendet werden können, um einen vollständig qualifizierten Domänennamen schneller zu finden.

In allen Fällen ist ein ordnungsgemäß konfiguriertes DNS die bevorzugte Methode zum Festlegen von Hostnamen. Wenn das DNS nicht verwendet werden kann, verwenden Sie die Datei 'hosts'. Die Verwendung des DNS oder der Datei 'hosts' sind die Standardmethoden zur Bereitstellung einer Namensauflösung für IP-Adressen. TADDM bietet Möglichkeiten, um diese Methoden außer Kraft zu setzen, aber da

alle anderen Methoden auf TADDM beschränkt sind, können sich dabei Namen ergeben, die mit Namen in anderen Managementsystemen inkonsistent sind.

Traceerstellung für eine Erkennung
Sie können die Phasen einer Erkennung in einem Traceprotokoll aufzeichnen, vom Beginn der Erkennung bis zur Aktualisierung des Änderungsprotokolls und zum Erstellen der Topologieabhängigkeiten. Jede Phase der Erkennung wird in einer zugeordneten Protokolldatei erfasst.

Phase der Erkennungsausführung und zugehörige Protokolldatei

Nachdem Sie eine Erkennung gestartet haben, wird jeder Erkennung eine eindeutige Kennung (die Ausführungs-ID) zugewiesen. Eine Zeitmarke im Format *YYYY-MM-DD-hh:mm:ss:SSS* ist die eindeutige Kennung für die Ausführung einer Erkennung, z. B. 20110517225225948. Das Element *YYYY-MM-DD* enthält die Datumsangabe im Format Jahr - Monat - Tag. Das Element *hh:mm:ss.sss* enthält die Zeitangabe im 24-Stunden-Format auf die Tausendstelsekunde genau. Im vorherigen Beispiel lautet das Datum 17.05.2011 (2011-05-17) und die Uhrzeit 22:52:25.948. Mithilfe dieser Kennung können Sie separate Protokolldateien für jeden Sensor im Verzeichnis `$COLLATION_HOME/log/sensors` erstellen. Die Zeitmarke wird innerhalb der Protokolldateien verwendet.

Während einer Erkennung überwacht der Prozessflussmanager den Status der Erkennung und den Status der Sensorereignisse. Darüber hinaus verwaltet der Prozessflussmanager die Übergabe von einem Service zum nächsten. Die Prozessflussaktivität wird in der Datei `$COLLATION_HOME/log/services/ProcessFlowManager.log` auf dem Erkennungs- oder Domänenserver gespeichert.

Die folgenden Beispiele zeigen verschiedene Aktivitäten, die vom Prozessflussmanager überwacht werden, und wie diese Informationen in der Protokolldatei gespeichert werden.

Start der Erkennung:

```
– 2011-05-17 22:53:01,643 ProcessFlowManager [RMI TCP Connection(42)-127.0.0.1]
INFO
processflowmgr.ProcessFlowManagerImpl - [ProcessFlowManagerImpl.I.0] startDisco
very()
started discovery with run id 2,011,051,722,525,948
– 2011-05-17 22:53:01,643 ProcessFlowManager [RMI TCP Connection(42)-127.0.0.1]
INFO
processflowmgr.ProcessFlowManagerImpl - [ProcessFlowManagerImpl.I.22] startDisco
very()
 setting the discoveryRun's run id to 2,011,051,722,525,948
– 2011-05-17 22:53:01,973 ProcessFlowManager [RMI TCP Connection(42)-127.0.0.1]
INFO
processflowmgr.ProcessFlowManagerImpl -
Discovery run, 2011051722525948 started with profile Level 2 Discovery
```

Abschluss der Erkennung:

```
– 2011-05-17 22:56:11,689 ProcessFlowManager [RMI TCP Connection(45)-127.0.0.1]
INFO
processflowmgr.ProcessFlowManagerImpl - [ProcessFlowManagerImpl.I.36]
discoveryDone(2,011,051,722,525,948) called by Discovery Manager
```

Erkennungsereignis:

```
– 2011-05-17 22:53:49,901 ProcessFlowManager [RMI TCP Connection(45)-127.0.0.1]
INFO
processflowmgr.ProcessFlowManagerImpl - [ProcessFlowManagerImpl.I.32]
discoveryProgress(2,011,051,722,525,948, Discovered - The CustomAppServerSensor
(JavaServer 9.156.47.175:36750) sensor discovered the following: CustomAppServer
Result,
 JavaServer,9.156.47.175:36750.) called by Discovery Manager
```

Phase des Topologieerstellungsprogramms und zugehörige Protokolldatei
Das Topologieerstellungsprogramm erstellt die Beziehungen und Abhängigkeiten zwischen den erkannten Elementen. Das Topologieerstellungsprogramm führt die Liste der Agenten aus, die in der Datei

$COLLATION_HOME/etc/TopologyBuilderConfigurationDefault.xml aufgeführt sind. Die Topologieagenten werden in festgelegten Intervallen ausgeführt. Ereignisse, die während einer Erkennung oder bei Abschluss einer Erkennung auftreten, können den Start des Topologieerstellungsprogramms jedoch ebenfalls auslösen. Jeder Agent führt eine bestimmte Task aus, z. B. Konsolidieren, Ermitteln von Abhängigkeiten, Erstellen von Abhängigkeitstabellen, und entfernt alte Informationen. Die Protokolle des Topologieerstellungsprogramms werden in den Dateien $COLLATION_HOME/log/services/TopologyBuilder.log und $COLLATION_HOME/log/agents/*.log auf dem Domänenserver, Synchronisationsserver und dem primären Speicherserver gespeichert.

Die folgenden Beispiele zeigen verschiedene Stufen beim Aufbau von Beziehungen und wie diese Informationen in der Protokolldatei gespeichert werden.

Start des Topologieerstellungsprogramms:

```
– 2011-05-17 22:56:11,717 TopologyBuilder [RMI TCP Connection(158)-127.0.0.1]
INFO cdb.TivoliStdMsgLogger
- CTJOT0400I Topology builder is starting.
```

Abschluss des Topologieerstellungsprogramms:

```
– 2011-05-17 23:16:39,429 TopologyBuilder [TopologyBuilderEngineThread$Dependen
cy@0.5]
 INFO engine.TopologyBuilderEngine - Topology agent completed :
 all normally in seconds 30.367
```

Übergang zum nächsten Topologieagenten:

```
– 2011-05-17 23:16:29,774 TopologyBuilder [TopologyBuilderEngineThread$Dependen
cy@0.5]
INFO cdb.TivoliStdMsgLogger - CTJOT0403I Topology builder agent class
com.ibm.cdb.topomgr.topobuilder.agents.ComputerSystemConsolidationAgent is stopp
ing.
– 2011-05-17 23:16:30,078 TopologyBuilder [TopologyBuilderEngineThread$Dependen
cy@0.5]
INFO cdb.TivoliStdMsgLogger - CTJOT0402I Topology builder agent class
com.ibm.cdb.topomgr.topobuilder.agents.ComputerSystemTypeAgent is starting.
```

Wenn Probleme auftreten, z. B. wenn das Topologieerstellungsprogramm blockiert, suchen Sie in der Protokolldatei nach dem zuletzt gestarteten Topologieagenten, um das Problem zu lösen. Wenn die Datei TopologyBuilder.log keine Einträge enthält, suchen Sie in den Einträgen der Datei TopologyManager.log nach der Zeitmarke des zuletzt gestarteten Agenten. Wenn Sie wissen, durch welche Agenten die Probleme verusacht werden, können Sie diese auch mithilfe der Datei $COLLATION_HOME/log/agents/agentName.log ermitteln.

Sonstige Services und Protokolldateien

Der Änderungsmanager verarbeitet Ereignisse und aktualisiert die Datensätze des Änderungsprotokolls. Diese Verarbeitung läuft unabhängig von der Erkennungsphase ab. Die zu verarbeitenden Ereignisse werden von anderen Services empfangen, z. B. von dem Prozess des Topologieerstellungsprogramms und dem Massenladeprogramm. Wenn Sie eine Topologieansicht öffnen, erstellt der Ansichtsmanager die Strukturen, die erforderlich sind, damit die Topologie in der grafischen Benutzerschnittstelle effizient dargestellt werden kann. Die Serviceprotokolle werden im Verzeichnis $COLLATION_HOME/log/services gespeichert. Jede Serviceprotokolldatei hat denselben Namen wie der zugehörige Service, z. B. services/ChangeManager.log.

Die folgenden Beispiele zeigen, wie diese Informationen in den Protokolldateien gespeichert werden.

Änderungsmanager (ChangeManager):

```
2011-05-19 13:22:42,342 ChangeManager [ChgWork-1] INFO changemgr.
ChangeManagerPersisterImpl -
[ChangeManagerPersister.I.3] Got a create or delete event
```

Ansichtsmanager (ViewManager):

```
2011-05-19 16:37:22,428 ViewManager [RMI TCP Connection(174)-127.0.0.1]
INFO viewmgr.ViewMetaLoader - [ViewMetaLoader.I.31] getViewMeta()
found view meta definition for view Business Application Topology
```

Caching der letzten erfolgreichen Berechtigungsnachweise
TADDM kann die letzten gültigen Zugriffsberechtigungsnachweise zwischenspeichern. Diese können in der nächsten Erkennung (Ebene 2 bzw. scriptbasierte Erkennung) wiederverwendet werden.

Bei der Ersterkennung eines Ziels durchläuft der TADDM-Server die Zugriffsliste und validiert jedes Element anhand des Erkennungsziels. Wenn die gültigen Berechtigungsnachweise gefunden werden, werden diese in einem Cache gespeichert und bei fortlaufenden Erkennungen desselben Verzeichnisziels wiederverwendet.

Ein Cache kann die folgenden beiden Werte speichern:

Berechtigungsnachweise
Dieser Wert wird in einem Cache gespeichert, wenn während der Erkennung die gültigen Berechtigungsnachweise für ein Erkennungsziel gefunden werden. Bei der nächsten Erkennung werden diese aus dem Cache ausgelesen und es wird geprüft, ob sie nach wie vor gültig sind. Falls sie immer noch gültig sind, werden sie für die Erkennung verwendet. Wenn sie nicht mehr gültig sind und der Rückgriff inaktiviert ist, wird die Information, dass der letzte Versuch fehlgeschlagen ist, auf dem Server gespeichert und die Erkennung wird gestoppt. Wenn der Rückgriff aktiviert ist, durchläuft der Server die Zugriffsliste und versucht, neue gültige Berechtigungsnachweise zu finden. Wenn Sie den Rückgriff aktivieren möchten, setzen Sie die Eigenschaft `com.ibm.cdb.security.auth.cache.fallback.failed` auf `true`.

Information, dass der letzte Versuch fehlgeschlagen ist (gemeinsam mit dem letzten Fehler)
Dieser Wert wird in einem Cache gespeichert, wenn während der Erkennung die gültigen Berechtigungsnachweise für ein Erkennungsziel nicht gefunden werden. Falls der Rückgriff inaktiviert ist, wird die Information, dass der letzte Versuch fehlgeschlagen ist, angezeigt und die Erkennung wird gestoppt. Wenn der Rückgriff aktiviert ist, durchläuft der Server die Zugriffsliste und versucht, neue gültige Berechtigungsnachweise zu finden. Wenn Sie den Rückgriff aktivieren möchten, setzen Sie die Eigenschaft `com.ibm.cdb.security.auth.cache.fallback.invalid` auf `true`.

Standardmäßig ist der Rückgriff in beiden Fällen aktiviert. Sie können das Rückgriffsverhalten und das Caching von Berechtigungsnachweisen anpassen, indem Sie die Caching-Eigenschaften für Zugriffsberechtigungsnachweise entsprechend festlegen.

Anmerkung: Berechtigungsnachweise werden nach der IP-Adresse, dem Positionstag, dem Berechtigungsnachweistyp und dem bei der Verbindung verwendeten Protokoll zwischengespeichert. Wenn ein Zugriffseintrag entfernt wird, werden alle zugehörigen Cacheeinträge ebenfalls entfernt. Der Cache für Berechtigungsnachweise kann mit dem neuen Dienstprogramm 'cachemgr' verwaltet werden.

Beschränkungen

- Das Caching von Berechtigungsnachweisen wird bei Erkennungen der Ebene 3 nicht verwendet. Es wird nur für Computersystemerkennungen der Ebene 2 und scriptbasierte Sensoren verwendet.
- Ein Cache überwacht keine Änderungen an der Bereichszugriffsbeschränkung. Fällt ein Erkennungsziel beispielsweise unter eine Bereichszugriffsbeschränkung und wird erkannt, zwischengespeichert und anschließend aus der bereichsorientierten Beschränkung entfernt, wird weiterhin der zwischengespeicherte Wert verwendet.
- Der zwischengespeicherte Wert hat Vorrang vor der Zugriffsliste im Profil. Wenn Sie beispielsweise eine Erkennung über die Hauptzugriffsliste ausführen und gültige Berechtigungsnachweise gespeichert sind, wird selbst dann weiterhin der zwischengespeicherte Wert verwendet, wenn Sie andere Berechtigungsnachweise in einem Profil angeben.

Sie können einen zwischengespeicherten Wert mit dem Dienstprogramm 'cachemgr' entfernen. Falls Sie häufig verschiedene Profile mit unterschiedlichen Zugriffseinträgen für dasselbe Erkennungsziel oder

denselben Bereich verwenden, können Sie für diese das Caching inaktivieren. Andernfalls werden in der Erkennung möglicherweise falsche Berechtigungsnachweise verwendet.

Übersicht über den Topologieerstellungsprozess

TADDM führt den Topologieerstellungsprozess in regelmäßigen Abständen aus. Bis der Topologieerstellungsprozess nach einer Erkennung oder nach einer Operation des Dienstprogramms zum Laden von Massendaten abgeschlossen ist, können in der TADDM-Datenbank nicht abgeglichene Objekte vorhanden sein und die Topologiebeziehungen können unvollständig sein.

Dieser Prozess läuft immer gleich ab, unabhängig davon, welchen TADDM-Implementierungstyp Sie verwenden.

Die Topologieerstellung umfasst die folgenden Operationen:

Bereinigen der TADDM-Datenbank
Bei dem Prozess werden alte Entitäten gelöscht, Abhängigkeiten mit fehlenden Quellen oder Zielen entfernt und andere Elemente entfernt, die ersetzt werden.

Erstellen von Abhängigkeiten zwischen Konfigurationselementen
Bei dem Prozess werden Abhängigkeiten zwischen miteinander kommunizierenden Prozessen erstellt, z. B. zwischen einer Anwendung und der zugrunde liegenden Datenbank, und zwischen sendenden und empfangenden WebSphere MQ-Warteschlangen. Außerdem werden Abhängigkeiten zwischen Teilen eines Anwendungsclusters oder einfach zwischen zwei Computersystemen erstellt.

Erstellen und Erweitern von Konfigurationselementen
Bei dem Prozess werden Informationen von vorhandenen Konfigurationselementen und Verbindungen zur synthetischen Erstellung neuer Konfigurationselemente verwendet. TADDM kann beispielsweise ein neues Konfigurationselement mit dem Namen "ApplicationServerClusters" auf Basis von Informationen erstellen, die aus früheren Erkennungen und Operationen des Dienstprogramms zum Laden von Massendaten stammen.

Erstellen von Informationen für Topologieansichten
Bei dem Prozess werden Informationen, die vom Datenmanagementportal zur schnelleren Anzeige von Topologieansichten verwendet werden können, generiert und gespeichert.

Daten exportieren
In diesem Prozess wird die TADDM-Datenbank abgefragt, um Informationen zum Konfigurationselement in externe Systeme zu exportieren. So wird beispielsweise die Integration in Registry Services als Topologieagent implementiert.

Protokolldateien und Protokollierung

Die Protokolldateien von TADDM und die Einrichtung der Protokollierung für die Fehlerbehebung werden im *Handbuch zur Fehlerbehebung* von TADDM beschrieben.

Umgebung sichern

In sicheren Umgebungen erzwingt TADDM die Authentifizierung, um vertrauliche Daten zu schützen.

Benutzeraccounts können über das Datenmanagementportal konfiguriert werden. Jeder Benutzer muss über einen gültigen Benutzeraccount verfügen, um über das Datenmanagementportal auf erkannte Informationen zu Netz- und Infrastrukturkomponenten zugreifen zu können.

Wird bei der Anmeldung an der Discovery Management Console die Option **Sichere Sitzung (SSL-Sitzung) aufbauen** ausgewählt, werden alle Daten (einschließlich der Benutzernamen und Kennwörter) vor einer Übertragung über das Netz verschlüsselt.

Anmerkung: Fix Pack 5 Wenn das Kontrollkästchen **Sichere Sitzung (SSL-Sitzung) aufbauen** beim Start der Discovery Management Console ausgewählt ist, muss der TADDM-Server im sicheren Modus ausgeführt werden. Dazu sollte die Eigenschaft `com.ibm.cdb.secure.server` auf 'true' gesetzt sein.

Während des Erkennungsprozesses verwendet der TADDM-Server das SSH-Protokoll (Secure Shell), damit eine sichere Kommunikation mit allen Computerhosts und anderen Einheiten möglich ist, die SSH unterstützen.

Der Server unterstützt die Authentifizierung sowohl durch SSH-Verschlüsselung als auch über Kennwörter bei der Anmeldung. Bei der Anmeldung wird die SSH-Authentifizierung über ein Kennwort verwendet; die Benutzernamen und Kennwörter, die Sie in der Zugriffsliste festlegen, werden für die Anmeldung an den Computerhosts verwendet, für die eine Erkennung durchgeführt werden soll.

Siehe auch „Eigenschaften für die Sicherheit” auf Seite 89.

Benutzerzugriff auf Konfigurationselemente steuern

TADDM steuert den Benutzerzugriff auf Konfigurationselemente durch Zugriffsobjektgruppen, Rollen und Berechtigungen.

Zugriffsrechte auf Konfigurationselemente werden wie folgt vergeben:

1. Konfigurationselemente werden in Zugriffsobjektgruppen zusammengefasst.
2. Rollen werden definiert, die Berechtigungsgruppen zusammenfassen.
3. Benutzer und Benutzergruppen werden definiert und jedem Benutzer/jeder Benutzergruppe werden Rollen zur Erteilung bestimmter Berechtigungen (für bestimmte Zugriffsobjektgruppen) zugewiesen.

Im Zusammenhang mit der Sicherheit in TADDM ist ein Benutzer eine Person, die Zugriff auf Konfigurationselemente hat, während eine Benutzergruppe aus mehreren Benutzern mit derselben Rolle beziehungsweise denselben Berechtigungen besteht.

Benutzer und Benutzergruppen werden im Datenmanagementportal erstellt. Der Zugriff von Benutzern und Benutzergruppen auf Konfigurationselemente wird durch die Rollen und Zugriffsobjektgruppen definiert, die Sie den Benutzern und Benutzergruppen zuweisen. Sie können diese Zuweisungen jederzeit ändern.

Berechtigungen
Eine Berechtigung gestattet einem Benutzer, eine Aktion auszuführen oder auf ein bestimmtes Konfigurationselement zuzugreifen. Berechtigungen sind in Rollen aufgeteilt und die Benutzer erhalten ihre jeweiligen Berechtigungen, indem ihnen Rollen mit den entsprechenden Berechtigungen zugewiesen werden.

TADDM stellt vier Berechtigungen bereit, die jeweils als Berechtigungen auf Datenebene oder auf Methodenebene klassifiziert werden.

Berechtigungen auf Datenebene

Die Berechtigungen 'Lesen' und 'Aktualisieren' gelten auf Datenebene.

Lesen
Der Benutzer kann sich Informationen zu einem Konfigurationselement anzeigen lassen.

Aktualisieren
Der Benutzer kann die Informationen zu einem Konfigurationselement ändern.

Berechtigungen auf Methodenebene

Die Berechtigungen 'Ermitteln' und 'Admin' gelten auf Methodenebene.

Ermitteln
Der Benutzer kann eine Erkennung starten, Erkennungsbereichsobjekte erstellen und aktualisieren oder zum Beispiel über das Bearbeitungsmenü (Edit) der Discovery Management Console neue Objekte erstellen.

Ein Benutzer ohne Erkennungsberechtigung kann sich nicht bei der Discovery Management Console anmelden und auch die Registerkarte 'Erkennung' im Datenmanagementportal nicht anzeigen.

Admin
Der Benutzer kann Benutzer, Rollen und Berechtigungen erstellen oder aktualisieren. Außerdem kann der Benutzer mit dem Autorisierungsmanager eine Berechtigungsrichtlinie konfigurieren.

Sicherheit auf Datenebene aktivieren

Sie können die Sicherheit auf Datenebene für AIX-, Linux-, Linux on System z- und Windows-Betriebssysteme aktivieren, indem Sie die Datei `collation.properties` entsprechend bearbeiten.

Zur Aktivierung der Sicherheit auf Datenebene (damit Sie selektiv Lese- und Aktualisierungsberechtigungen erteilen können) müssen Sie folgende Schritte ausführen:

1. Suchen Sie in der Datei `collation.properties` nach folgender Zeile und ändern Sie dort den Wert der Eigenschaft von `false` in `true`:

   ```
   com.collation.security.enabledatalevelsecurity=false
   ```

2. Speichern Sie die Datei.
3. Stoppen Sie den TADDM-Server.
4. Starten Sie den TADDM-Server erneut.

Anmerkung: In einer Streaming-Server-Implementierung müssen Sie die Datei `collation.properties` auf jedem Speicherserver aktualisieren und alle Speicherserver erneut starten.

Über die Erstellung von Zugriffsobjektgruppen können Sie Berechtigungen gezielter setzen. Bei Aktivierung der Sicherheit auf Datenebene können die wichtigen TADDM-Ressourcen über Zugriffsobjektgruppen gesichert werden. Falls die Sicherheit auf Datenebene aktiviert ist, können die Benutzer nur die KEs (Konfigurationselemente) in den Zugriffsobjektgruppen ändern, für die sie über Aktualisierungsberechtigung verfügen.

Sekundäre Ressourcen, wie Ressourcen für die physische Region einschließlich des Attributs `SiteInfo`, werden beim Erstellen einer Zugriffsobjektgruppe nicht angezeigt.

Rollen

Als Rolle wird eine Reihe von Berechtigungen bezeichnet, die einem Benutzer zugewiesen werden kann. Beim Zuweisen einer Rolle werden bestimmte Zugriffsfunktionen gewährt.

Wenn Sie einem Benutzer eine Rolle zuweisen, müssen Sie mindestens eine Zugriffsobjektgruppe für diese Rolle angeben. Sie können also den Bereich der Rolle auf die Zugriffsobjektgruppen begrenzen, die für den Benutzer geeignet sind.

Beispiel: Da Sarah für die NT-Server und Workstations Ihres Unternehmens verantwortlich ist, weisen Sie ihr die Supervisorrolle für eine Zugriffsobjektgruppe zu, die diese Systeme enthält. Da Werner für die Linux-Systeme zuständig ist, weisen Sie ihm die Supervisorrolle für eine Zugriffsobjektgruppe zu, die diese Systeme enthält. Obwohl Sarah und Werner nun dieselbe Rolle haben (da beide dieselben Operationen ausführen), haben sie dennoch Zugriff auf unterschiedliche Ressourcen.

Anmerkung: Wenn Sie einen Synchronisationsserver verwenden, müssen alle Rollen für jede TADDM-Domäne erstellt und die Domänenserver dann mit dem Synchronisationsserver synchronisiert werden.

Vordefinierte Rollen

TADDM stellt die folgenden vordefinierten Rollen bereit:

Operator
Diese Rolle verfügt über Leseberechtigung.

Supervisor
Diese Rolle verfügt über Lese-, Aktualisierungs- und Erkennungsberechtigungen.

Administrator
Diese Rolle verfügt über Lese-, Aktualisierungs-, Erkennungs- und Administratorberechtigungen.

Weitere Rollen, die erstellt werden können

Sie können weitere Rollen erstellen, wenn Sie noch andere Berechtigungskombinationen zuweisen müssen. Die folgenden Kombinationen könnten besonders nützlich sein:

Lesen + Aktualisieren
Lese- und Aktualisierungsberechtigung für Objekte in zugeordneten Zugriffsobjektgruppen.

Lesen + Aktualisieren + Verwalten
Lese- und Aktualisierungsberechtigung für Objekte in zugeordneten Zugriffsobjektgruppen sowie Berechtigung zur Erstellung von Benutzern, Rollen und Berechtigungen.

Zugriffsobjektgruppen
TADDM verwaltet den Zugriff auf Konfigurationselemente nicht auf Einzelbasis. Stattdessen werden die Konfigurationselemente in Gruppen, so genannten Zugriffsobjektgruppen, zusammengefasst. Bei einer Zugriffsobjektgruppe handelt es sich um mehrere Konfigurationselemente, die aus Sicherheitsgründen kollektiv verwaltet werden.

Die Sicherheit jeder Zugriffsobjektgruppe wird dann mittels der Erstellung von Rollen und dem Zuweisen dieser Rollen an Benutzer garantiert. Die Rolle gilt nur für die Zugriffsobjektgruppen, die Sie beim Zuweisen der Rolle zu einem Benutzer angeben. Mithilfe von Zugriffsobjektgruppen begrenzen Sie also den Bereich einer Rolle.

Bei der Installation von TADDM wird die Zugriffsobjektgruppe mit dem Namen `DefaultAccessCollection` erstellt, die alle Konfigurationselemente enthält. Alle Benutzer verfügen standardmäßig über Lese- und Aktualisierungsberechtigungen für diese Zugriffsobjektgruppe, sofern keine Sicherheit auf Datenebene aktiviert ist.

Anmerkung: Benutzer verfügen nicht über die Berechtigung zum Lesen und Aktualisieren von Zugriffsobjektgruppen. Sie können nur einzelne Konfigurationselemente lesen und aktualisieren. Lese- und Aktualisierungszugriff haben die Benutzer hingegen auf Zugriffsobjektgruppen, die Mitglieder der ihnen zugewiesenen Zugriffsobjektgruppen sind.

Sicherheitsrichtlinien zurücksetzen
Wenn die Sicherheitsrichtlinien (Berechtigungen, Rollen und Zugriffsobjektgruppen) auf ihre Standardeinstellungen zurückgesetzt werden sollen, müssen Sie dazu zwei Dateien austauschen. Beim Zurücksetzen von Sicherheitsrichtlinien müssen Sie jedoch alle Benutzer löschen und neu erstellen.

Informationen zu diesem Vorgang

Die Sicherheitsrichtlinien sind im Verzeichnis `$COLLATION_HOME/var/policy` in den folgenden beiden Dateien gespeichert, die auch zum Initialisieren der Sicherheitsrichtlinien verwendet werden:

- `AuthorizationPolicy.xml`
- `AuthorizationRoles.xml`

Nach der Initialisierung der Sicherheitsrichtlinien werden diese Dateien umbenannt und in demselben Verzeichnis gespeichert. Beispielsweise wurden folgende Dateien umbenannt:

- `AuthorizationPolicy.backup.xml`
- `AuthorizationRoles.backup.xml`

Die Standardversionen der Dateien mit den bereitgestellten Sicherheitsrichtlinien befinden sich ebenfalls in demselben Verzeichnis. Bei den folgenden Dateien handelt es sich um die Standardversionen:

- `DefaultPolicy.xml`
- `DefaultRoles.xml`

Vorgehensweise

So stellen Sie die Standardsicherheitsrichtlinien wieder her:

1. Um die aktuellen Richtliniendateien zu speichern, benennen Sie sie um oder verschieben Sie sie in ein anderes Verzeichnis.
2. Löschen Sie alle von Ihnen erstellten Benutzer.
3. Löschen Sie das Verzeichnis `$COLLATION_HOME/var/ibmsecauthz`.

4. Erstellen Sie eine Kopie der Datei `DefaultPolicy.xml` unter dem Namen `AuthorizationPolicy.xml`.
5. Erstellen Sie eine Kopie der Datei `DefaultRoles.xml` unter dem Namen `AuthorizationRoles.xml`.
6. Starten Sie den Server erneut.
7. Erstellen Sie Benutzer nach Bedarf.

Lockouts

Mithilfe von Lockouts können Sie einen einzelnen Benutzer oder alle Benutzer aus TADDM aussperren, wenn die konfigurierte Anzahl der zulässigen fehlgeschlagenen Anmeldeversuche überschritten wird. Durch den Einsatz der Lockoutfunktion ist eine bessere Authentifizierungssteuerung möglich und diese Funktion unterstützt Sie bei der Abwehr von Brute-Force-Attacken zur Kennwortentschlüsselung.

Ein lokales Lockout wird ausgelöst, wenn ein einzelner Benutzer die konfigurierte Anzahl fehlgeschlagener Anmeldeversuche überschreitet. Der Benutzer kann sich daraufhin für einen konfigurierten Zeitraum nicht mehr bei TADDM anmelden.

Wenn ein globales Lockout ausgelöst wird, können sich für einen konfigurierten Zeitraum überhaupt keine Benutzer bei TADDM anmelden. Ein globales Lockout wird durch eine der folgenden beiden Situationen ausgelöst:

- Die Anzahl aktiver Lockouts für verschiedene Benutzer überschreitet die konfigurierte Anzahl der maximal zulässigen globalen Lockouts.
- Die Anzahl fehlgeschlagener Anmeldeversuche für eindeutige Benutzernamen überschreitet den konfigurierten Grenzwert.

Bestehende Sitzungen sind von einem ausgelösten Lockout nicht betroffen.

Durch die Konfiguration der entsprechenden Eigenschaften in der Datei `collation.properties` können Sie die Anzahl der zulässigen fehlgeschlagenen Anmeldeversuche und den Zeitraum angeben, in dem ein Lockout aktiv bleibt. Weitere Informationen zu diesen Eigenschaften finden Sie im Abschnitt „Lockouteigenschaften” auf Seite 85.

Wenn die Zeitspanne für ein globales Lockout abgelaufen ist, werden alle derzeit laufenden lokalen Lockouts automatisch gelöscht.

In einer Synchronisationsserverimplementierung steuert der Synchronisationsserver die Sicherheit sämtlicher TADDM-Domänen. Alle Lockouts, die auf dem Domänenserver aktiv waren, bevor er mit dem Synchronisationsserver verbunden wurde, werden gelöscht, wenn die Synchronisierung zwischen dem Domänenserver und dem Synchronisationsserver aktiviert ist.

Die fehlgeschlagenen Anmeldeversuche, mit denen die Gesamtzahl berechnet wird, können auf verschiedene Weisen erfolgt sein - beispielsweise in der API der Befehlszeilenschnittstelle, in der Java-API, mit Tools (Scripts), SOAP, REST, über die Discovery Management Console oder mit dem Datenmanagementportal. Die Lockoutfunktion wird in Integrationen ausgeführt, die die TADDM-API verwenden. Bei Anmeldungen mit Single Sign-on oder bei datenbankbasierten Integrationen (zum Beispiel Tivoli Common Reporting) findet sie jedoch keine Anwendung.

Ein TADDM-Serveradministrator kann ein lokales oder globales Lockout mit dem Script `$COLLATION_HOME/bin/lockmgr.sh` löschen. Sie können das Script über die folgenden Server ausführen:

- Domänenserver in einer Domänenserver-Implementierung
- Synchronisationsserver in einer Synchronisationsserver-Implementierung
- Primärer Speicherserver in einer Streaming-Server-Implementierung

Sie können das Script `lockmgr.sh` mit den folgenden Optionen ausführen:

`lockmgr.sh -s`
Zeigt den Lockoutstatus an.

`lockmgr.sh -g`
Löscht ein aktives globales Lockout.

lockmgr.sh -u *Benutzername*
Löscht ein aktives lokales Lockout für einen bestimmten Benutzer.

lockmgr.sh -h
Zeigt den Hilfetext für das Script `lockmgr.sh` an.

Verschlüsselung

Verschlüsselung ist der Prozess der Datentransformation in eine unverständliche Form, die so erfolgt, dass die ursprünglichen Daten entweder nicht mehr oder nur mithilfe eines Entschlüsselungsprozesses wiederhergestellt werden können.

Fix Pack 5 TADDM verwendet die Eigenschaft '*com.collation.security.algo.aes.keylength*', um den Algorithmus (AES 128 oder AES 256) vom '*FIPS-konformen IBMJCEFIPS*' Sicherheitsprovider zu bestimmen, mit dem die folgenden Elemente verschlüsselt werden:

- Kennwörter, einschließlich der Einträge in den Dateien `collation.properties` und `userdata.xml`
- In der Datenbank gespeicherte Zugriffslisteneinträge

Beispiel:

Diese Eigenschaft definiert die Schlüssellänge für AES `-com.collation.security.algo.aes.keylength=128`.

Bei der ersten Installation von TADDM wird ein Verschlüsselungsschlüssel generiert und die Verschlüsselung von Kennwörtern erfolgt mithilfe dieses neuen Verschlüsselungsschlüssels. Der Verschlüsselungsschlüssel befindet sich standardmäßig in der Datei `etc/TADDMSec.properties`.

Position des TADDM-Verschlüsselungsschlüssels ändern

Ändern Sie in der Datei `collation.properties` den Wert der Eigenschaft `com.collation.security.key`, um die Position des Verschlüsselungsschlüssels zu ändern. Sie können für die Eigenschaft relativ zum Verzeichnis `$COLLATION_HOME` eine andere Position wählen.

Bewahren Sie zur Vermeidung eines eventuellen Datenverlusts eine Sicherungskopie des Verschlüsselungsschlüssels an einem separaten Standort auf. Der Schlüssel kann im Falle eines Problems mit dem ursprünglichen Exemplar wiederhergestellt werden.

TADDM-Verschlüsselungsschlüssel in einer Domänenserverimplementierung ändern

Anmerkung: TADDM unterstützt die Änderung des Verschlüsselungsschlüssels nach der Installation in einer Implementierung mit Streaming- und Synchronisationsserver nicht.

Zur Änderung des TADDM-Verschlüsselungsschlüssels in einer Domänenserverimplementierung müssen Sie das Script `bin/changekey.sh` (oder eine funktional entsprechende Stapelscriptdatei) verwenden. Mithilfe dieses Scripts können verschlüsselte Einträge in den Dateien `collation.properties` und `userdata.xml` sowie in der Datenbank gespeicherte Zugriffslisteneinträge migriert werden. Stellen Sie bei der Verwendung des Scripts `bin/changekey.sh` sicher, dass Sie als der Benutzer ohne Rootberechtigung angemeldet sind, der während der Installation definiert wurde.

Nach der erfolgreichen Verwendung dieses Scripts müssen Sie TADDM erneut starten.

Format für die Ausführung des Scripts

```
./changekey.sh $COLLATION_HOME Benutzer_mit_Administratorberechtigung Administratorkennwort
```

Beispiel

```
./changekey.sh /opt/IBM/taddm/dist administrator taddm
```

FIPS-Kompatibilität

Sie können TADDM für den Betrieb in einem Modus konfigurieren, der FIPS-kompatible Algorithmen zur Verschlüsselung verwendet, indem Sie die FIPSMode-Eigenschaft **com.collation.security.FIPS-Mode** auf `true` setzen.

Setzen Sie die Eigenschaft **com.collation.security.FIPSMode** in den folgenden Dateien auf `true`:

- `$COLLATION_HOME/dist/etc/collation.properties`
- `$COLLATION_HOME/dist/sdk/etc/collation.properties`
- `sdk/etc/collation.properties` jeder TADDM SDK-Installation, die eine Verbindung mit dem FIPS-konformen TADDM herstellt.

Der Standardwert der Eigenschaft **com.collation.security.FIPSMode** ist `false`.

Im FIPS-Modus verwendet TADDM folgende für FIPS 140-2 genehmigte Verschlüsselungsanbieter:

- IBMJCEFIPS (Zertifikat 376)
- IBMJSSEFIPS (Zertifikat 409)

Weitere Informationen zu den Zertifikaten 376 und 409 finden Sie auf der Website des National Institute of Standards and Technology (NIST), http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2004.htm.

Der FIPS-Modus kann mit Ausnahme der folgenden Erkennungen mit allen TADDM-Erkennungstypen verwendet werden:

- SNMP-Erkennung der Ebene 2
- i5/OS-Erkennung der Ebene 2
- ZEnterprise-Erkennung der Ebene 2
- VMware ESXi-Erkennung der Ebene 2
- VMware Virtual Center-Erkennung der Ebene 3
- JBoss-Erkennung der Ebene 3
- Oracle Application Server-Erkennung der Ebene 3
- WebLogic-Erkennung der Ebene 3
- SAP CCMS- und SLD-Erkennung der Ebene 3
- EMC-Erkennung der Ebene 3
- Fix Pack 1 Sybase-Erkennung der Ebene 3
- Erkennungen der Ebene 2 und 3, für die Windows Management Instrumentation (WMI) oder PowerShell-Sitzung (PowerShell-Sitzung wird in TADDM 7.3.0.2 oder höher unterstützt) nur zur Erkennung von Windows-Plattformen verwendet wird, wenn Windows TADDM-Server, Windows-Gateways und Windows-Erkennungsziele nicht im FIPS-konformen Modus ausgeführt werden. Informationen zur Konfiguration von Windows-Servern im FIPS-konformen Modus finden Sie in der Windows-Dokumentation, zum Beispiel unter http://support.microsoft.com/kb/811833.

TADDM-Sensoren, die unter einem FIPS-konformen Modus ausgeführt werden und SSH verwenden, können keine Verbindung zu Servern herstellen, die nur das SSHv1- bzw. das SSHv2-Protokoll mit ihrer zu schwachen Verschlüsselung unterstützen. TADDM kann die FIPS-Konformität der SSH-Implementierung auf den Zielservern nicht überprüfen. Sie müssen selbst überprüfen, ob die SSH-Implementierungen in Ihrer Umgebung FIPS-konform sind.

Wenn Sie im FIPS-Modus das TADDM-SDK und die Discovery Management Console im sicheren Modus verwenden, wird nur IBM Java unterstützt.

Zugehörige Konzepte

„Konformität mit dem Standard SP800-131” auf Seite 21
Sie können TADDM so konfigurieren, dass es den Sicherheitsstandard SP800-131a des National Institute of Standards and Technology (NIST) unterstützt.

Fix Pack 8 Kennwortrichtlinie

Bei der Kennwortrichtlinie handelt es sich um eine Gruppe von Regeln, mit denen die Verwendung von Kennwörtern und deren Verwaltung in TADDM gesteuert wird. Diese Regeln sollen sicherstellen, dass Benutzer ihre Kennwörter regelmäßig ändern und die Kennwörter die syntaktischen Kennwortanforderungen des Unternehmens erfüllen.

In TADDM 7.3.0.8 können Kennwortrichtlinien mit *Regeln für die Kennwortsicherheit* definiert werden, die feststellen, ob ein neues Kennwort gültig ist.

Bei einer *Regel für die Kennwortsicherheit* handelt es sich um eine Regel, der ein Kennwort entsprechen muss, wie beispielsweise die Mindestlänge von Kennwörtern und die Anzahl an unterschiedlichen Zeichen, die zulässig und nicht zulässig sind. Sie können die Standards und Regeln für Kennwörter angeben, z. B.:

Mindestlänge von 15 Zeichen

Es müssen mindestens zwei der folgenden Zeichentypen vorhanden sein:

- Großbuchstaben
- Kleinbuchstaben
- Zahlen
- Sonderzeichen

Wichtig: Die folgenden Eigenschaften für die Kennwortrichtlinie müssen im primären Speicherserver und allen Erkennungsservern in der Datei `collation.properties` konfiguriert werden. Stellen Sie sicher, dass die konfigurierten Werte für die Eigenschaften der Kennwortrichtlinie auf allen Servern identisch sind.

- Zur Aktiviert der Kennwortrichtlinie setzen Sie die folgende Eigenschaft auf '`true`'. Der Standardwert ist '`false`', was bedeutet, dass die Kennwortrichtlinie nicht aktiviert ist.

  ```
  com.collation.passwordpolicy=false
  ```

- Wenn Sie die Mindestlänge der Zeichen für das Kennwort ändern möchten, ändern Sie die folgenden Eigenschaft. Der Standardwert ist 15.

  ```
  com.collation.passwordpolicy.minlength=15
  ```

- Wenn Sie die Mindestanzahl von Zeichentypen im Kennwort aus den Bereichen Großbuchstaben, Kleinbuchstaben, Zahlen oder Sonderzeichen festlegen möchten, ändern Sie die folgende Eigenschaft. Der Standardwert ist 2.

  ```
  com.collation.passwordpolicy.MinCharTypes=2
  ```

Sobald die Kennwortrichtlinie konfiguriert ist, ist das neue Kennwort mit der neuen Kennwortrichtlinie konform. Das Ablaufdatum des Kennworts ist aktuell standardmäßig auf 90 Tage ab dem Datum gesetzt, an dem der Benutzer oder Administrator ein Kennwort geändert hat oder ein neuer Benutzer vom Administrator erstellt wurde.

Anmerkung:

- Diese Kennwortrichtlinie gilt nur für die Authentifizierung des dateibasierten TADDM-Repositorys. Weitere Informationen finden Sie im Abschnitt 'Planung der Sicherheit' im *Installationshandbuch*.
- Diese Kennwortrichtlinie gilt nicht für interne Benutzer (_topomgr, _discmgr , _pfm).
- Ein vorhandenes Kennwort und das Ablaufdatum eines Kennworts werden erst geändert, wenn die Benutzer, ein Administrator oder Bediener ihr Kennwort mindestens einmal ändern, nachdem die neue Kennwortrichtlinie konfiguriert wurde.

Konformität mit dem Standard SP800-131

Sie können TADDM so konfigurieren, dass es den Sicherheitsstandard SP800-131a des National Institute of Standards and Technology (NIST) unterstützt.

Der Sicherheitsstandard SP800-131a erfordert im Vergleich zu anderen Standards (wie FIPS 140-2) längere Schlüssel und eine stärkere Verschlüsselung. Außerdem setzt er das Protokoll Transport Layer Security (TLS) v1.2 voraus. Weitere Informationen finden Sie im Abschnitt http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf.

Zur Aktivierung des SP800-131a-Modus müssen Sie die Eigenschaft `com.ibm.jsse2.sp800-131` in den folgenden Dateien auf `strict` setzen:

- `$COLLATION_HOME/dist/etc/collation.properties`
- `$COLLATION_HOME/dist/sdk/etc/collation.properties`
- `sdk/etc/collation.properties` jeder TADDM SDK-Installation, aus der eine Verbindung mit der SP800-131-konformen TADDM-Instanz hergestellt wird.

Standardmäßig ist die Eigenschaft `com.ibm.jsse2.sp800-131` nicht gesetzt.

Der SP800-131a-konforme Modus wird von den gleichen TADDM-Erkennungstypen unterstützt wie der FIPS-Modus.

Im SP800-131-Modus verwendet TADDM das sicherste SSL-Protokoll (TLS v1.2) der verschlüsselten Kommunikation. Hierzu müssen jedoch folgende Voraussetzungen erfüllt sein.

- Wenn Sie das Datenmanagementportal über den Web-SSL-Port (HTTPS-Port) verwenden wollen, müssen Sie Ihren Web-Browser so einrichten, dass er das TLS v1.2-Protokoll unterstützt.
- Wenn Sie das TADDM-SDK und die Discovery Management Console im sicheren Modus verwenden, müssen Sie das TLS v1.2-Protokoll in Ihrer Java Runtime Environment aktivieren. Beachten Sie auch, dass nur IBM Java unterstützt wird.
- Wenn Ihr SSL-Zertifikat nicht mit dem SP800-131a-Standard konform ist, muss es neu generiert werden. Die hierzu erforderlichen Schritte sind im Abschnitt „Angepasste SSL-Zertifikate zur Verwendung in TADDM installieren” auf Seite 31 beschrieben.

Zugehörige Konzepte

„FIPS-Kompatibilität” auf Seite 20
Sie können TADDM für den Betrieb in einem Modus konfigurieren, der FIPS-kompatible Algorithmen zur Verschlüsselung verwendet, indem Sie die FIPSMode-Eigenschaft **`com.collation.security.FIPSMode`** auf `true` setzen.

Sicherheit für eine Synchronisationsserverimplementierung

Falls Sie eine Synchronisationsserverimplementierung verwenden, müssen bei der Konfiguration des Synchronisationsservers für Ihre Umgebung bestimmte Änderungen an der Sicherheit vorgenommen werden.

Wenn Sie die dateibasierte TADDM-Registry verwenden und eine TADDM-Domäne zu einem Synchronisationsserver hinzugefügt wird, müssen Sie für diesen Synchronisationsserver alle bereits vorhandenen Benutzer einer Domäne einschließlich der zugewiesenen Rollen und der Zugriffsrechte auf Zugriffsobjektgruppen erneut erstellen. Wenn Sie ein Lightweight Directory Access Protocol (LDAP) oder die Benutzerregistry von eingebundenen WebSphere-Repositorys verwenden, müssen Sie zum Synchronisationsserver die Berechtigungen für alle Benutzer mit Zugriff auf TADDM hinzufügen.

Wenn Sie zum Synchronisationsserver eine Domäne hinzufügen, wird die Authentifizierung und Autorisierung für die neue Domäne an den Synchronisationsserver delegiert.

Anmeldungen an der Domäne werden vom Synchronisationsserver verarbeitet. Außerdem werden die Methodenaufrufe des Sicherheitsmanagers durch den Synchronisationsserver verarbeitet.

Die folgende Liste fasst sonstige Sicherheitsinformationen zusammen, die Ihnen für die Konfiguration Ihres Synchronisationsservers bekannt sein müssen:

- Damit TADDM fehlerfrei arbeiten kann, muss das Datenmanagementportal auf dem Synchronisationsserver aktiv sein. Eine TADDM-Domäne delegiert Sicherheitsoperationen an das Datenmanagementportal und diese Delegation wird alle 2,5 Minuten aktualisiert. Wenn die Delegation nach 5 Minuten nicht aktualisiert wurde, delegiert die TADDM-Domäne keine Sicherheitsoperationen mehr und fährt fort, als ob der Synchronisationsserver nicht vorhanden wäre. In diesem Fall müssen die TADDM-Benutzer-

schnittstellen erneut gestartet werden, um die Sitzungen mit dem Synchronisationsserver erneut aufzubauen.

- In allen folgenden Fällen muss die TADDM-Benutzerschnittstelle erneut gestartet werden, um die Sitzungen mit dem richtigen Synchronisationsserver erneut aufzubauen:
  - Die Domäne, in der die Benutzerschnittstelle aktiv ist, wird zu dem Datenmanagementportal, das auf einem Synchronisationsserver ausgeführt wird, hinzugefügt.
  - Die Benutzerschnittstelle wird in einer Domäne geöffnet, während die Domäne mit einem Datenmanagementportal verbunden ist, der Synchronisationsserver ist jedoch später nicht mehr verfügbar (zum Beispiel durch einen Neustart des Synchronisationsservers oder bei Netzproblemen).
- Rollen, Berechtigungen und Zugriffsobjektgruppen, die im TADDM-Server gespeichert sind, werden über die Domäne mit dem Synchronisationsserver synchronisiert. Die Zuordnungen von Benutzern zu Rollen sind nicht synchronisiert.
- Die für die Domäne erstellten Rollen können vom Synchronisationsserver verwendet werden, nachdem diese Objekte zwischen der Domäne und dem Synchronisationsserver synchronisiert wurden.
- Die Benutzer werden nicht mit dem Synchronisationsserver synchronisiert.
- Eine zentrale Benutzerregistry, zum Beispiel LDAP oder eingebundene WebSphere-Repositorys, ist die empfohlene Authentifizierungsmethode für den Synchronisationsserver. Bei einer zentralen Benutzerregistry werden die Kennwörter an einem Ort gespeichert.
- Zugriffsobjektgruppen können sich nicht auf Domänen erstrecken.
- Die Synchronisation erfolgt von der Domäne an den Synchronisationsserver. Objekte, die im Synchronisationsserver erstellt wurden, werden nicht an die Domäne weitergegeben.
- Erstellen und belegen Sie Zugriffsobjektgruppen in der Domäne und synchronisieren Sie diese mit dem Synchronisationsserver.
- Erstellen Sie Rollen in der Domäne und synchronisieren Sie diese mit dem Synchronisationsserver.
- Berechtigen Sie die Benutzer des Synchronisationsservers für den Zugriff auf Zugriffsobjektgruppen von mehreren Domänen aus.

Sicherheit für eine Streaming-Server-Implementierung

Falls Sie eine Streaming-Server-Implementierung verwenden, werden die Authentifizierung und die Autorisierung an den primären Speicherserver delegiert.

Wenn Sie die dateibasierte TADDM-Registry verwenden, müssen Sie TADDM-Benutzer auf dem primären Speicherserver erstellen und autorisieren. Wenn Sie ein Lightweight Directory Access Protocol (LDAP) oder die Benutzerregistry von eingebundenen WebSphere-Repositorys verwenden, müssen Sie TADDM-Benutzer auf dem primären Speicherserver autorisieren. Der bevorzugte Registrytyp für die TADDM-Authentifizierung ist eine zentrale Benutzerregistry, z. B. eine LDAP-Registry oder eine Registry eines eingebundenen WebSphere-Repositorys.

Anmeldungen an den Erkennungsservern und sekundären Speicherservern werden auf dem primären Speicherserver verarbeitet. Deshalb wird die Benutzerauthentifizierung mit dem Benutzerregistry ausgeführt, für das der primäre Speicherserver konfiguriert ist. Außerdem werden die Funktionen des Sicherheitsmanagers durch den primären Speicherserver verarbeitet.

Damit TADDM fehlerfrei arbeiten kann, muss der primäre Speicherserver aktiv sein.

Wenn der primäre Speicherserver gestoppt und erneut gestartet wird, muss eine TADDM-Benutzerschnittstelle erneut gestartet werden, um die Sitzungen mit dem primären Speicherserver erneut aufzubauen.

Konfiguration für LDAP

Sie können einen externen LDAP-Server für die Benutzerauthentifizierung konfigurieren.

Vorbereitende Schritte
Konfigurieren Sie für die Authentifizierung in einem LDAP-Benutzerregister ein LDAP V2- oder V3-Register.

Informationen zu diesem Vorgang

Bei Verwendung von LDAP und/oder VMM werden die LDAP-Benutzer und/oder -Gruppen immer in LDAP/VMM gespeichert und müssen nicht in TADDM erstellt werden. Über TADDM werden lediglich den LDAP-Benutzern und -Gruppen Rollen zugeordnet. Nur diese Zuordnungen der Benutzer/Gruppen zu den Rollen, die sogenannten Berechtigungen, müssen in TADDM erstellt und gespeichert werden. Bei der Administrator-ID handelt es sich um einen speziellen internen TADDM-Benutzer, der unabhängig von der konfigurierten Benutzerregistry immer mit dateibasierter Sicherheit verarbeitet wird. Dieser Benutzer kann immer für die erste Zuordnung von Rollen zu LDAP-Benutzern und -Gruppen verwendet werden.

Vorgehensweise

Gehen Sie wie folgt vor, um LDAP oder VMM für die Benutzerauthentifizierung zu verwenden:

1. Konfigurieren Sie TADDM für die Verwendung der LDAP-Registry. Konfigurieren Sie dazu die entsprechenden Eigenschaften in der Datei 'collation.properties'.
2. Melden Sie sich mit der TADDM-Administrator-ID beim Datenmanagementportal an.
3. Führen Sie einen der folgenden Schritte aus:
   - Durchsuchen Sie im Fensterbereich 'Benutzer' die LDAP-Registry über das Feld **Benutzer suchen** nach dem entsprechenden Benutzer.
   - Durchsuchen Sie im Fensterbereich 'Benutzergruppen' die LDAP-Registry über das Feld **Benutzergruppe suchen** nach der entsprechenden Benutzergruppe.

   Anmerkung: Als Suchergebnisse sind die von der LDAP-Registrysuche zurückgegebenen Namen der Benutzer oder Gruppen aufgeführt. Auf diesem Weg können keine Benutzer erstellt oder von LDAP in TADDM kopiert werden. Die Liste soll angeben, welche TADDM-Berechtigungen für die Benutzer erstellt werden müssen.
4. Ordnen Sie den aufgelisteten Benutzern (oder Gruppen) die erforderlichen TADDM-Rollen zu.

   Nur diese Berechtigungen, nicht die LDAP-Benutzer (oder Gruppen), werden in TADDM gespeichert.

Nächste Schritte
Gehen Sie wie folgt vor, um SSL für LDAP zu konfigurieren:

1. Suchen Sie in der Datei `collation.properties` nach der folgenden Eigenschaft und ändern Sie ihren Wert von `false` in `true`:

   ```
   com.collation.security.auth.ldapUseSSL
   ```
2. Konfigurieren Sie die folgenden Truststore- und Keystore-Eigenschaften wie gewünscht:

   ```
   com.collation.security.auth.ldapClientKeyStore
   ```

   ```
   com.collation.security.auth.ldapClientKeyStorePassphrase
   ```

   ```
   com.collation.security.auth.ldapClientTrustStore
   ```

   ```
   com.collation.security.auth.ldapClientTrustStorePassphrase
   ```
3. Ändern Sie gegebenenfalls den Port, an dem der LDAP-Server für SSL-Verbindungen empfangsbereit ist, indem Sie die folgende Eigenschaft konfigurieren:

   ```
   com.collation.security.auth.ldapPortNumber
   ```

Konfiguration für eingebundene WebSphere-Repositorys

Wenn Sie eine Tivoli WebSphere-Anwendung für eine zentrale Benutzerregistry konfiguriert haben, die eingebundene WebSphere-Repositorys verwendet, können Sie eine Konfiguration für eingebundene WebSphere-Repositorys in einer Registry für eingebundene Repositorys durchführen.

TADDM-Server für die Verwendung eingebundener WebSphere-Repositorys konfigurieren

Das eingebundene WebSphere-Repository ist ein flexibles Meta-Repository innerhalb von WebSphere, das mehrere Arten von Benutzerregistrys, einschließlich Microsoft Active Directory, unterstützt.

Vorbereitende Schritte

Die Konfiguration von TADDM für die Verwendung eingebundener WebSphere-Repositorys ist erforderlich, wenn Sie andere Tivoli-Produkte in Ihrer Umgebung verwenden und die einmalige Anmeldung zwischen TADDM und einem oder mehreren der folgenden Produkte benötigen:

- IBM Tivoli Change and Configuration Management Database (CCMDB) oder IBM SmartCloud Control Desk (SCCD)
- IBM Tivoli Business Service Manager

Für TADDM sind zusätzliche Services erforderlich, die nicht Bestandteil eines WebSphere-Standardpakets sind; wenn TADDM für die Verwendung von eingebundenen Repositorys konfiguriert werden soll, müssen Sie daher eine der folgenden WebSphere-Installationen verwenden:

- WebSphere Application Server Network Deployment (wie mit CCMDB oder SCCD installiert)
- WebSphere Application Server (wie zusammen mit IBM Tivoli Business Service Manager installiert)

Unterstützte Versionen dieses Produkts finden Sie im Abschnitt „Unterstützte Versionen” auf Seite 177.

Vor Ausführung dieser Konfigurationsschritte muss der über eingebundene WebSphere-Repositorys realisierte Authentifizierungservice bereits auf einem WebSphere Application Server Network Deployment-Server konfiguriert sein. Weitere Informationen finden Sie in der Dokumentation zu IBM Tivoli Change and Configuration Management Database (CCMDB) oder IBM SmartCloud Control Desk (SCCD).

Informationen zu diesem Vorgang

Diese Konfiguration ermöglicht die einmalige Anmeldung zwischen Tivoli-Anwendungen mithilfe von WebSphere-LTPA-Tokens (LTPA = Lightweight Third-Party Authentication). Bei einer Konfiguration von TADDM für die Verwendung derselben eingebundenen WebSphere-Repositorys, die auch von CCMDB bzw. SCCD verwendet werden, wird die einmalige Anmeldung für den Launch-in-Context zwischen IBM Tivoli CCMDB bzw. IBM SCCD und TADDM unterstützt.

Soll TADDM automatisch für die Verwendung eingebundener WebSphere-Repositorys konfiguriert werden, müssen Sie TADDM installieren und dabei **Eingebundene WebSphere-Repositorys** als Benutzerregistry auswählen.

Diese Konfiguration wird auf allen TADDM-Servertypen in allen Implementierungen unterstützt.

Vorgehensweise

Gehen Sie folgendermaßen vor, um die Konfiguration manuell auszuführen:

1. Stoppen Sie den TADDM-Server.
2. Geben Sie das von diesem TADDM-Server verwendete Benutzermanagementmodul an. Die folgenden Werte sind gültig:

   file
   Dieser Wert wird für eine dateibasierte Benutzerregistry verwendet. (Standardwert)

   ldap
   Dieser Wert wird für eine LDAP-Benutzerregistry verwendet.

   vmm
   Dieser Wert wird für eine Benutzerregistry verwendet, die die eingebundenen Repositorys von WebSphere Application Server verwendet.

Als Beispiel dient die Datei $COLLATION_HOME/etc/collation.properties:

```
com.collation.security.usermanagementmodule=vmm
```

3. Legen Sie den WebSphere-Hostnamen und den Port in der Datei collation.properties fest.
   Beispiele dafür sind:

   ```
   com.collation.security.auth.websphereHost=localhost
   com.collation.security.auth.webspherePort=2809
   ```

   Verwenden Sie bei der Angabe des WebSphere-Ports in der Datei collations.properties die folgende Eigenschaft com.collation.security.auth.webspherePort. Als WebSphere-Port muss der Port des WebSphere-Servers für das Bootprogramm angegeben werden. Für WebSphere Application Server und die integrierte Version von WebSphere Application Server ist 2809 der Standardport, für WebSphere Application Server Network Deployment, das von IBM Tivoli CCMDB bzw. IBM SCCD verwendet wird, ist 9809 der Standardport.
4. Legen Sie den Benutzernamen und das Kennwort des WebSphere-Administrators in der Datei collation.properties fest.
   Beispiele dafür sind:

   ```
   com.collation.security.auth.VMMAdminUsername=administrator
   com.collation.security.auth.VMMAdminPassword=password
   ```
5. Nehmen Sie die folgende Änderung an der Konfigurationsdatei für Authentifizierungsservices vor:
   - Bei Linux-, AIX- und Linux on zSeries-Betriebssystemen befindet sich die Datei im Pfad $COLLATION_HOME/etc/ibmessclientauthncfg.properties.
   - Bei Windows-Betriebssystemen befindet sich die Datei in folgendem Pfad: %COLLATION_HOME%\etc\ibmessclientauthncfg.properties.

   Setzen Sie in der Eigenschaft authnServiceURL den vollständig qualifizierten Domänennamen des Systems ein, auf dem Ihre WebSphere-Instanz installiert ist, sowie den HTTP-Port der WebSphere-Instanz.

   ```
   # This is the URL for the Authentication Service
   authnServiceURL=http://localhost:9080/TokenService/services/Trust
   ```
6. Kopieren Sie die WebSphere-Dateien orb.properties und iwsorbutil.jar in die durch Ihre TADDM-Installation verwendete JRE.

   Gehen Sie beispielsweise in einer TADDM-Linux-Installation wie folgt vor:

   a) Kopieren Sie dist/lib/websphere/6.1/orb.properties in dist/external/ jdk-Linux-i686/jre/lib/.

   b) Kopieren Sie dist/lib/websphere/6.1/iwsorbutil.jar in dist/external/ jdk-Linux-i686/jre/lib/ext/.
7. Legen Sie den WebSphere-Hostnamen und den Port in der Datei sas.client.props fest:
   - Bei Linux-, AIX- und Linux on zSeries-Betriebssystemen befindet sich die Datei in folgendem Pfad: $COLLATION_HOME/etc/sas.client.props.
   - Bei Windows-Betriebssystemen befindet sich die Datei in folgendem Pfad: %COLLATION_HOME%\etc\sas.client.props, Beispiel:

     ```
     com.ibm.CORBA.securityServerHost=host1.austin.ibm.com
     com.ibm.CORBA.securityServerPort=2809
     ```

   Anmerkung: Für WebSphere Application Server und die integrierte Version von WebSphere Application Server ist 2809 der Standardport, für WebSphere Application Server Network Deployment, das von IBM Tivoli CCMDB bzw. IBM SCCD verwendet wird, ist 9809 der Standardport.
8. Geben Sie in der Datei sas.client.props den Benutzernamen und das Kennwort des WebSphere-Administrators an.
   Beispiele dafür sind:

```
# RMI/IIOP user identity
com.ibm.CORBA.loginUserid=administrator
com.ibm.CORBA.loginPassword=password
```

9. So verschlüsseln Sie das Anmeldekennwort in der Datei `sas.client.props`:
   a) Kopieren Sie die Datei `sas.client.props` zurück auf den TADDM-Server in das Verzeichnis `$COLLATION_HOME/etc`.
   b) Verschlüsseln Sie das Kennwort je nach Betriebssystem, auf dem WebSphere installiert ist, wie folgt:
      - Unter Linux, AIX und Linux on System z:

        Verwenden Sie den Befehl `PropFilePasswordEncoder.sh`.
      - Unter Windows:

        Verwenden Sie `PropFilePasswordEncoder.bat`, zum Beispiel

        ```
        C:\WebSphere\profiles\AppSrv01\bin\PropFilePasswordEncoder C:\temp\sas
                .client.props com.ibm.CORBA.loginPassword
        ```
   c) Kopieren Sie die Datei `sas.client.props` zurück auf den TADDM-Server in das Verzeichnis `etc`.
10. Starten Sie den TADDM-Server.

Nächste Schritte

Nach Abschluss der Installation können Sie mit dem im lokalen dateibasierten TADDM-Repository definierten Standardbenutzer mit Administratorberechtigung weitere TADDM-Benutzer (einschließlich TADDM-Administratoren) konfigurieren. Diese TADDM-Benutzer werden anhand von eingebundenen WebSphere-Repositorys authentifiziert.

Es gibt Sicherheitskonfigurationen für Tivoli CCMDB bzw. IBM SCCD, mit denen Gruppen und Gruppenzugehörigkeiten in den Maximo-Anwendungen für Benutzer und Gruppen erstellt und gepflegt werden können.

Wenn Tivoli CCMDB bzw. IBM SCCD entsprechend konfiguriert ist, verwendet TADDM nicht das Repository von Tivoli CCMDB bzw. IBM SCCD, sondern ein eigenes Repository. Benutzer müssen sowohl in Tivoli CCMDB bzw. IBM SCCD/Maximo als auch in TADDM erstellt werden.

TADDM kann so konfiguriert werden, dass Benutzer- und Gruppendefinitionen in externen Benutzerregistrys über eingebundene WebSphere-Repositorys verwendet werden. Dagegen kann TADDM keine in Tivoli CCMDB gespeicherten Benutzer- und Gruppendefinitionen verwenden, da diese nicht von eingebundenen WebSphere-Repositorys unterstützt werden.

LTPA-Schlüssel für den Authentifizierungsservice aktualisieren

Wenn Sie bei Einsatz eingebundener WebSphere-Repositorys die einmalige Anmeldung verwenden, müssen die LTPA-Schlüssel (LTPA - Lightweight Third-Party Authentication) für den Authentifizierungsservice mit den von den eingebundenen WebSphere-Repositorys verwendeten Schlüsseln synchronisiert sein.

Vorgehensweise

Bei Änderungen an den von eingebundenen WebSphere-Repositorys verwendeten LTPA-Schlüsseln müssen Sie die vom Authentifizierungsservice verwendeten Schlüssel wie folgt resynchronisieren:

1. Exportieren Sie die neuen WebSphere-LTPA-Schlüssel:
   a) Wählen Sie in der WebSphere-Administrationskonsole **Secure administration, applications, and infrastructure** > **Authentication mechanisms and expiration** (Verwaltung, Anwendungen und Infrastruktur schützen > (Authentifizierungsverfahren und Ablauf) aus.
   b) Geben Sie für **Cross-cell single sign-on** (Zellenübergreifende einmalige Anmeldung) einen Dateinamen und ein Kennwort für die Datei an, die die exportierten LTPA-Schlüssel enthalten soll.

2. Wechseln Sie in einer Eingabeaufforderung in das Verzeichnis `bin` des entsprechenden WebSphere-Profils.
3. Führen Sie den folgenden **`wsadmin`**-WebSphere-Befehl aus:

```
wsadmin> $AdminTask importESSLTPAKeys {-pathname Pfadname -password Kennwort}
```

   Dabei sind *Pfadname* und *Kennwort* die Werte, die Sie beim Export der LTPA-Schlüssel als Dateiname und Kennwort angegeben haben.
4. Starten Sie den WebSphere-Server erneut.

Authentifizierungskanal sichern
Wenn Sie TADDM für die Verwendung eingebundener WebSphere-Repositorys konfigurieren, können Sie die Kommunikation zwischen dem Authentifizierungsclient und dem Authentifizierungsservice sichern.

Informationen zu diesem Vorgang

In TADDM wird ein Authentifizierungsservice verwendet, der die einmalige Anmeldung (Single Sign-on, SSO) unterstützt. Der Authentifizierungsservice wird während der Installation von IBM Tivoli Change and Configuration Management Database (IBM SmartCloud Control Desk (SCCD)) oder IBM Tivoli Business Service Manager installiert.

Unterstützte Versionen dieses Produkts finden Sie im Abschnitt „Unterstützte Versionen” auf Seite 177.

Es gibt zwei Mechanismen zur Sicherung der Kommunikation zwischen einem Authentifizierungsclient und einem Authentifizierungsservice:

- SSL
- Clientauthentifizierung

Authentifizierungskanal für SSL konfigurieren
Sie können die Kommunikation schützen, indem Sie die WebSphere-Unterzeichnerzertifikate für die Konfiguration von SSL zwischen dem Authentifizierungsclient und dem Authentifizierungsserver verwenden.

Vorgehensweise

Gehen Sie folgendermaßen vor, um eine Konfiguration für SSL zwischen dem Authentifizierungsclient und dem Authentifizierungsserver zu erstellen:

1. Führen Sie eine der folgenden Aktionen aus:
   a) Bei Verwendung der von Tivoli Integrated Portal installierten WebSphere-Instanz: Wählen Sie **SSL certificate and key mgmt** > **Manage endpoint security configurations** > **Node1** > **Key stores and certificates** > **NodeDefaultTrustStore** > **Signer certificates** (SSL-Zertifikat und Schlüsselmanagement > Sicherheitskonfigurationen für Endpunkte verwalten > Node1 > Schlüsselspeicher und Zertifikate > NodeDefaultTrustStore > Unterzeichnerzertifikate) aus.
   b) Bei Verwendung der von Tivoli Change and Configuration Management Database (CCMDB) oder IBM SmartCloud Control Desk installierten WebSphere-Instanz: Wählen Sie **SSL certificate and key mgmt** > **Manage endpoint security configurations** > **ctgNode01** > **Key stores and certificates** > **NodeDefaultTrustStore** > **Signer certificates** (SSL-Zertifikat und Schlüsselmanagement > Sicherheitskonfigurationen für Endpunkte verwalten > ctgNode01 > Schlüsselspeicher und Zertifikate > NodeDefaultTrustStore > Unterzeichnerzertifikate) aus.
2. Exportieren Sie die WebSphere-Unterzeichnerzertifikate in Dateien ('dummyclientsigner' beispielsweise in die Datei `signer1.cert` und 'dummyserversigner' in die Datei `signer2.cert`). Wenn Sie sich nicht sicher sind, welche Zertifikate exportiert werden sollen, müssen Sie alle Unterzeichnerzertifikate exportieren.
3. Kopieren Sie die `.cert`-Dateien auf den TADDM-Server. Erstellen Sie einen Zertifikatsspeicher für vertrauenswürdige Zertifikate (Truststore) und importieren Sie die WebSphere-Unterzeichnerzertifikate wie folgt:

```
$COLLATION_HOME/external/jdk-Linux-i686/jre/bin/keytool \
-genkey -alias truststore -keystore truststore.jks
```

```
$COLLATION_HOME/external/jdk-Linux-i686/jre/bin/keytool \
-import -trustcacerts -alias default -file signer1.cert -keystore truststore.jks
$COLLATION_HOME/external/jdk-Linux-i686/jre/bin/keytool \
-import -trustcacerts -alias dummyserversigner -file signer2.cert -keystore truststore.jks
```

4. Fügen Sie Kennwort und Pfad des Truststores in die Einträge der Datei `$COLLATION_HOME/etc/collation.properties` ein:

```
com.collation.security.auth.ESSClientTrustStore=/opt/IBM/taddm/dist/etc/truststore.jks
com.collation.security.auth.ESSClientTrustPwd=Kennwort
```

5. Aktualisieren Sie die URL des Tivoli-Authentifizierungsservice in der Datei `ibmessclien-tauthncfg.properties`, sodass HTTPS und Port 9443 verwendet werden. Stellen Sie dabei sicher, dass der WebSphere-Hostname korrekt ist ('localhost' wird durch diesen Hostnamen ersetzt) und dem HTTP-Eintrag ein Kommentarzeichen vorangestellt ist.

```
# This is the URL for the ESS Authentication Service
#authnServiceURL=http://localhost:9080/TokenService/services/Trust
authnServiceURL=https://localhost:9443/TokenService/services/Trust
```

Clientauthentifizierung konfigurieren

Aktivieren Sie zur Konfiguration der Clientauthentifizierung zwischen dem Authentifizierungsclient und dem Authentifizierungsserver die WebSphere-Sicherheitsfunktion für Anwendungen.

Vorbereitende Schritte

Nach Aktivierung der WebSphere-Anwendungssicherheit können Sie dem bei der TADDM-Installation angegebenen WebSphere-Benutzer mit Administratorberechtigung die Rolle 'TrustClientRole' zuordnen. Dies bedeutet zusätzliche Sicherheit für den Authentifizierungsservice, da nur die Benutzer sich für den Authentifizierungsservice authentifizieren können, die zur Gruppe 'TrustClientRole' gehören.

Vorgehensweise

So fügen Sie dem bei der TADDM-Installation angegebenen WebSphere-Benutzer mit Administratorberechtigung die Rolle 'TrustClientRole' zu:

1. Melden Sie sich an der WebSphere Administration Console an.
2. Klicken Sie in der Registerkarte **Security** (Sicherheit) auf **Enterprise Applications** (Unternehmensanwendungen).

   Das Teilfenster **Enterprise Applications** (Unternehmensanwendungen) wird angezeigt.
3. Klicken Sie in der Namensspalte der Tabelle 'Enterprise Applications' (Unternehmensanwendungen) auf die Authentifizierungsserviceanwendung (authnsvc_ctges).

   Das Teilfenster **Enterprise Applications > authnsvc_ctges** (Unternehmensanwendungen > authnsvc_ctges) wird angezeigt.
4. Klicken Sie in der Liste 'Detailed Properties' (Ausführliche Eigenschaften) im Teilfenster **Enterprise Applications > authnsvc_ctges** (Unternehmensanwendungen > authnsvc_ctges) auf **Security role to user/group mapping** (Zuordnung von Sicherheitsrolle zu Benutzer/Gruppe).

   Das Teilfenster **Enterprise Applications > authnsvc_ctges > Security role to user/group mapping** (Unternehmensanwendungen > authnsvc_ctges > Zuordnung von Sicherheitsrolle zu Benutzer/Gruppe) wird angezeigt.
5. Gehen Sie in der Tabelle im Teilfenster **Enterprise Applications > authnsvc_ctges > Security role to user/group mapping** (Unternehmensanwendungen > authnsvc_ctges > Zuordnung von Sicherheitsrolle zu Benutzer/Gruppe) wie folgt vor:
   - Klicken Sie in der Tabelle das Kontrollkästchen neben TrustClientRole an.
   - Löschen Sie die Markierung aus dem Kontrollkästchen **Everyone** (Alle).
   - Klicken Sie auf **Lookup Users** (Benutzer suchen) oder **Lookup Groups** (Gruppen suchen). Das Teilfenster **Enterprise Applications > authnsvc_ctges > Security role to user/group mapping > Lookup users or groups** (Unternehmensanwendungen > authnsvc_ctges > Zuordnung von Sicherheitsrolle zu Benutzer/Gruppe > Benutzer oder Gruppen suchen) wird angezeigt.

- Gehen Sie im Teilfenster **Enterprise Applications > authnsvc_ctges > Security role to user/group mapping > Lookup users or groups** (Unternehmensanwendungen > authnsvc_ctges > Zuordnung von Sicherheitsrolle zu Benutzer/Gruppe > Benutzer oder Gruppen suchen) wie folgt vor:
  - Suchen Sie mithilfe der Zeichenfolgeeingabefelder 'Begrenzung' und 'Suchen' nach Benutzern oder Gruppen. Wenn eine Gruppe oder ein Benutzer gefunden wurde, wird dieser in der Verfügbarkeitsliste angezeigt.
  - Wählen Sie aus der Verfügbarkeitsliste den gewünschten Benutzer oder die gewünschte Gruppe aus.
  - Klicken Sie auf **Verschieben**, um diesen Benutzer bzw. diese Gruppe in die Liste **Ausgewählt** zu verschieben.
- Klicken Sie auf **OK**. Das Teilfenster **Enterprise Applications > authnsvc_ctges > Security role to user/group mapping** (Unternehmensanwendungen > authnsvc_ctges > Zuordnung von Sicherheitsrolle zu Benutzer/Gruppe) wird angezeigt.
- Inaktivieren Sie im Teilfenster **Enterprise Applications > authnsvc_ctges > Security role to user/group mapping** (Unternehmensanwendungen > authnsvc_ctges > Zuordnung von Sicherheitsrolle zu Benutzer/Gruppe) das Kontrollkästchen **Everyone** (Alle).
- Klicken Sie auf **OK**. Das Teilfenster **Enterprise Applications > authnsvc_ctges** (Unternehmensanwendungen > authnsvc_ctges) wird angezeigt.
- Klicken Sie auf **Speichern**, um die Konfiguration zu speichern. Das Teilfenster **Enterprise Applications** (Unternehmensanwendungen) wird angezeigt.
- Klicken Sie auf **OK**. Das Teilfenster **Enterprise Applications > authnsvc_ctges** (Unternehmensanwendungen > authnsvc_ctges) wird angezeigt.

Konfiguration für Microsoft Active Directory

Sie können Microsoft Active Directory unter Verwendung von LDAP oder unter Verwendung eingebundener WebSphere-Repositorys als Authentifizierungsmethode für TADDM verwenden. Falls die einmalige Anmeldung an TADDM erforderlich ist, sollten Sie eingebundene WebSphere-Repositorys verwenden.

Informationen zu diesem Vorgang

Sie können die in der Active Directory-Registry definierten Benutzer verwenden, ohne neue Benutzer definieren zu müssen, indem Sie TADDM für die Verwendung von Active Directory konfigurieren. Dabei können Sie TADDM so konfigurieren, dass Active Directory als LDAP-Registry verwendet wird, oder Sie können TADDM für die Verwendung eingebundener WebSphere-Repositorys und anschließend eingebundene WebSphere-Repositorys für Active Directory konfigurieren.

Wird Active Directory bei der Installation von TADDM installiert, können Sie TADDM so konfigurieren, dass ein Benutzer aus Active Directory als TADDM-Administrator verwendet wird. Der Administrator kann den Zugriff auf TADDM konfigurieren und anderen Benutzern Zugriff auf TADDM-Objekte und -Services erteilen.

Diese Konfiguration wird auf allen TADDM-Servertypen in allen Implementierungen unterstützt.

Vorgehensweise

Führen Sie eine der folgenden Aktionen aus:

- So konfigurieren Sie Microsoft Active Directory für die Verwendung von LDAP:
  a. Konfigurieren Sie TADDM für LDAP. Weitere Informationen hierzu finden Sie im Abschnitt „Konfiguration für LDAP” auf Seite 23.
  b. Stellen Sie sicher, dass bei der Verwendung von Active Directory die Eigenschaft **`com.collation.security.auth.ldapFollowReferrals`** in der Datei `collation.properties` auf *true* gesetzt wird.
- So konfigurieren Sie Microsoft Active Directory für die Verwendung eingebundener WebSphere-Repositorys:

a. Konfigurieren Sie TADDM für eingebundene WebSphere-Repositorys. Weitere Informationen zur Konfiguration von TADDM für eingebundene WebSphere-Repositorys finden Sie im Abschnitt „TADDM-Server für die Verwendung eingebundener WebSphere-Repositorys konfigurieren" auf Seite 25.

b. Konfigurieren Sie eingebundene WebSphere-Repositorys für Microsoft Active Directory. Weitere Informationen zur Konfiguration unterstützter Entitätstypen in einer Konfiguration mit eingebundenen Repositorys finden Sie im Abschnitt **TADDM-Server für eingebundene WebSphere-Repositorys konfigurieren** im *WebSphere Application Server Information Center* unter http://www-01.ibm.com/support/knowledgecenter/SSAW57_6.1.0/com.ibm.websphere.nd.doc/info/ae/ae/twim_entitytypes.html.

TADDM-Web-Services schützen

Sie können den HTTP-Port für TADDM inaktivieren, indem Sie die Eigenschaft `com.ibm.cdb.secure.tomcat` (TADDM 7.3.0) oder die Eigenschaft `com.ibm.cdb.secure.liberty` (TADDM 7.3.0.1 und höher) in der Datei `collation.properties` auf `true` setzen. Außerdem können Sie mit dem Flag `com.ibm.cdb.http.ssl.protocol` ein sichereres SSL-Protokoll einrichten.

Der Standardwert der Eigenschaften `com.ibm.cdb.secure.tomcat` und `com.ibm.cdb.secure.liberty` ist `false`. Wenn der HTTP-Port inaktiviert ist, ist der Zugriff auf TADDM nur über den HTTPS-Port möglich. Beispiel: `https://example.com:9431`.

Einschränkung: Wenn Sie TADDM in der Streaming-Server-Implementierung installiert haben und Ihre Erkennungsserver und sekundären Speicherserver betriebsbereit sind, können Sie die Eigenschaft `com.ibm.cdb.secure.tomcat` oder die Eigenschaft `com.ibm.cdb.secure.liberty` auf `true` setzen. In diesem Fall ist der HTTP-Port inaktiviert und Sie können TADDM im sicheren Modus verwenden. Wenn Sie jedoch einen neuen Erkennungsserver oder sekundären Speicherserver zu Ihrer Implementierung hinzufügen möchten, müssen Sie den HTTP-Port temporär aktivieren, da das TADDM-Installationsprogramm das HTTPS-Protokoll nicht unterstützt. Gehen Sie folgendermaßen vor, um den sicheren Modus temporär zu inaktivieren:

1. Ändern Sie den Wert der Eigenschaft `com.ibm.cdb.secure.tomcat` oder der Eigenschaft `com.ibm.cdb.secure.liberty` in `false`.
2. Starten Sie den TADDM-Server erneut.
3. Installieren Sie einen neuen Erkennungsserver oder sekundären Speicherserver.
4. Ändern Sie den Wert der Eigenschaft `com.ibm.cdb.secure.tomcat` oder der Eigenschaft `com.ibm.cdb.secure.liberty` in `true`.
5. Starten Sie den TADDM-Server erneut.

Der Standardwert der Eigenschaft `com.ibm.cdb.http.ssl.protocol` ist TLS. Die sicheren Werte sind TLS, `TLSv1.1` und `TLSv1.2`. Wenn Sie die sichersten Protokolle (`TLSv1.1` oder `TLSv1.2`) verwenden wollen, müssen Sie Ihren Web-Browser so einrichten, dass er diese Protokolle unterstützt.

Angepasste SSL-Zertifikate zur Verwendung in TADDM installieren

Sie können Ihre eigenen angepassten SSL-Zertifikate installieren und diese bei TADDM verwenden.

Vorgehensweise

1. Erstellen Sie eine Sicherungskopie der folgenden Schlüsselspeicherdatei:
   - `$COLLATION_HOME/etc/serverkeys`
   - `$COLLATION_HOME/etc/jssecacerts.cert`
2. Wechseln Sie in das Verzeichnis `$COLLATION_HOME/etc`, öffnen Sie eine Befehlszeile und geben Sie die Parameter `keytool` und TADDM `sslpassphrase` mit den Werten wie folgt ein:
   - Betriebssystem Linux:

```
keytool=../external/jdk-Linux-x86_64/bin/keytool
pass=XXXXXXXX30374
```

- Betriebssystem Windows:

```
set keytool=..\external\jdk-Windows-i386-64\bin\keytool.exe
set pass=XXXXXXXX30374
```

Der Wert des Parameters `pass` ist der Wert der Eigenschaft `com.collation.sslpassphrase`, die in der Datei `collation.properties` angegeben ist.

3. Entfernen Sie das selbst signierte Zertifikat und den Schlüssel aus TADDM, indem Sie die folgenden Befehle ausführen:
   - Betriebssystem Linux:

```
$keytool -delete -alias collation -noprompt -keystore jssecacerts.cert
-storepass $pass
$keytool -delete -alias collation -noprompt -keystore serverkeys -storepass
$pass
```

   - Betriebssystem Windows:

```
%keytool% -delete -alias collation -noprompt -keystore jssecacerts.cert
-storepass %pass%
%keytool% -delete -alias collation -noprompt -keystore serverkeys -storepass
%pass%
```

4. Generieren Sie den SSL-Schlüssel mit den erforderlichen Werten für den CN, die Gültigkeit, den Algorithmus und andere Parameter und speichern Sie diesen in der Datei `serverkeys`.
   Sie können beispielsweise den folgenden Befehl ausführen:
   - Betriebssystem Linux:

```
$keytool -genkey -alias collation -keystore serverkeys -validity 3650
-keyAlg RSA -sigalg SHA256WithRSA -keypass $pass -storepass $pass -dname
"CN=John Public, OU=Engineering, OU=NA, o=Company, L=Manhattan,
S=New York, c=US"
```

   - Betriebssystem Windows:

```
%keytool% -genkey -alias collation -keystore serverkeys -validity 3650
-keyAlg RSA -sigalg SHA256WithRSA -keypass %pass% -storepass %pass% -dname
"CN=John Public, OU=Engineering, OU=NA, o=Company, L=Manhattan,
S=New York, c=US"
```

5. Erstellen Sie eine andere Sicherungskopie der Datei `serverkeys`in der Sie den generierten SSL-Schlüssel gespeichert haben.
6. Generieren Sie die Zertifikatssignieranforderung (CSR-Datei), indem Sie den folgenden Befehl ausführen:
   - Betriebssystem Linux:

```
$keytool -certreq -alias collation -storepass $pass -file
/tmp/certreq.csr -keystore serverkeys
```

   - Betriebssystem Windows:

```
%keytool% -certreq -alias collation -storepass %pass% -file
C:\temp\certreq.csr -keystore serverkeys
```

7. Verwenden Sie die CSR-Datei, um das SSL-Zertifikat von der offiziellen Zertifizierungsstelle abzurufen. Speichern Sie das Zertifikat auf Ihrem TADDM-Server, z. B. im Verzeichnis `tmp` im Betriebssystem Linux oder im Verzeichnis `C:\temp` im Betriebssystem Windows.

   Anmerkung: Es gibt zwei Arten von Zertifikaten: '**Einzelzertifikate**' und '**Vollständige Zertifikatsketten**'.

8. Fix Pack 6

Für den Import des empfangenen Zertifikats ('Einzelzertifikat' oder 'Vollständige Zertifikatskette') in die Dateien `serverkeys` und `jssecacerts.cert` unter TADDM führen Sie die folgenden Befehle aus:

Wichtig: Geben Sie für den Parameter `-file` den Pfad zur Datei an, in der Sie das SSL-Zertifikat im vorherigen Schritt gespeichert haben, beispielsweise `/tmp/cert.crt` auf dem Linux-Betriebssystem.

Einzelzertifikate

- Betriebssystem Linux:

```
$keytool -import -trustcacerts -alias root -noprompt -keystore
serverkeys -storepass $pass -keypass $pass -file /tmp/CAcert.cer
```

```
$keytool -import -trustcacerts -alias intermediate -noprompt -keystore
serverkeys -storepass $pass -keypass $pass -file
/tmp/IntermediateCAcert.cer
```

```
$keytool -import -trustcacerts -alias server -noprompt -keystore
serverkeys -storepass $pass -keypass $pass -file /tmp/serverCAcert.cer
```

```
$keytool -import -trustcacerts -alias root -noprompt -keystore
jssecacerts.cert -storepass $pass -keypass $pass -file /tmp/CAcert.cer
```

```
$keytool -import -trustcacerts -alias intermediate -noprompt -keystore
jssecacerts.cert -storepass $pass -keypass $pass -file
/tmp/IntermediateCAcert.cer
```

```
$keytool -import -trustcacerts -alias server -noprompt -keystore
jssecacerts.cert -storepass $pass -keypass $pass -file
/tmp/serverCAcert.cer
```

- Betriebssystem Windows:

```
%keytool% -import -trustcacerts -alias root -noprompt -keystore
serverkeys -storepass %pass% -keypass %pass% -file C:\temp\CAcert.cer
```

```
%keytool% -import -trustcacerts -alias intermediate -noprompt -keystore
serverkeys -storepass %pass% -keypass %pass% -file
C:\temp\IntermediateCAcert.cer
```

```
%keytool% -import -trustcacerts -alias server -noprompt -keystore
serverkeys -storepass %pass% -keypass %pass% -file
C:\temp\serverCAcert.cer
```

```
%keytool% -import -trustcacerts -alias root -noprompt -keystore
jssecacerts.cert -storepass %pass% -keypass %pass% -file
C:\temp\CAcert.cer
```

```
%keytool% -import -trustcacerts -alias intermediate -noprompt -keystore
jssecacerts.cert -storepass %pass% -keypass %pass% -file
C:\temp\IntermediateCAcert.cer
```

```
%keytool% -import -trustcacerts -alias server -noprompt -keystore
jssecacerts.cert -storepass %pass% -keypass %pass% -file
C:\temp\serverCAcert.cer
```

Vollständige Zertifikatsketten

- Betriebssystem Linux:

```
$keytool –import –trustcacerts -alias collation -noprompt –keystore
serverkeys -storepass $pass -keypass $pass -file /tmp/cert_chain.crt
```

```
$keytool –import –trustcacerts -alias collation -noprompt –keystore
jssecacerts.cert -storepass $pass -keypass $pass -file
```

```
/tmp/cert_chain.crt
```

- Betriebssystem Windows:

```
%keytool% -import -trustcacerts -alias collation -noprompt -keystore
serverkeys -storepass %pass% -keypass %pass% -file
C:\temp\cert_chain.crt
```

```
%keytool% -import -trustcacerts -alias collation -noprompt -keystore
jssecacerts.cert -storepass %pass% -keypass %pass% -file
C:\temp\cert_chain.crt
```

9. Starten Sie den TADDM-Server erneut.

Nächste Schritte

Bewahren Sie die Sicherungskopien der Datei `serverkeys` auf, die Sie in Schritt 4 erstellt haben, sowie die Datei, in der Sie das SSL-Zertifikat in Schritt 7 gespeichert haben. Sie benötigen diese Dateien, wenn Sie das Zertifikat ersetzen oder erneuern müssen. Gehen Sie wie folgt vor, um das Zertifikat zu ersetzen oder zu erneuern:

1. Wiederholen Sie die Schritte 2 und 3.
2. Stellen Sie die Datei `serverkeys` wieder her.
3. Wiederholen Sie die Schritte 8 und 9.

TADDM-Server verwalten

Damit Sie TADDM für die Erkennung konfigurieren können, muss Ihnen die Verwaltung der TADDM-Server vertraut sein, die zahlreiche Tasks umfasst.

TADDM-Serverstatus überprüfen

Sie können den Status des TADDM-Servers in der Administratorkonsole oder mithilfe des Befehls **`control`** überprüfen.

Status in der Administratorkonsole überprüfen

Wenn Sie den Status in der Administratorkonsole überprüfen möchten, öffnen Sie einen Web-Browser und geben Sie den URL und die Portnummer des Systems ein, auf dem der TADDM-Server installiert ist. Der folgende URL ist ein Beispiel:

```
http://system.company.com:9430
```

Die Administratorkonsole wird geöffnet und darin eine Liste der Komponenten des TADDM-Servers und deren Status angezeigt.

Status mit dem Befehl `control` überprüfen

Wenn Sie den Status mithilfe des Befehls **`control`** überprüfen möchten, führen Sie folgende Schritte aus:

1. Melden Sie sich als Benutzer ohne Rootberechtigung an, der während des Installationsprozesses definiert wurde.
2. Wechseln Sie in einer Eingabeaufforderung zu dem Verzeichnis, in dem der TADDM-Server installiert ist.
3. Führen Sie folgende Befehle aus:
   - Unter AIX, Linux und Linux on System z:

```
$COLLATION_HOME/bin/control status
```

- Unter Windows:

```
%COLLATION_HOME%\bin\control.bat status
```

Je nach der Implementierung, die Sie haben, und nach dem Servertyp, auf dem TADDM in der jeweiligen Implementierung ausgeführt wird, wird folgende Ausgabe angezeigt:

Synchronisationsserverimplementierung

Synchronisationsserver

- TADDM 7.3.0:

```
DbInit: Started
Tomcat: Started
EcmdbCore: Started

TADDM: Running
```

- TADDM 7.3.0.1 und höher:

```
DbInit: Started
Liberty: Started
EcmdbCore: Started

TADDM: Running
```

Domänenserver

- TADDM 7.3.0:

```
Discover: Started
DbInit: Started
Tomcat: Started
Topology: Started
DiscoverAdmin: Started
Proxy: Started
EventsCore: Started

TADDM: Running
```

- TADDM 7.3.0.1 und höher:

```
Discover: Started
DbInit: Started
Liberty: Started
Topology: Started
DiscoverAdmin: Started
Proxy: Started
EventsCore: Started

TADDM: Running
```

Streaming-Server-Implementierung

Speicherserver

- TADDM 7.3.0:

```
TADDM: Starting
EtaddmCore: Started
DbInit: Started
Tomcat: Started

TADDM: Running
```

- TADDM 7.3.0.1 und höher:

```
TADDM: Starting
EtaddmCore: Started
DbInit: Started
Liberty: Started
```

```
TADDM: Running
```

Erkennungsserver

- TADDM 7.3.0:

```
Discover: Started
Tomcat: Started
DiscoverAdmin: Started
ProxyLite: Started
EventsCore: Started

TADDM: Running
```

- TADDM 7.3.0.1 und höher:

```
Discover: Started
Liberty: Started
DiscoverAdmin: Started
ProxyLite: Started
EventsCore: Started

TADDM: Running
```

TADDM-Server starten

Wenn Sie bei der Installation die Option **Server bei Systemstart starten** ausgewählt haben, wird der TADDM-Server bei jedem Systemstart automatisch gestartet.

Informationen zu diesem Vorgang

Wichtig: Bevor der TADDM-Server gestartet wird, muss ein lokaler oder ferner Datenbankserver gestartet werden und aktiv sein. Der TADDM-Server kann nicht fehlerfrei initialisiert oder ausgeführt werden, wenn die Datenbank nicht verfügbar ist.

Vorgehensweise

Gehen Sie folgendermaßen vor, um den TADDM-Server manuell zu starten:

1. Melden Sie sich als Benutzer ohne Rootberechtigung an, der während des Installationsprozesses definiert wurde.
2. Öffnen Sie die Eingabeaufforderung.
3. Wechseln Sie in das Verzeichnis, in dem der TADDM-Server installiert ist.
4. Führen Sie das Start-Script mit einem der folgenden Befehle aus:
   - Unter Linux, AIX und Linux on System z:

     ```
     $COLLATION_HOME/bin/control start
     ```

   - Unter Windows:

     ```
     %COLLATION_HOME%\bin\startServer.bat
     ```

   Beim Start des Servers auf einem Windows-System kann folgende Zeitüberschreitungsnachricht angezeigt werden: `Error 1053: The service did not respond to the start or control request in a timely fashion` (Fehler 1053: Der Service hat nicht rechtzeitig auf die Start- oder Steueranforderung geantwortet). Dieser Fehler tritt auf, da der TADDM-Server beim Start möglicherweise die zulässige Startzeit überschreitet. Sie können diese Nachricht ignorieren. Der Startvorgang wird fortgesetzt, bis er abgeschlossen ist.

   Wenn Sie den TADDM-Server mit Rootberechtigungen installiert haben, können Sie den TADDM-Server manuell starten, indem Sie folgendes Script ausführen:

   ```
   /etc/init.d/collation start
   ```

TADDM-Server stoppen

Sie können den TADDM-Server und die zugehörigen Erkennungsprozesse manuell stoppen.

Vorgehensweise

So stoppen Sie den TADDM-Server manuell:

1. Melden Sie sich als Benutzer ohne Rootberechtigung an, der während des Installationsprozesses definiert wurde.
2. Öffnen Sie die Eingabeaufforderung.
3. Wechseln Sie in das Verzeichnis, in dem der TADDM-Server installiert ist.
4. Führen Sie das Stopp-Script mit einem der folgenden Befehle aus:
   - Unter Linux, AIX und Linux on System z:

     ```
     $COLLATION_HOME/bin/control stop
     ```

   - Unter Windows:

     ```
     %COLLATION_HOME%\bin\stopServer.bat
     ```

   Wenn Sie den TADDM-Server mit Rootberechtigungen installiert haben, können Sie den TADDM-Server manuell stoppen, indem Sie folgendes Script ausführen:

   ```
   /etc/init.d/collation stop
   ```

Nächste Schritte

Einige Sensoren werden in ihrer eigenen Java Virtual Machine (JVM) ausgeführt. Wenn Sie während eines Erkennungsprozesses das Steuerscript (`./control stop`) verwenden, um TADDM zu stoppen, müssen diese zusätzlichen, als lokale Anker bezeichneten JVMs eventuell manuell gestoppt werden. Andernfalls kann ein nicht erwartetes Verhalten auftreten. Bestimmte Erkennungsvorgänge können beispielsweise mit einer schlechteren Leistung erfolgen.

Geben Sie den folgenden Befehl ein, um sicherzustellen, dass der lokale Anker nicht mehr aktiv ist:

```
% ps -ef |grep -i anchor
```

Dieser Befehl erkennt alle aktiven lokalen Anker-Prozesse. Die Ausgabe ähnelt dem folgenden Codebeispiel:

```
 coll   23751  0.0  0.0  6136  428 ?   S  Jun02   0:00 /bin/sh
    local-anchor.sh 8494 <Weitere Informationen finden Sie hier>
```

Wenn ein Prozess aktiv ist, stoppen Sie ihn mit dem folgenden Befehl:

```
-  % kill -9 23751
```

Wenn Sie den Befehl ausgeführt haben, prüfen Sie mit folgendem Befehl, ob der Prozess gestoppt wurde:

```
 % ps -ef |grep -i anchor
```

Daten sichern

Sichern Sie Ihre Daten regelmäßig, um sie bei einem Systemausfall wiederherstellen zu können.

Vorbereitende Schritte

Stoppen Sie vor einer Datensicherung den TADDM-Server.

Vorgehensweise

Speichern Sie alle Dateien in dem Verzeichnis, in dem der TADDM-Server installiert ist.

- Unter Linux, AIX und Linux on System z ist `/opt/IBM` der Standardpfad zum Verzeichnis.
- Bei Windows-Betriebssystemen ist `C:\opt\IBM` der Standardpfad zum Verzeichnis.

Nächste Schritte
Schlagen Sie zur Sicherung der Datenbankdateien in der Dokumentation nach, die vom Datenbankanbieter geliefert wird.

Daten wiederherstellen

Nach einem Systemausfall können Sie die Konfigurations-, Daten- und Datenbankdateien wiederherstellen. Auf diese Weise können Sie den Normalbetrieb ab dem Zeitpunkt der letzten Sicherung vor dem Ausfall wiederaufnehmen.

Vorgehensweise

So stellen Sie die Daten aus Sicherungsdatenträgern wieder her:

1. Führen Sie eine der folgenden Aktionen aus:
   - Stellen Sie das Verzeichnis `/opt/IBM` wieder her und starten Sie TADDM erneut.
   - Stellen Sie das Verzeichnis `C:\opt\IBM` wieder her und starten Sie TADDM erneut.
2. Suchen Sie die Sicherungskopie der Datendateien.
3. Öffnen Sie die Eingabeaufforderung.
4. Wechseln Sie in das Verzeichnis, in dem der TADDM-Server installiert ist.
5. Kopieren Sie die Sicherungskopie der Datendateien in das Installationsverzeichnis.
6. Schließen Sie das Fenster mit Eingabeaufforderung.
7. Starten Sie den TADDM-Server.

Nächste Schritte
Wenn die Datenbank vom Systemausfall betroffen war, stellen Sie die Datenbankdateien mithilfe der Dokumentation Ihres Datenbankanbieters wieder her.

Erkennungsbereiche, Profile und angepasste Servervorlagen zwischen TADDM-Servern kopieren

Mithilfe des Befehls **`datamover.sh|bat`** können Sie Erkennungsbereiche, Profile und angepasste Servervorlagen zwischen TADDM-Servern kopieren.

Sie können Erkennungsbereiche, Profile und angepasste Servervorlagen (alle Entitäten) exportieren oder angeben, welche Entität auf einem Server exportiert werden soll. Anschließend können Sie die Entität oder die Entitäten auf dem Zielserver importieren.

Einschränkung: Zur Wahrung der Datenintegrität müssen Sie die Daten zwischen denselben Versionen der TADDM-Server verschieben.

Gehen Sie folgendermaßen vor, um die Entitäten zwischen TADDM-Servern zu kopieren:

1. Führen Sie den folgenden Befehl auf dem Quellenserver aus, um die erforderliche Entität oder die erforderlichen Entitäten in eine Datei zu exportieren:

   ```
   datamover.sh|bat -u Benutzer -p Kennwort -a Aktion [-t Typ ] [-f Dateiname]
   ```

   Dabei gilt:

   Benutzer
   Der TADDM-Benutzername.

   Kennwort
   Das TADDM-Benutzerkennwort.

   Aktion
   Geben Sie eine der folgenden Aktionen an: `import`, `export` oder `help`.

Optional: ***Typ***
Geben Sie eine der folgenden Aktionen an: `all`, `scope`, `profile` oder `template`. Der Standardwert ist `all`.

Optional: ***Dateiname***
Geben Sie einen Dateinamen an. Der Standardwert ist `dtamover.xml`.

Die standardmäßigen Erkennungsprofile werden nicht exportiert, während alle angepassten Servervorlagen, vom Benutzer erstellten Profile und Bereiche hingegen exportiert werden können.

Nach der Ausführung des Befehls werden die Informationen zu den exportierten Entitäten angezeigt. Bei der Ausgabedatei `exporthost.xml` werden beispielsweise die folgenden Informationen bereitgestellt:

```
6 Bereiche exportiert
1 Profil exportiert
57 Vorlagen exportiert
```

2. Kopieren Sie die Datei oder die Dateien auf den Zielserver, führen Sie den Befehl **`datamover.sh|bat`** aus und importieren Sie die Entität oder die Entitäten.

   Beim Importieren von Entitäten gelten die folgenden Regeln:

   - Wenn ein Bereich oder ein Profil mit demselben Namen bereits auf dem Server vorhanden ist, wird der importierte Bereich oder das importierte Profil umbenannt. Die Datei wird in *Name*_TADDM umbenannt.
   - Wenn eine Vorlage mit demselben Namen bereits auf dem Server vorhanden ist, wird die Vorlage mit der vorhandenen Vorlage zusammengeführt.

Discovery Management Console implementieren

Nachdem Sie sichergestellt haben, dass der TADDM-Server verfügbar ist, kann die Discovery Management Console implementiert werden.

Vorgehensweise

So implementieren Sie die Discovery Management Console:

1. Stellen Sie den Benutzern die URL (einschließlich der Portnummer) des Systems zur Verfügung, auf dem der TADDM-Server installiert ist.

   Ihre URL-Eingabe könnte beispielsweise wie folgt aussehen:

   ```
   http://system.company.com:9430
   ```

2. Teilen Sie den Benutzern ihren Benutzernamen und ihr Kennwort mit.
3. Geben Sie an, ob die Benutzer Secure Sockets Layer (SSL) verwenden sollten.

   Weisen Sie bei Verwendung von SSL die Benutzer an, anhand der Instruktionen auf der Installations- und Startseite der Discovery Management Console einen Truststore für den TADDM-Server zu speichern. Weitere Informationen finden Sie im TADDM-*Installationshandbuch*.

   Wichtig: SSL sollte für die gesamte Kommunikation zwischen der Discovery Management Console und dem TADDM-Server verwendet werden.
4. Die Benutzer müssen über eine unterstützte Installation der Java Runtime Environment auf dem System verfügen, mit dem die Discovery Management Console angezeigt wird.

   Weitere Informationen zu Clientvoraussetzungen finden Sie im TADDM-*Installationshandbuch*.
5. Verweisen Sie Benutzer auf das TADDM-*Benutzerhandbuch*; es enthält Informationen zum Starten der Discovery Management Console.

TADDM-Kommunikation konfigurieren

Zur Einrichtung der TADDM-Kommunikation müssen alle erforderlichen Services, Verbindungen und Firewalls konfiguriert werden.

TADDM-Services

Die TADDM-Konnektivität lässt sich in drei Bereiche einteilen:

Öffentliche Konnektivität
Öffentliche Konnektivität bezieht sich auf die Netzkonnektivität von außerhalb der TADDM-Infrastruktur, beispielsweise Verbindungen zum TADDM-Server über das Datenmanagementportal, die Discovery Management Console oder API-Clients. Dies ist die höchste Ebene der Konnektivität.

Inter-Server-Konnektivität
Inter-Server-Konnektivität bezieht sich auf die Netzkonnektivität zwischen Elementen der TADDM-Kerninfrastruktur, also zwischen Erkennungsservern und Speicherservern. Dies ist die mittlere Ebene der Konnektivität.

Lokale Konnektivität
Mit lokaler Konnektivität ist die Netzkonnektivität zwischen den lokalen Services eines Systems gemeint. Dies ist die niedrigste Ebene der Konnektivität.

Die Konnektivität kann für jeden Service während der Installation oder später durch eine Änderung der Konfigurationseigenschaften in der Konfigurationsdatei `collation.properties` konfiguriert werden.

Standardschnittstelle der Services
Zur Konfiguration einer Standardüberwachungsschnittstelle für alle Services stellen Sie in der Datei `collation.properties` die Eigenschaft `com.ibm.cdb.global.hostname` ein.

Tabelle 2. Standardschnittstelleneinstellungen für Services

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für globale Services	com.ibm.cdb.global.hostname	0.0.0.0

Vom Kommunikationstyp abhängige Überwachungsschnittstelle
Wenn Sie empfangsbereite Schnittstellen separat für die Services jedes Konnektivitätsbereichs konfigurieren möchten, ändern Sie die entsprechende Eigenschaft in der Datei `collation.properties`.

Tabelle 3. Standardschnittstelleneinstellungen für Services

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für öffentliche Konnektivitätsservices	com.ibm.cdb.public.hostname	Definiert durch com.ibm.cdb.global.hostname
Host für Inter-Server-Konnektivitätsservices	com.ibm.cdb.interserver.hostname	Definiert durch com.ibm.cdb.global.hostname
Host für lokale Konnektivitätsservices	com.ibm.cdb.local.hostname	127.0.0.1

Anmerkung: Wenn keine Schnittstelle bzw. eine Schnittstelle mit dem Wert `0.0.0.0.` angegeben ist, muss eine lokale externe Netzschnittstelle für die Kommunikation mit sich selbst geöffnet sein. Ist eine Schnittstelle angegeben, so muss diese für die Kommunikation mit sich selbst geöffnet sein.

Überwachungsschnittstelle für bestimmte Services

Sie können für jeden Service während der Installation oder später durch Änderung der entsprechenden Eigenschaft in der Datei `collation.properties` einen separaten TCP-Port konfigurieren.

Konfiguration der Serviceschnittstelle
Wenn Sie für jeden Service eine bestimmte Überwachungsschnittstelle konfigurieren möchten, ändern Sie die entsprechende Eigenschaft mit dem Suffix `host` in der Datei `collation.properties`.

Beispiel für den Service TopologyManager:

```
com.ibm.cdb.service.TopologyManager.host=192.168.1.5
```

Anmerkung: Diese Namenskonvention gilt nicht für die Registrys öffentlicher oder Inter-Server-Services.

Konfiguration des Serviceports

Wenn Sie für jeden Service eine bestimmte Überwachungsschnittstelle konfigurieren möchten, ändern Sie die entsprechende Eigenschaft mit dem Suffix `port` in der Datei `collation.properties`.

Das folgende Beispiel bezieht sich auf den Service TopologyManager:

```
com.ibm.cdb.service.TopologyManager.port=9550
```

Konfiguration für SSL-Services

Wenn Sie für jeden SSL-Service eine bestimmte Überwachungsschnittstelle bzw. einen Port konfigurieren möchten, ändern Sie die entsprechende Eigenschaft mit dem Infix `secure` in der Datei `collation.properties`.

Das folgende Beispiel bezieht sich auf den Service SecureApiServer:

- `com.ibm.cdb.service.SecureApiServer.secure.host=192.168.1.5`
- `com.ibm.cdb.service.SecureApiServer.secure.port=9531`

Konfiguration der Schnittstelle für das Webportal (HTTP und HTTPS)

Zur Konfiguration einer Überwachungsschnittstelle für ein Webportal (HTTP und HTTPS) ändern Sie die Eigenschaft `com.ibm.cdb.service.web.host` in der Datei `collation.properties`.

Anmerkung: Der HTTP- und HTTPS-Host wird im Gegensatz zu anderen Services durch Änderung einer Eigenschaft konfiguriert.

Datenbankverbindungen

Ändern Sie zur Konfiguration einer bestimmten Datenbankverbindung die Eigenschaften `com.collation.db.port` und `com.collation.db.server` in der Datei `collation.properties`.

Beispiele dafür sind:

- `com.collation.db.port=65432`
- `com.collation.db.server=9.156.47.156`

DNS-Verbindungen

Wenn Sie für die Kommunikation vollständig qualifizierte Domänennamen (FQDN) verwenden möchten, müssen Sie sicherstellen, das der an der Kommunikation beteiligte Host den FQDN des DNS-Service auflösen kann.

Sensorverbindungen

Die Konfiguration der vom Pingsensor und vom Portsensor für die Verbindungen verwendeten Ports wird in der Dokumentation zum Pingsensor und Portsensor erläutert. Vergewissern Sie sich, dass die Ports für die zu erkennenden Services geöffnet sind.

Tabelle 4. Standardports für Ping- und Portsensor

Portname	**Standardport**	**Protokoll**
SSH	22	TCP
Telnet	23	TCP

Tabelle 4. Standardports für Ping- und Portsensor (Forts.)		
Portname	**Standardport**	**Protokoll**
DNS	53	TCP
WMI	135	TCP
Fix Pack 2 PowerShell	5985, 5986	TCP
LDAP	389	TCP
SMB	445	TCP
Oracle	1521	TCP
CiscoWorks	1741	TCP

Ankerverbindungen

TADDM kann für die Verbindung zu einem Ankerserver einen der folgenden Verbindungstypen verwenden: *ssh* oder *direct*. Wenn Sie einen bestimmten Ankerverbindungstyp konfigurieren möchten, ändern Sie den Wert der Eigenschaft `com.collation.discover.anchor.connectType` in der Datei `collation.properties` in *ssh* bzw. *direct*.

Wenn Sie einen bestimmten Ankerverbindungstyp für eine bestimmte Adresse konfigurieren möchten, ändern Sie in der Datei `collation.properties` die Eigenschaft `com.collation.discover.anchor.connectType` mit der IP-Adresse als Suffix. Beispiel:

```
com.collation.discover.anchor.connectType.1.2.3.4=direct
```

Port 8497 ist als Standardport für die Verbindung mit einem Ankerserver festgelegt. Sie können diesen Port über die Discovery Management Console konfigurieren.

- Im Modus *ssh* müssen die Ports in einer öffentlichen Schnittstelle, auf die der TADDM-Server und der Ankerverbindungsport in einer Loopback-Schnittstelle auf dem System zugreifen, auf dem sich der Ankerserver befindet, für die SSH-Kommunikation geöffnet werden.
- Im Modus *direct* müssen die Ports in einer öffentlichen Schnittstelle, auf die der TADDM-Server zugreift, für die SSH-Kommunikation und die Ankerverbindung geöffnet werden.

Gateway-Verbindungen

TADDM kann über SSH eine Verbindung zu einem Gateway-Server herstellen.

Auf dem Gateway muss der Host-SSH-Port in einer öffentlichen Schnittstelle, auf die der TADDM-Server zugreift, für die Kommunikation geöffnet sein.

Hostnamen eines Servers in einen vollständig qualifizierten Domänennamen auflösen

Damit die erfolgreiche Kommunikation zwischen Servern sichergestellt ist, muss der Host-Server seinen Hostnamen unter Verwendung der Auflösungsbibliothek des Betriebssystems in einen vollständig qualifizierten Domänennamen (FQDN) auflösen können. Eine der folgenden Bedingungen muss erfüllt sein:

- In der Suchreihenfolge der Hostauflösung des Betriebssystems muss der Domänennamensserver (DNS) Vorrang vor lokalen Dateien haben. In der Dokumentation des Betriebssystems finden Sie Anweisungen zur Konfiguration dieser Einstellung.
- In der Hostdatei muss der vollständig qualifizierte Domänenname des TADDM-Servers vor dem Kurznamen stehen.

Falls keine dieser Bedingungen erfüllt werden kann, können Sie die Eigenschaft `com.collation.serverID` in der Datei `collation.properties` auf das Internet Protocol oder den Hostnamen des TADDM-Servers setzen. Stellen Sie außerdem sicher, dass die Server-ID (ServerID) in 'Synchronization Server / Enterprise Server > Data Management Portal > Domain Management > Domain Host Name' (Syn-

chronisationsserver/Unternehmensserver > Datenmanagementportal > Domänenverwaltung > Domänenhostname) auf denselben Wert gesetzt ist.

Ephemere Ports

Die TADDM-Kommunikation schließt die Verwendung ephemerer Ports ein. Diese Ports sind temporär und gelten speziell für ein Betriebssystem. Jedes Betriebssystem verfügt über einen definierten Portnummernbereich, aus dem bestimmte Ports zufällig ausgewählt werden. Diese Ports werden nicht von TADDM definiert. Informationen zum Portbereich und zur erforderlichen Konfiguration sowie weitere Einzelheiten finden Sie in der Dokumentation zu dem Betriebssystem, das Sie verwenden.

Firewalls konfigurieren

Zur Einrichtung der TADDM-Kommunikation müssen die erforderlichen Firewalls konfiguriert werden. Welche Schritte für diese Task im Einzelnen erforderlich sind, hängt davon ab, ob eine Domänenserver-, Streaming-Server- oder Synchronisationsserverimplementierung konfiguriert ist.

Die Informationen zur Firewallkonfiguration sind in Tabellen aufgeführt. In jeder Tabelle ist die Richtung der Kommunikation angegeben. Auf dem Zielsystem muss der Zielservice-Port in der Firewall als Quelle für abgehende Verbindungen und als Ziel für eingehende Verbindungen offen sein. Auf dem Quellensystem muss der Zielservice-Port in der Firewall als Ziel für abgehende Verbindungen und als Quelle für eingehende Verbindungen offen sein.

Wichtig: Auch über Low-Level-Clients müssen Services für höhere Konnektivitätsebenen bereitstehen. Öffentliche Services müssen zum Beispiel auch für die Inter-Server-Konnektivität geöffnet sein.

Wenn in der Tabelle für die Richtung Loopback angegeben ist, muss an dieser Schnittstelle die gesamte Kommunikation geöffnet sein. Wenn Sie die Standardportkonfiguration ändern, müssen Sie darauf achten, dass Sie die entsprechenden Ports für Ihre Umgebungskonfiguration öffnen.

Konfiguration von Firewalls in einer Domänenserverimplementierung

Die Firewalls in einer Domänenserverimplementierung müssen so konfiguriert werden, dass bestimmte Ports für die Kommunikation geöffnet sind.

In der folgenden Abbildung ist die TADDM-Kommunikation in einer Domänenserverimplementierung dargestellt.

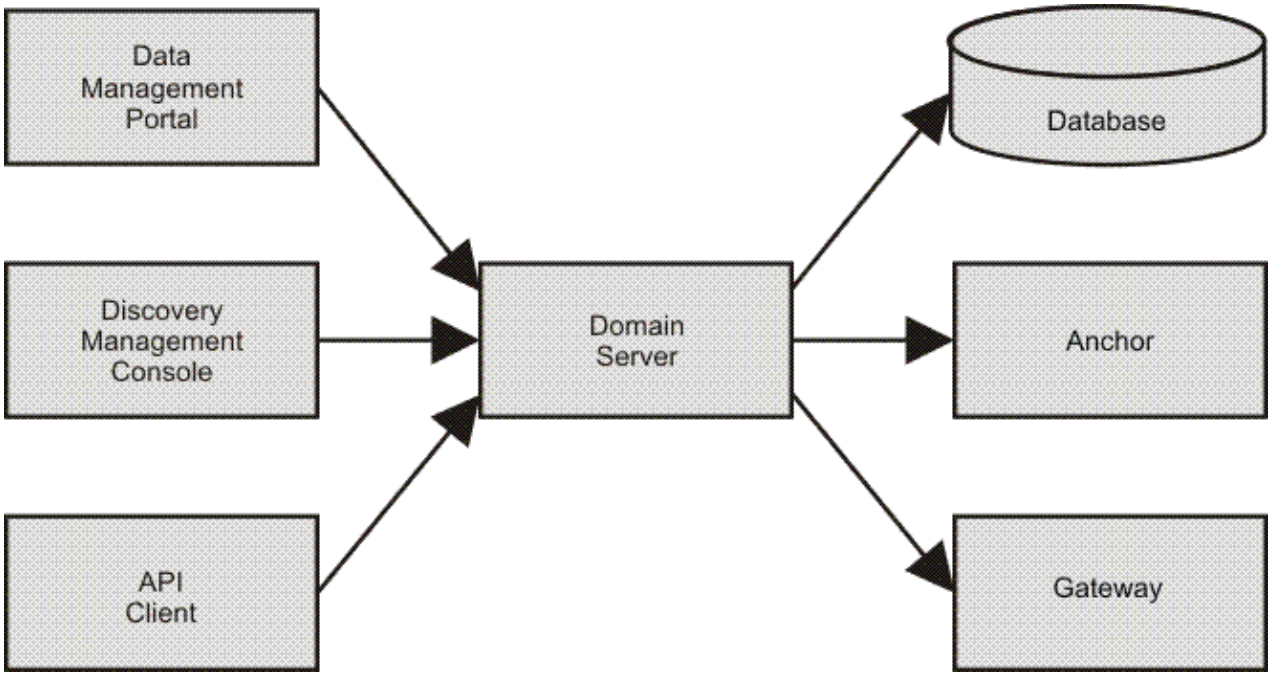

Abbildung 1. TADDM-Kommunikation in einer Domänenserverimplementierung

Konnektivitätsservices

Für eine Domänenerverimplementierung können Sie öffentliche und lokale Konnektivitätsservices sowie Inter-Server-Konnektivitätsservices konfigurieren.

Öffentliche Konnektivitätsservices

In der folgenden Tabelle sind die Standardhosteinstellungen für die öffentlichen Konnektivitätsservices des Domänenservers aufgeführt.

Tabelle 5. Standardhosteinstellungen für die öffentlichen Konnektivitätsservices des Domänenservers

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für öffentlichen Service	com.ibm.cdb.public.hostname	Definiert durch com.ibm.cdb.global.hostname

In der folgenden Tabelle sind die Standardporteinstellungen für die öffentlichen Konnektivitätsservices des Domänenservers aufgeführt.

Tabelle 6. Standardporteinstellungen für die öffentlichen Konnektivitätsservices des Domänenservers

Name	Konfigurationseigenschaft	Protokoll	Standardport
API-Server-Port	com.ibm.cdb.service.ApiServer.port	TCP	9530
Sicherer API-Server-Port	com.ibm.cdb.service.SecureApiServer.secure.port	TCP	9531
HTTP-Port (ohne SSL)	com.ibm.cdb.service.web.port	TCP	9430
HTTPS-Port (mit SSL)	com.ibm.cdb.service.web.secure.port	TCP	9431
Kommunikationsport für GUI-Server	com.ibm.cdb.service.ClientProxyServer.port	TCP	9435
SSL-Kommunikationsport für GUI-Server	com.ibm.cdb.service.SecureClientProxyServer.secure.port	TCP	9434
Port für öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port	TCP	9433

Lokale Konnektivitätsservices

Die Ports für die lokalen Services werden nicht explizit festgelegt. An der Schnittstelle, die für lokale Services definiert ist, müssen alle Ports geöffnet sein. Standardschnittstelle ist die Loopback-Schnittstelle.

In der folgenden Tabelle sind die Standardhosteinstellungen für die lokalen Konnektivitätsservices des Domänenservers aufgeführt.

Tabelle 7. Standardhosteinstellungen für die lokalen Konnektivitätsservices des Domänenservers

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für lokalen Service	com.ibm.cdb.local.hostname	127.0.0.1

Kommunikation in der Domänenserverimplementierung konfigurieren

Für eine erfolgreiche Kommunikation in der Domänenserverimplementierung müssen öffentliche und lokale Konnektivitätsservices konfiguriert werden.

Die folgende Tabelle zeigt die Elemente, die Sie in der Domänenserverimplementierung verbinden können, und die Ports, die Sie für eine erfolgreiche Kommunikation öffnen müssen.

Kommunikation zwischen dem Datenbankserver und dem Domänenserver

Tabelle 8. Kommunikation zwischen dem Datenbankserver und dem Domänenserver

Element A	Port	Richtung	Element B	Konfigurationseigenschaft
Datenbankserver	5000	←	Domänenserver	

Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und dem Domänenserver

Tabelle 9. Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und dem Domänenserver

Element A	Port	Richtung	Element B	Konfigurationseigenschaft
Discovery Management Portal	9433	→	Domänenserver - Öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port
	9435	→	Domänenserver - ClientProxyServer	com.ibm.cdb.service.ClientProxyServer.port
	9434	→	Domänenserver - SecureClientProxyServer	com.ibm.cdb.service.SecureClientProxyServer.secure.port
API-Clients	9433	→	Domänenserver - Öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port
	9530	→	Domänenserver - API-Server	com.ibm.cdb.service.ApiServer.port
	9531	→	Domänenserver - Sicherer API-Server	com.ibm.cdb.service.SecureApiServer.secure.port
Webportal- und Datenmanagementportal-Clients	9430	→	Domänenserver - Web	com.ibm.cdb.service.web.port
	9431	→	Domänenserver - Sicheres Web	com.ibm.cdb.service.web.secure.port

Kommunikation zwischen dem Anker und Gateway und dem Domänenserver

Tabelle 10. Kommunikation zwischen dem Anker und Gateway und dem Domänenserver

Element A	Port	Richtung	Element B	Konfigurationseigenschaft
Anker (im ssh-Modus) - SSH	22	←	Domänenserver (im ssh-Modus)	
Anker (im Direktmodus) - SSH		←	Domänenserver (im Direktmodus)	
Anker (im ssh-Modus) - SSH-Tunnelweiterleitung	8497	↔	Domänenserver (im ssh-Modus)	
Anker (im Direktmodus) - direct		←	Domänenserver (im Direktmodus)	

Tabelle 10. Kommunikation zwischen dem Anker und Gateway und dem Domänenserver (Forts.)				
Element A	**Port**	**Rich-tung**	**Element B**	**Konfigurationseigenschaft**
Gateway - SSH	22	←	Domänenserver	

Lokale Kommunikation

Tabelle 11. Kommunikationskonfiguration der lokalen Konnektivität für einen Domänenserver		
Lokale Kommunikation	**Richtung**	**Konfigurationseigenschaft**
Domänenserver - Lokale Service-Registry	↔	com.ibm.cdb.local.hostname
Domänenserver - Lokale Services		
Domänenserver - 127.0.0.1		

Konfiguration von Firewalls in einer Streaming-Server-Implementierung
Die Firewalls in einer Streaming-Server-Implementierung müssen so konfiguriert werden, dass bestimmte Ports für die Kommunikation geöffnet sind.

In der folgenden Abbildung ist die TADDM-Kommunikation in einer Streaming-Server-Implementierung dargestellt.

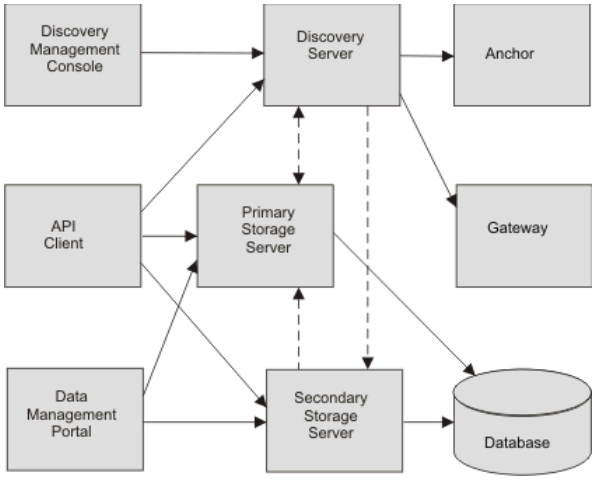

Abbildung 2. TADDM-Kommunikation in einer Streaming-Server-Implementierung

Konnektivitätsservices
Für eine Streaming-Server-Implementierung können Sie öffentliche und lokale Konnektivitätsservices sowie Inter-Server-Konnektivitätsservices konfigurieren.

Wichtig: Die Standardports der weiter unten in diesem Abschnitt angegebenen Eigenschaften gelten nur für diejenigen Eigenschaften, die in der Datei `collation.properties` aufgelistet sind. Ist eine Eigenschaft in dieser Datei nicht festgelegt bzw. auskommentiert, so gilt für sie ein Zufallsport. Zur Sicherstellung eines erfolgreichen Systemstarts sollten Sie daher insbesondere darauf achten, dass die Eigenschaft `com.ibm.cdb.service.RegistriesURLProvider.port` in der Datei `collation.properties` definiert ist.

Öffentliche Konnektivitätsservices

In der folgenden Tabelle sind die Standardhosteinstellungen für die öffentlichen Konnektivitätsservices des primären und des sekundären Speicherservers sowie des Erkennungsservers aufgeführt.

Tabelle 12. Standardhosteinstellungen für die öffentlichen Konnektivitätsservices des primären und des sekundären Speicherservers sowie des Erkennungsservers

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für öffentlichen Service	com.ibm.cdb.public.hostname	Definiert durch com.ibm.cdb.global.hostname

In der folgenden Tabelle sind die Standardporteinstellungen für die öffentlichen Konnektivitätsservices des primären und des sekundären Speicherservers sowie des Erkennungsservers aufgeführt.

Tabelle 13. Standardporteinstellungen für die öffentlichen Konnektivitätsservices des primären und des sekundären Speicherservers sowie des Erkennungsservers

Name	Konfigurationseigenschaft	Protokoll	Standardport
API-Server-Port	com.ibm.cdb.service.ApiServer.port	TCP	9530
Sicherer API-Server-Port	com.ibm.cdb.service.SecureApiServer.secure.port	TCP	9531
HTTP-Port (ohne SSL)	com.ibm.cdb.service.web.port	TCP	9430
HTTPS-Port (mit SSL)	com.ibm.cdb.service.web.secure.port	TCP	9431
Kommunikationsport für GUI-Server	com.ibm.cdb.service.ClientProxyServer.port	TCP	9435
SSL-Kommunikationsport für GUI-Server	com.ibm.cdb.service.SecureClientProxyServer.secure.port	TCP	9434
Port für öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port	TCP	9433

Inter-Server-Konnektivitätsservices

In der folgenden Tabelle sind die Standardhosteinstellungen für die Inter-Server-Konnektivitätsservices des primären und des sekundären Speicherservers aufgeführt.

Tabelle 14. Standardhosteinstellungen für die Inter-Server-Konnektivitätsservices des primären und des sekundären Speicherservers

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für Inter-Server-Service	com.ibm.cdb.interserver.hostname	Definiert durch com.ibm.cdb.global.hostname

In der folgenden Tabelle sind die Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des primären Speicherservers aufgeführt.

Tabelle 15. Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des primären Speicherservers

Name	Konfigurationseigenschaft	Protokoll	Standardport
Port für Topologiemanager	com.ibm.cdb.service.TopologyManager.port	TCP	9550
Port für Sicherheitsmanager	com.ibm.cdb.service.SecurityManager.port	TCP	9540
Port für Registry-URL-Provider	com.ibm.cdb.service.RegistriesURLProvider.port	TCP	9560

Tabelle 15. Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des primären Speicherservers (Forts.)

Name	Konfigurationseigenschaft	Protokoll	Standard-port
Port für Inter-Server-Service-Registry	com.ibm.cdb.service.registry.interserver.port	TCP	4160

In der folgenden Tabelle sind die Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des sekundären Speicherservers aufgeführt.

Tabelle 16. Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des sekundären Speicherservers

Name	Konfigurationseigenschaft	Protokoll	Standard-port
Port für Topologiemanager	com.ibm.cdb.service.TopologyManager.port	TCP	9550
Port für Registry-URL-Provider	com.ibm.cdb.service.RegistriesURLProvider.port	TCP	9560
Port für Inter-Server-Service-Registry	com.ibm.cdb.service.registry.interserver.port	TCP	4160

Lokale Konnektivitätsservices

Die Ports für die lokalen Services werden nicht explizit festgelegt. An der Schnittstelle, die für lokale Services definiert ist, müssen alle Ports geöffnet sein. Standardschnittstelle ist die Loopback-Schnittstelle.

In der folgenden Tabelle sind die Standardhosteinstellungen für die lokalen Konnektivitätsservices des primären und des sekundären Speicherservers sowie des Erkennungsservers aufgeführt.

Tabelle 17. Standardhosteinstellungen für die lokalen Konnektivitätsservices des primären und des sekundären Speicherservers sowie des Erkennungsservers

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für Services des lokalen Konnektivitätsbereichs	com.ibm.cdb.local.hostname	127.0.0.1

Kommunikation in der Streaming-Server-Implementierung konfigurieren

Für eine erfolgreiche Kommunikation in der Streaming-Server-Implementierung müssen öffentliche, Inter-Server- und lokale Konnektivitätsservices konfiguriert werden.

Die folgende Tabelle zeigt die Elemente, die Sie in der Streaming-Server-Implementierung verbinden können, und die Ports, die Sie für eine erfolgreiche Kommunikation öffnen müssen.

Inter-Server-Kommunikation

Tabelle 18. Kommunikationskonfiguration der Inter-Server-Konnektivität in der Streaming-Server-Implementierung

Element A	Port	Richtung	Element B	Konfigurationseigenschaft	TLS-Unterstützung
Erkennungsserver			**Primärer Speicherserver**		
	9433	→	Primärer Speicherserver	com.ibm.cdb.service.registry.public.port	Ja
	4160	→	Primärer Speicherserver - Inter-Server-Service-Registry	com.ibm.cdb.service.registry.interserver.port	Nein
	9560	→	Primärer Speicherserver - Registrys URLProvider	com.ibm.cdb.service.Registries URLProvider.port	Ja
	9540	→	Primärer Speicherserver - SecurityManager	com.ibm.cdb.service.Security Manager.port	Ja
	9550	→	Primärer Speicherserver - TopologyManager	com.ibm.cdb.service.Topology Manager.port	Ja
	9430	←	Primärer Speicherserver - Web	com.ibm.cdb.service.web.port	Nein
Erkennungsserver			**Sekundärer Speicherserver**		
	4160	→	Sekundärer Speicherserver - Inter-Server-Service-Registry	com.ibm.cdb.service.registry.interserver.port	Nein
	9560	→	Sekundärer Speicherserver - Registrys URLProvider	com.ibm.cdb.service.Registries URLProvider.port	Ja
	9550	→	Sekundärer Speicherserver - TopologyManager	com.ibm.cdb.service.Topology Manager.port	Ja

Tabelle 18. Kommunikationskonfiguration der Inter-Server-Konnektivität in der Streaming-Server-Implementierung (Forts.)

Element A	Port	Richtung	Element B	Konfigurationseigenschaft	TLS-Unterstützung
Sekundärer Speicherserver			**Primärer Speicherserver**		
	4160	→	Primärer Speicherserver - Inter-Server-Service-Registry	com.ibm.cdb.service.registry.interserver.port	Nein
	9560	→	Primärer Speicherserver - Registrys URLProvider	com.ibm.cdb.service.Registries URLProvider.port	Ja
	9540	→	Primärer Speicherserver - SecurityManager	com.ibm.cdb.service.Security Manager.port	Ja
	9550	→	Primärer Speicherserver - TopologyManager	com.ibm.cdb.service.Topology Manager.port	Ja
Datenbankserver	5000	←	**Primärer Speicherserver**		Nein
Datenbankserver	5000	←	**Sekundärer Speicherserver**		Nein

Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und den TADDM-Servern

Tabelle 19. Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und den TADDM-Servern

Element A	Port	Richtung	Element B	Konfigurationseigenschaft	TLS-Unterstützung
Discovery Management Portal	9433	→	Erkennungsserver - Öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port	Ja
	9435	→	Erkennungsserver - ClientProxyServer	com.ibm.cdb.service.ClientProxyServer.port	Nein
	9434	→	Erkennungsserver - SecureClient ProxyServer	com.ibm.cdb.service.SecureClientProxy Server.secure.port	Ja

Tabelle 19. Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und den TADDM-Servern (Forts.)

Element A	Port	Richtung	Element B	Konfigurationseigenschaft	TLS-Unterstützung
API-Clients	9433	→	• Erkennungsserver - Öffentliche Service-Registry • Primärer Speicherserver - Öffentliche Service-Registry • Sekundärer Speicherserver - Öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port	Ja
	9530	→	• Erkennungsserver - API-Server • Primärer Speicherserver - API-Server • Sekundärer Speicherserver - API-Server	com.ibm.cdb.service.ApiServer.port	Nein
	9531	→	• Erkennungsserver - Sicherer API-Server • Primärer Speicherserver - Sicherer API-Server • Sekundärer Speicherserver - Sicherer API-Server	com.ibm.cdb.service.SecureApiServer.secure.port	Ja

Tabelle 19. Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und den TADDM-Servern (Forts.)					
Element A	**Port**	**Richtung**	**Element B**	**Konfigurationseigenschaft**	**TLS-Unterstützung**
Webportal- und Datenmanagementportal-Clients	9430	→	• Erkennungsserver - Web • Primärer Speicherserver - Web • Sekundärer Speicherserver - Web	com.ibm.cdb.service.web.port	Nein
	9431	→	• Erkennungsserver - Sicheres Web • Primärer Speicherserver - Sicheres Web • Sekundärer Speicherserver - Sicheres Web	com.ibm.cdb.service.web.secure.port	Ja

Kommunikation zwischen dem Anker und Gateway und dem Erkennungsserver

Tabelle 20. Kommunikation zwischen dem Anker und Gateway und dem Erkennungsserver				
Element A	**Port**	**Richtung**	**Element B**	**Konfigurationseigenschaft**
Anker (im ssh-Modus) - SSH	22	←	Erkennungsserver (im ssh-Modus)	
Anker (im Direktmodus) - SSH		←	Erkennungsserver (im Direktmodus)	
Anker (im ssh-Modus) - SSH-Tunnelweiterleitung	8497	↔	Erkennungsserver (im ssh-Modus)	
Anker (im Direktmodus) - direct		←	Erkennungsserver (im Direktmodus)	
Gateway - SSH	22	←	Erkennungsserver	

Lokale Kommunikation

Tabelle 21. Kommunikationskonfiguration der lokalen Konnektivität in der Streaming-Server-Implementierung		
Lokale Kommunikation	**Richtung**	**Konfigurationseigenschaft**
Erkennungsserver		

Tabelle 21. Kommunikationskonfiguration der lokalen Konnektivität in der Streaming-Server-Implementierung (Forts.)

Lokale Kommunikation	**Richtung**	**Konfigurationseigenschaft**
Erkennungsserver - Lokale Service-Registry	⟷	com.ibm.cdb.local.hostname
Erkennungsserver - Lokale Services		
Erkennungsserver - 127.0.0.1		
Primärer Speicherserver		
Primärer Speicherserver - Lokale Service-Registry	⟷	com.ibm.cdb.local.hostname
Primärer Speicherserver - Lokale Services		
Primärer Speicherserver - 127.0.0.1		
Sekundärer Speicherserver		
Sekundärer Speicherserver - Lokale Service-Registry	⟷	com.ibm.cdb.local.hostname
Sekundärer Speicherserver - Lokale Services		
Sekundärer Speicherserver - 127.0.0.1		

Konfiguration von Firewalls in einer Synchronisationsserverimplementierung
Die Firewalls in einer Synchronisationsserverimplementierung müssen so konfiguriert werden, dass bestimmte Ports für die Kommunikation geöffnet sind.

In der folgenden Abbildung ist die TADDM-Kommunikation in einer Synchronisationsserverimplementierung dargestellt.

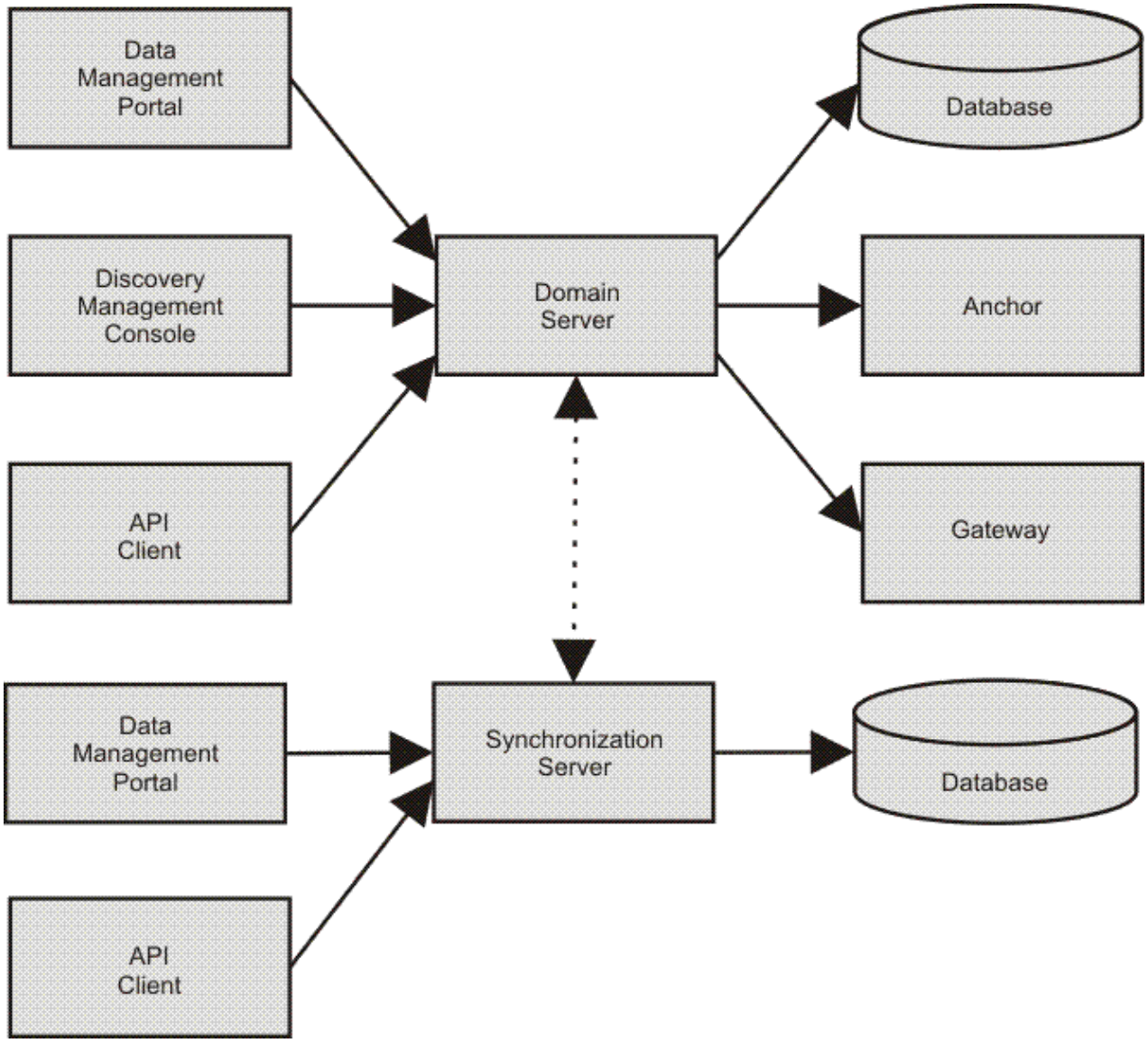

Abbildung 3. TADDM-Kommunikation in einer Synchronisationsserverimplementierung

Konnektivitätsservices konfigurieren

Für eine Synchronisationsserverimplementierung können Sie öffentliche und lokale Konnektivitätsservices sowie Inter-Server-Konnektivitätsservices konfigurieren.

Wichtig: Die Standardports der weiter unten in diesem Abschnitt angegebenen Eigenschaften gelten nur für diejenigen Eigenschaften, die in der Datei `collation.properties` aufgelistet sind. Ist eine Eigenschaft in dieser Datei nicht festgelegt bzw. auskommentiert, so gilt für sie ein Zufallsport. Zur Sicherstellung eines erfolgreichen Systemstarts sollten Sie daher insbesondere darauf achten, dass die Eigenschaft `com.ibm.cdb.service.RegistriesURLProvider.port` in der Datei `collation.properties` definiert ist.

Öffentliche Konnektivitätsservices

In der folgenden Tabelle sind die Standardhosteinstellungen für die öffentlichen Konnektivitätsservices des Domänen- und des Synchronisationsservers aufgeführt.

Tabelle 22. Standardhosteinstellungen für die öffentlichen Konnektivitätsservices des Domänen- und des Synchronisationsservers

Name	Konfigurationseigenschaft	Standardschnittstelle
Host für öffentlichen Service	com.ibm.cdb.public.hostname	Definiert durch com.ibm.cdb.global.hostname

In der folgenden Tabelle sind die Standardporteinstellungen für die öffentlichen Konnektivitätsservices des Domänenservers aufgeführt.

Tabelle 23. Standardhosteinstellungen für die öffentlichen Konnektivitätsservices des Domänenservers

Name	Konfigurationseigenschaft	Protokoll	Standardport
API-Server-Port	com.ibm.cdb.service.ApiServer.port	TCP	9530
Sicherer API-Server-Port	com.ibm.cdb.service.SecureApiServer.secure.port	TCP	9531
HTTP-Port (ohne SSL)	com.ibm.cdb.service.web.port	TCP	9430
HTTPS-Port (mit SSL)	com.ibm.cdb.service.web.secure.port	TCP	9431
Kommunikationsport für GUI-Server	com.ibm.cdb.service.ClientProxyServer.port	TCP	9435
SSL-Kommunikationsport für GUI-Server	com.ibm.cdb.service.SecureClientProxyServer.secure.port	TCP	9434
Port für öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port	TCP	9433

Folgende Tabelle zeigt die Standardporteinstellungen für die öffentlichen Konnektivitätsservices des Synchronisationsservers.

Tabelle 24. Standardporteinstellungen für die öffentlichen Konnektivitätsservices des Synchronisationsservers

Name	Konfigurationseigenschaft	Protokoll	Standardport
API-Server-Port	com.ibm.cdb.service.ApiServer.port	TCP	9530
Sicherer API-Server-Port	com.ibm.cdb.service.SecureApiServer.secure.port	TCP	9531
HTTP-Port (ohne SSL)	com.ibm.cdb.service.web.port	TCP	9430
HTTPS-Port (mit SSL)	com.ibm.cdb.service.web.secure.port	TCP	9431
Port für öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port	TCP	9433

Inter-Server-Konnektivitätsservices

In der folgenden Tabelle sind die Standardhosteinstellungen für die Inter-Server-Konnektivitätsservices des Domänen- und des Synchronisationsservers aufgeführt.

Tabelle 25. Standardhosteinstellungen für die Inter-Server-Konnektivitätsservices des Domänen- und des Synchronisationsservers		
Name	**Konfigurationseigenschaft**	**Standardschnittstelle**
Host für Inter-Server-Service	com.ibm.cdb.interserver.hostname	Definiert durch com.ibm.cdb.global.hostname

In der folgenden Tabelle sind die Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des Domänenservers aufgeführt.

Tabelle 26. Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des Domänenservers			
Name	**Konfigurationseigenschaft**	**Protokoll**	**Standardport**
Port für Topologiemanager	com.ibm.cdb.service.TopologyManager.port	TCP	9550
Port für Sicherheitsmanager	com.ibm.cdb.service.SecurityManager.port	TCP	9540
Port für Registry-URL-Provider	com.ibm.cdb.service.RegistriesURLProvider.port	TCP	9560
Port für Inter-Server-Service-Registry	com.ibm.cdb.service.registry.interserver.port	TCP	4160

In der folgenden Tabelle sind die Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des Synchronisationsservers aufgeführt.

Tabelle 27. Standardporteinstellungen für die Inter-Server-Konnektivitätsservices des Synchronisationsservers			
Name	**Konfigurationseigenschaft**	**Protokoll**	**Standardport**
Port für Registry-URL-Provider	com.ibm.cdb.service.RegistriesURLProvider.port	TCP	9560
Port für Unternehmenssicherheitsmanager	com.ibm.cdb.service.EnterpriseSecurityManager.port	TCP	9570
Port für Inter-Server-Service-Registry	com.ibm.cdb.service.registry.interserver.port	TCP	4160

Lokale Konnektivitätsservices

Die Ports für die lokalen Services werden nicht explizit festgelegt. An der Schnittstelle, die für lokale Services definiert ist, müssen alle Ports geöffnet sein. Standardschnittstelle ist die Loopback-Schnittstelle.

In der folgenden Tabelle sind die Standardhosteinstellungen für die lokalen Konnektivitätsservices des Domänen- und des Synchronisationsservers aufgeführt.

Tabelle 28. Standardhosteinstellungen für die lokalen Konnektivitätsservices des Domänen- und des Synchronisationsservers		
Name	**Konfigurationseigenschaft**	**Standardschnittstelle**
Host für lokalen Service	com.ibm.cdb.local.hostname	127.0.0.1

Kommunikation in der Synchronisationsserverimplementierung konfigurieren

Für eine erfolgreiche Kommunikation in der Synchronisationsserverimplementierung müssen öffentliche, Inter-Server- und lokale Konnektivitätsservices konfiguriert werden.

Die folgende Tabelle zeigt die Elemente, die Sie in der Synchronisationsserverimplementierung verbinden können, und die Ports, die Sie für eine erfolgreiche Kommunikation öffnen müssen.

Inter-Server-Kommunikation

Tabelle 29. Kommunikationskonfiguration der Inter-Server-Konnektivität in der Synchronisationsserverimplementierung

Element A	Port	Richtung	Element B	Konfigurationseigenschaft
Domänenserver			**Synchronisationsserver**	
	4160	→	Synchronisationsserver - InterServer-Service-Registry	com.ibm.cdb.service.registry.interserver.port
	9560	→	Synchronisationsserver - RegistriesURLProvider	com.ibm.cdb.service.RegistriesURLProvider.port
	9570	→	Synchronisationsserver - EnterpriseSecurityManager	com.ibm.cdb.service.EnterpriseSecurityManager.port
Domänenserver			**Synchronisationsserver**	
Domänenserver - Inter-Server-Service-Registry	4160	←	Synchronisationsserver	com.ibm.cdb.service.registry.interserver.port
Domänenserver - Registries URLProvider	9560	←		com.ibm.cdb.service.RegistriesURLProvider.port
Domänenserver - Security Manager	9540	←		com.ibm.cdb.service.SecurityManager.port
Domänenserver - Topology Manager	9550	←		com.ibm.cdb.service.TopologyManager.port
Datenbankserver	5000	←	**Domänenserver**	
Datenbankserver	5000	←	**Synchronisationsserver**	

Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und den Domänen- und Synchronisationsservern

Tabelle 30. Kommunikation zwischen Discovery Management Portal, API-Clients sowie Webportal- und Datenmanagementportal-Clients und den Domänen- und Synchronisationsservern

Element A	Port	Richtung	Element B	Konfigurationseigenschaft
Discovery Management Portal	9433	→	Domänenserver - Öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port
	9435	→	Domänenserver - ClientProxyServer	com.ibm.cdb.service.ClientProxyServer.port
	9434	→	Domänenserver - SecureClientProxyServer	com.ibm.cdb.service.SecureClientProxyServer.secure.port
API-Clients	9433	→	• Domänenserver - Öffentliche Service-Registry • Synchronisationsserver - Öffentliche Service-Registry	com.ibm.cdb.service.registry.public.port
	9530	→	• Domänenserver - API-Server • Synchronisationsserver - API-Server	com.ibm.cdb.service.ApiServer.port
	9531	→	• Domänenserver - Sicherer API-Server • Synchronisationsserver - Sicherer API-Server	com.ibm.cdb.service.SecureApiServer.secure.port
Webportal- und Datenmanagementportal-Clients	9430	→	• Domänenserver - Web • Synchronisationsserver - Web	com.ibm.cdb.service.web.port
	9431	→	• Domänenserver - Sicheres Web • Synchronisationsserver - Sicheres Web	com.ibm.cdb.service.web.secure.port

Kommunikation zwischen dem Anker und Gateway und dem Domänenserver

Tabelle 31. Kommunikation zwischen dem Anker und Gateway und dem Domänenserver

Element A	Port	Richtung	Element B	Konfigurationseigenschaft
Anker (im ssh-Modus) - SSH	22	←	Domänenserver (im ssh-Modus)	
Anker (im Direktmodus) - SSH		←	Domänenserver (im Direktmodus)	

Tabelle 31. Kommunikation zwischen dem Anker und Gateway und dem Domänenserver (Forts.)				
Element A	**Port**	**Rich-tung**	**Element B**	**Konfigurationseigenschaft**
Anker (im ssh-Modus) - SSH-Tunnel-weiterleitung	8497	⟷	Domänenserver (im ssh-Modus)	
Anker (im Direktmodus) - direct		⟵	Domänenserver (im Direktmodus)	
Gateway - SSH	22	⟵	Domänenserver	

Lokale Kommunikation

Tabelle 32. Kommunikationskonfiguration der lokalen Konnektivität in der Synchronisationsserverimplementierung		
Lokale Kommunikation	**Richtung**	**Konfigurationseigenschaft**
Domänenserver		
Domänenserver - Lokale Service-Registry	⟷	com.ibm.cdb.local.hostname
Domänenserver - Lokale Services		
Domänenserver - 127.0.0.1		
Synchronisationsserver		
Synchronisationsserver - Lokale Service-Registry	⟷	com.ibm.cdb.local.hostname
Synchronisationsserver - Lokale Services		
Synchronisationsserver - 127.0.0.1		

Referenzinformationen zu TADDM-Servereigenschaften

Die Datei `collation.properties` enthält Eigenschaften für den TADDM-Server. Sie können einige dieser Eigenschaften bearbeiten.

Die Datei `collation.properties` befindet sich im Verzeichnis `$COLLATION_HOME/etc`. Die Datei enthält Begleittext zu allen Eigenschaften.

Wenn Sie die Datei `collation.properties` aktualisieren, speichern Sie die Datei und starten Sie den Server erneut, damit die Änderungen wirksam werden.

Bereichsorientierte und nicht bereichsorientierte Eigenschaften

Die Datei `collation.properties` enthält zwei Arten von Eigenschaften: bereichsorientierte und nicht bereichsorientierte.

Bereichsorientierte Eigenschaft
Eine Eigenschaft, an die Sie eine IP-Adresse oder den Namen einer Bereichsgruppe anhängen können. Die IP-Adresse oder der Name der Bereichsgruppe bestimmen die Abhängigkeit der Eigenschaft vom Host, der erkannt wird. Die Namen von Bereichsgruppen dürfen keine Leerzeichen, Hochkommas ('), Punkte (.) und Schrägstriche (/) enthalten.

Nicht bereichsorientierte Eigenschaft
Eine Eigenschaft, die Sie nicht auf ein bestimmtes Objekt beschränken können.

Bei den folgenden Eigenschaften handelt es sich beispielsweise um nicht bereichsorientierte Eigenschaften:

- `com.collation.log.filesize`
- `com.collation.discover.agent.command.lsof.Linux`

Die Eigenschaft `com.collation.discover.agent.command.lsof.Linux` kann jedoch eine bereichsorientierte Eigenschaft sein, wenn Sie eine IP-Adresse oder den Namen einer Bereichsgruppe an die Eigenschaft anhängen, wie in den folgenden Beispielen dargestellt:

- Beispiel für das Anhängen der IP-Adresse `129.42.56.212`:

```
com.collation.discover.agent.command.lsof.Linux.129.42.56.212=sudo lsof
```

- Beispiel für das Anhängen einer Bereichsgruppe namens "scope1":

```
com.collation.discover.agent.command.lsof.Linux.scope1=sudo lsof
```

Eigenschaften, die nicht geändert werden dürfen
Die Datei `collation.properties` enthält einige Eigenschaften, bei deren Änderung das System unter Umständen funktionsunfähig wird.

Die folgenden Eigenschaften dürfen nicht geändert werden:

com.collation.version
Gibt die Produktversion an.

com.collation.branch
Gibt den Codezweig an.

com.collation.buildnumber
Gibt die Buildnummer an. Diese Nummer wird durch den Erstellungsprozess festgelegt.

com.collation.oalbuildnumber
Gibt die Buildnummer für einen anderen Erstellungsprozess an.

com.collation.SshWeirdReauthErrorList=`Permission denied`
Diese Eigenschaft muss auf `Permission denied` (Berechtigung verweigert) gesetzt sein.

Diese Eigenschaft ist erforderlich, da Windows-Systeme gültige Anmeldeversuche beliebig ablehnen können. Sie können versuchen, die Benutzer/Kennwort-Paare zu verwenden, die bei vorherigen Erkennungsläufen erfolgreich waren.

Caching-Eigenschaften für Zugriffsberechtigungsnachweise
Diese Eigenschaften gelten für das Caching von Zugriffsberechtigungsnachweisen.

com.ibm.cdb.security.auth.cache.disabled=`false`
Der Standardwert ist `false`.

Diese Eigenschaft legt fest, ob das Caching für Berechtigungsnachweise inaktiviert ist.

Bei dieser Eigenschaft handelt es sich um eine bereichs- und profilorientierte Eigenschaft. Sie können eine IP-Adresse, den Namen einer Bereichsgruppe oder einen Profilnamen anhängen. Darüber hinaus können Sie die Eigenschaft in der Profilkonfiguration der Discovery Management Console festlegen.

com.ibm.cdb.security.auth.cache.fallback.failed=`true`
Der Standardwert ist `true`.

Diese Eigenschaft aktiviert den Rückgriff, wenn ein Cache gültige Berechtigungsnachweise enthält, beim Abruf jedoch keine Validierung vornehmen kann. Wenn der Rückgriff aktiviert ist und die zwischengespeicherten Berechtigungsnachweise nicht mehr gültig sind, durchläuft der Cache alle verfügbaren Zugriffseintragstypen, bis eine Übereinstimmung gefunden wird.

Bei dieser Eigenschaft handelt es sich um eine bereichs- und profilorientierte Eigenschaft. Sie können eine IP-Adresse, den Namen einer Bereichsgruppe oder einen Profilnamen anhängen.

Bei den folgenden Einträgen handelt es sich um Beispiele für Einträge, die in der Datei `collation.properties` enthalten sein können:

```
com.ibm.cdb.security.auth.cache.fallback.failed=false
com.ibm.cdb.security.auth.cache.fallback.failed.10.160.160.11=true
com.ibm.cdb.security.auth.cache.fallback.failed.ScopeA=true
com.ibm.cdb.security.auth.cache.fallback.failed.GroupA=true
com.ibm.cdb.security.auth.cache.fallback.failed.Level_2_Discovery=false
```

Sie können diese Eigenschaft auch in der Discovery Management Console in der Profilkonfiguration auf der Registerkarte **Plattformeigenschaften** festlegen.

com.ibm.cdb.security.auth.cache.fallback.invalid=`true`
Der Standardwert ist `true`.

Diese Eigenschaft aktiviert den Rückgriff, wenn ein aus dem Cache gelesener Eintrag einen ungültigen Versuch enthält (der letzte Zugriff ist fehlgeschlagen, da keine gültigen Berechtigungsnachweise gefunden wurden). Wenn der Rückgriff aktiviert ist, durchläuft der Cache alle verfügbaren Zugriffseintragstypen, bis eine Übereinstimmung gefunden wird.

Es handelt sich um eine bereichs- und profilorientierte Eigenschaft, an die eine IP-Adresse, der Name einer Bereichsgruppe oder ein Profilname angehängt werden können.

Bei den folgenden Einträgen handelt es sich um Beispiele für Einträge, die in der Datei `collation.properties` enthalten sein können:

```
com.ibm.cdb.security.auth.cache.fallback.invalid=false
com.ibm.cdb.security.auth.cache.fallback.invalid.10.160.160.11=true
com.ibm.cdb.security.auth.cache.fallback.invalid.ScopeA=true
com.ibm.cdb.security.auth.cache.fallback.invalid.GroupA=true
com.ibm.cdb.security.auth.cache.fallback.invalid.Level_2_Discovery=false
```

Sie können diese Eigenschaft auch in der Discovery Management Console in der Profilkonfiguration auf der Registerkarte **Plattformeigenschaften** festlegen.

Fix Pack 5 **com.ibm.cdb.security.auth.cache.itm.disabled=`true`**
Der Standardwert ist `true`.

Diese Eigenschaft legt fest, ob das Caching für Berechtigungsnachweise für die OSLC-Erkennung aktiviert ist.

Bei dieser Eigenschaft handelt es sich um eine bereichs- und profilorientierte Eigenschaft. Sie können eine IP-Adresse, den Namen einer Bereichsgruppe oder einen Profilnamen anhängen. Darüber hinaus können Sie die Eigenschaft in der Profilkonfiguration der Discovery Management Console festlegen.

Zugehörige Konzepte

„Caching der letzten erfolgreichen Berechtigungsnachweise” auf Seite 13
TADDM kann die letzten gültigen Zugriffsberechtigungsnachweise zwischenspeichern. Diese können in der nächsten Erkennung (Ebene 2 bzw. scriptbasierte Erkennung) wiederverwendet werden.

API-Porteigenschaften
Diese Eigenschaften gelten für API-Ports.

com.ibm.cdb.service.ApiServer.port=9530
Der Standardwert ist 9530. Hier muss ein ganzzahliger Wert angegeben werden.

Diese Eigenschaft gibt den Port an, der vom API-Server auf Nicht-SSL-Anforderungen überwacht wird. Der Wert kann auf jeden beliebigen Port auf dem Server gesetzt werden. Für alle Clients, die Verbindungen über das API herstellen, muss für Nicht-SSL-Verbindungen dieser Port angegeben werden.

com.ibm.cdb.service.SecureApiServer.secure.port=9531
Der Standardwert ist 9531. Hier muss ein ganzzahliger Wert angegeben werden.

Diese Eigenschaft gibt den Port an, der vom API-Server auf SSL-Anforderungen überwacht wird. Der Wert kann auf jeden beliebigen Port auf dem Server gesetzt werden. Für alle Clients, die Verbindungen über das API herstellen, muss für SSL-Verbindungen dieser Port angegeben werden.

Agenteneigenschaften bereinigen

Die Bereinigungsagenten entfernen alle verwaisten Aliasnamen und Konfigurationselemente oder korrigieren die fehlenden Zeilen in den Tabellen. Die meisten der Agenten lesen Eigenschaften, die in der Datei `collation.properties` definiert sind.

AliasesCleanupAgent

Der Agent entfernt aus der Tabelle ALIASES die Aliasnamen, die nicht mehr mit den Namensattributen des Konfigurationselements übereinstimmen. Er entfernt auch die Aliasnamen und Zeilen aus der Tabelle PERSOBJ, die über keine zugehörigen Konfigurationselemente verfügen. Der Agent liest die folgenden Eigenschaften aus der Datei `collation.properties`:

Fix Pack 2 **com.ibm.cdb.topomgr.topobuilder.deleteAliasesWithoutMaster**

Der Standardwert ist `true`.

Die Eigenschaft gibt an, ob die Aliasnamen,die nicht über den entsprechenden Masteraliasnamen verfügen, aus der Tabelle ALIASES gelöscht werden. Standardmäßig ist der Löschvorgang aktiviert.

com.ibm.cdb.topomgr.topobuilder.max.row.fetch

Der Standardwert ist `1000`.

Die Eigenschaft konfiguriert die Stapelgröße, mit der Aliasnamen aus der Tabelle ALIASES abgerufen werden.

Wenn Sie die Eigenschaft auf `-1` setzen, überprüft der Agent keine Aliasnamen.

com.ibm.cdb.topomgr.topobuilder.max.row.delete

Der Standardwert ist `5000`.

Die Eigenschaft konfiguriert die Stapelgröße, die zum Löschen von Aliasnamen verwendet wird.

Wenn Sie die Eigenschaft auf `-1` setzen, entfernt der Agent keine Aliasnamen, sondern meldet nur die beschädigten Namen.

com.ibm.cdb.topomgr.topobuilder.agents.AliasesCleanupAgent.maxNumber OfMastersToScan

Der Standardwert ist `1000`.

Die Eigenschaft konfiguriert die Anzahl der Konfigurationselemente, für die die Prüfung von Aliasnamen während einer einzelnen Ausführung des Agenten erforderlich ist.

com.ibm.cdb.topomgr.topobuilder.cleanupOrphanedAliasesAndPersobj

Der Standardwert ist `true`. Der Agent führt die Bereinigung aus.

Die Eigenschaft aktiviert oder deaktiviert die Bereinigung der Aliasnamen in der Tabelle ALIASES und der GUIDs in der Tabelle PERSOBJ, die über keine zugehörigen Konfigurationselemente verfügen.

com.ibm.cdb.topomgr.topobuilder.DelayToRemoveAliases

Der Standardwert ist 12 (Stunden). Die verwaisten Aliasnamen, die älter als 12 Stunden sind, werden vom Agenten entfernt.

Die Eigenschaft definiert die Zeit in Stunden, nach denen die Aliasnamen ohne ein zugehöriges Konfigurationselement vom Agenten entfernt werden. Es werden neue Aliasnamen geschützt, die möglicherweise über kein zugehöriges Konfigurationselement verfügen, da das Speichern des Konfigurationselements nicht beendet ist.

Diese Eigenschaft ist mit Vorsicht zu verwenden. Legen Sie keinen kleineren Wert fest.

AliasesJnTableCleanupAgent

Dieser Agent entfernt alte Zeilen aus der Tabelle ALIASES_JN. Diese Tabelle enthält den Verlauf der Änderungen an der Tabelle ALIASES. Sie wird verwendet, um potenzielles übergeordnetes Zusammenführen von Konfigurationselementen in der Datenbank zu finden. Der Agent liest die folgenden Eigenschaften aus der Datei `collation.properties`:

Fix Pack 2 **com.ibm.cdb.topomgr.topobuilder.agents.AliasesJnTableCleanupAgent.maxRow**
Der Standardwert ist `5000`. Der Standardwert sollte nicht geändert werden.

Diese Eigenschaft gibt die maximale Anzahl an Zeilen an, die gleichzeitig vom Agent gelöscht wird.

com.ibm.cdb.topomgr.topobuilder.agents.AliasesJnTableCleanupAgent.remove OlderThanDays
Der Standardwert ist `30` (Tage).

Diese Eigenschaft entfernt Zeilen, die älter als die angegebene Zeit sind. Standardmäßig werden Zeilen entfernt, die älter als 30 Tage sind.

Wenn Sie diese Eigenschaft auf `0` oder einen niedrigeren Wert setzen, ist der Agent inaktiviert.

com.ibm.cdb.topomgr.topobuilder.agents.AliasesJnTableCleanupAgent.timeout
Der Standardwert ist `1800` (Sekunden).

Diese Eigenschaft gibt die Zeit an, nach der der Agent das Zeitlimit überschritten hat. Wenn die angegebene Zeit nicht ausreicht, um alle alten Zeilen zu löschen, versucht der Agent, sie bei der nächsten Ausführung zu löschen.

DependencyCleanupAgent

Der Agent entfernt die ruhenden Relationship-Objekte. Der Agent liest die folgenden Eigenschaften aus der Datei `collation.properties`:

com.ibm.cdb.topomgr.topobuilder.agents.DependencyCleanupAgent.timeout
Der Standardwert ist `600` (Sekunden). Nach dieser Zeit entfernt der Agent keine Objekte mehr, selbst wenn noch welche übrig sind.

com.ibm.cdb.topomgr.topobuilder.agents.DependencyCleanupAgent. removeOlderThanDays
Der Standardwert ist `90` (Tage). Ältere Relationship-Objekte werden als ruhende Objekte behandelt.

ObjectsWithoutAliasesCleanupAgent

Der Agent entfernt die Konfigurationselemente, die nicht über die Aliasnamen aus der Tabelle ALIASES verfügen. Der Agent liest die folgenden Eigenschaften aus der Datei `collation.properties`:

com.ibm.cdb.topomgr.topobuilder.agents.ObjectsWithoutAliasesCleanupAgent. maxToRemove
Der Standardwert ist `1000`.

Die Eigenschaft begrenzt die Anzahl der Konfigurationselemente, die der Agent während einer Ausführung entfernt. Wenn Sie die Eigenschaft auf `-1` setzen, wird der Agent beendet, ohne eine Bereinigung durchzuführen, und es wird die Nachricht `ObjectsWithoutAliasesCleanupAgent is disabled` angezeigt.

PersobjCleanupAgent

Der Agent korrigiert alle fehlenden Zeilen in der Tabelle PERSOBJ. In der Datei `collation.properties` wird keine Konfiguration vorgenommen. Der Agent zeigt in der Zusammenfassung an, wie viele Zeilen korrigiert wurden, wie aus dem folgenden Beispiel ersichtlich wird:

```
2012-08-22 18:12:21,500 TopologyBuilder [TopologyBuilderEngineThread$Cleanup@4.0]
INFO agents.PersobjCleanupAgent - Fixed 10 rows in PERSOBJ table
```

StorageExtentCleanupAgent

Der Agent entfernt die ruhenden StorageExtent-Objekte. Der Agent liest die folgenden Eigenschaften aus der Datei `collation.properties`:

com.ibm.cdb.topomgr.topobuilder.agents.StorageExtentCleanupAgent.timeout
Der Standardwert ist `1800` (Sekunden). Nach dieser Zeit entfernt der Agent keine Objekte mehr, selbst wenn noch welche übrig sind.

com.ibm.cdb.topomgr.topobuilder.agents.StorageExtentCleanupAgent.removeOlderThanDays
Der Standardwert ist 1 (Tag). StorageExtent-Objekte, die über einen Tag älter als ihre übergeordneten ComputerSystem-Objekte sind, werden als ruhende Objekte behandelt.

VlanInterfaceCleanupAgent

Der Agent entfernt die ruhenden SVlanInterface-Objekte. Der Agent liest die folgenden Eigenschaften aus der Datei `collation.properties`:

com.ibm.cdb.topomgr.topobuilder.agents.VlanInterfaceCleanupAgent.timeout
Der Standardwert ist `1800` (Sekunden). Nach dieser Zeit entfernt der Agent keine Objekte mehr, selbst wenn noch welche übrig sind.

com.ibm.cdb.topomgr.topobuilder.agents.VlanInterfaceCleanupAgent.removeOlderThanDays
Der Standardwert ist 1 (Tag). VlanInterface-Objekte, die über einen Tag älter als ihre übergeordneten Vlan-Objekte sind, werden als ruhende Objekte behandelt.

Befehle, die höhere Berechtigungen erfordern können
Diese Eigenschaften geben die von TADDM verwendeten Betriebssystembefehle an, für deren Ausführung auf dem Zielsystem eine höhere Berechtigungsstufe wie 'root' (oder 'superuser') erforderlich ist.

Für gewöhnlich wird 'sudo' auf UNIX- und Linux-Systemen verwendet, um eine Berechtigungserhöhung bereitzustellen. Statt sudo können Sie auch folgende Alternativlösungen nutzen:

- Aktivieren Sie die Zugriffsberechtigung 'setuid' für das ausführbare Zielprogramm.
- Fügen Sie der Gruppe, die dem ausführbaren Zielprogramm zugeordnet ist, den Erkennungsserviceaccount hinzu.
- Verwenden Sie 'root' für den Erkennungsserviceaccount (diese Methode sollte nicht die erste Wahl sein)

Für jede Eigenschaft kann sudo global konfiguriert werden, was bedeutet, dass der Befehl mit sudo auf allen Betriebssystemzielen ausgeführt wird oder auf eine bestimmte IP-Adresse oder Bereichsgruppe beschränkt wird.

Wichtig: 'sudo' muss auf allen Zielsystemen, für die eine Berechtigungserhöhung erforderlich ist, mit der Option 'NOPASSWD' konfiguriert werden. Andernfalls blockiert die Erkennung, bis sudo aufgrund einer Zeitlimitüberschreitung beendet wird.

com.collation.discover.agent.command.hastatus.Linux=sudo /opt/VRTSvcs/bin/hastatus
com.collation.discover.agent.command.haclus.Linux=sudo /opt/VRTSvcs/bin/haclus
com.collation.discover.agent.command.hasys.Linux=sudo /opt/VRTSvcs/bin/hasys
com.collation.discover.agent.command.hares.Linux=sudo /opt/VRTSvcs/bin/hares
com.collation.discover.agent.command.hagrp.Linux=sudo /opt/VRTSvcs/bin/hagrp
com.collation.discover.agent.command.hatype.Linux=sudo /opt/VRTSvcs/bin/hatype
com.collation.discover.agent.command.hauser.Linux=sudo /opt/VRTSvcs/bin/hauser

- Diese Eigenschaften sind zur Erkennung von Veritas-Clusterkomponenten erforderlich.
- Zur Ausführung dieser Befehle ohne sudo muss der TADDM-Service-Account auf dem Ziel zur Veritas-Administratorgruppe gehören.

com.collation.discover.agent.command.vxdisk=vxdisk
com.collation.discover.agent.command.vxdg=vxdg
com.collation.discover.agent.command.vxprint=vxprint
com.collation.discover.agent.command.vxlicrep=vxlicrep
com.collation.discover.agent.command.vxupgrade=vxupgrade

- Diese Eigenschaften erkennen Veritas-Standardspeicherinformationen sowie zusätzliche Veritas-spezifische Informationen wie Datenträgergruppe, Veritas-Datenträger, Plexes und untergeordnete Datenträger.

Fix Pack 6 **com.collation.discover.agent.command.zlogin=sudo zlogin**

com.collation.platform.os.command.ps.SunOS=/usr/ucb/ps axww
com.collation.platform.os.command.psEnv.SunOS=/usr/ucb/ps axwweee
com.collation.platform.os.command.psParent.SunOS=ps -elf -o ruser,pid,ppid,comm
com.collation.platform.os.command.psUsers.SunOS=/usr/ucb/ps auxw

- Diese Eigenschaften sind zur Erkennung von Prozessinformationen zu Solaris-Systemen erforderlich.

  Sie können eine bestimmte Solaris-Version angeben, indem Sie die SunOS-Versionsnummer an den Eigenschaftsnamen anhängen. So gilt beispielsweise die folgende Eigenschaft für Solaris 10:

  ```
  com.collation.platform.os.command.ps.SunOS5.10=sudo /usr/ucb/ps axww
  ```

com.collation.platform.os.command.ps.Linux=ps axww
com.collation.platform.os.command.psEnv.Linux=ps axwweee
com.collation.platform.os.command.psParent.Linux=ps -ax -o ruser,pid,ppid,comm
com.collation.platform.os.command.psUsers.Linux=ps auxw

- Diese Eigenschaften sind zur Erkennung von Prozessinformationen zu Linux-Systemen erforderlich.

com.collation.platform.os.command.ps.AIX=ps axww
com.collation.platform.os.command.psEnv.AIX=ps axwweee
com.collation.platform.os.command.psParent.AIX=ps -elf -o ruser,pid,ppid,comm
com.collation.platform.os.command.psUsers.AIX=ps auxw

- Diese Eigenschaften sind zur Erkennung von Prozessinformationen zu AIX-Systemen erforderlich.

com.collation.platform.os.command.ps.HP-UX=sh UNIX95= ps -elfx -o pid,tty,state,time,args
com.collation.platform.os.command.psEnv.HP-UX=ps -eflx
com.collation.platform.os.command.psParent.HP-UX=sh UNIX95= ps -elfx -o ruser,pid,ppid,comm
com.collation.platform.os.command.psUsers.HP-UX=ps -efx

- Diese Eigenschaften sind zur Erkennung von Prozessinformationen zu HP-UX-Systemen erforderlich.

com.collation.discover.agent.command.lsof.Vmnix=lsof
com.collation.discover.agent.command.lsof.Linux=lsof
com.collation.discover.agent.command.lsof.SunOS.1.2.3.4=sudo lsof
com.collation.discover.agent.command.lsof.Linux.1.2.3.4=sudo lsof
com.collation.discover.agent.command.lsof.HP-UX=lsof
com.collation.discover.agent.command.lsof.AIX=lsof

- Diese Eigenschaften sind zur Erkennung von Prozess- oder Portinformationen erforderlich.

  Sie können eine bestimmte Solaris-Version angeben, indem Sie die SunOS-Versionsnummer an den Eigenschaftsnamen anhängen. So gilt beispielsweise die folgende Eigenschaft für Solaris 10:

  ```
  com.collation.discover.agent.command.lsof.SunOS5.10=sudo /usr/local/bin/lsof
  ```

com.collation.discover.agent.command.dmidecode.Linux=dmidecode
com.collation.discover.agent.command.dmidecode.Linux.1.2.3.4=sudo dmidecode
Fix Pack 6

- Diese Eigenschaften sind zur Erkennung von UUID, Hersteller, Modell- und Seriennummer auf Linux-Systemen erforderlich.

com.collation.discover.agent.command.vmcp.Linux=
Mit dieser Eigenschaft kann eine Gastbenutzer-ID auf einem virtuellen Linux-Zielsystem erkannt werden, das unter einem z/VM-Betriebssystem aktiv ist.

com.collation.discover.agent.command.cat.SunOS=cat
com.collation.discover.agent.command.cat.SunOS.1.2.3.4=sudo cat

- Diese Eigenschaft ist zur Erkennung von Konfigurationsinformationen für eine Check Point-Firewall auf Solaris-Systemen erforderlich.

com.collation.discover.agent.command.interfacesettings.SunOS=sudo ndd
com.collation.discover.agent.command.interfacesettings.Linux=sudo mii-tool
com.collation.discover.agent.command.interfacesettings.SunOS.1.2.3.4=sudo ndd
com.collation.discover.agent.command.interfacesettings.Linux.1.2.3.5=sudo mii-tool
com.collation.discover.agent.command.interfacesettings.HP-UX=lanadmin
com.collation.discover.agent.command.interfacesettings.AIX=netstat

- Diese Eigenschaften sind zur Erkennung erweiterter Netzschnittstelleninformationen (z. B. Schnittstellenübertragungsrate) erforderlich.

com.collation.discover.agent.command.adb.HP-UX=adb
com.collation.discover.agent.command.adb.HP-UX.1.2.3.4=sudo adb

- Diese Eigenschaft ist zur Erkennung von Prozessorinformationen zu HP-Systemen erforderlich.

com.collation.discover.agent.command.kmadmin.HP-UX=kmadmin
com.collation.discover.agent.command.kmadmin.HP-UX.1.2.3.4=sudo /usr/sbin/kmadmin

- Diese Eigenschaft ist zur Erkennung von Kernelmodulen auf HP-Systemen erforderlich.

com.collation.platform.os.command.partitionTableListing.SunOS=prtvtoc

- Diese Eigenschaft ist zur Erkennung von Partitionstabelleninformationen zu Solaris-Systemen erforderlich.

com.collation.platform.os.command.lvm.lvdisplay.1.2.3.4=sudo lvdisplay -c
com.collation.platform.os.command.lvm.vgdisplay.1.2.3.4=sudo vgdisplay -c
com.collation.platform.os.command.lvm.pvdisplay.1.2.3.4=sudo pvdisplay -c

- Diese Eigenschaften sind zur Erkennung von Datenträgerinformationen erforderlich.

com.collation.platform.os.command.lputil.SunOS.1.2.3.4=sudo /usr/sbin/lpfc/lputil

- Diese Eigenschaft ist zur Erkennung von Emulex Fibre Channel HBA-Informationen auf Solaris-Systemen erforderlich.

com.collation.platform.os.command.crontabEntriesCommand.SunOS=crontab -l
com.collation.platform.os.command.crontabEntriesCommand.Linux=crontab -l -u
com.collation.platform.os.command.crontabEntriesCommand.AIX=crontab -l
com.collation.platform.os.command.crontabEntriesCommand.HP-UX=crontab -l

- Diese Eigenschaften sind zur Erkennung von **crontab**-Einträgen erforderlich. Sie können diese Eigenschaften für einen bestimmten Bereich angeben, indem Sie eine IP-Adresse oder den Namen einer Bereichsgruppe an die Eigenschaft anfügen. Im folgenden Beispiel wird eine IP-Adresse angefügt:

  ```
  com.collation.platform.os.command.crontabEntriesCommand.AIX.1.2.3.4=crontab -l
  ```

com.collation.platform.os.command.filesystems.Linux=df -kTP
com.collation.platform.os.command.filesystems.SunOS=df -k | grep -v 'No such file or directory' | grep -v 'Input/output error' | awk '{print $1, $2, $4, $6}'
com.collation.platform.os.command.filesystems.AIX=df -k | grep -v 'No such file or directory' | grep -v 'Input/output error' | awk '{print $1, $2, $3, $7}
com.collation.platform.os.command.filesystems.HP-UX=df -kP | grep -v 'No such file or directory' | grep -v 'Input/output error' | grep -v Filesystem

- Diese Eigenschaften sind zur Erkennung von Dateisystemen erforderlich.

com.collation.platform.os.command.fileinfo.ls=sudo ls
com.collation.platform.os.command.fileinfo.ls.1.2.3.4=sudo ls
com.collation.platform.os.command.fileinfo.cksum=sudo cksum
com.collation.platform.os.command.fileinfo.cksum.1.2.3.4=sudo cksum
com.collation.platform.os.command.fileinfo.dd=sudo dd
com.collation.platform.os.command.fileinfo.dd.1.2.3.4=sudo dd

- Diese Eigenschaften sind für die privilegierte Dateierfassung erforderlich.
- Die privilegierte Dateierfassung wird in Situationen verwendet, in denen der Erkennungsserviceaccount keinen Lesezugriff auf Anwendungskonfigurationsdateien hat, die für die Erkennung erforderlich sind.

com.collation.discover.agent.WebSphereVersionAgent.versionscript=sudo
Diese Eigenschaft kann für den Zugriff auf die WebSphere-Datei `versionInfo.sh` aktiviert werden, wenn der Erkennungsbenutzer keinen Zugriff auf das WebSphere Application Server-System hat.

com.collation.platform.os.command.fileinfo.OnlyDirectoryRecursive
Dieses Flag ändert die Art und Weise, wie die Konfigurationsdateien erkannt werden. Der Standardwert ist `False`.

Wenn Sie dieses Flag in `True` ändern, verwendet dieser Mechanismus den Suchbefehl nicht, um den Inhalt eines Verzeichnisses rekursiv zu erkennen.

Wenn Sie das Flag auf `False` setzen, verwendet dieser Mechanismus den Suchbefehl, um eine Datei rekursiv aufzuspüren, ohne dass die genaue Position der Datei angegeben wird.

Eigenschaften für den Kontextmenüservice und den Datenintegrationsservice
Diese Eigenschaften gelten für den Kontextmenüservice (Context Menu Service, CMS) und den Datenintegrationsservice (Data Integration Service, DIS).

com.ibm.cdb.DisCmsIntegration.enabled=true
Der Standardwert ist `true`.

Diese Eigenschaft gibt an, ob der Topologieerstellungsagent CMSDISAgent für eine regelmäßige Aktualisierung der TADDM-Daten aktiviert werden soll, die in der Datenbank des Kontextmenüservice und des Datenintegrationsservice registriert sind.

com.ibm.cdb.DisCmsIntegration.dbUser=*Benutzer*
Diese Eigenschaft gibt die Datenbankbenutzer-ID für die Datenbank des Kontextmenüservice und des Datenintegrationsservice an.

com.ibm.cdb.DisCmsIntegration.dbPassword=*Kennwort*
Diese Eigenschaft gibt das Datenbankbenutzerkennwort für die Datenbank des Kontextmenüservice und des Datenintegrationsservice an.

com.ibm.cdb.DisCmsIntegration.dbUrl=*URL*
Diese Eigenschaft gibt die Datenbank-URL für die Datenbank des Kontextmenüservice und des Datenbankintegrationsservice an.

com.ibm.cdb.DisCmsIntegration.dbDriver=*Treiber*
Diese Eigenschaft gibt den Datenbanktreiber für den Kontextmenüservice und den Datenintegrationsservice an.

com.ibm.cdb.DisCmsIntegration.changehistory.days_previous=30
Der Standardwert ist `30`.

Diese Eigenschaft gibt den Zeitraum in Tagen an, aus dem Änderungsprotokolle in den Änderungsberichten für den Kontextmenüservice und den Datenintegrationsservice angezeigt werden sollen.

Datenbankeigenschaften
Diese Eigenschaften gelten für die TADDM-Datenbank.

com.collation.db.password=Kennwort
Diese Eigenschaft gibt das Datenbankkennwort des Datenbankbenutzers an; es wird auf dem TADDM-Server gespeichert.

com.collation.db.archive.password=Kennwort
Diese Eigenschaft gibt das Datenbankkennwort für den Datenbankarchivbenutzer an; es wird auf dem TADDM-Server gespeichert.

com.ibm.cdb.db.max.retries
Diese Eigenschaft gibt die Anzahl der möglichen Wiederholungsversuche bei der Herstellung der Datenbankverbindung an.

com.ibm.cdb.db.timeout
Diese Eigenschaft gibt die Wartezeit in Millisekunden zwischen den Wiederholungsversuchen an.

com.ibm.cdb.db.connection.ssl.enable=false
Diese Eigenschaft legt fest, ob die Datenbankverbindung für den Datenbankbenutzer im SSL-Modus hergestellt wird.

Der Standardwert ist `false`.

com.ibm.cdb.db.connection.ssl.truststore.file=Dateiname
Diese Eigenschaft gibt die Truststore-Datei für den Datenbankbenutzer für die Herstellung der Datenbankverbindung im SSL-Modus an. Die Truststore-Datei muss sich im Verzeichnis `$COLLATION_HOME/etc/` befinden.

com.ibm.cdb.db.connection.ssl.truststore.password=Kennwort
Diese Eigenschaft gibt das Kennwort für die Truststore-Datei des Datenbankbenutzers für die Herstellung der Datenbankverbindung im SSL-Modus an.

com.ibm.cdb.db.archive.connection.ssl.enable=false
Diese Eigenschaft legt fest, ob die Datenbankverbindung für den Benutzer der Archivdatenbank im SSL-Modus hergestellt wird.

Der Standardwert ist `false`.

com.ibm.cdb.db.archive.connection.ssl.truststore.file=Dateiname
Diese Eigenschaft gibt die Truststore-Datei für den Benutzer der Archivdatenbank für die Herstellung der Datenbankverbindung im SSL-Modus an. Die Truststore-Datei muss sich im Verzeichnis `$COLLATION_HOME/etc/` befinden.

com.ibm.cdb.db.archive.connection.ssl.truststore.password=Kennwort
Diese Eigenschaft gibt das Kennwort für die Truststore-Datei des Benutzers der Archivdatenbank für die Herstellung der Datenbankverbindung im SSL-Modus an.

So verschlüsseln Sie die Datenbankkennwörter in der Datei `collation.properties`:

1. Bearbeiten Sie den Datenbankbenutzer und/oder archivieren Sie das Benutzerkennwort im Klartext.
2. Stoppen Sie den TADDM-Server.
3. Führen Sie die Datei `encryptprops.sh` oder `encryptprops.bat` aus (im Verzeichnis `$COLLATION_HOME/bin`). Dieses Script verschlüsselt die Kennwörter.
4. Starten Sie den TADDM-Server erneut.

Eigenschaften für die Erkennung
Diese Eigenschaften gelten für die Erkennung allgemein. Die TADDM-Servereigenschaften, die sich auf einen bestimmten Sensor beziehen, sind in den *Referenzinformationen zu Sensoren* für TADDM für den jeweiligen Sensor beschrieben.

Fix Pack 4 **com.discover.anchor.maxChannelNumber**
Diese Eigenschaft gibt die maximale Anzahl Kanäle an, die in der SSH-Sitzung zwischen dem TADDM-Server und dem Anker gleichzeitig geöffnet sind. Wenn die Anzahl der geöffneten Kanäle zu hoch ist, wird die Erkennung auf einem solchen Anker möglicherweise blockiert und es kommt bei den in einem solchen Bereich enthaltenen Sensoren zu einer Zeitlimitüberschreitung. Steuern Sie in einem solchen Fall mit dieser Eigenschaft die Anzahl der geöffneten Kanäle.

Der Standardwert ist `50`.

Fix Pack 4 **com.collation.platform.os.copyToLocal.preferScpCommand**

Diese Eigenschaft gibt an, ob mit dem externen **scp**-Befehl Dateien von fernen Hosts, normalerweise Erkennungsziele, auf den TADDM-Server kopiert werden. Der externe **scp**-Befehl wird in der Eigenschaft `com.collation.platform.os.scp.command` definiert. Setzen Sie diese Eigenschaft auf `true`, um die Verwendung des externen **scp**-Befehl zu aktivieren.

Der Standardwert für diese Eigenschaft ist `false`.

Anmerkung: Diese Eigenschaft gilt nur für die SSH-Sitzungen, die mit einer schlüsselbasierten Anmeldung eingerichtet wurden (siehe „Erkennung mit Secure Shell (SSH) konfigurieren" auf Seite 110). Im Falle einer Authentifizierung mit einem Benutzernamen und einem Kennwort wird der interne **scp**-Befehl unabhängig vom Wert der Eigenschaft `com.collation.platform.os.copyToLocal.preferScpCommand` verwendet.

Diese Eigenschaft ist eine bereichsorientierte Eigenschaft. Sie können eine IP-Adresse oder den Namen einer Bereichsgruppe an die Eigenschaft anhängen. Beispiele dafür sind:

```
com.collation.platform.os.copyToLocal.preferScpCommand.12.234.255.4=true
```

com.collation.platform.os.scp.command

Diese Eigenschaft gibt den Pfad zum Betriebssystembefehl **scp** an. Er kann verwendet werden, wenn ein interner SSH-Client ausfällt und deshalb keine Dateien zwischen dem TADDM-Server und fernen Hosts, üblicherweise Erkennungsziele, ausgetauscht werden. Sie können auch einen alternativen Befehl verwenden, der aber die gleiche Syntax wie der Befehl **scp** haben muss.

Beispielwert: `/usr/local/bin/scp`.

Fix Pack 3 **com.collation.platform.session.ssh.winAuth**

Diese Eigenschaft gibt an, ob bei Verwendung einer SSH-Sitzung eine Anmeldung mit Windows-Berechtigungsnachweisen versucht wird. Der Standardwert ist `true`.

Sie können den Wert auf `false` setzen, wenn das Risiko besteht, dass während der Erkennung Versuche stattfinden, sich mit Windows-Berechtigungsnachweisen bei Nicht-Windows-Servern anzumelden. So kann verhindert werden, dass Windows Active Directory-Konten ausgesperrt werden.

Fix Pack 3 **com.collation.platform.os.ignoreL2InterfaceDescription**

Diese Eigenschaft gibt die Beschreibungen von erkannten L2Interfaces an, die bei der Computersystemsignaturberechnung ignoriert werden sollen. Geben Sie beispielsweise folgenden Wert an, wenn Microsoft Load Balancer Interface nicht zur Berechnung der Signatur eines Computersystems verwendet werden soll:

```
com.collation.platform.os.ignoreL2InterfaceDescription=Microsoft Load
Balancer Interface
```

Der Wert dieser Eigenschaft wird wie ein regulärer Ausdruck behandelt. Das heißt, dass Sie mehrere Schnittstellenbeschreibungen hinzufügen können, ohne ein Trennzeichen, z. B. ein Komma, angeben zu müssen.

Fix Pack 3 **com.ibm.cdb.topomgr.topobuilder.agents.ConnectionDependencyAgent2.dependencyPlaceholders**

Wird diese Eigenschaft auf `true` gesetzt, werden Platzhalter-App-Server für nicht erkannte Abhängigkeiten erstellt.

Anmerkung: Diese Eigenschaft ist standardmäßig nicht in der Datei `collation.properties` enthalten. Sie müssen sie hinzufügen.

Wenn Sie den Wert zum ersten Mal auf `true` setzen, müssen Sie TADDM erneut starten, um erweiterte Attribute für LogicalConnection- und SSoftwareServer-Klassen zu aktivieren. Die erweiterten Attribute sind für eine ordnungsgemäße Ausführung dieser Funktion notwendig.

Weitere Informationen zu Platzhaltern finden Sie im Abschnitt „Für Erkennung von Platzhaltern konfigurieren" auf Seite 122.

com.collation.platform.session.EncodingOverRide
Diese Eigenschaft gibt den Typ der Codierung an, die während einer Erkennungssitzung verwendet wird. Sie ist besonders hilfreich, wenn Ihre Zielserver eine andere Codierung als die auf dem TADDM-Server verwenden.

Der Wert dieser Eigenschaft ist der Name der Codierung, z. B. UTF-8. Sie ist nicht standardmäßig in der Datei `collation.properties` enthalten. Sie müssen sie dort hinzufügen.

Sie können auch einen Bereich oder eine IP-Adresse zu dieser Eigenschaft hinzufügen. Beispiele dafür sind:

```
com.collation.platform.session.EncodingOverRide.37.53.105.24=UTF-8
```

com.collation.discover.anchor.forceDeployment=true
Der Standardwert ist `true`.

Diese Eigenschaft gibt an, ob die Anker für den erkannten Bereich beim Start der Erkennung implementiert werden sollen.

Bei Angabe von `false` werden die Anker nur implementiert, wenn eine der folgenden Situationen vorliegt:

- Eine der IP-Adressen aus dem Bereich kann nicht mit Ping überprüft werden.
- Port 22 kann nicht über eine der erkannten IP-Adressen erreicht werden.

Falls verkettete Anker vorhanden sind, gilt diese Bedingung für alle Anker in der Kette. Wenn ein Anker in der Kette durch eine Bedingung eingeschränkt ist, müssen die vorherigen Anker die Bedingung erfüllen, damit alle Anker implementiert werden können.

com.collation.discover.anchor.lazyDeployment=false
Der Standardwert ist `false`.

Diese Eigenschaft gibt an, ob die für einen Sensor erforderlichen Dateien bei der Implementierung eines Ankers kopiert werden sollen (Wert `false`) oder beim Start des Sensors, für den die Dateien erforderlich sind (Wert `true`).

Für den IBM WebSphere-Sensor bestehen beispielsweise Abhängigkeiten im Verzeichnis `dist/lib/websphere`. Dieses Verzeichnis hat eine Größe von 130 MB. Wird für diese Eigenschaft `false` angegeben, werden die Abhängigkeitsdaten bei der Implementierung des Ankers auf den Zielhost kopiert. Bei Angabe von `true` werden die Daten beim Start der Ausführung des WebSphere-Sensors im Anker kopiert. Wird kein WebSphere-Sensor im Anker ausgeführt, werden die 130 MB nicht an den fernen Host gesendet.

com.collation.discover.DefaultAgentTimeout=600000
Der Wert lautet `600000` (Millisekunden), dies entspricht 10 Minuten.

Diese Eigenschaft gibt das Zeitlimit für Sensoren in Millisekunden an. Das Standardzeitlimit sollte nicht geändert werden. Stattdessen können Sie das Zeitlimit für einzelne Sensoren angeben.

Fügen Sie die folgende Zeile in der Datei `collation.properties` hinzu, um den Zeitlimitwert für einen bestimmten Sensor zu überschreiben:

```
com.collation.discover.agent.Sensorname*Sensor.timeout=
  Zeit_in_Millisekunden*
```

Hier ein Beispiel:

```
com.collation.discover.agent.OracleSensor.timeout=1800000
```

com.collation.IpNetworkAssignmentAgent.defaultNetmask=*IP-Beginn-IP-Ende/net-mask*[, ...]

Diese Eigenschaft definiert, wie während einer Erkennung der Ebene 1 erkannte IP-Adressen generierten Teilnetzen zugeordnet werden sollen. Teilnetze werden nicht durch eine Erkennung der Ebene 1 erkannt. Stattdessen werden IpNetwork-Objekte erstellt, die alle Schnittstellen enthalten, die keinem bei einer Erkennung der Ebene 2 oder 3 erkannten Teilnetz zugeordnet sind. Diese Konfigurati-

onseigenschaft gibt an, wie viele IpNetwork-Objekte erstellt werden und wie viele Knoten die einzelnen Teilnetze enthalten sollen. (Dies gilt auch für alle während einer Erkennung der Ebene 2 oder 3 erkannten Schnittstellen, die aus irgendwelchen Gründen keinem erkannten Teilnetz zugeordnet werden können.)

Der Wert für diese Eigenschaft besteht aus einer einzigen Zeile, die einen oder mehrere durch Kommas getrennte Einträge enthält. Jeder Eintrag beschreibt einen IP-Adressbereich in der IPv4-Schreibweise mit Trennzeichen sowie mit einer in Form einer ganzen Zahl zwischen 8 und 31 angegebenen Teilnetzmaske. Anschließend werden erkannte Schnittstellen in dem angegebenen Bereich erstellten Teilnetzen zugeordnet, deren Größe den in der Teilnetzmaske angegebenen Wert nicht überschreitet.

Der folgende Wert definiert beispielsweise zwei Teilnetzadressbereiche mit verschiedenen Teilnetzmasken:

```
9.0.0.0-9.127.255.255/23, 9.128.0.0-9.255.255.255/24
```

Die angegebenen Adressbereiche können sich überschneiden. Wenn eine erkannte IP-Adresse mit mehr als einem definierten Bereich übereinstimmt, wird sie dem ersten passenden Teilnetz in der Liste des Eigenschaftswerts zugeordnet.

Nach der Erstellung oder Änderung dieser Konfigurationseigenschaft und dem Neustart des TADDM-Servers verwenden alle nachfolgenden Erkennungen der Ebene 1 die definierten Teilnetze. Um bereits vorhandene IP-Schnittstellenobjekte in der TADDM-Datenbank neu zuzuordnen, rufen Sie das Verzeichnis `$COLLATION_HOME/bin` auf und führen Sie einen der folgenden Befehle aus:

- **`adjustL1Networks.sh`** (Linux- und UNIX-Systeme)
- **`adjustL1Networks.bat`** (Windows-Systeme)

Wird der Wert nicht korrekt angegeben, werden die entsprechenden Nachrichten nur bei Ausführung des Befehlszeilendienstprogramms **`adjustL1Networks.sh`** (Linux und UNIX-Systeme) oder **`adjustL1Networks.bat`** (Windows-Systeme) angezeigt. Andernfalls werden die Nachrichten in der Datei `TopologyBuilder.log` im Verzeichnis `$COLLATION_HOME/log/services` und in der Datei `IpNetworkAssignmentAgent.log` im Verzeichnis `$COLLATION_HOME/log/agents` abgelegt.

Dieses Script ordnet alle während Erkennungen der Ebene 1 erkannten IP-Schnittstellenobjekte nach der Beschreibung in der Konfigurationseigenschaft den entsprechenden Teilnetzen neu zu. Alle generierten IP-Netzobjekte, die keine Schnittstellen enthalten, werden dann aus der Datenbank gelöscht. Nach Ausführung des Scripts werden in der TADDM-Schnittstelle aufgrund der geänderten Objekte möglicherweise mehrere Benachrichtigungen über geänderte Komponenten angezeigt. Sie können diese Benachrichtigung durch Aktualisierung des Fensters löschen.

Anmerkung: Stellen Sie vor Ausführung dieses Befehls sicher, dass der TADDM-Server aktiv ist und keine Erkennung oder Massenladeoperation durchgeführt wird. Auf dem Synchronisationsserver wird dieses Script nicht unterstützt.

com.collation.number.persist.discovery.run=30
Der Standardwert ist 30.

Gibt die Anzahl der Erkennungen an, für die Informationen im Erkennungsprotokoll des Datenmanagementportals und der Discovery Management Console gespeichert werden.

Um den Standardwert in einer Streaming-Server-Implementierung zu ändern, geben Sie den neuen Wert auf dem primären Speicherserver ein.

com.collation.platform.os.hostappdescriptorfiles.dir="*Pfad*"
Gibt den vollständig qualifizierten Pfad des Verzeichnisses an, in dem die Dateien des Komponentenanwendungsdeskriptors für Computersysteme (Hosts) implementiert sind. Diese Eigenschaft ist erforderlich, wenn Geschäftsanwendungen unter Verwendung von Anwendungsdeskriptoren Computersysteme hinzugefügt werden sollen. Sie können diese Eigenschaft für einen bestimmten Hostnamen oder eine bestimmte IP-Adresse definieren, sodass für jeden Host ein eigener Pfad angegeben werden kann. Im Folgenden zwei Beispiele für die Pfadangabe des Hostanwendungsdeskriptors:

- Linux- und UNIX-Systeme: `/home/taddm/hostappdescriptors`

- Windows-Systeme: `c://taddm//hostappdescriptors`

com.collation.platform.session.GatewayForceSsh
Gibt an, ob erzwungen werden soll, dass das Gateway unabhängig vom Anker agiert. Gültige Werte sind *true* und *false*. Setzen Sie den Wert auf `true`, um Cygwin-Probleme zu beheben, wenn sich das Gateway und der Anker auf demselben System befinden. Wenn der Wert auf 'true' gesetzt ist, werden Daten zwischen dem Gateway und dem Anker mithilfe einer SSH-Sitzung anstatt einer lokalen Sitzung übertragen.

com.collation.rediscoveryEnabled=`false`
Der Standardwert ist `false`.

Diese Eigenschaft gilt für die erneute Erkennung eines bereits erkannten Konfigurationselements. Diese Funktion ist im Datenmanagementportal verfügbar.

Einschränkung: Die erneute Erkennung kann die Berechtigungsnachweise aus einem angepassten Profil nicht verwenden, sondern verwendet die Berechtigungsnachweise aus der globalen Liste.

Anmerkung:

Soll die erneute Erkennung in einer Domänenserverimplementierung aktiviert werden, setzen Sie den Wert auf dem Domänenserver auf `true`.

Soll die erneute Erkennung in einer Streaming-Server-Implementierung aktiviert werden, müssen Sie den Wert auf dem Erkennungs- und Speicherserver auf `true` setzen.

Erneute Erkennung in einer Streaming-Server-Implementierung
Bei Verwendung der erneuten Erkennung in einer Streaming-Server-Implementierung kann ein Konfigurationselement zwar von verschiedenen Erkennungsservern erkannt, jedoch nur von dem Erkennungsserver erneut erkannt werden, von dem es zuletzt erkannt wurde. Bei mehreren Erkennungsservern werden die Informationen zur erneuten Erkennung eines Konfigurationselements von den einzelnen Erkennungsservern überschrieben.

Bei einer Aktivierung der erneuten Erkennung auf dem Erkennungsserver werden für jedes erkannte Objekt zusätzliche Informationen zur erneuten Erkennung erstellt.

Bei einer Aktivierung der erneuten Erkennung auf dem Speicherserver werden die einzelnen erkannten Objekte mit zusätzlichen Informationen zur erneuten Erkennung gespeichert.

Wird die erneute Erkennung auf dem Erkennungsserver aktiviert, auf dem Speicherserver dagegen inaktiviert, stehen die Informationen zur erneuten Erkennung nicht in der TADDM-Datenbank zur Verfügung. Darüber hinaus müssen Sie sicherstellen, dass für den Erkennungs- und den Speicherserver dieselben Berechtigungsnachweise verwendet werden.

com.ibm.cdb.discover.sensor.sys.utilization.workingdir=`/tmp/taddm`
Der Standardwert ist `/tmp/taddm`.

Diese Eigenschaft gibt den Rootpfad für die IBM Tivoli Utilization-Sensorenscripts an, die auf dem Zielsystem ausgeführt werden sollen. Wird dieser Wert nicht angegeben, wird der Pfad verwendet, der durch die Eigenschaft `com.ibm.cdb.taddm.script.path` definiert ist.

com.ibm.cdb.locationTag
Gibt für jedes Konfigurationselement (Configuration Item, CI), das auf dem TADDM-Server erstellt wurde, das Attribut für den Positionstag an. Mit dem Attribut für den Positionstag, das die Position eines Konfigurationselements identifiziert, werden statische Positionstags unterstützt. Vor der Angabe dieses Tags müssen Sie den Wert von `com.ibm.cdb.locationTaggingEnabled` auf `true` setzen.

com.ibm.cdb.locationTaggingEnabled=`false`
Der Standardwert ist `false`.

Gibt an, ob die Funktion des Positions-Taggings aktiviert ist. Setzen Sie den Wert dieser Eigenschaft auf `true`, wenn Sie:

- für jedes Konfigurationselement (Configuration Item, CI), das auf dem TADDM-Server erstellt wurde, ein Attribut für den Positionstag angeben möchten. Details finden Sie in der Beschreibung der Eigenschaft `com.ibm.cdb.locationTag`.

- unter Verwendung der Befehlszeilenschnittstelle einen dynamischen Positionstag für Konfigurationselemente angeben möchten, die während einer einzelnen Erkennung erstellt wurden. Dynamische Positionstags überschreiben bereits vorhandene Positionstags (statische Positionstags).
- einen dynamischen Positionstag für Konfigurationselemente angeben möchten, die beim Laden von Daten mit dem Massenladeprogramm erstellt wurden.
- bei der Ausführung eines BIRT-Berichts zum Filtern der Daten und Berichtsinformationen, die auf die betreffende Position beschränkt werden sollen, einen Positionstagwert angeben möchten.
- einen Positionstagwert für Konfigurationselemente erstellen möchten, die während eines Topologieerstellungsprozesses erstellt wurden.

com.ibm.cdb.taddm.host
Gibt den Aliasnamen des TADDM-Server-Hosts an. Wird dieser Wert nicht angegeben, wird der Hostname des Systems verwendet. Wenn der TADDM-Server den Systemhostnamen nicht auflösen kann oder dieser in den lokalen Host aufgelöst wird, müssen Sie diese Eigenschaft manuell angeben.

com.ibm.cdb.taddm.script.path=/tmp/taddm
Der Standardwert ist `/tmp/taddm`.

Diese Eigenschaft gibt den Stammverzeichnispfad für die Sensorenscripts an, die auf dem Zielsystem ausgeführt werden sollen. Dort wird eine Unterverzeichnisstruktur in folgendem Format erstellt: `Hostalias/Erkennungsnummer/Sensorname`. Der Name von `Hostalias` wird aus der Eigenschaft `com.ibm.cdb.taddm.host` abgerufen. Wird diese Eigenschaft nicht angegeben, wird der Hostname des Systems verwendet. Damit zwischen gleichzeitig ablaufenden Erkennungen auf demselben Erkennungsserver unterschieden werden kann, wird dem Verzeichnis `Erkennungsnummer` eine Nummer zugeordnet. Beim Speichern der Erkennungsscripts und der Erkennungsergebnisse wird diese Verzeichnisstruktur verwendet.

com.collation.discover.agent.signature.ignore.*1.2.3.4*=true
Mit dieser Eigenschaft kann eine IP-Adresse bei der Berechnung der Signatur ausgelassen werden.

Bei einigen Konfigurationen kann sich die Signatur eines Computersystems als nicht eindeutig erweisen, in welchem Fall es beim Abgleich mit den bestehenden Einträgen in der TADDM-Datenbank zu Problemen kommt. Dies passiert zum Beispiel gelegentlich, wenn Sie virtuelle Maschinen mit virtuellen Netzkarten verwenden, deren Hardware- und IP-Adressen gültig sind. In einem solchen Fall müssen Sie die Signaturberechnung ausschließen und andere Namensregeln (z. B. ProductModel, Manufacturer oder Serial Number) verwenden.

Fügen Sie für jede IP-Adresse, die ignoriert werden soll, die Eigenschaft `com.collation.discover.agent.signature.ignore.`*`1.2.3.4`*`=true` hinzu, wobei *1.2.3.4* die zu ignorierende IP-Adresse ist.

Sollen mehrere IP-Adressen ignoriert werden, erstellen Sie einen Erkennungsbereich. Fügen Sie der Datei `collation.properties` in diesem Fall die Eigenschaft `com.collation.discover.agent.signature.ignore.`*`blacklist`*`=true` hinzu, wobei *blacklist* der Erkennungsbereich mit den zu ignorierenden IP-Adressen ist.

Erweiterte Eigenschaften für die Erkennung

Die erweiterten Erkennungseigenschaften legen die Pufferkapazität für das Speichern von Arbeitselementen fest. Außerdem können Sie über diese die Anzahl der Neustarts bestimmter Erkennungselementen oder den Zeitwert für die Ausgabe der Statistikdaten in einem Protokoll angeben. Ändern Sie diese Eigenschaften nur, wenn Sie den Erkennungsprozess im Detail optimieren müssen.

com.ibm.cdb.discover.buffers.workitem.capacity=64
Der Standardwert ist 64. Dieser Wert ist jedoch immer der doppelte Wert der Eigenschaft `com.collation.discover.dwcount`, die standardmäßig auf 32 gesetzt ist.

Diese Eigenschaft gibt die Pufferkapazität für das Speichern von Erkennungsarbeitselementen an. Mit ihr wird der Speicherbedarfs des Erkennungsprozesses begrenzt, damit Fehler aufgrund einer Speicherknappheit (OutOfMemory) vermieden werden können. Für jede Erkennung wird ein neuer Sensor gestartet.

Dieser Wert darf nicht niedriger als die Anzahl der Erkennungsworker sein, die in der Eigenschaft com.collation.discover.dwcount angegeben ist, da einige davon andernfalls im Leerlauf verbleiben.

com.ibm.cdb.discover.buffers.workitem.maxresets=10
Der Standardwert ist 10.

Diese Eigenschaft gibt an, wie oft ein Sensor im Fall eines nicht erwarteten Fehlers erneut gestartet werden kann, zum Beispiel bei einem Ausfall einer TADDM-JVM, die für die Erkennung verantwortlich ist.

Die Anzahl der Neustarts für ein Element des Erkennungsprozesses wird außerdem auch durch com.ibm.cdb.discover.runrestartlimit begrenzt. Dort ist die Anzahl der Erkennungsneustarts angegeben.

com.ibm.cdb.discover.buffers.seed.capacity=100
Der Standardwert ist 100.

Diese Eigenschaft gibt die Pufferkapazität für das Speichern von Seed-Arbeitselementen an. Mit ihr wird der Speicherbedarfs des Erkennungsprozesses begrenzt, damit Fehler aufgrund einer Speicherknappheit (OutOfMemory) vermieden werden können.

com.ibm.cdb.discover.buffers.result.capacity=100
Der Standardwert ist 100.

Diese Eigenschaft gibt die Pufferkapazität für das Speichern von Ergebnisarbeitselementen an. Mit ihr wird der Speicherbedarfs des Erkennungsprozesses begrenzt, damit Fehler aufgrund einer Speicherknappheit (OutOfMemory) vermieden werden können. Für jedes Ergebnisarbeitselement kann ein neuer Sensor gestartet werden.

Verwenden Sie für diese Angabe den Wert der Eigenschaft com.ibm.cdb.discover.buffers.discovered.capacity.

com.ibm.cdb.discover.buffers.result.maxresets=10
Der Standardwert ist 10.

Diese Eigenschaft gibt an, wie oft ein Erkennungsprozess einen neuen Sensor für ein Ergebnisarbeitselement im Fall eines nicht erwarteten Fehlers starten kann, zum Beispiel bei einem Ausfall einer TADDM-JVM, die für die Erkennung verantwortlich ist.

Die Anzahl der Neustarts für ein Element des Erkennungsprozesses wird außerdem auch durch com.ibm.cdb.discover.runrestartlimit begrenzt. Dort ist die Anzahl der Erkennungsneustarts angegeben.

com.ibm.cdb.discover.buffers.discovered.capacity=100
Der Standardwert ist 100.

Diese Eigenschaft gibt die Pufferkapazität für das Speichern von erkannten Arbeitselementen an. Jedes erkannte Arbeitselement stellt ein Erkennungsergebnis dar, das in der Datenbank gespeichert wird.

Dieser Wert darf nicht niedriger als die Anzahl der Schreibthreads für die Datenbank sein, die in der Eigenschaft com.collation.discover.observer.topopumpcount angegeben ist.

com.ibm.cdb.discover.buffers.statistics.interval.seconds=60
Der Standardwert ist 60. Geben Sie den Wert in Sekunden an.

Diese Eigenschaft gibt den Zeitwert für das Speichern der Pufferstatistik der Erkennung in einem Protokoll an. Das Protokoll befindet sich in /log/services/DiscoveryState.log.

com.ibm.cdb.discover.buffers.timeout.interval.seconds=600
Der Standardwert beträgt 600, also zehn Minuten. Geben Sie den Wert in Sekunden an.

Diese Eigenschaft gibt den Zeitwert für das Prüfen der Arbeitselemente im Hinblick auf eine Zeitlimitüberschreitung an.

com.ibm.cdb.discover.runcontroller.statistics.interval.seconds=60
Der Standardwert ist 60. Geben Sie den Wert in Sekunden an.

Diese Eigenschaft gibt den Zeitwert für das Speichern der Ausführungsstatistik der Erkennung in einem Protokoll an. Das Protokoll befindet sich in `/log/services/DiscoveryRunController.log`.

com.ibm.cdb.discover.runrestartlimit=11
Der Standardwert ist 11.

Diese Eigenschaft gibt an, wie oft eine nicht initialisierte Erkennung nach einem Fehler erneut gestartet werden kann. Die Erkennung befindet sich im nicht initialisierten Status, wenn der Prozess noch nicht für alle Elemente des Erkennungsbereichs gestartet wurde.

com.collation.discovery.oracle.tablelimit=1000
Der Standardwert ist 1000. Die Eigenschaft unterstützt nur positive Werte.

Diese Eigenschaft steuert die Menge der Tabellen, die vom Oracle-Sensor erkannt werden.

Eigenschaften für die gleichzeitige Erkennung
Diese Eigenschaften gelten für die gleichzeitige Erkennung.

com.collation.discover.concurrent.discovery=true
Der Standardwert ist `true`.

Über diese Eigenschaft wird die gleichzeitige Erkennung aktiviert.

com.collation.discover.max.concurrent.discoveries=10
Der Standardwert ist 10.

Diese Eigenschaft gibt die maximale Anzahl an Erkennungsläufen an, die gleichzeitig ausgeführt werden können.

Eigenschaften für die asynchrone Erkennung
Diese Eigenschaften gelten für die asynchrone Erkennung.

com.ibm.cdb.discover.asd.AsyncDiscoveryResultsDirectory=var/asdd
Der Standardwert ist `var/asdd`; dieser Pfad ist relativ zum Verzeichnis `com.collation.home`.

Diese Eigenschaft gibt an, wo sich das Stammverzeichnis für die Archivdateien auf dem TADDM-Server befindet, die die Ergebnisse einer asynchronen Erkennung enthalten. Dabei kann es sich um einen relativen oder einen absoluten Pfad handeln. Dabei ist ein relativer Pfad ein Pfad relativ zum Verzeichnis `com.collation.home`.

com.ibm.cdb.discover.asd.ProcessUnreachableIPs=false
Der Standardwert ist `false`.

Über diese Eigenschaft wird die Verarbeitung von nicht erreichbaren IP-Adressen aktiviert, die in einer asynchronen Erkennung verwendet werden. Soll die Verarbeitung dieser Adressen aktiviert werden, muss die Eigenschaft auf `true` gesetzt werden.

com.ibm.cdb.tarpath=tar
Der Standardwert ist `tar`.

Diese Eigenschaft gibt den Pfad des Befehls **`tar`** *auf dem TADDM-Server* bei einer asynchronen Erkennung an.

Auf Betriebssystemen wie AIX oder Linux ist diese Eigenschaft normalerweise nicht erforderlich, da der Befehl **`tar`** bereits installiert und verfügbar ist. Auf einem TADDM-Server unter Windows hingegen muss ein tar-Programm eines anderen Herstellers installiert und der vollständige Pfadname des Programms angegeben werden, wenn ein Scriptpaket für die asynchrone Erkennung generiert oder Erkennungsarchivdateien verarbeitet werden sollen.

Das folgende Beispiel zeigt, wie der Pfad des **tar**-Befehls auf dem TADDM-Server für das Betriebssystem AIX angegeben wird:

```
com.ibm.cdb.tarpath=tar
```

com.ibm.cdb.targettarpath=tar
Der Standardwert ist tar.

Diese Eigenschaft gibt den Pfad des Befehls **tar** auf dem Zielsystem bei einer asynchronen Erkennung an.

Auf Zielbetriebssystemen wie AIX oder Linux ist diese Eigenschaft normalerweise nicht erforderlich, da der Befehl **tar** bereits installiert und verfügbar ist. Auf Solaris-Betriebssystemen müssen Sie hingegen aufgrund der Einschränkung, die in Bezug auf die Länge von Dateinamen besteht, das Archivdienstprogramm 'gtar' verwenden und den Pfad zum Dienstprogramm angeben.

Die folgenden Beispiele zeigen, wie der Pfad des Befehls **tar** *auf dem Zielsystem* für die jeweiligen Betriebssysteme angegeben werden muss:

Unter AIX
com.ibm.cdb.targettarpath.AIX=tar

Unter Solaris
com.ibm.cdb.targettarpath.SunOS=/usr/sfw/bin/gtar

Eigenschaften für die scriptbasierte Erkennung
Diese Eigenschaften gelten für die scriptbasierte Erkennung.

Fix Pack 4 **com.ibm.cdb.discover.enableOutputFileSplittingProcess=true**
Der Standardwert ist true.

Diese Eigenschaft gibt an, ob die Hauptausgabedatei, die während einer scriptbasierten Erkennung erstellt wird, in kleinere Dateien aufgeteilt wird. Standardmäßig wird die Datei aufgeteilt. Diese Einstellung verhindert Leistungsprobleme, wenn die Ausgabedatei groß ist. Siehe auch die Eigenschaft com.ibm.cdb.discover.numberOfLinesForOutputFileSplittingProcess.

Fix Pack 4 **com.ibm.cdb.discover.numberOfLinesForOutputFileSplittingProcess=10000**
Der Standardwert ist 10000.

Diese Eigenschaft wird nur aktiviert, wenn die Eigenschaft com.ibm.cdb.discover.enableOutputFileSplittingProcess auf true gesetzt ist.
Diese Eigenschaft gibt die ungefähre Anzahl der Zeilen an, die in den kleineren Ausgabedateien zulässig sind, die durch Aufteilung der Hauptausgabedatei erstellt wurden. Die genaue Anzahl Zeilen wird durch das Dateiformat bestimmt. Nach der angegebenen Anzahl von Zeilen wird die Datei nur geteilt, wenn das Ende einer aussagefähigen Gruppe von Daten erreicht ist, um sicherzustellen, dass das vollständige Dateiformat korrekt ist. Dies bedeutet bei einem Wert von 10000, dass die kleineren Dateien beispielsweise 10200 Zeilen haben können.

com.ibm.cdb.taddm.asd.prefix=sh
Der Standardwert ist sh.

Diese Eigenschaft gibt ein Präfix an, das dem Script hinzugefügt werden soll, das während einer Erkennung ausgeführt wird. Beispiel: *Präfix* script.sh. Es handelt sich um eine bereichsorientierte Eigenschaft, an die die IP-Adresse oder der Name einer Bereichsgruppe angehängt werden kann.

com.ibm.cdb.discover.DeleteScriptDiscoveryOutputs=true
Der Standardwert ist true.

Diese Eigenschaft gibt an, ob die bei einer scriptbasierten Erkennung generierte Scriptausgabe zur Verarbeitung durch die Sensoren auf den TADDM-Server kopiert werden soll. Die Ausgabe ist unter Umständen für die Fehlerbehebung hilfreich, standardmäßig wird sie jedoch nach Abschluss des Erkennungsvorgangs gelöscht. Wenn Sie diese Eigenschaft auf false setzen, wird die Scriptausgabe nicht gelöscht.

com.ibm.cdb.discover.DeleteRemoteBeforeScriptsRun=false
Der Standardwert ist false.

Diese Eigenschaft legt fest, ob TADDM vor der erneuten Ausführung einer Erkennung alle Ausgaben entfernt, die im fernen Verzeichnis von früheren Erkennungen zurückgeblieben sind.

com.ibm.cdb.discover.PreferScriptDiscovery=`false`
Der Standardwert ist `false`.

Über diese Eigenschaft wird die scriptbasierte Erkennung aktiviert; sie hat nur auf Sensoren Auswirkung, die die scriptbasierte Erkennung unterstützen. Bei Angabe von `true` wird die scriptbasierte Erkennung aktiviert.

com.ibm.cdb.discover.smallFileSizeLimit=1048576
Der Standardwert ist 1048576 (1024*1024 - 1 MB).

Diese Eigenschaft legt für Kopieroperationen das Dateigrößenlimit in Byte fest, durch das eine Kontrollsummenprüfung ausgelöst wird. Dateien mit einer Größe unter diesem Limit werden ohne Berechnung der Kontrollsumme kopiert. Ab dieser Größe werden Dateien nur kopiert, wenn sie noch nicht im Zielverzeichnis vorhanden sind bzw. ihre Kontrollsumme nicht mit der lokalen (Quellen-)Datei übereinstimmt.

Das Limit können Sie mit folgenden Werten inaktivieren:

- `0` - Bei Kopieroperationen wird immer eine Kontrollsumme berechnet.
- `-1` - Bei Kopieroperationen wird nie eine Kontrollsumme berechnet.

Eigenschaften für Erkennung mit IBM Tivoli Monitoring (altes Verfahren)

Diese Eigenschaften gelten für die Erkennung mit IBM Tivoli Monitoring (altes Verfahren).

Altes Integrationsverfahren

Dieser Abschnitt bezieht sich auf ein veraltetes Verfahren zur TADDM-Integration mit IBM Tivoli Monitoring. Ab TADDM Version 7.3.0 sollte die Integration mit IBM Tivoli Monitoring 6.3 mithilfe von OSLC Automation erfolgen. Das Integrationsverfahren unter Verwendung des IBM Tivoli Monitoring Scope-Sensors ist veraltet und steht in künftigen Releases nicht mehr zur Verfügung. Weitere Informationen zu Eigenschaften, die zur Konfiguration des Erkennungsprozesses mithilfe von OSLC Automation verwendet werden, finden Sie in den Abschnitten „TADDM über OSLC Automation mit IBM Tivoli Monitoring integrieren” auf Seite 178 und „Eigenschaften für die Erkennung mit OSLC Automation Session” auf Seite 79.

Eigenschaften mit Auswirkungen auf die TADDM-Vorgehensweise bei der Erkennung von Tivoli Monitoring-Endpunkten

Für eine TADDM-Erkennung der Ebene 2 und 3 ist normalerweise ein Domänenserver (in einer Domänen- oder Synchronisationsserverimplementierung) bzw. ein Erkennungsserver (in einer Streaming-Server-Implementierung) erforderlich, um mit einer der folgenden Methoden eine direkte Verbindung zu einem Zielsystem herzustellen:

- Secure Shell (SSH) für UNIX-basierte Zielsysteme
- Windows Management Instrumentation (WMI) für Windows-Systeme

Damit diese Methoden verwendet werden können, muss der Domänen- oder Erkennungsserver die Benutzerberechtigung (Benutzerkonto und Kennwort) kennen.

Bei einer Erkennung mit IBM Tivoli Monitoring kann TADDM Informationen der Ebene 2 (und einige Informationen der Ebene 3) zu Zielsystemen erkennen, für die keine Benutzerberechtigung verfügbar ist. Die Sensoren werden in der Tivoli Monitoring-Infrastruktur ausgeführt und nur die Berechtigungsnachweise für Tivoli Enterprise Portal Server sind erforderlich. Nachdem der IBM Tivoli Monitoring Scope-Sensor konfiguriert und ausgeführt wurde, kann Tivoli Monitoring bei zukünftigen Erkennungen der Ebene 2 standardmäßig verwendet werden. Für den Fall, dass dieses Standardverhalten in Ihrer Umgebung nicht erwünscht ist, bietet TADDM die Möglichkeit über folgende Servereigenschaften festzulegen, ob Tivoli Monitoring oder eine direkte Verbindung (SSH oder WMI) für die Erkennung verwendet werden soll. Diese Eigenschaften können auf globaler Ebene oder für einen bestimmten Bereich oder ein bestimmtes Erkennungsprofil festgelegt werden.

com.ibm.cdb.session.allow.ITM=true
Der Standardwert ist `true` und bedeutet, dass TADDM IBM Tivoli Monitoring für die Erkennung von Tivoli Monitoring-Endpunkten verwenden kann.

Diese Eigenschaft gibt an, ob TADDM IBM Tivoli Monitoring für die Erkennung von Tivoli Monitoring-Endpunkten verwenden kann.

Wenn stattdessen eine direkte Verbindung zu einem Tivoli Monitoring-Endpunkt hergestellt werden soll, legen Sie den Wert `false` fest.

Sie können mit dieser Eigenschaft auch einen benutzerdefinierten Erkennungsbereich angeben, wie in folgendem Beispiel dargestellt:

com.ibm.cdb.session.allow.ITM.*IP-Adresse*=false
Im folgenden Beispiel wird angegeben, dass TADDM den Erkennungsbereich `10.20.30.40` verwendet und eine direkte Verbindung zum Endpunkt herstellt, selbst wenn dieser durch Tivoli Monitoring überwacht wird:

```
com.ibm.cdb.session.allow.ITM.10.20.30.40=false
```

com.ibm.cdb.session.prefer.ITM=true
Der Standardwert ist `true` und bedeutet, dass TADDM IBM Tivoli Monitoring für die Erkennung von Tivoli Monitoring-Endpunkten verwendet.

Diese Eigenschaft gibt an, ob TADDM IBM Tivoli Monitoring als bevorzugte Methode für die Erkennung von Tivoli Monitoring-Endpunkten verwendet, vorausgesetzt, dass die Erkennung mit IBM Tivoli Monitoring für die Endpunkte zulässig ist. Wenn TADDM IBM Tivoli Monitoring für die Erkennung verwendet und die Erkennung nicht erfolgreich ist, verwendet TADDM stattdessen anschließend eine direkte Verbindung zu den Endpunkten. Wenn umgekehrt die Erkennung mit IBM Tivoli Monitoring nicht die bevorzugte Methode ist und die direkte Verbindung zu dem Endpunkt nicht erfolgreich ist, versucht TADDM stattdessen anschließend über IBM Tivoli Monitoring eine Verbindung zu den Endpunkten herzustellen, und auch in diesem Fall nur unter der Voraussetzung, dass die Erkennung mit IBM Tivoli Monitoring für die Endpunkte zulässig ist.

Sie können mit dieser Eigenschaft auch einen benutzerdefinierten Erkennungsbereich angeben, wie in folgendem Beispiel dargestellt:

com.ibm.cdb.session.prefer.ITM.*IP-Adresse*=false
Im folgenden Beispiel wird angegeben, dass TADDM den Erkennungsbereich `10.20.30.40` verwendet und eine direkte Verbindung zu den Tivoli Monitoring-Endpunkten herstellt:

```
com.ibm.cdb.session.prefer.ITM.10.20.30.40=false
```

com.ibm.cdb.session.prefer.ITM.Level_3_Discovery=false
Der Standardwert ist `false` und bedeutet, dass TADDM eine direkte Verbindung zu den Tivoli Monitoring-Endpunkten herstellt, falls Sie ein Erkennungsprofil der Ebene 3 verwenden. Bei allen anderen Erkennungsebenen verwendet TADDM stattdessen IBM Tivoli Monitoring für die Erkennung von Tivoli Monitoring-Endpunkten, was von den Werten der folgenden Eigenschaften abhängt:

- **com.ibm.cdb.session.allow.ITM**
- **com.ibm.cdb.session.prefer.ITM**

Diese Eigenschaft gibt an, ob TADDM IBM Tivoli Monitoring für die Erkennung von Tivoli Monitoring-Endpunkten mit einem Erkennungsprofil der Ebene 3 verwendet.

Wenn Sie den Wert auf `true` setzen, kann TADDM IBM Tivoli Monitoring für die Erkennung von Tivoli Monitoring-Endpunkten mit einem Erkennungsprofil der Ebene 3 verwenden.

Eigenschaften zur Optimierung der Verbindung zwischen dem TADDM-Server und dem Portalserver

Bei einer Erkennung der Ebene 2 mit IBM Tivoli Monitoring verwendet TADDM die folgenden TADDM-Servereigenschaften zur Optimierung der Verbindung zwischen dem TADDM-Server und dem Tivoli Enterprise Portal Server, falls es bei der Verbindung zu Verzögerungen kommt:

com.collation.discover.agent.ITM.CmdWrapperSelectionPattern=
Diese Eigenschaft gibt die Befehle an, die in ein Script eingeschlossen werden müssen, wenn eine Erkennung in einer IBM Tivoli Monitoring-Umgebung ausgeführt wird.

com.collation.platform.session.ITMSessionConnectionCooldownPeriod=60000
Diese Eigenschaft gibt das Zeitintervall in Millisekunden an, bis die Verbindung zum Tivoli Enterprise Portal Server nach der Ermittlung eines Fehlers reinitialisiert wird.

com.collation.platform.session.ITMSessionConnectionRetryLimit=5
Diese Eigenschaft gibt die Anzahl der Versuche für den Zugriff auf eine Verbindung an, für den Fall, dass die einleitende Verbindung fehlgeschlagen ist, bevor eine Fehlermeldung ausgegeben wird.

com.collation.platform.session.ITMSessionNumProgressChecks=600
Diese Eigenschaft gibt an, wie oft der Fortschritt beim Aufbau einer Verbindung überprüft wird, bevor die Verbindung als fehlgeschlagen gilt.

com.collation.platform.session.ITMSessionProgressCheckInterval=1000
Diese Eigenschaft gibt das Zeitintervall in Millisekunden zwischen den Überprüfungen des Verbindungsfortschritts an.

Eigenschaften für die Erkennung mit OSLC Automation Session

Diese Eigenschaften gelten für die Erkennung mit OSLC Automation Session.

Eigenschaften für die Integration über OSLC

com.ibm.cdb.topobuilder.integration.oslc.automationprovider
Diese Eigenschaft gibt die URL-Adressen derjenigen OSLC Execute Automation-Service-Provider an, die nicht in Jazz SM Registry Services registriert sind.

Folgendes Beispiel zeigt die URL-Adressen von OSLC Execute Automation-Service-Providern für ITM:

```
com.ibm.cdb.topobuilder.integration.oslc.automationprovider=
http://<AUTOMATION_PROVIDER_INSTALLATION_HOST>:15210/itmautomationprovider
```

Folgendes Beispiel zeigt, wie die URL-Adressen für mehrere OSLC Execute Automation-Service-Provider angegeben werden:

```
com.ibm.cdb.topobuilder.integration.oslc.automationprovider.1=
http://9.1.1.1:15210/itmautomationprovider
com.ibm.cdb.topobuilder.integration.oslc.automationprovider.2=
http://9.2.2.2:15210/itmautomationprovider
```

com.ibm.cdb.topobuilder.integration.oslc.automation.scope.alwaysrefresh=`false`
Der Standardwert ist `false`.

Diese globale Eigenschaft gibt an, ob OSLCAutomationAgent die Bereichsgruppen bei jeder Ausführung neu erstellt. Für die Neuerstellung der Bereichsgruppen ist eine Verbindung mit Jazz SM Registry Services oder OSLC Execute Automation-Service-Providern (bzw. beidem) erforderlich.

Wenn diese Eigenschaft auf `true` gesetzt ist, erstellt der Agent die Bereichsgruppen neu, selbst wenn sich der vom OSLC Execute Automation-Service-Provider bereitgestellte Automationsplan seit der letzten Ausführung des Agenten nicht geändert hat.

com.ibm.cdb.topobuilder.integration.oslc.frsurl
Diese Eigenschaft gibt die IP-Adresse der Jazz SM Registry Services (FRS) an, die bei der Integration mit anderen Produkten über OSLC verwendet wird. Die Adresse der Jazz SM Registry Services muss das folgende Format haben:

```
protocol://IP_oder_Hostname:Port
```

Diese Eigenschaft wird auch von OSLCAgent verwendet.

com.ibm.cdb.topobuilder.integration.oslc.automation.frsurl
Diese Eigenschaft gibt die IP-Adresse in Form des vollständigen Pfads zur Registrierungsobjektgruppe der Jazz SM Registry Services (FRS) an. Sie kann verwendet werden, wenn Jazz SM Registry Services nicht den standardmäßigen Servicepfad `/oslc` verwendet.

Eigenschaften für die Erkennung mit Automation Session

com.ibm.cdb.session.OSLCAutomation.deleteSudoFromCommands=false
Der Standardwert ist 'false'.

Bei dieser Eigenschaft handelt es sich um eine bereichsorientierte Eigenschaft, die für Ziele auf 'true' gesetzt werden kann, für die sudo aus Befehlen entfernt werden muss, die an OSLC gesendet werden

Verwendungsbeispiel:

```
com.ibm.cdb.session.OSLCAutomation.deleteSudoFromCommands=true
com.ibm.cdb.session.OSLCAutomation.deleteSudoFromCommands.9.100.100.200=true
com.ibm.cdb.session.OSLCAutomation.deleteSudoFromCommands.scope_name1=true
```

com.ibm.cdb.session.oslcautomation.pluginId=com.ibm.cdb.session.oslcautomation_1.0.0
Der Standardwert ist `com.ibm.cdb.session.oslcautomation_1.0.0`.

Diese Eigenschaft gibt die OSGi-Bundle-ID des OSLC Automation Session-Plug-ins an.

com.ibm.cdb.session.itm.endpointClass=com.collation.platform.session.oslcautomation.OSLCAutomationEndpoint
Der Standardwert ist `com.collation.platform.session.oslcautomation.OSLCAutomationEndpoint`.

Diese Eigenschaft gibt die zu verwendende Endpunktklasse an.

com.ibm.cdb.session.allow.OSLCAutomation=true
Der Standardwert ist `true`.

Diese bereichsorientierte Eigenschaft gibt an, ob TADDM für die Erkennung OSLC Automation Session verwenden kann.

Verwendungsbeispiel:

```
com.ibm.cdb.session.allow.OSLCAutomation=true
com.ibm.cdb.session.allow.OSLCAutomation.9.100.1.0=true
com.ibm.cdb.session.allow.OSLCAutomation.scope_set2=true
```

com.ibm.cdb.session.prefer.OSLCAutomation=true
Der Standardwert ist `true`.

Diese bereichsorientierte Eigenschaft gibt an, ob OSLC Automation Session für eine Erkennung die bevorzugte Sitzung ist. Der Wert dieser Eigenschaft hat Vorrang vor allen anderen bevorzugten Werten (beispielsweise einer ITM-Standardsitzung).

Verwendungsbeispiel:

```
com.ibm.cdb.session.prefer.OSLCAutomation=true
com.ibm.cdb.session.prefer.OSLCAutomation.9.100.100.200=true
com.ibm.cdb.session.prefer.OSLCAutomation.scope_name1=true
```

com.ibm.cdb.session.oslcautomation.timeout.httpconnect=60000
Der Standardwert ist `60000` (60 Sekunden). Der Wert wird in Millisekunden angegeben.

Diese globale Eigenschaft gibt das Zeitlimit für die Verbindung mit dem OSLC Execute Automation-Service-Provider an.

com.ibm.cdb.session.oslcautomation.timeout.httpread=240000
Der Standardwert ist `240000` (4 Minuten). Der Wert wird in Millisekunden angegeben.

Diese globale Eigenschaft gibt das Zeitlimit für das Lesen der Daten vom OSLC Execute Automation-Service-Provider an.

com.ibm.cdb.session.oslcautomation.request.async.maxretries=60
Der Standardwert ist `60`.

Diese globale Eigenschaft gibt die maximale Anzahl aufeinanderfolgender Anforderungen bei asynchron generierten automatischen Ergebnissen (AutomationResults) an.

com.ibm.cdb.session.oslcautomation.request.async.delay=10000
Der Standardwert ist `10000` (10 Sekunden). Der Wert wird in Millisekunden angegeben.

Diese globale Eigenschaft gibt die Verzögerung zwischen aufeinanderfolgenden Anforderungen bei asynchron generierten automatischen Ergebnissen (AutomationResults) an.

Anmerkung: Fix Pack 4 Falls sich wegen Zeitlimitüberschreitungen keine SSH-Sitzung mit dem Server herstellen lässt, versuchen Sie, den optimalen Einstellungswert für die nachfolgende Eigenschaft zu konfigurieren:

com.collation.mindterm.Ssh2Preferences= hello-timeout=30; alive = 25; compression= 9

com.collation.discover.agent.app.packagedapp.mysap.SLDServerPortList = 51200
Mit Hilfe dieser Eigenschaft können Sie den SLD-Port ändern und festlegen, dass der angegebene Port zur Sensorkonfiguration hinzugefügt werden soll.

com.ibm.cdb.security.auth.cache.itm.disabled=true
Der Standardwert ist `true`.

Diese Eigenschaft legt fest, ob das Caching für Berechtigungsnachweise für die OSLC-Erkennung aktiviert ist.

Bei dieser Eigenschaft handelt es sich um eine bereichs- und profilorientierte Eigenschaft. Sie können eine IP-Adresse, den Namen einer Bereichsgruppe oder einen Profilnamen anhängen. Darüber hinaus können Sie die Eigenschaft in der Profilkonfiguration der Discovery Management Console festlegen.

Eigenschaften für die Anpassung der DNS-Suche

Diese Eigenschaften gelten für die Anpassung der DNS-Suche.

com.collation.platform.os.disableDNSLookups=false
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Wird diese Eigenschaft auf `true` gesetzt, sind DNS-Suchvorgänge für den TADDM-Server inaktiviert.

com.collation.platform.os.disableRemoteHostDNSLookups=false
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Wenn Sie die Eigenschaft in `true` ändern, werden Namenssuchen (nur DNS) auf erkannten fernen Hosts inaktiviert. Über diese Eigenschaft wird erzwungen, dass alle Namenssuchläufe auf dem TADDM-Server erfolgen.

com.collation.platform.os.command.fqdn=nslookup $1 | grep Name | awk '{print $2}'
Der Standardwert ist `nslookup $1 | grep Name | awk '{print $2}'`.

Mit diesem Befehl wird der vollständig qualifizierte Domänenname gefunden (fqdn: fully-qualified domain name). In den meisten Fällen wird diese Eigenschaft nicht benötigt, da der standardmäßige Algorithmus für den vollständig qualifizierten Domänennamen in den meisten Produktionsumgebungen funktioniert. Wenn diese Eigenschaft nicht benötigt wird, muss sie in Kommentarzeichen gesetzt werden. In Umgebungen, in denen sich der vollständig qualifizierte Domänenname aus dem Hostnamen ableitet, wird jedoch die Aktivierung dieser Eigenschaft empfohlen. Sie müssen diese Eigenschaft beispielsweise aktivieren, wenn Hostnamen im DNS als Aliasnamen konfiguriert sind.

Wenn diese Eigenschaft verwendet wird, muss das DNS verfügbar und korrekt konfiguriert sein. Andernfalls schlägt der Befehl **nslookup** wahrscheinlich fehl oder hat eine lange Antwortzeit.

Bei ihrer Aktivierung wird diese Eigenschaft nur auf dem TADDM-Server verwendet. Derzeit werden nur die Betriebssysteme AIX und Linux unterstützt. Auf TADDM-Servern unter Windows wird diese Eigenschaft nicht unterstützt.

Eigenschaften für die grafische Benutzerschnittstelle

Diese Eigenschaften gelten für die grafische Benutzerschnittstelle von TADDM.

Fix Pack 3 **com.ibm.cdb.gui.supportedJRE.warning=true**
Diese Eigenschaft gibt an, ob die Warnung CTJTG0034E beim Start der Discovery Management Console angezeigt wird. In dieser Nachricht werden Sie gewarnt, dass Sie eine nicht unterstützte Version der Java Runtime Environment verwenden. Wenn Sie TADDM zusammen mit der nicht unterstützten Version der Java Runtime Environment verwenden möchten und diese Nachricht nicht angezeigt werden soll, setzen Sie diese Eigenschaft auf `false`.

Der Standardwert für diese Eigenschaft lautet `true`.

Eigenschaften für den JVM-Speicher der grafischen Benutzerschnittstelle

Diese Eigenschaften gelten für den JVM-Speicher der grafischen Benutzerschnittstelle.

com.collation.gui.initial.heap.size=128m

Der Standardwert ist `128m`.

Anfangsgröße des Heapspeichers für die TADDM-Benutzerschnittstelle.

com.collation.gui.max.heap.size=512m

Der Standardwert ist `512m` .

Maximale Größe des Heapspeichers für die TADDM-Benutzerschnittstelle.

Diese Eigenschaften sind für kleine TADDM-Domänen angemessen. Zur Dimensionierung werden die folgenden Kategorien von TADDM-Servern verwendet (basierend auf baugleichen Servern):

- Klein: bis zu 1000 baugleiche Server
- Mittel: 1000 - 2500 baugleiche Server
- Groß: 2500 - 5000 baugleiche Server

Durch die Vergrößerung dieser Werte für mittlere und große Umgebungen wird das Leistungsverhalten einiger Operationen in der Benutzerschnittstelle verbessert. Einige Ansichten werden nicht fehlerfrei aufgebaut, wenn für TADDM zu diesem Zeitpunkt nicht genügend Speicher zur Verfügung steht.

Für eine mittlere Umgebung:

com.collation.gui.initial.heap.size=256m

Der Standardwert ist `256m`.

com.collation.gui.max.heap.size=768m

Der Standardwert ist `768m`.

Für eine große Umgebung:

com.collation.gui.initial.heap.size=512m

Der Standardwert ist `512m`.

com.collation.gui.max.heap.size=1024m

Der Standardwert ist `1024m`.

Eigenschaften der Ports der grafischen Benutzerschnittstelle

Diese Eigenschaften gelten für die Ports der grafischen Benutzerschnittstelle.

com.collation.tomcatshutdownport=9436 (nur TADDM 7.3.0)
Der Standardwert ist 9436.

Dieser Port wird für den Tomcat-Systemabschlussbefehl verwendet.

com.ibm.cdb.service.web.port=9430
Der Standardwert ist 9430.

Der HTTP-Port wird ohne SSL verwendet.

com.ibm.cdb.service.web.secure.port=9431
Der Standardwert ist 9431.

Der HTTPS-Port wird mit SSL verwendet.

com.ibm.cdb.service.ClientProxyServer.port=9435
Der Standardwert ist 9435.

Der RMI-Datenport wird ohne SSL verwendet.

com.ibm.cdb.service.SecureClientProxyServer.secure.port=9434
Der Standardwert ist 9434.

Der RMI-Datenport wird mit SSL verwendet.

com.ibm.cdb.service.registry.public.port=9433
Der Standardwert ist 9433.

Der öffentliche Port für die Service-Registry.

Eigenschaften für LDAP
Diese Eigenschaften gelten für LDAP.

Für die Benutzerauthentifizierung kann ein externer LDAP-Server verwendet werden. Sowohl die anonyme Authentifizierung als auch die Authentifizierung durch Kennwörter werden durch einen externen LDAP-Server unterstützt.

In der Datei `collation.properties` können der Hostname, die Portnummer, der Basis-DN, der eindeutige Name für Bindung und das Kennwort (für die Kennwortauthentifizierung) des LDAP-Servers konfiguriert werden. Sie können auch ein bestimmtes Namensattribut konfigurieren, das mit der Benutzer-ID (UID) übereinstimmt und nach dem dann gesucht werden kann.

Die LDAP-Konfiguration wird für Implementierungen empfohlen, in denen Synchronisations- und Domänenserver verwendet werden. Konfigurieren Sie in einer Unternehmensumgebung den Domänen- und den Synchronisationsserver so, dass beide dieselbe Benutzerregistry verwenden. Beim Anmelden an einem Domänenserver, der mit einem Synchronisationsserver verbunden ist, wird die Anmeldung auf dem Synchronisationsserver verarbeitet. Tritt zwischen dem Synchronsations- und einem Domänenserver ein Problem mit der Netzverbindung auf und ist der Domänenserver so konfiguriert, dass er dieselbe Benutzerregistry wie der Synchronisationsserver verwendet, können Sie sich ohne Rekonfiguration erfolgreich am Domänenserver anmelden.

com.collation.security.auth.ldapAuthenticationEnabled=true
Der Standardwert ist `true`.

Über diese Eigenschaft wird die LDAP-Authentifizierung aktiviert.

com.collation.security.auth.ldapBaseDN=ou=People,dc=ibm,dc=com
Der Standardwert ist `ou=People,dc=ibm,dc=com`.

Diese Eigenschaft definiert den LDAP-Basis-DN. Der LDAP-Basis-DN ist der Ausgangspunkt für alle LDAP-Suchvorgänge.

com.collation.security.auth.ldapBaseGroupDN
In der Datei `collation.properties` ist diese Eigenschaft standardmäßig als Kommentar gekennzeichnet.

Diese Eigenschaft definiert den LDAP-Stammzweig zum Durchsuchen von Gruppen, der sich vom Stammzweig für alle LDAP-Abfragen unterscheiden kann. Wenn Sie mehrere LDAP-Stammzweige für die Suche nach Gruppen angeben möchten, trennen Sie die Namen der Zweige mit dem Zeichen ";".

Wenn Sie für diese Eigenschaft keinen Wert angeben, ist der Standardwert der Wert der Eigenschaft `com.collation.security.auth.ldapBaseDN`.

com.collation.security.auth.ldapBindDN=uid=ruser,dc=ibm,dc=com
Der Standardwert ist `uid=ruser,dc=ibm,dc=com`.

Wenn die einfache Authentifizierung verwendet wird, definiert diese Eigenschaft die Benutzer-ID, die für die Authentifizierung in LDAP verwendet wird.

Wichtig:

- Wenn ein Wert für `com.collation.security.ldapBindDN` nicht angegeben oder die Eigenschaft mit Kommentarzeichen versehen ist, wird eine anonyme Verbindung mit LDAP versucht. Das

folgende Beispiel zeigt, wie die Eigenschaft mit dem Nummernzeichen (#) als Kommentar gekennzeichnet wird:

```
#com.collation.security.auth.ldapBindDN=uid=ruser,
dc=ibm,dc=com
```

- Wenn für `com.collation.security.auth.ldapBindDN` ein Wert angegeben ist, wird die einfache Authentifizierung verwendet und
- für `com.collation.security.auth.ldapBindPassword` muss ebenfalls ein Wert angegeben werden.

com.collation.security.auth.ldapBindPassword=ruser
Der Standardwert ist `ruser`.

Wenn die einfache Authentifizierung verwendet wird, definiert diese Eigenschaft das Benutzerkennwort, das für die Authentifizierung in LDAP verwendet wird.

com.collation.security.auth.ldapClientKeyStore=*Schlüsselspeicherpfad*
Die Eigenschaft definiert die Position des Schlüsselspeichers, der die Zertifikate auf dem TADDM-Server enthält. Der Speicher muss das Clientzertifikat enthalten, damit der TADDM-Server beim LDAP-Server authentifiziert werden kann.

com.collation.security.auth.ldapClientKeyStorePassphrase=*Schlüsselspeicherkennphrase*
Optional: Diese Eigenschaft definiert das Kennwort für den Schlüsselspeicher.

com.collation.security.auth.ldapClientTrustStore=*TS-Pfad*
Die Eigenschaft definiert die Position des Truststores, der die Zertifikate auf dem TADDM-Server enthält. Der Speicher muss das LDAP-Serverzertifikat enthalten.

com.collation.security.auth.ldapClientTrustStorePassphrase=*TS-Kennphrase*
Optional: Diese Eigenschaft definiert das Kennwort für den Truststore.

com.collation.security.auth.ldapGroupMemberAttribute=member
Der Standardwert ist `member`.

Diese Eigenschaft definiert den Namen des Attributs, das die Mitglieder einer Gruppe in LDAP enthält.

com.collation.security.auth.ldapGroupNamingAttribute=cn
Der Standardwert ist `cn`.

Diese Eigenschaft definiert den Namen des Attributs, das für die Benennung von Benutzergruppen in LDAP verwendet wird.

com.collation.security.auth.ldapGroupObjectClass=groupofnames
Der Standardwert ist `groupofnames`.

Diese Eigenschaft definiert die Klasse, die die Benutzergruppen in LDAP repräsentiert.

com.collation.security.auth.ldapHostName=ldap.ibm.com
Der Standardwert ist `ldap.ibm.com`.

Diese Eigenschaft definiert den Hostnamen für den LDAP-Server.

com.collation.security.auth.ldapPortNumber=389
Der Standardwert ist 389.

Diese Eigenschaft definiert den Port für den LDAP-Server.

com.collation.security.auth.ldapUIDNamingAttribute=uid
Der Standardwert ist `uid`.

Diese Eigenschaft definiert den Namen des Attributs, das für die Benennung von Benutzern in LDAP verwendet wird.

com.collation.security.auth.ldapUserObjectClass=person
Der Standardwert ist `person`.

Diese Eigenschaft definiert den Namen der Klasse, die die Benutzer in LDAP repräsentiert.

com.collation.security.auth.ldapUseSSL=false
Der Standardwert ist `false`.

Die Eigenschaft wird für die Aktivierung der Authentifizierung in einer LDAP-Benutzerregistry mit einer SSL-Verbindung verwendet.

com.collation.security.usermanagementmodule=ldap
Der Standardwert ist `ldap`.

Diese Eigenschaft definiert das durch den TADDM-Server verwendete Benutzermanagementmodul. Gültige Werte sind:

- `file` (für eine dateibasierte Benutzerregistry). Der Standardwert ist 'true'.
- `ldap` (für eine LDAP-Benutzerregistry)
- `vmm` (für eine Benutzerregistry, die eingebundene WebSphere Application Server-Repositorys verwendet)

Lockouteigenschaften
Diese Eigenschaften gelten für Lockouts.

com.collation.security.lockout.treshold=3
Der Standardwert ist 3.

Diese Eigenschaft gibt die Gesamtzahl der fehlgeschlagenen Anmeldeversuche für einen bestimmten Benutzer an, bei der ein lokales Lockout für den betreffenden Benutzer ausgelöst wird.

com.collation.security.lockout.timeout=30
Der Standardwert ist 30.

Diese Eigenschaft gibt die Dauer in Minuten an, für die der Benutzer, durch den das lokale Lockout ausgelöst wurde, nach der Auslösung eines lokalen Lockouts aus TADDM ausgesperrt wird.

com.collation.security.lockout.globalthreshold=100
Der Standardwert ist 100.

Diese Eigenschaft gibt die Anzahl gleichzeitiger Einzelbenutzer-Lockouts an, die ein globales Lockout auslöst.

com.collation.security.lockout.globaltimeout=30
Der Standardwert ist 30.

Diese Eigenschaft gibt die Dauer in Minuten an, für die alle Benutzer bei Auslösung eines globalen Lockouts aus TADDM ausgesperrt werden.

com.collation.security.lockout.failedloginthreshold=1000
Der Standardwert ist 1000.

Diese Eigenschaft gibt die Gesamtzahl der fehlgeschlagenen Anmeldeversuche für eindeutige Benutzer an, bei der ein globales Lockout ausgelöst wird.

Protokollierungseigenschaften
Diese Eigenschaften gelten für die Protokollierung.

com.collation.log.filesize=20MB
Der Standardwert ist 20MB.

Die maximale Größe der Protokolldatei. Wenn die Datei diese Größe erreicht, wird eine neue Protokolldatei erstellt. Die aktuelle Protokolldatei wird mit der Dateiendung *.N* gespeichert. *N* ist die Nummer 1 in der Wertegruppe der Eigenschaft **com.collation.log.filecount**. Sie legen fest, wie viele Protokolldateien erstellt und erhalten werden, bevor die Dateien durch die Eigenschaft **com.collation.log.filecount** turnusmäßig wechseln.

Sie können die Anzahl an Bytes direkt oder in Kilobyte (KB) oder Megabyte (MB) eingeben.

Bei den folgenden Beispielen handelt es sich um gültige Werte der Größe von Protokolldateien:

- 1000000

- 512 KB
- 10 MB

com.collation.log.filecount=5
Der Standardwert ist 5.

Die Anzahl der von Ihnen verwalteten Protokolldateien.

com.collation.log.level.vm.*vmName*=INFO
Der Standardwert ist INFO

Legt die Protokollebene für jedes virtuelle System fest.

vmName ist ein virtuelles Java-System mit einem zugeordneten TADDM-Servicenamen. Die nachfolgende Liste enthält weitere gültige Optionen:

- Topology
- DiscoverAdmin
- EventsCore
- Proxy
- Discover
- EcmdbCore
- StorageService
- DiscoveryService

Die nachfolgende Liste enthält weitere gültige Optionen:

- FATAL
- ERROR
- WARNING
- INFO
- DEBUG (Beim Einstellen der DEBUG-Option verlangsamt sich die Systemleistung.)
- TRACE (Ein Einstellen der TRACE-Option führt zur Protokollierung von Kennwörtern.)

Leistungseigenschaften
Diese Eigenschaften gelten für die TADDM-Leistung.

com.collation.discover.dwcount=32
Der Standardwert ist 32. Hier muss ein ganzzahliger Wert angegeben werden.

Diese Eigenschaft wirkt sich auf die Erkennungsrate aus. Bei einem Worker-Thread der Erkennung handelt es sich um einen Thread, der Sensoren ausführt. Diese Eigenschaft gibt an, wie viele Worker-Threads der Erkennung gleichzeitig ausgeführt werden können, und sie gilt nur für einen Erkennungsserver in einer Streaming-Server-Implementierung oder für einen Domänenserver in einer Domänenserver-Implementierung.

Bei einer Erkennung mit IBM Tivoli Monitoring (altes Verfahren mit IBM Tivoli Monitoring Scope-Sensor) muss der Wert auf 16 gesetzt werden. Für alle anderen Erkennungstypen liegt der gültige Wertebereich zwischen 32 und 160.

com.collation.discover.observer.topopumpcount=16
Der Standardwert ist 16. Hier muss ein ganzzahliger Wert angegeben werden.

Diese Eigenschaft wirkt sich auf die Speicherrate der Erkennungsergebnisse in der TADDM-Datenbank aus. Sie gibt die Anzahl der Writer-Threads an, die für die Kommunikation mit der TADDM-Datenbank erstellt werden.

Bei einem Erkennungsserver in einer Streaming-Server-Implementierung steuert diese Eigenschaft die Anzahl der Threads, mit denen der Erkennungsserver Erkennungsergebnisse an den Speicherserverpool sendet.

Bei einem Speicherserver in einer Streaming-Server-Implementierung steuert diese Eigenschaft die Anzahl der Threads, die ursprünglich gestartet wurden, um Erkennungsergebnisse von den Erkennungsserverinstanzen zu erhalten. Diese Threads nutzen eine Datenbankverbindung vom Verbindungspool für die Kommunikation mit der TADDM-Datenbank. Wenn die gepoolte JDBC-Verbindung nicht verfügbar ist und eine Anforderung zum Speichern von Daten empfangen wird, erstellen dies e Threads eine nicht gepoolte Verbindung. Wenn die Datenbank nicht bei jedem Speichern oder Anfordern eines Objekts aus der Datenbank eine neue Datenbankverbindung erstellen, sondern eine gepoolte Verbindung verwendet werden soll, legen Sie 'topopumpcount' auf dem Speicherserver als Summe der topopumpcount-Werte aller Erkennungsserver fest. Der Standardwert beträgt zwar 16, aber wenn die Summe der topopumpcount-Werte für alle Erkennungsserver den Wert von 'topopumpcount' auf den Speicherservern übersteigt, auf denen die Ergebnisse gespeichert werden, können Sie die Anzahl der Datenbankthreads erhöhen.

Bei einem Domänenserver in einer Domänenserver-Implementierung steuert diese Eigenschaft die Anzahl der Threads, die Erkennungsergebnisse von den Worker-Threads der Erkennung erhalten.

Die Threads verwenden dann eine Datenbankverbindung aus dem Verbindungspool für die Kommunikation mit der TADDM-Datenbank (beispielsweise zum Speichern der Ergebnisse und für den Abruf der Daten). Wenn keine weiteren in Pools zusammengefassten JDBC-Verbindungen mehr vorhanden sind, erstellt der Thread eine Verbindung, die nicht in einem Pool zusammengefasst ist.

com.ibm.cdb.discover.observer.topopump.threshold=0.7
com.ibm.cdb.discover.observer.topopump.threshold.*Name_der_Topologieagentengruppe*=0.7
Der Standardwert ist `0,7`. Der Wert muss eine Gleitkommakonstante sein.

Diese Eigenschaft gibt den Bruchteil der Datenbank-Writer-Threads an, die Sie starten können, wenn die Topologieagenten ausgeführt werden. Sie können den Schwellenwert für eine bestimmte Agentengruppe gesondert oder alle Werte gemeinsam angeben. Ist für eine Agentengruppe kein Wert definiert, wird der allgemeine Schwellenwert verwendet. Dieser Wert ermöglicht die Begrenzung der Threads, die die Erkennungsergebnisse in der TADDM-Datenbank speichern, wenn die Topologieagenten ausgeführt werden.

com.ibm.cdb.typesServiceRefreshInterval=120
Der Standardwert ist `120`. Als Mindestwert kann `30`, als Maximalwert `1800` angegeben werden.

Diese Eigenschaft gibt das Aktualisierungsintervall in Sekunden an, in dem Komponententypen bei der Erstellung einer angepassten Abfrage, bei der Anzeige eines Änderungsprotokolls oder bei der Anzeige von Informationen zu einem Komponentenvergleich aktualisiert werden sollen.

com.ibm.cdb.ea.metaRefreshFrequency=20
Der Standardwert ist `20`. Hier muss ein ganzzahliger Wert angegeben werden.

Diese Eigenschaft gibt das Aktualisierungsintervall in Sekunden für die Aktualisierung der Informationen zu den definierten erweiterten Attributen (z. B. auf den Speicherservern) an.

Fix Pack 8 Eigenschaften der Kennwortrichtlinie

Diese Kennwortrichtlinie gilt nur für die Authentifizierung des dateibasierten TADDM-Repositorys. Weitere Einzelheiten zur Sicherheit und Authentifizierung finden Sie im Abschnitt 'Planung der Sicherheit' im *Installationshandbuch*.

Wichtig: Die folgenden Eigenschaften für die Kennwortrichtlinie müssen im primären Speicherserver und allen Erkennungsservern in der Datei `collation.properties` konfiguriert werden. Stellen Sie sicher, dass die konfigurierten Werte für die Eigenschaften der Kennwortrichtlinie auf allen Servern identisch sind.

com.collation.passwordpolicy=false
Der Standardwert ist `false`.

Mit dieser Eigenschaft wird die Kennwortrichtlinie für die Authentifizierung des dateibasierten Repositorys aktiviert. Zur Aktiviert der Kennwortrichtlinie setzen Sie diese Eigenschaft auf '`true`'.

com.collation.passwordpolicy.minlength=15
Der Standardwert ist 15.

Mit dieser Eigenschaft wird die Mindestlänge der Zeichen für das Kennwort geändert.

com.collation.passwordpolicy.MinCharTypes=2
Der Standardwert ist 2.

Mit dieser Eigenschaft wird erzwungen, dass das Kennwort mindestens zwei der folgenden Zeichentypen enthält:

- Großbuchstaben
- Kleinbuchstaben
- Zahlen
- Sonderzeichen

SSH-Eigenschaften (Secure Shell)
Diese Eigenschaften gelten für die Secure Shell (SSH).

Fix Pack 1 **com.ibm.cdb.platform.SshVersionSessionSkipList**
Diese Eigenschaft gibt die Versionen von SSH-Servern an, für die die Sitzung nicht eingerichtet wird. Bei solchen Servern wird der Sitzungssensor fehlerfrei beendet.

Der Wert dieser Eigenschaft ist eine durch Kommas getrennte Liste, z. B. `Cisco, Data ONTAP, SSH-2.0-OpenSSH_5.9 PKIX FIPS, OpenSSH_OA`.

com.collation.SshLogInput=false
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Wenn Sie diesen Wert auf `true` setzen, wird die SSH-Eingabe protokolliert.

com.collation.SshPort=22
Der Standardwert ist 22. Hier muss ein ganzzahliger Wert angegeben werden.

Diese Eigenschaft gibt den Port an, den der Server für alle SSH-Verbindungen verwendet.

com.collation.SshSessionCommandTimeout=120000
Der Standardwert ist `120000`. Hier muss ein ganzzahliger Wert angegeben werden.

Dieser Wert zeigt die zulässige Dauer (in Millisekunden) zur Ausführung des SSH-Befehls an. Wird diese Eigenschaft von einem Agenten aus verwendet, muss ihr Wert kleiner als der Wert für die Eigenschaft **AgentRunnerTimeout** sein, damit er wirksam ist.

com.collation.SshWeirdReauthErrorList=Permission denied

Diese Eigenschaft ermöglicht die Wiederholung der Benutzername/Kennwort-Paare, die zuvor bei Erkennungsläufen erfolgreich verwendet wurden. Die Eigenschaft ist erforderlich, weil Windows-Systeme in zufälliger Abfolge gültige Anmeldeversuche ablehnen. Die Eigenschaft muss die Einstellung `Permission denied` (Berechtigung abgelehnt) aufweisen. Diese Eigenschaft darf nicht geändert werden.

com.collation.WmiInstallProviderTimeout=240000
Der Standardwert ist `240000`. Hier muss ein ganzzahliger Wert angegeben werden.

Dieser Wert zeigt die zulässige Wartedauer (in Millisekunden) bis zur Ausführung des WMI InstallProvider-Scripts an.

com.collation.SshSessionReuseSuppressList
Einige Versionen des SSH-Servers unterstützen die Wiederverwendung von Verbindungen nicht so, wie durch TADDM implementiert. Diese SSH-Serverversionen müssen dieser Eigenschaft hinzugefügt werden, damit TADDM Ziele, auf denen diese Versionen ausgeführt werden, erfolgreich erkennen kann.

Der Wert dieser Eigenschaft ist eine durch Kommas getrennte Liste. Es reicht völlig aus, nur den Anfang der SSH-Serverversion anzugeben. Beispiel: `SSH-2.0-BoKS_SSH_6`.

Die SSH-Serverversionen finden Sie in der Protokolldatei des Sitzungssensors.

Eigenschaften für die Sicherheit
Diese Eigenschaften gelten für die Sicherheit.

Fix Pack 3 **com.ibm.cdb.secure.server=false**
Der Standardwert ist `false`.

Diese Eigenschaft gibt an, ob alle TADDM-Services von den öffentlichen und externen RMI-Registrys sicher sind. Ist sie auf `true` gesetzt, werden alle öffentlichen Services, die nicht sicher sind (ClientProxyServer und API-Server), in die interne RMI-Registry verschoben. Außerdem wird das SSL-Protokoll für externe Services (z. B. RegistriesURLProvider, SecurityManager und TopologyManager) erzwungen.

Wenn Sie diese Eigenschaft auf `true` setzen, müssen Sie auch die Eigenschaften `com.collation.security.enablesslforconsole` und `com.collation.security.enforceSSL` auf `true` setzen.

Diese Eigenschaft kann sich auf die Integration mit anderen Produkten auswirken, die nicht gesicherte Verbindungen mit TADDM herstellen.

Wenn Sie den Standardwert dieser Eigenschaft ändern, tun Sie es an folgenden Positionen:

- `$COLLATION_HOME/dist/etc/collation.properties`
- `$COLLATION_HOME/dist/sdk/etc/collation.properties`
- `sdk/etc/collation.properties` jeder TADDM-SDK-Installation

Fix Pack 5 **com.ibm.cdb.rmi.registry.secure=false**
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Zum Aktivieren des sicheren Registry-Modus setzten Sie dieses Flag auf `true`.

Wenn der Server im sicheren Registry-Modus (`com.ibm.cdb.rmi.registry.secure=true`) ausgeführt wird, wird der folgende Port mit einem SSL-Protokoll geschützt: `com.ibm.cdb.service.registry.public.port (Standardwert: 9433)`

Wenn der Server im sicheren Registry-Modus (`com.ibm.cdb.rmi.registry.secure=true`) ausgeführt wird, muss das Kontrollkästchen 'Sichere Sitzung (SSL) aufbauen' beim Starten der Data Management Console aktiviert sein.

Fix Pack 1 **com.ibm.cdb.secure.liberty=false**
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Zur Inaktivierung nicht sicherer Ports setzen Sie dieses Flag auf `true`.

com.collation.security.privatetruststore=true
Der Standardwert ist `true`.

Gültige Werte sind `true` oder `false`. Wenn SSL aktiviert ist, muss dieser Wert auf `true` gesetzt werden.

com.collation.security.enablesslforconsole=true
Der Standardwert ist `true`.

Gültige Werte sind `true` oder `false`.

com.collation.security.enabledatalevelsecurity=false
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Wenn der Zugriff auf TADDM-Objektgruppen nach Benutzer oder Benutzergruppe eingeschränkt werden soll, müssen Sie diesen Wert auf `true` setzen.

com.collation.security.enforceSSL=false
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Legen Sie für das Flag `true` fest, um nicht gesicherte Verbindungen zu inaktivieren und die Verwendung von SSL-Verbindungen zu erzwingen.

com.collation.security.usermanagementmodule=file
Der Standardwert ist `file`.

Für diese Eigenschaft stehen drei Optionen zur Verfügung:

- `file` (für eine dateibasierte TADDM-Benutzerregistry)
- `ldap` (für eine LDAP-Benutzerregistry)
- `vmm` (für eine Benutzerregistry, die eingebundene WebSphere Application Server-Repositorys verwendet)

com.collation.security.auth.sessionTimeout=240
Der Standardwert ist 240. Hier muss ein ganzzahliger Wert angegeben werden.

com.collation.security.auth.searchResultLimit=100
Der Standardwert ist 100. Hier muss ein ganzzahliger Wert angegeben werden.

Verwenden Sie diese Eigenschaft, wenn es viele Benutzer gibt.

Wichtig: Wenn ein LDAP- oder WebSphere Federated-Repository mehr als 100 Benutzer enthält, erhöhen Sie diesen Wert, damit die erwartete Anzahl von Benutzern unterstützt wird. Beispiel: `com.collation.security.auth.searchResultLimit=150`

com.collation.security.auth.websphereHost=localhost
Der Standardwert ist `localhost`.

Geben Sie den vollständig qualifizierten Domänennamen des Systems ein, das die Funktionen der eingebundenen Repositorys von WebSphere Application Server hostet.

com.collation.security.auth.websperePort=2809
Der Standardwert ist 2809.

Hierbei muss es sich um einen ganzzahligen Wert handeln. Dieser Wert kennzeichnet den WebSphere-Systemport.

com.ibm.cdb.service.SecurityManager.port=9540
Bei anderen Servern als einem Synchronisationsserver:

Der Standardwert ist 9540.

Gibt den vom Sicherheitsmanager verwendeten Firewall-Port an.

Bei einem Synchronisationsserver:

Standardmäßig ist diese Eigenschaft nicht festgelegt.

Domänen verwenden für die Kommunikation mit einem Synchronisationsserver einen Port, der im Parameter **`com.collation.EnterpriseSecurityManager.port`** angegeben ist. Der Standardwert für diese Eigenschaft lautet 19433.

com.collation.cdm.analytics.authorizedRole=

Das Fenster **Analyse** kann auf eine bestimmte Rolle eingeschränkt werden. Diese Eigenschaft ist standardmäßig nicht in der Datei `collation.properties` definiert und das Fenster **Analyse** steht jedem zur Verfügung. Der Wert der Eigenschaft muss der Name der Rolle sein, die auf das Fenster zugreifen darf.

Der Zugriff auf die folgenden Bereiche des Fensters **Analyse** kann von der jeweiligen Rolle abhängig sein:

- Fix Pack 2 Gruppierungsmuster
- Bestandsübersicht
- Anwendungsübersicht
- Serviceübersicht
- Systembestand
- Software-Serverbestand

- BIRT-Berichte

com.collation.security.discoverOutsideScope=true
Der Standardwert ist `true`.

Gültige Werte sind `true` oder `false`. Wenn Sie die Erkennung von Elementen außerhalb des Bereichs inaktivieren wollen, setzen Sie dieses Flag auf `false`.

com.ibm.cdb.secure.tomcat=falsch (nur TADDM 7.3.0)
Der Standardwert ist `false`.

Gültige Werte sind `true` oder `false`. Zur Inaktivierung nicht sicherer Ports setzen Sie dieses Flag auf `true`.

com.ibm.cdb.http.ssl.protocol=TLS
Der Standardwert ist TLS.

Diese Eigenschaft ändert das vom Web-SSL-Port (HTTPS-Port) verwendete SSL-Protokoll, standardmäßig 9431. Sie können den Port mit der Eigenschaft `com.ibm.cdb.service.web.secure.port` festlegen.

Eine Liste der unterstützten Werte finden Sie in der IBM Java 7-Dokumentation unter http://www-01.ibm.com/support/knowledgecenter/SSYKE2_7.0.0/com.ibm.java.security.component.70.doc/security-component/jsse2Docs/protocols.html. Wenn Sie die sichersten Protokolle (z. B. `TLS v1.1` oder `TLS v1.2`) verwenden wollen, müssen Sie Ihren Web-Browser so einrichten, dass er diese Protokolle unterstützt. Zu reglementierende Protokolle beeinträchtigen u. U. auch die Integration mit anderen Produkten, die sich über den Web-SSL-Port mit TADDM verbinden.

Fix Pack 5 Wenn `com.ibm.cdb.http.ssl.protocol=TLSv1.2` und JAVA7 auf der Clientseite verwendet werden, müssen die folgenden Einstellungen aktualisiert werden:

```
<JAVA_HOME>/jre/lib/security/java.security
jdk.tls.disabledAlgorithms=SSLv2, SSLv3, TLSv1, TLSv1.1
```

Außerdem sollten TLSv1 und TLSv1.1 im Browser inaktiviert sein.

com.ibm.cdb.ssl.protocol=TLS
Diese Eigenschaft wird standardmäßig nicht zur Datei `collation.properties` hinzugefügt. Wird sie nicht hinzugefügt, hat sie den Standardwert TLS. Um diese Eigenschaft zu ändern, müssen Sie sie manuell mit dem neuen Wert zur Datei `collation.properties` hinzufügen.

Diese Eigenschaft ändert das SSL-Protokoll, das von folgenden Ports verwendet wird:

- Dem Port, an dem der API-Server für SSL-Anforderungen empfangsbereit ist, standardmäßig 9531. Sie können den Port mit der Eigenschaft `com.ibm.cdb.service.SecureApiServer.secure.port` festlegen.
- Dem RMI-Datenport, der mit SSL verwendet wird, standardmäßig 9434. Sie können den Port mit der Eigenschaft `com.ibm.cdb.service.SecureClientProxyServer.secure.port` festlegen.

Eine Liste der unterstützten Werte finden Sie in der IBM Java 7-Dokumentation unter http://www-01.ibm.com/support/knowledgecenter/SSYKE2_7.0.0/com.ibm.java.security.component.70.doc/security-component/jsse2Docs/protocols.html. Wenn Sie die sichersten Protokolle (z. B. `TLS v1.1` oder `TLS v1.2`) verwenden wollen, müssen Sie Ihren Web-Browser so einrichten, dass er diese Protokolle unterstützt. Zu strikte Protokolle können sich zudem auf die Integration mit anderen Produkten auswirken, die über die aufgelisteten Ports Verbindungen mit TADDM herstellen.

com.ibm.cdb.http.ssl.ciphers=
Chiffrierwerte werden für den LibertyServer festgelegt und die Kommunikation erfolgt nur an den angegebenen Chiffrierwerten. Andernfalls werden die Standardschlüssel ausgewählt, was schwache Algorithmen sein könnten.

com.ibm.cdb.rmi.ssl.protocol=
Die Eigenschaft `com.ibm.cdb.rmi.ssl.protocol` erleichtert die Aktivierung eines bestimmten Protokolls für die SSL-Verbindung, die in `com.ibm.cdb.ssl.protocol` erstellt wurde.

`com.ibm.cdb.rmi.ssl.protocol` muss aus der unterstützten Protokollliste in `com.ibm.cdb.ssl.protocol` stammen.

com.ibm.cdb.rmi.ssl.ciphers=
Mit dieser Eigenschaft können Sie die Verschlüsselungsalgorithmen für den RMI-Datenport und den Port festlegen, an dem dieser API-Server empfangsbereit ist.

Eigenschaften für temporäre Verzeichnisse
Diese Eigenschaften gelten für die Verwendung temporärer Verzeichnisse.

In temporären Verzeichnissen werden von TADDM unter bestimmten Bedingungen temporäre Dateien gespeichert. So werden in temporären Verzeichnissen beispielsweise Ankerprotokolldateien, Erkennungsscripts, die Ergebnisse eines Erkennungsvorgangs sowie die Informationen, die für manche Sensoren zur Ausführung einer Erkennung erforderlich sind, gespeichert. TADDM verwendet drei temporäre Verzeichnisse: `ANCHOR_DIR`, `ASD_TEMP_DIR` und `TADDM_TEMP_ROOT`.

com.ibm.cdb.taddm.anchor.root=.
Der Standardwert ist `.\`.

Dieser Eintrag gibt den Pfad des Verzeichnisses `ANCHOR_DIR` an, auf dem der Ankerserver implementiert ist. Es handelt sich um eine bereichsorientierte Eigenschaft, an die die IP-Adresse, der Name des Bereichs oder das Betriebssystem angehängt werden kann. Beispiel: `com.ibm.cdb.taddm.anchor.root.SunOS=`.

Für Windows-Systeme werden der folgende Eigenschaftsname und Standardwert verwendet:

```
com.ibm.cdb.taddm.anchor.root.Windows=%windir%\\temp\\
```

Der Eigenschaftswert verwendet Variablen, die auf Zielhosts aufgelöst werden. Die Variablen für Linux, AIX und SunOS müssen mit einem Dollarzeichen ($) als Präfix versehen werden. Variablen für Windows müssen in Prozentzeichen (%) eingeschlossen werden. Beispiel: `com.ibm.cdb.taddm.anchor.root=$TMP/taddmdirs/anchor` und `com.ibm.cdb.taddm.anchor.root.Windows=%TEMP%\\taddmdirs\\anchor`.

Wenn es sich bei dem aufgelösten Eigenschaftswert um einen relativen Verzeichnispfad handelt, wird er mit folgendem Präfix versehen:

- `%windir%\\temp\\` - für Windows
- Ausgangsverzeichnis - für AIX-, Linux- und SunOS-Systeme

Der Datei wird das Verzeichnis `taddmversion/anchor` als Suffix angehängt. Beispiel: `/home/taddmusr/taddm7.2.1/anchor` und `c:\Windows\Temp\taddm7.2.1\anchor`.

com.ibm.cdb.taddm.asd.temp
Dieser Eintrag gibt den Pfad des Verzeichnisses `ASD_TEMP_DIR` an, in dem Erkennungsscripts und die Ergebnisse von Erkennungsvorgängen gespeichert werden. Es handelt sich hier um eine bereichsorientierte Eigenschaft, die angepasst werden kann, indem die IP-Adresse oder das Betriebssystem angehängt werden.

In dem angegebenen Pfad wird das Verzeichnis `taddm`*`Version`*`/asd/` erstellt. Beispiel: `/tmp/taddm7.2.1/asd/`. Bei Angabe eines neuen Pfads müssen alle Benutzer eine Zugriffsberechtigung auf diesen Pfad haben.

com.ibm.cdb.taddm.file.temp=.
Der Standardwert ist `.\`.

Dieser Eintrag gibt den Pfad des Verzeichnisses `TADDM_TEMP_ROOT` an, in dem verschiedene Sensoren temporäre Daten speichern, die für die Ausführung einer Erkennung erforderlich sind. So speichern beispielsweise DB2®- und WebLogic-Sensoren temporäre Daten in diesem Verzeichnis.

Das Verzeichnis `TADDM_TEMP_ROOT` wird im Ausgangsverzeichnis in `taddm`*`Version`*`/temp/` erstellt. Beispiel: `/home/taddmusr/taddm7.2.1/temp/`.

Eigenschaften für das Topologieerstellungsprogramm
Diese Eigenschaften gelten für das Topologieerstellungsprogramm.

com.collation.topobuilder.RuntimeGcUnknownServerRetentionSpan=5
Der Standardwert ist 5.

Diese Eigenschaft gibt in Tagen an, wie lange unbekannte Prozesse erhalten bleiben sollen. Maximal kann ein Wert von 14 angegeben werden. Unbekannte Prozesse bestimmen, wann die angepassten Serverschablonen benötigt werden. Ohne regelmäßige Bereinigung häufen sich diese jedoch. Dadurch kann es zu Leistungsproblemen mit der Topologie kommen. Das Element für z/OS-Adressräume wird durch diese Verarbeitung nicht entfernt.

com.collation.topobuilder.RuntimeGcThreadCount=
Der Standardwert ist 4.

Durch diese Eigenschaft wird dem RuntimeGC-Agenten eine Parallelverarbeitung hinzugefügt, mit der die Leistung verbessert werden kann.

com.collation.topobuilder.agent.DerivedAppToAppDependencyAgent.ServiceDependency.enabled
Der Standardwert ist `false`.

Mit dieser Eigenschaft wird angegeben, ob der Topologieagent `DerivedAppToAppDependency` eine Abhängigkeit zwischen Geschäftsanwendungen erstellt, wenn ihre Mitglieder sich in einer Serviceabhängigkeit befinden.

Um dem Agenten die Erstellung einer solchen Abhängigkeit zu ermöglichen, setzen Sie die Eigenschaft auf `true` (wahr).

Eigenschaften für den Topologiemanager.
Diese Eigenschaften gelten für den Topologiemanager.

com.ibm.JdoQuery.FetchBatchSize=500
Der Standardwert ist 500.

Die Stapelgröße ist eine konfigurierbare Eigenschaft, die der Eigenschaft **kodo.FetchBatchSize** entspricht. Diese Eigenschaft stellt die Anzahl der Zeilen dar, die jeweils beim Blättern durch die Ergebnismenge einer Abfrage abgerufen werden soll.

com.ibm.cdb.service.TopologyManager.port=9550
Der Standardwert ist 9550.

Gibt den durch den Topologiemanager verwendeten Firewall-Port an.

Eigenschaften für den Anzeigemanager
Diese Eigenschaften gelten für den Anzeigemanager.

Fix Pack 2 **com.ibm.taddm.hideNetworkConnectionUnusedColumns.enabled**
Der Standardwert ist `false`.

Diese Eigenschaft gibt an, ob folgende Spalten der Registerkarte **Network Connections** im Data Management Portal angezeigt werden:

- **Flüsse**
- **Pakete**
- **Oktette**
- **Zuerst gesehen**
- **Zuletzt gesehen**

Um diese Spalten auszublenden, setzen Sie diese Eigenschaft auf `true` (wahr).

com.collation.view.maxnodes=500
Der Standardwert ist 500. Hier muss ein ganzzahliger Wert angegeben werden.

Diese Eigenschaft gibt die maximale Anzahl der Knoten an, die in einem Topologiediagramm im Datenmanagementportal angezeigt werden können. Wenn Sie die Eigenschaft auf einen höheren Wert setzen, können Sie umfangreichere Topologien anzeigen. Dadurch kann sich allerdings der Speicherbedarf erhöhen.

Überprüfen der Datenintegrität

Sie können den Befehl **verify-data** ausführen, um die Datenintegrität der Konfigurationselemente in der TADDM-Datenbank zu prüfen. Sie können Beziehungen, Übernahmezuordnungen, Duplikate und übergeordnetes Zusammenführen prüfen.

Vorbereitende Schritte

Führen Sie keine Erkennung, kein Massenladeprogramm und keine Synchronisation aus, wenn die Reparaturoption aktiviert ist. Das Datenintegritätstool analysiert ein großes Datenvolumen und die Verarbeitung kann unter Umständen einige Zeit dauern, insbesondere, wenn die Reparaturoption aktiviert ist. Der TADDM-Server muss betriebsbereit sein; stellen Sie sicher, dass er keine Tasks ausführt.

Informationen zu diesem Vorgang

Das Tool zur Überprüfung der Datenintegrität dokumentiert und repariert Datenintegritätsprobleme von Konfigurationselementen in der TADDM-Datenbank. Das ausführbare Script befindet sich im Verzeichnis `$COLLATION_HOME/bin`. Die Ergebnisse des Tools werden in der Datei `verify-data.log` dokumentiert und protokolliert. Sie können das Tool stoppen und zu einem beliebigen Zeitpunkt erneut ausführen.

Beziehungen prüfen

Sie können Überprüfungen für Beziehungsabfragen und Fremdschlüssel in allen Tabellen für Modelle und Schnittmengen durchführen.

Informationen zu diesem Vorgang

Wenn die Option zum Reparieren aktiviert ist, werden durch die Überprüfung von Beziehungen untergeordnete Objekte entfernt, wenn sich in der Datenbank kein übergeordnetes Objekt befindet, und es werden ungültige Fremdschlüsselwerte für Beziehungen gelöscht, die als nicht enthalten definiert sind. Möglicherweise wird auch eine erhebliche Anzahl von Konfigurationselementen einer niedrigeren Ebene gelöscht. Wenn die Elemente jedoch keinem übergeordneten Objekt zugeordnet sind, können sie sicher entfernt werden.

Prozedur

Zum Überprüfen von Beziehungen führen Sie einen der folgenden Befehle aus:

- **verify-data.sh -v ro [-a repair]**
- **verify-data.bat -v ro [-a repair]**

Zuordnung der Übernahme prüfen

Bei der Überprüfung der Zuordnung der Übernahme werden alle Tabellen abgefragt, die einer Konfigurationselementklasse zugeordnet sind, und es wird überprüft, ob alle Tabellen in jeder Zeile einen Eintrag enthalten.

Informationen zu diesem Vorgang

Wenn die Option zum Reparieren aktiviert ist, werden die Datensätze erneut erstellt.

Prozedur

Zum Überprüfen der Zuordnung der Übernahme führen Sie einen der folgenden Befehle aus:

- **verify-data.sh -v io [-a repair]**
- **verify-data.bat -v io [-a repair]**

Duplikate prüfen
Bei der Prüfung auf Duplikate werden doppelte Konfigurationselemente auf Basis von Feldwerten für die Namenskonvention in der Datenbank gesucht.

Informationen zu diesem Vorgang
Wenn die Option zum Reparieren aktiviert ist, werden doppelte Objekte zusammengeführt. Nach der Zusammenführung verbleibt das permanente Objekt in der Datenbank und das temporäre Objekt wird gelöscht.

Die Zusammenführung erfolgt parallel durch mehrere Threads. Die Standardanzahl der Threads beträgt 5. Sie können die Anzahl der Threads in der Datei `collation.properties` ändern, indem Sie das Attribut **`com.ibm.cdb.topomgr.dataverification.generator.ThreadCount`** auf eine geeignete Zahl setzen. Beispiel:

- **`com.ibm.cdb.topomgr.dataverification.generator.ThreadCount=10`**

Nach der Änderung der Anzahl der Threads müssen Sie den TADDM-Server erneut starten.

Beim Zusammenführen der Objekte können einige Fehler auftreten. Die Ursache der Fehler ist in einer Protokolldatei enthalten.

- `ERROR_INVALID_DURABLE_GUID`
- `ERROR_INVALID_TRANSIENT_GUID`

Der Fehler tritt auf, weil Aliasnamen in der Aliastabelle fehlen oder ein Objekt ungültig ist. Sie müssen warten, bis die Bereinigungsagenten die ungültigen Objekte gelöscht haben.

Prozedur

Für die Überprüfung auf Duplikate führen Sie eine der folgenden Optionen aus:

- **`verify-data.sh -v dup [-a repair]`**
- **`verify-data.bat -v dup [-a repair]`**

Übergeordnetes Zusammenführen prüfen
Bei der Prüfung des übergeordneten Zusammenführens werden die Daten verwendet, die in der Tabelle ALIASES_JN zur Suche und Dokumentation von GUIDs mit einer großen Zahl von wichtigen Änderungen an Aliasnamen zusammengestellt werden.

Informationen zu diesem Vorgang

Die Tabelle ALIASES_JN enthält den Verlauf der Änderungen an der Tabelle ALIASES. Das übergeordnete Zusammenführen ist eine Situation, in der wenige Objekte die zugehörigen übergeordneten Objekte in das gleiche Modellobjekt ändern. Untergeordnete Objekte werden dann in einigen übergeordneten Objekten in Gruppen zusammengefasst. Es wird kein übergeordnetes Zusammenführen ermittelt, das vor der Installation von TADDDM 7.2.1 Fixpack 3 auftrat, da es keine erforderlichen Daten in der Tabelle ALIASES_JN gibt. Die Prüfung enthält keine Option zur Reparatur, da möglicherweise falsche positive Ergebnisse gefunden und dokumentiert werden.

Standardmäßig ist die ausführliche Aufzeichnung für die Klassen 'ComputerSystem', 'AppServer' und für Betriebssystemklassen sowie für alle anderen daraus übernommenen Klassen aktiviert. Wenn Sie die Aufzeichnung für andere Klassen aktivieren möchten, können Sie die folgende Eigenschaft in der Datei `collation.properties` bearbeiten:

```
com.ibm.tivoli.namereconciliation.service.overmergeClasses
```

Im Anschluss finden Sie ein Beispiel der Eigenschaft, die für die Suche nach den Klassen 'ComputerSystem', 'AppServer' und nach Betriebssystemklassen angegeben ist:

```
com.ibm.tivoli.namereconciliation.service.overmergeClasses=
ComputerSystem,AppServer,OperatingSystem
```

Bedeutung der Aktionen, die für die Ausführung des Befehls verwendet werden:

- s1s2s1 - Die Überprüfung sucht Konfigurationselemente, deren Werte für Namensattribute in einer Schleife geändert werden. Es wird beispielsweise ein Computersystem mit einer Signatur A, anschließend Signatur B und dann wieder Signatur A ermittelt.

  Es gibt ein Szenario, das wie ein s1s2s1-Verifizierungstyp angezeigt wird, aber tatsächlich kein übergeordnetes Zusammenführen ist. Beispiel: Eine VM wird in ESX A gehostet, dann in ESX B verschoben und wieder zurück in ESX A verschoben.

  Um dieses Szenario zu vermeiden, konfigurieren Sie die folgende Eigenschaft in der Datei 'collation.properties': **com.collation.overmerge_exclusion_al_name_rule_guid = 8125A88D7F7E3CD38E9639955DD19383**

  Beim Abrufen von Daten des Typs 's1s2s1verification' wird mit dieser Eigenschaft das zuvor beschriebene Szenario ausgeschlossen.
- s1s2s3 - Die Überprüfung sucht Konfigurationselemente, die eine Reihe von Änderungen für angegebene Namensattribute enthalten.
- m1m2m1 - Die Überprüfung sucht Konfigurationselemente, deren GUIDs oft die zugehörige Master-GUID geändert haben. Es wird beispielsweise ein Alias A mit der Master-GUID B ermittelt, der später der Master-GUID C neu zugeordnet wurde und anschließend erneut der Master-GUID B zugeordnet wird.
- m1m2m3 - Die Überprüfung sucht Konfigurationselemente, deren GUIDs die zugehörigen Master-GUIDs ein paar Mal geändert haben.
- WinCSLinCSWinCS - Die Überprüfung sucht Konfigurationselemente, deren Typ einige Male geändert wurde. Es wird beispielsweise ein Computersystem ermittelt, das ursprünglich als 'WindowsComputerSystem' gespeichert wurde, anschließend in 'LinuxUnitaryComputerSystem' aktualisiert und dann erneut als 'WindowsComputerSystem' aktualisiert wurde.

Prozedur

Führen Sie zur Überprüfung des übergeordneten Zusammenführens einen der folgenden Befehle aus:

- **verify-data.sh -v om [-a <Aktion>] [-p <Klasse>] [-from <Zeitmarke>] [-to <Zeitmarke>]**
- **verify-data.bat -v om [-a <Aktion>] [-p <Klasse>] [-from <Zeitmarke>] [-to <Zeitmarke>]**

  Dabei gilt:

  – **<Aktion>**: `s1s2s1`, `s1s2s3`, `m1m2m1`, `m1m2m3`, `WinCSLinCSWinCS`
  – **<Klasse>**: Jede Klasse aus dem TADDM-Modell, z. B. ComputerSystem.
  – **<Zeitmarke>**: Zeitmarke im Format `JJJJ-MM-TT HH24:MI:SI`.

Beispiel

```
verify-data.sh -v om -a s1s2s1 m1m2m1 WinCSLinCSWinCS
-p ComputerSystem -from 2012-11-13 14:50:00 -to 2012-11-14 14:50:01
```

Mit diesem Befehl werden übermäßige Zusammenführungen des Typs s1s2s1, m1m2m1 und WinCSLinCSWinCS für die Klasse 'ComputerSystem' und für alle Klassen, die daraus übernommen werden, gesucht, die zwischen 2012-11-13 14:50:00 und 2012-11-14 14:50:01 erstellt wurden.

Problem des übergeordneten Zusammenführens lösen

Ein übergeordnetes Zusammenführen tritt auf, wenn wenige Objekte die zugehörigen übergeordneten Objekte in das gleiche Modellobjekt ändern. Untergeordnete Objekte werden dann in einigen übergeordneten Objekten in Gruppen zusammengefasst.

Vorgehensweise

1. Führen Sie die Überprüfung auf übergeordnetes Zusammenführen aus.

2. Prüfen Sie die aufgelisteten Konfigurationselemente. Die Prüfung kann diese fälschlicherweise als übergeordnetes Zusammenführen melden.
3. Korrigieren Sie die Konfiguration in Umgebungen, die möglicherweise die Ursache eines übergeordneten Zusammenführens sind. Die Konfigurationsprobleme können die gleiche Kennung, Seriennummer, VMID und weitere Namensattribute für Konfigurationselemente betreffen.
4. Entfernen Sie die übergeordnet zusammengeführten Objekte aus der TADDM-Datenbank.
5. Führen Sie eine Erkennung der gelöschten Objekte aus und prüfen Sie die Ergebnisse.
6. Entfernen Sie alle Datensätze aus der Tabelle ALIASES_JN, nachdem Sie die Probleme mit dem übergeordneten Zusammenführen behoben haben.

Cache für Berechtigungsnachweise verwalten - Dienstprogramm 'cachemgr'

Mit dem Befehl **cachemgr.sh** oder **cachemgr.bat** können Sie den Inhalt des Cache für Berechtigungsnachweise auflisten und löschen.

Befehlssyntax

`cachemgr -h | -u` *Benutzer* `-p` *Kennwort* `(-l|-r)` *valid|invalid|all* `[[ -s` *IP|scope|scope group|range|subnet* `] [ -a` *Adressraum* `] [ -n` *Name_der_Zugriffsberechtigungsnachweise* `] [ -c` *Typ* `] [ -d` *jjjj/mm/tt* `] [ -k` *Schlüssel* `] [ -t` *Positionstag* `]]`

Parameter

-a, --addressSpace ***Adressraum***
Gibt den Namen des Adressraums an.

-c, --class ***Typ***
Gibt den Typ eines ausgewählten Zugriffseintrags an, der durch den Namen der jeweiligen Klasse beschrieben wird, die den Zugriffseintrag implementiert.

-d, --date ***jjjj/mm/tt***
Gibt den Schwellenwert für das Datum an, mit dem Einträge ausgewählt werden, die nicht bis zu einer angegebenen Zeit geändert wurden. Das Format ist jjjj/mm/tt.

-h, --help
Zeigt den Hilfetext an.

-k, --key ***Schlüssel***
Gibt den Schlüssel eines ausgewählten Cacheeintrags an.

-l, --list ***valid|invalid|all***
Gibt die Auflistungsoperation an, die über die folgenden Argumente gesteuert wird:

- *valid* - listet nur gültige Authentifizierungsversuche auf, die in einem Cache zwischengespeichert sind.
- *invalid* - listet nur ungültige Authentifizierungsversuche auf, die in einem Cache zwischengespeichert sind.
- *all* - listet sowohl gültige als auch ungültige Authentifizierungsversuche auf, die in einem Cache zwischengespeichert sind.

-n, --name ***Name_der_Zugriffsberechtigungsnachweise***
Gibt den Namen der Zugriffsberechtigungsnachweise an (wie in der Zugriffsliste).

-p, --password ***Kennwort***
Gibt das Kennwort für den Benutzer an, der sich beim TADDM-Server anmeldet.

-r, --remove ***valid|invalid|all***
Gibt die Operation zum Entfernen an, die über die folgenden Argumente gesteuert wird:

- *valid* - entfernt nur gültige Authentifizierungsversuche, die in einem Cache zwischengespeichert sind.
- *invalid* - entfernt nur ungültige Authentifizierungsversuche, die in einem Cache zwischengespeichert sind.

- *all* - entfernt sowohl gültige als auch ungültige Authentifizierungsversuche, die in einem Cache zwischengespeichert sind.

-s, --scope ***IP|scope|scope group|range|subnet***
Gibt den Bereich eines Zugriffseintrags an. Er wird über die folgenden Argumente gesteuert:

- *IP*
- *scope*
- *scope group*
- *range*
- *subnet*

-t, --locationTag ***Positionstag***
Gibt den Positionstag eines ausgewählten Zugriffseintrags an.

-u, --username ***Benutzername***
Gibt den Benutzer an, der sich beim TADDM-Server anmeldet.

Beispiele

- Der folgende Befehl listet alle ungültigen Authentifizierungsversuche für Computer auf, die zum Bereich "ScopeSet" gehören:

```
cachemgr.sh -u user -p password -l invalid -s ScopeSet
```

Dieser Befehl generiert die folgende Ausgabe:

```
Die folgenden Einträge entsprechen den bereitgestellten Kriterien:
CachedAuthEntry
    guid: 3B954CE4CFBF346C8DF538F09F1F7FFD
    keyString: SSH!9.128.109.144!!com.collation.platform.security.auth.HostAuth!
null!
    lastModified: Thursday, 5 September 2013 11:00:38
    Berechtigung: ungültig. Fehlernachricht: CTJTP1190E Der Server hat den Autori
sierungsprozess
nicht abgeschlossen.
CachedAuthEntry
    guid: ACC2F35A66D3379BAC13FC606C5A08A3
    keyString: SSH!9.128.109.145!!com.collation.platform.security.auth.HostAuth!
null!
    lastModified: Thursday, 5 September 2013 11:00:38
    Berechtigung: ungültig. Fehlernachricht: CTJTP1190E Der Server hat den Autori
sierungsprozess
nicht abgeschlossen.
```

- Der folgende Befehl löscht ungültige Authentifizierungsversuche im IP-Bereich 9.123.149.10 bis 9.123.149.12 und den Zugriffseintrag com.collation.platform.security.auth.HostAuth:

```
cachemgr.sh -u user -p password -r invalid -s 9.123.149.10-9.123.149.12
-c com.collation.platform.security.auth.HostAuth
```

Dieser Befehl generiert die folgende Ausgabe:

```
Authentifizierungseinträge erfolgreich aus Cache entfernt (2).
```

Rückgabecodes des Dienstprogramms 'cachemgr'
Wenn Sie ein cron-Script oder ein anderes Script schreiben, von dem das Dienstprogramm 'cachemgr' aufgerufen wird, zeigen die folgenden Rückgabecodes an, wie das Programm beendet wurde.

0
Das Programm wurde erfolgreich ausgeführt.

1
Ein ungültiger Befehlszeilenparameter wurde angegeben. Entweder ist der Parameter selbst falsch oder die mit ihm bereitgestellten Daten sind nicht korrekt. Korrigieren Sie den Befehl und versuchen Sie es erneut.

2
Ein Befehlszeilenparameter für das Datum wurde nicht im erwarteten Format angegeben.

3
Entweder kann die angegebene Bereichsdefinition nicht in eine IP-Adresse aufgelöst werden oder der angegebene Zugriffseintrag ist ungültig.

4
Es ist ein unbekannter Fehler aufgetreten. Wechseln Sie zum Protokollverzeichnis und öffnen Sie die Datei `cachemgr.log`, um nach weiteren Informationen zu suchen.

5
Die Berechtigungen (Erkennung) des angegebenen Benutzers reichen nicht für die Ausführung der Operation aus.

6
Die Datenbank enthielt keine Einträge, die den angegebenen Kriterien entsprachen.

Vorbereitung für die Erkennung

Um eine optimale Informationserfassung bei den von TADDM (Tivoli Application Dependency Discovery Manager) durchgeführten Erkennungsläufen in Ihrer Umgebung zu erreichen, müssen Sie einige Konfigurationstasks zur Vorbereitung der Umgebung für die Erkennung ausführen.

Informationen zu diesem Vorgang
Die jeweils erforderlichen Konfigurationstasks hängen von Art und Umfang der Erkennung ab, die in Ihrer Umgebung unterstützt werden muss.

Nächste Schritte

Neben der Konfiguration Ihrer Umgebung für die Erkennung müssen auch die TADDM-Sensoren entsprechend konfiguriert werden. Informationen hierzu finden Sie im Handbuch TADDM *Sensoren*.

Informationen zur Ausführung einer Erkennung (einschließlich der Definition eines Bereichs und der Festlegung eines Zeitplans) finden Sie im TADDM-*Benutzerhandbuch*.

Anmelde-ID für Benutzer konfigurieren

Für die erfolgreiche Ausführung von Erkennungen benötigt TADDM einen interaktiven Benutzer. Für diesen Benutzer müssen Sie die Anmelde-ID konfigurieren.

Für alle Erkennungssitzungen, auch für Sitzungen zwischen Server und Gateway und Gateway und Ziel, wird die Anmelde-ID eines interaktiven Benutzers im nicht interaktiven Modus verwendet. Der Benutzer muss interaktiv sein, damit die Befehle ausgeführt werden können. Dennoch werden die Befehle nicht wirklich im interaktiven Modus ausgeführt, sondern der Benutzer führt den Befehl aus und wartet dann nur noch auf die Ergebnisse.

Legen Sie den Benutzer dazu in `/etc/passwd` wir folgt fest:

```
TADDM-Benutzer:x:100:100::/export/home/TADDM-Benutzer:/bin/sh
```

Dabei ist *TADDM-Benutzer* der Name des TADDM-Benutzers.

Konfiguration für alternative Erkennungsmethoden

Sie können auch alternative Erkennungsmethoden verwenden, z. B. die asynchrone Erkennung, die scriptbasierte Erkennung oder die Erkennung mit IBM Tivoli Monitoring.

Notes:

1. Die asynchrone und scriptbasierte Erkennung wird nur unterstützt, wenn auf dem Zielcomputersystem das Betriebssystem AIX, FreeBSD, HP NonStop, Linux (nur x86-Systeme), Solaris oder Windows aktiv ist.
2. Auf einem Zielcomputersystem mit Solaris funktioniert die scriptbasierte Erkennung unter Umständen nicht, wenn SunSSH 1.0 verwendet wird.

Umgebung für eine asynchrone Erkennung konfigurieren
Soll eine asynchrone Erkennung ausgeführt werden, müssen Sie zunächst die Erkennung konfigurieren.

Informationen zu diesem Vorgang

Für die Konfiguration einer asynchronen Erkennung müssen Sie ein Erkennungsscriptpaket generieren, das Paket auf das Zielsystem kopieren und anschließend das Script auf dem Zielsystem ausführen. Die Ausgabe des Erkennungsscripts wird in Form einer Archivdatei erstellt, die das Ergebnis der Erkennung enthält. Diese Archivdatei muss anschließend in den TADDM-Server verschoben werden.

Anmerkung: Wenn Sie die Erkennung für die Ausführung im asynchronen Modus konfiguriert und dann TADDM aktualisiert haben, müssen Sie erneut ein Erkennungsscriptpaket generieren, da sich die ID des Sensor-Plug-ins ändern kann.

Vorgehensweise

1. Um ein Erkennungsscriptpaket zu generieren, müssen Sie im Verzeichnis `$COLLATION_HOME/bin` einen der folgenden Befehle eingeben:
   - **Reguläres Verfahren**

     ```
     makeASDScriptPackage OUTPUT_DIR UNAME [IPADDRESS] [PACKING_METHOD]
     ```

     OUTPUT_DIR
     Der Verzeichnispfad des Scriptpakets.

     UNAME
     Das Betriebssystem des Zielsystems, auf dem das Script ausgeführt werden soll. Gültige Werte sind `AIX`, `Linux`, `SunOS`, `FreeBSD`, `Windows` und `NONSTOP_KERNEL`.

     ***IPADDRESS* (optional)**
     Die IP-Adresse des Zielsystems, auf dem das Script ausgeführt werden soll.

     Die Scripts, die bei der asynchronen Erkennung verwendet werden, stützen sich auf Informationen aus den TADDM-Servereigenschaften, die in der Datei `collation.properties` definiert sind und von denen einige bereichsorientiert sein können.

     Bereichsorientierte Eigenschaft
     Eine Eigenschaft, an die Sie eine IP-Adresse oder den Namen einer Bereichsgruppe anhängen können. Die IP-Adresse oder der Name der Bereichsgruppe bestimmen die Abhängigkeit der Eigenschaft vom Host, der erkannt wird. Die Namen von Bereichsgruppen dürfen keine Leerzeichen, Hochkommas ('), Punkte (.) und Schrägstriche (/) enthalten.

     Wenn Sie einige der TADDM-Servereigenschaften durch die Angabe eines Bereichs angepasst haben, sollten Sie die Option *IPADDRESS* in den Befehl **makeASDScriptPackage** einfügen.

     ***PACKING_METHOD* (optional)**
     Die Methode, mit der die Dateien zu einem Paket zusammengefasst werden. Gültige Werte sind `tar` und `zip`.

     Wird keine Methode angegeben, wird sie vom Betriebssystem vorgegeben. Auf Betriebssystemen wie Linux wird beispielsweise die tar-Methode verwendet.

     Standardmäßig wird der Systempfad nach dem Archivierungsdienstprogramm durchsucht. Fügen Sie bei Bedarf der Datei `collation.properties` die Eigenschaft `com.ibm.cdb.tarpath` hinzu und geben Sie einen anderen Pfad für das Archivierungsdienstprogramm an.

Auf Solaris-Betriebssystemen müssen Sie aufgrund der Einschränkung, die in Bezug auf die Länge von Dateinamen besteht, das Archivdienstprogramm 'gtar' verwenden und den Pfad zum Dienstprogramm angeben.

Das folgende Beispiel zeigt, wie der Pfad des **tar**-Befehls auf dem TADDM-Server für das Betriebssystem AIX angegeben wird:

```
com.ibm.cdb.tarpath=tar
```

Die folgenden Beispiele zeigen, wie der Pfad des Befehls **tar** *auf dem Zielsystem* für die jeweiligen Betriebssysteme angegeben werden muss:

Unter AIX
`com.ibm.cdb.targettarpath.AIX=tar`

Unter Solaris
`com.ibm.cdb.targettarpath.SunOS=/usr/sfw/bin/gtar`

Um beispielsweise ein Erkennungsscriptpaket für das AIX-Betriebssystem zu generieren, geben Sie den folgenden Befehl ein:

```
./makeASDScriptPackage /tmp AIX
```

Mit diesem Befehl wird im Verzeichnis `tmp` das AIX-Scriptpaket `/tmp/taddm_AIX.tar` erstellt.

- **Erweitertes Verfahren**

```
makeASDScriptPackage --outputDir AUSGANGSVERZEICHNIS --uname BENUTZERNAME
[--ipAddress IP-ADRESSE] [--packingMethod KOMPRIMIERUNGSMETHODE] [--sensors SENSOR]
```

--outputDir *AUSGANGSVERZEICHNIS*
Lesen Sie die Beschreibung des *OUTPUT_DIR*-Parameters für das reguläre Verfahren.

--uname *BENUTZERNAME*
Lesen Sie die Beschreibung des *UNAME*-Parameters für das reguläre Verfahren.

[--ipAddress *IP-ADRESSE*] (optional)
Lesen Sie die Beschreibung des *IPADDRESS*-Parameters für das reguläre Verfahren.

[--packingMethod *KOMPRIMIERUNGSMETHODE*] (optional)
Lesen Sie die Beschreibung des *PACKING_METHOD*-Parameters für das reguläre Verfahren.

[--sensors *SENSOR*] (optional)
Der Name des Sensors, den Sie in Ihr Paket einschließen möchten. Die folgende Tabelle enthält die Sensornamen, die in diesem Befehl verwendet werden müssen.

*Tabelle 33. Sensornamen, die im Befehl **makeASDScriptPackage** verwendet werden.*

Sensor	**Im Befehl verwendeter Name**
Apache-Sensor	apacheserver
Citrix XenServer-Sensor	xenserver
FreeBSD-Computersystemsensor	computersystem
Generic Server-Sensor	genericserver
HP NonStop-Computersystemsensor	computersystem
IBM AIX-Computersystemsensor	computersystem
IBM DB2-Sensor	db2

*Tabelle 33. Sensornamen, die im Befehl **makeASDScriptPackage** verwendet werden. (Forts.)*

Sensor	**Im Befehl verwendeter Name**
IBM Lotus Domino-Server-Sensor	dominoserverinitial
IBM Tivoli Utilization-Sensor	utilization
IBM WebSphere MQ Server-Sensor	mqserver
IBM WebSphere-Sensor	webspherescript
JBoss Application Server 7-Sensor	jboss7
KVM-Sensor	kvm
Linux-Computersystemsensor	computersystem
Microsoft Exchange-Sensor	exchange
Microsoft IIS Web-Server-Sensor	iisserver
Oracle-Sensor	oracle
Solaris-Computersystemsensor	computersystem
WebLogic SSH-Sensor	weblogiclaunchersensor
Windows-Computersystemsensor	computersystem

Der Sensor für asynchrone Erkennung wird standardmäßig jedem Paket hinzugefügt. Alle Betriebssystemsensoren haben den Namen `computersystem`. Sie werden auf der Basis des `--uname`-Parameters unterschieden. Sie können beispielsweise folgende Parameter angeben:

```
[...] --uname Linux --sensors computersystem
```

Linux-Computersystemsensor wird zum Paket hinzugefügt.

Um beispielsweise ein Erkennungsscriptpaket für das AIX-Betriebssystem zu generieren, geben Sie den folgenden Befehl ein:

```
./makeASDScriptPackage --outputDir /tmp --uname AIX --sensors computersystem
```

Mit diesem Befehl wird im Verzeichnis `tmp` das AIX-Scriptpaket `/tmp/taddm_AIX.tar` erstellt.

2. Kopieren Sie das Scriptpaket aus dem Verzeichnis *OUTPUT_DIR* in das Zielsystem und dekomprimieren Sie das Scriptpaket.
3. Erteilen Sie als Rootbenutzer auf UNIX-Systemen oder Administrator auf dem Windows-System allen Scriptdateien Ausführungsberechtigungen. Wird das Erkennungsscript als Benutzer ohne Rootberechtigung oder Benutzer ohne Administratorberechtigung ausgeführt, führen einige Sensorscripts unter Umständen keine erfolgreiche Erkennung durch oder der Sensor erkennt Daten nur in einem begrenzten Umfang.
4. Führen Sie das Script **scriptsRunner.sh** für die UNIX-Ziele bzw. das Script **scriptsRunner.bat** für das Windows-Ziel aus.
5. Verschieben Sie die Archivdatei, die als Ergebnis generiert wird (z. B. `/tmp/taddm${`*`Version`*`}/asd/taddmasd-${`*`Hostname`*`}-${`*`Ausführungszeitmarke`*`}.tar`) in das Verzeichnis auf dem TADDM-Server, das durch die Eigenschaft `com.ibm.cdb.discover.asd.AsyncDiscoveryResultsDirectory` in der Datei `collation.properties` angegeben ist.

6. Setzen Sie in der Datei `collation.properties` die Eigenschaft `com.ibm.cdb.discover.asd.ProcessUnreachableIPs` auf `true`.
7. Fix Pack 6 Stellen Sie sicher, dass die Sensoren für die asynchrone Erkennung (`ASDPingSensor` und `ASDSensor`) in Ihrem Erkennungsprofil aktiviert sind.

   Standardmäßig ist nur 'ASDSensor' in Erkennungsprofilen der Ebene 2 und 3 aktiviert.
8. Erstellen Sie einen Bereich mit der IP-Adresse des Zielsystems.

Nächste Schritte

Führen Sie die Erkennung aus. Hierfür ist keine Rootberechtigung erforderlich.

Kann der Ping-, Port- oder Sitzungssensor bei der Erkennung nicht auf das Zielsystem zugreifen, wird das Zielsystem als nicht erreichbar eingestuft. Wird die Eigenschaft `com.ibm.cdb.discover.asd.ProcessUnreachableIPs` auf `true` gesetzt, wird der Sensor für die asynchrone Erkennung ausgeführt, um die Erkennungsarchivdatei für das Zielsystem zu verarbeiten. Die Archivdatei wird nur verarbeitet, wenn die IP-Adresse aus dem Erkennungsbereich mit der IP-Adresse des Systems übereinstimmt, von dem die Archivdatei erstellt wurde. Anhand des Inhalts dieser Archivdatei werden die Sensoren für die Verarbeitung ihrer Scriptausgaben terminiert. Nach der Verarbeitung der Archivdatei wird sie in *`tar-Dateiname`*`.tar_DONE` umbenannt, damit sie nicht erneut verarbeitet wird.

Die Erkennungsarchivdatei wird nur einmal verarbeitet. Ist ein Sensor nicht für die Verarbeitung der Scriptausgabe zu dem Zeitpunkt, zu dem die Archivdatei verarbeitet wird, aktiviert, wird auch bei Ausführung einer zweiten Erkennung mit aktiviertem Sensor eine zuvor verarbeitete Archivdatei nur dann verarbeitet, wenn Sie folgende Schritte ausführen:

1. Benennen Sie die Archivdatei wieder in ihren ursprünglichen Namen um. Beispiel: Entfernen Sie `_DONE` aus dem Dateinamen.
2. Die Datei `.processed` im Verzeichnis `$COLLATION_HOME/var/asdd` enthält eine Liste der verarbeiteten Archivdateien. Entfernen Sie den Namen der Archivdatei aus der Datei `.processed`.

Innerhalb eines Erkennungsvorgangs können zwar mehrere Archivdateien aus verschiedenen Systemen, aber nur jeweils eine Archivdatei pro Zielsystem verarbeitet werden. Sind für ein Zielsystem mehrere Archivdateien vorhanden, wird nur die mit der neuesten Zeitmarke verarbeitet.

Um innerhalb eines Erkennungsvorgangs mehrere Archivdateien aus verschiedenen Systemen zu erkennen, müssen Sie die einzelnen Archivdateien in das Verzeichnis kopieren, das über die Eigenschaft `com.ibm.cdb.discover.asd.AsyncDiscoveryResultsDirectory` angegeben ist. Dabei müssen Sie im Erkennungsbereich die IP-Adressen der einzelnen Zielsysteme aufnehmen.

Da das Erkennungsscript die Erkennungsarchivdatei mithilfe des **`tar`**-Befehls erstellt, müssen Sie bei einem TADDM-Server, der unter Windows aktiv ist, das tar-Programm eines anderen Anbieters installieren, damit TADDM die Dateien aus der Archivdatei extrahieren kann. Das Verzeichnis für das tar-Programm wird über die Eigenschaft `com.ibm.cdb.tarpath` definiert.

Als Referenz wird die TAR-Implementierung '`bsdtar`' eines Drittanbieters unterstützt, wenn sie über die oben genannte Eigenschaft für den TADDM-Server konfiguriert wird, der auf einem Windows-Betriebssystem ausgeführt wird.

Einschränkung: Ihr tar-Programm muss lange Dateipfade unterstützen. GNU Tar 1.13 wird nicht unterstützt, da es lange Dateinamen möglicherweise abschneidet.

Der Prozess zum manuellen Starten der Erkennung kann mit den folgenden Eigenschaften automatisiert werden:

Tabelle 34.

Eigenschaftsname	Zulässige Werte	Beschreibung
com.ibm.cdb.discover.asd.autodiscovery.enabled	True/false	Bei 'true' wird die Option für die Verarbeitung der gespeicherten ASD-Ergebnisdateien aktiviert. Der Standardwert ist 'false'.

Tabelle 34. (Forts.)		
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**
com.ibm.cdb.discover.asd.autodiscovery.asdScope	<Bereichsname> Default = ASD	Der Thread wird das in diesem Bereich erwähnte Ziel auswählen, um die Ergebnisdatei zu verarbeiten. Wenn diese Eigenschaft nicht angegeben ist, wird der standardmäßige ASD-Bereich verarbeitet.
com.ibm.cdb.discover.asd.autodiscovery.asdProfile	<Profilname> Standardwert = ASD	Der Thread wird die in diesem Profil erwähnten Sensoren auswählen, um die Ergebnisdatei zu verarbeiten. Wenn diese Eigenschaft nicht angegeben ist, wird das ASD-Standardprofil verarbeitet.
com.ibm.cdb.discover.asd.autodiscovery.filesThreshold	<Dateischwellenwert> Standardwert=20	Mindestanzahl der für den Thread erforderlichen Dateien, um die Verarbeitung zu starten. Der Thread verarbeitet das Ergebnis, wenn entweder der Dateischwellenwert oder der Zeitschwellenwert erreicht wird.
com.ibm.cdb.discover.asd.autodiscovery.timeThreshold	<Zeitschwellenwert> Standardwert=60 (Sekunden)	Zeitschwellenwert (in Sekunden), nach dem der Thread die Ergebnisdateien verarbeitet, auch wenn der Dateischwellenwert nicht erfüllt ist.

Konfiguration für die scriptbasierte Erkennung
Für die Ausführung einer scriptbasierten Erkennung müssen Sie zunächst die Erkennung konfigurieren.

Informationen zu diesem Vorgang

Im Vergleich zu regulären Sensoren sind Sensoren, die im scriptbasierten Modus ausgeführt werden, einfacher nachzuvollziehen, denn alle vom Sensor verwendeten Befehle stehen in einem einzigen Script, das angezeigt werden kann. Eine Liste der Sensoren, die den scriptbasierten Modus unterstützen, und Informationen zu Einschränkungen, die für einige der Sensoren gelten, finden Sie im Abschnitt *Sensoren, die scriptbasierte und asynchrone Erkennung unterstützen* in den TADDM *Referenzinformationen zu Sensoren*.

Prozedur

- Konfigurieren Sie den Sensor; dabei haben Sie die folgenden Möglichkeiten:
  - **Alle Sensoren aktivieren, die scriptbasierte Erkennung unterstützen**
    Sollen alle Sensoren, die die scriptbasierte Erkennung unterstützen, global aktiviert werden, öffnen Sie die Datei `collation.properties` und setzen Sie die Eigenschaft `com.ibm.cdb.discover.PreferScriptDiscovery` auf `true`.
  - **Alle Sensoren aktivieren, die scriptbasierte Erkennung in einem bestimmten Erkennungsprofil unterstützen**
    Gehen Sie wie folgt vor, um für ein bestimmtes Erkennungsprofil alle Sensoren zu aktivieren, die scriptbasierte Erkennung unterstützen:
    1. Wählen Sie in Discovery Management Portal das Erkennungsprofil aus, für das Sie den scriptbasierten Modus aktivieren möchten.
    2. Setzen Sie auf der Registerkarte **Plattformeigenschaften** den Wert der Eigenschaft `com.ibm.cdb.discover.PreferScriptDiscovery` auf `true`.
  - **Einen Sensor aktivieren, der scriptbasierte Erkennung in einem Erkennungsprofil unterstützt**
    Soll in einem Erkennungsprofil ein bestimmter Sensor, der die scriptbasierte Erkennung unterstützt, aktiviert werden, müssen Sie die Konfiguration dieses Sensors in dem entsprechenden Erkennungsprofil aktualisieren. Gehen Sie hierzu wie folgt vor:

1. Gehen Sie in Discovery Management Portal zu dem Erkennungsprofil, das den Sensor enthält, den Sie aktivieren möchten.
2. Wählen Sie den Sensor auf der Registerkarte **Sensorkonfiguration** aus und klicken Sie auf **Neu**.
3. Geben Sie im Fenster **Konfiguration erstellen** den Namen der Konfiguration an und wählen Sie die Option **Scriptbasierte Erkennung ausführen** aus.
4. Klicken Sie auf **OK**, um die Konfiguration zu speichern.

Nächste Schritte

TADDM für die Auswahl von Nicht-Standardbenutzern für die Erkennung konfigurieren

Standardmäßig wird nur der Benutzer, der vom Script angefordert wird, für die Erkennung verwendet. Wenn es bei der Ausführung einer Erkennung Probleme mit dem Standardbenutzer gibt und Sie einen anderen Benutzer haben, der alle erforderlichen Berechtigungen besitzt, können Sie TADDM so konfigurieren, dass es diesen Benutzer für die Erkennung auswählt.

Anmerkung: Verwenden Sie die folgende Konfiguration mit Vorsicht. Wenn ein Benutzer, der nicht alle erforderlichen Berechtigungen besitzt, für eine Erkennung verwendet wird, kann die Erkennung fehlschlagen oder einige der Ziele werden möglicherweise nicht erkannt.

Bearbeiten Sie in der Datei `plugin.xml`, die sich in einem Paket für jeden Sensor im Verzeichnis `COLLATION_HOME/osgi/plugins` befindet, die Knotendefinition `script`, z. B. wie im Snippet `plugin.xml` für den IBM WebSphere MQ Server-Sensor:

```
<scriptset>
   <ostype>AIX</ostype>
   <mainScript name="sensorCommon.sh" />
   <script name="script.sh" authClassName="com.collation.platform.security.
auth.MQServerAuth" authMode="preferred" hostAuthFallback="true”/>
</scriptset>
```

Folgende Eigenschaften können definiert werden:

authMode

Bestimmt, wie TADDM die Zugriffslisteneinträge mit dem durch `authClassName` angegebenen Typ behandelt. Folgende Werte sind verfügbar:

- `single` - nur ein vom Script angeforderter Benutzer wird verwendet. Dies ist der Standardwert.
- `preferred` - zunächst wird der vom Script bevorzugte Benutzer verwendet; steht dieser nicht zur Verfügung bzw. schlägt die Anmeldung damit fehl, so werden die verbleibenden Zugriffslisteneinträge des angegebenen Typs verwendet.
- `regular` - die Zugriffslisteneinträge werden in der angegebenen Reihenfolge und ohne Beachtung der Benutzervorgabe verwendet.

hostAuthFallback

Bestimmt, ob TADDM im Falle von Problemen bei der Herstellung der Verbindung mit dem Ziel unter einem bestimmten `authClassName` bzw. unter dem vom Script vorgegebenen Benutzer (oder beidem) auf die Sitzung zurückgreift, die von dem generischen Benutzer für die Verbindung mit diesem Ziel verwendet wird. Folgende Werte sind verfügbar:

- `false` - dies ist der Standardwert.
- `true`.

Erkennung mit IBM Tivoli Monitoring konfigurieren (altes Verfahren)

TADDM kann mithilfe einer IBM Tivoli Monitoring-Infrastruktur der Version 6.2.1 oder höher Erkennungen der Ebene 1 und 2 und einige Erkennungen der Ebene 3 durchführen.

Altes Integrationsverfahren

Dieser Abschnitt bezieht sich auf ein veraltetes Verfahren zur TADDM-Integration mit IBM Tivoli Monitoring. Ab TADDM Version 7.3.0 sollte die Integration mit IBM Tivoli Monitoring 6.3 mithilfe von OSLC Automation erfolgen. Das Integrationsverfahren unter Verwendung des IBM Tivoli Monitoring Scope-Sensors ist veraltet und steht in künftigen Releases nicht mehr zur Verfügung. Weitere Informationen

zur Konfiguration einer Erkennung mithilfe von OSLC Automation finden Sie im Abschnitt „Erkennung mit OSLC Automation Session konfigurieren” auf Seite 107.

Wenn Sie IBM Tivoli Monitoring 6.2.1-TIV-ITM-FP0001, 6.2.2-TIV-ITM-FP0002 oder höher verwenden, können Sie Tivoli Monitoring-Endpunkte mit dem Tivoli Enterprise Portal Server erkennen. Diese Fixpacks beheben APAR IZ63983, wodurch die Tivoli Monitoring-Leistung bei TADDM-Erkennungen verbessert wird. Wenn Sie ältere Releases oder Versionen von IBM Tivoli Monitoring zur Ausführung von TADDM-Erkennungen mit dem Tivoli Enterprise Portal Server verwenden, kann dies zu einer überhöhten Prozessor- und Netzauslastung führen, vor allem bei Tivoli Monitoring-Komponenten.

Anmerkung: Die Erkennung mit IBM Tivoli Monitoring ist nur möglich, wenn unter Microsoft SQL Server und unter DB2 die Tivoli Enterprise Portal Server-Datenbank installiert ist. Bei Verwendung der Apache Derby-Datenbank als Tivoli Enterprise Portal Server-Datenbank funktioniert diese Erkennung nicht.

Spezifische TADDM-Servereigenschaften für die Erkennung mit Tivoli Monitoring

Informationen zu den für die Erkennung mit IBM Tivoli Monitoring spezifischen TADDM-Servereigenschaften einschließlich der Eigenschaften mit Auswirkungen auf die TADDM-Vorgehensweise bei der Erkennung von Tivoli Monitoring-Endpunkten finden Sie unter „Eigenschaften für Erkennung mit IBM Tivoli Monitoring (altes Verfahren)” auf Seite 77.

In einem Erkennungsprofil können Sie die TADDM-Servereigenschaften konfigurieren, die Auswirkungen auf die TADDM-Vorgehensweise bei der Erkennung von Tivoli Monitoring-Endpunkten haben. Führen Sie dazu folgende Schritte aus, je nachdem, ob Sie ein benutzerdefiniertes Profil oder das Standardprofil verwenden:

Eigenschaften für ein benutzerdefiniertes Profil konfigurieren

1. Starten Sie die Discovery Management Console.
2. Öffnen Sie das Dialogfeld **Erkennungsprofile**.
3. Klicken Sie auf das Erkennungsprofil, dass Sie konfigurieren möchten.
4. Klicken Sie auf die Registerkarte **Plattformeigenschaften**.
5. Ändern Sie den Wert der Eigenschaft, die Sie aktualisieren möchten, und aktivieren Sie das Kontrollkästchen **Eingeschlossen** für diese Eigenschaft.
6. Speichern Sie die Änderungen.

Eigenschaften für das Standardprofil konfigurieren

Bearbeiten Sie die Datei `$COLLATION_HOME/etc/collation.properties`, indem Sie die gewünschte Eigenschaft hinzufügen (oder bearbeiten), wie in folgendem Beispiel dargestellt, wobei *Erkennungsprofil* für den Profilnamen steht:

```
com.ibm.cdb.session.allow.ITM.Erkennungsprofil=true
```

Beispielsweise gibt folgende Eigenschaft an, dass TADDM das Erkennungsprofil "Utilization Discovery" (Auslastungserkennung) verwendet und IBM Tivoli Monitoring für die Erkennung von Tivoli Monitoring-Endpunkten einsetzt:

```
com.ibm.cdb.session.allow.ITM.Utilization_Discovery=true
```

Anmerkung: Ersetzen Sie in der Datei `collation.properties` das Leerzeichen zwischen "Utilization" und "Discovery" im Profilnamen durch einen Unterstrich.

Weitere konfigurierbare TADDM-Servereigenschaften

In den folgenden Konfigurationstipps sind weitere TADDM-Servereigenschaften beschrieben, die möglicherweise ebenfalls konfiguriert werden sollen:

- Für die folgende, für Windows-Systeme spezifische Eigenschaft muss der Standardwert `true` festgelegt werden, um die Erkennung von Windows-Zielsystemen mithilfe von IBM Tivoli Monitoring zu aktivieren.

Wenn hingegen der Wert `false` festgelegt, kann TADDM keine IBM Tivoli Monitoring-Sitzung für Windows-Zielsysteme einrichten.

```
com.collation.AllowPrivateGateways=true
```

- Während der Erkennung kann es beim Tivoli Enterprise Portal Server zu einer hohen Prozessorauslastung kommen. Um diese zu minimieren, können Sie die Anzahl der Erkennungs-Worker-Threads begrenzen, die während der Erkennung ausgeführt werden. Legen Sie auf dem TADDM-Server die folgende Servereigenschaft fest:

```
com.collation.discover.dwcount=16
```

- In einer umfangreichen IBM Tivoli Monitoring-Umgebung überschreitet der IBM Tivoli Monitoring Scope-Sensor möglicherweise das Zeitlimit vor Abschluss der Verarbeitung. Legen Sie die folgenden Servereigenschaften fest, um eine längere Verarbeitungszeit zuzulassen:

```
com.collation.platform.session.ITMSessionNumProgressChecks=3600
com.collation.discover.agent.ITMScopeSensor.timeout=3600000
```

Erkennung mit OSLC Automation Session konfigurieren
TADDM kann über OSLC Erkennungen der Ebene 2 sowie einige Erkennungen der Ebene 3 durchführen.

Vorbereitende Schritte
Für die Konfiguration der Erkennung in den von den OSLC Execute Automation-Service-Providern bereitgestellten Bereichsgruppen müssen folgende Voraussetzungen erfüllt sein:

- Mindestens ein OSLC Execute Automation-Service-Provider muss installiert sein und funktionieren.
- Zwischen TADDM und dem OSLC Execute Automation-Service-Provider muss eine Verbindung bestehen.

Vorgehensweise

Zur Ausführung einer Erkennung mit OSLC Automation Session führen Sie die folgenden Schritte aus:

1. Fügen Sie der Zugriffsliste die Berechtigungsnachweise für den Zugriff auf das Produkt hinzu, das Sie integrieren möchten. Erstellen Sie dazu einen neuen Zugriffslisteneintrag des Typs "`Integration`">"`OSLC Automation`".

   Wenn Sie TADDM mit ITM integrieren, müssen Sie die Berechtigungsnachweise für ITM TEPS bereitstellen. Während der Erkennung wird durch die Zugriffslisteneinträge für OSLC Automation und den Typ des ITM-Zugriffslisteneintrags die Kompatibilität mit früheren Versionen sichergestellt.
2. Prüfen Sie den Erkennungsbereich. Die Bereichsgruppen werden vom OSLCAutomationAgent in regelmäßigen Abständen erstellt. Die neuen Bereichsgruppen sind auf der Registerkarte **Scope Sets** (Bereichsgruppen) aufgeführt. Wenn Sie TADDM mit ITM integrieren, wird eine Bereichsgruppe pro ITM TEMS erstellt. Sie können OSLCAutomationAgent mit dem folgenden Befehl ausführen:

   ```
   /taddm/dist/support/bin/runtopobuild.sh -a OSLCAutomationAgent
   ```
3. Konfigurieren Sie Erkennungseigenschaften, die die Verwendung von OSLC Automation Session zulassen. Die Eigenschaften können Sie in der Datei `collation.properties` oder in einem neuen benutzerdefinierten Erkennungsprofil festlegen.
   - Datei `collation.properties`:

     ```
     com.ibm.cdb.session.prefer.OSLCAutomation=true
     com.ibm.cdb.session.allow.OSLCAutomation=true
     ```

     Beispiele für bereichsorientierte Eigenschaften:

     ```
     com.ibm.cdb.session.prefer.OSLCAutomation.9.222.222.124=false
     com.ibm.cdb.session.prefer.OSLCAutomation.Level_3_Discovery=false
     ```

- Benutzerdefiniertes Erkennungsprofil: Erstellen Sie in Discovery Management Console ein neues Erkennungsprofil und konfigurieren Sie die Einstellungen auf der Registerkarte **Platform Properties** (Plattformeigenschaften) wie folgt:

```
com.ibm.cdb.session.allow.OSLCAutomation=true
com.ibm.cdb.session.prefer.OSLCAutomation=true
```

4. Führen Sie mit einer der folgenden Methoden eine reguläre Erkennung der von OSLCAutomationAgent erstellten Bereiche aus:
   - Mit dem Standardprofil der Ebene 2 oder 3, wenn die Datei `collation.properties` so konfiguriert ist, dass OSLCAutomation Session unterstützt wird.
   - Mit dem neuen Erkennungsprofil mit korrekt konfigurierten **Plattformeigenschaften**.

Zugehörige Verweise

„Eigenschaften für die Erkennung mit OSLC Automation Session" auf Seite 79
Diese Eigenschaften gelten für die Erkennung mit OSLC Automation Session.

„Befehlszeilenschnittstelle für OSLCAutomationAgent" auf Seite 192
OSLCAutomationAgent wird für die Erfassung der Daten von OSLC Execute Automation-Service-Providern verwendet. Mit den entsprechenden Befehlen können Sie den Agenten manuell ausführen und die von ihm erstellten Bereichsgruppen aktualisieren.

Erkennungsebene konfigurieren

Die Erkennungsebene muss konfiguriert werden.

Konfiguration für die Erkennung der Ebene 1
Für eine Erkennung der Ebene 1 (Erkennung ohne Berechtigungsnachweis), bei der der TCP/IP-Stack durchsucht wird, um Basisinformationen zu aktiven Computersystemen zu sammeln, ist eine Minimalkonfiguration erforderlich.

Informationen zu diesem Vorgang

Für eine Erkennung der Ebene 1 ist die Konfiguration der Netzeinheiten in Ihrer Umgebung erforderlich, die der TADDM-Server erkennen soll.

Vorgehensweise

Gehen Sie hierzu folgendermaßen vor:

1. Zeichnen Sie je nach SNMP-Version die folgenden Daten zur Verwendung mit dem TADDM-Server auf:
   - Zeichnen Sie für SNMP V1 und V2 die Zeichenfolge SNMP MIB2 GET COMMUNITY auf.
   - Zeichnen Sie für SNMP V3 den SNMP-Benutzernamen und das SNMP-Kennwort auf.
2. Weisen Sie die Berechtigungen für MIB2-Systeme, IP, Schnittstellen und erweiterte Schnittstellen zu.

Konfiguration für die Erkennung der Ebene 2
Neben den Voraussetzungen für eine Erkennung der Ebene 1 muss für die Erkennung der Ebene 2 Unterstützung für die Erkennung ausführlicher Informationen zur Hostkonfiguration konfiguriert werden.

Vorbereitende Schritte
Handelt es sich bei den Zielsystemen um IBM Tivoli Monitoring-Endpunkte, die vom IBM Tivoli Monitoring Scope-Sensor erkannt werden, werden die Berechtigungen dieser Zielsysteme für eine Erkennung der Ebene 2 nicht benötigt. Weitere Informationen finden Sie in folgenden Quellen:

- „TADDM mit IBM Tivoli Monitoring integrieren (altes Verfahren)" auf Seite 193
- „Erkennung mit IBM Tivoli Monitoring konfigurieren (altes Verfahren)" auf Seite 105
- TADDM *Sensorreferenz* mit Informationen zum IBM Tivoli Monitoring Scope-Sensor

Informationen zu diesem Vorgang

Für die Zielbetriebssysteme (Computersystem), die von TADDM erkannt werden sollen, müssen Sie mindestens folgende Software konfigurieren:

Secure Shell (SSH)
Sie können entweder OpenSSH verwenden oder die SSH-Version, die mit dem Betriebssystem geliefert wird. Weitere Informationen zu Windows-Betriebssystemen finden Sie unter „Abhängigkeit von Windows Management Instrumentation (WMI)" auf Seite 120.

SUNWscpu (nur Solaris-Umgebung)
Installieren Sie das SUNWscpu-Paket (Source Compatibility), damit alle Informationen zu Prozessen zur Verfügung stehen.

LiSt Open Files (lsof)
Zur Bereitstellung umfassender Abhängigkeitsdaten müssen Sie das Programm 'LiSt Open Files' (lsof) in Übereinstimmung mit den Voraussetzungen installieren, die im TADDM-Wiki unter https://github.com/TADDM/taddm-wiki/wiki/Generic-Server-Sensor-(lsof) im Abschnitt *lsof requirements* (lsof-Voraussetzungen) genannt sind.

Service-Accounts erstellen

Sie müssen auf allen Computersystemen, die mittels SSH-schlüsselbasierten und kennwortbasierten Verbindungen erkannt werden sollen, einen Service-Account anlegen. Dies ist die primäre Methode für die Erkennung der Computersysteme (Server) in Ihrem Netz.

Informationen zu diesem Vorgang

Erstellen Sie zur Vereinfachung der Erkennungskonfiguration denselben Service-Account auf jedem Zielcomputersystem, das erkannt werden soll. Der Service-Account muss den Zugriff auf alle Ressourcen auf dem Zielcomputersystem ermöglichen, das von TADDM erkannt werden muss. Der Service-Account muss auf jedem Zielcomputersystem Schreibzugriffsrechte auf das eigene Ausgangsverzeichnis haben. Dieses Verzeichnis muss über ca. 20 MB freien Speicherplatz verfügen. In diesem Verzeichnis können während einer Erkennung Scripts und temporäre Ergebnisdateien gespeichert werden. Nach Abschluss der Erkennung werden die Dateien wieder gelöscht.

Sie können einen Service-Account ohne Root-Berechtigung verwenden. Allerdings ist für einige Betriebssystembefehle, die bei der Erkennung verwendet werden, unter Umständen eine höhere Berechtigung wie 'root' (oder 'superuser') erforderlich, damit sie auf dem Zielcomputersystem ausgeführt werden können.

Vorgehensweise

Führen Sie eine der folgenden Prozeduren aus, um einen Service-Account auf dem Zielcomputersystem zu erstellen:

1. Bei einem Linux-, Solaris-, AIX- und Linux on zSeries-Betriebssystem gehen Sie davon aus, dass der Name des Service-Accounts `coll` lautet und erstellen den Service-Account mit folgenden Befehlen:

   ```
   # mkdir -p /export/home/coll
   # useradd -d /export/home/coll -s /bin/sh \
      -c "Service Account" -m coll
   # chown -R coll /export/home/coll
   ```

2. Erstellen Sie für ein Windows-Computersystem einen Service-Account, der ein Mitglied der Gruppe des lokalen Administrators ist.

   Dieser Account kann ein lokaler oder ein Domänenaccount sein. Da TADDM für die Erkennung auf WMI angewiesen ist, muss der Account Zugriff auf alle WMI-Objekte auf dem lokalen Computer haben. Der Service-Account muss auf dem Windows-Gateway und auf allen Windows-Zielcomputersystemen erstellt werden.

   Anmerkung: Der Service-Account muss über Schreib-/Lesezugriff auf das Verzeichnis `\WINDOWS\system32` oder `\WINDOWS\system64` und dessen Unterverzeichnisse haben. Auf Windows Server 2008-Systemen verfügen neue Benutzer nicht automatisch über den erforderlichen Zugriff, daher muss er dem Service-Account explizit erteilt werden.

Erkennung mit Secure Shell (SSH) konfigurieren

Der TADDM-Server kann eine Verbindung zu OpenSSH (Version 1 oder 2) oder zu der SSH-Version herstellen, die mit dem Betriebssystem geliefert wird.

Der TADDM-Server unterstützt folgende Authentifizierungsmethoden:

- SSH2-schlüsselbasierte Anmeldung (RSA- oder DSA-Schlüssel) und SSH1-schlüsselbasierte Anmeldung (nur RSA)
- Eingabe des Benutzernamens und Kennworts mit SSH2 oder SSH1

Obwohl Sie frei zwischen den Authentifizierungsmethoden wählen können, wird die SSH2-schlüsselbasierte Anmeldung bevorzugt verwendet. Der Server probiert automatisch jede Methode in der aufgeführten Reihenfolge aus und verwendet die Methode, die zuerst erfolgreich funktioniert. Der TADDM-Server verwendet dann für diesen Host dieselbe Methode während der gesamten Dauer des Erkennungslaufs.

Anmerkung: Bei der SSH2-schlüsselbasierten Anmeldung, probiert der TADDM-Server die Anmeldung nur mit einem der Schlüssel, entweder RSA oder DSA, je nachdem, welcher Schlüssel auf dem TADDM-Server gefunden wird. Wenn beide Schlüssel vorhanden sind, wird nur RSA verwendet.

Schlüsselpaare für die schlüsselbasierte Anmeldung am TADDM-Server erstellen

Sie können unter Verwendung des SSH-Protokolls (Secure Shell) ein Schlüsselpaar mit einem öffentlichen und einem privaten Schlüssel erstellen, das für die schlüsselbasierte Anmeldung am TADDM-Server verwendet wird.

Informationen zu diesem Vorgang

Abhängig von der jeweiligen SSH-Version verwendet die SSH-schlüsselbasierte Anmeldung die Schlüssel aus Tabelle 35 auf Seite 110:

Tabelle 35. SSH-Schlüssel

SSH-Version/Algorithmus	Privater Schlüssel	Öffentlicher Schlüssel
`Openssh/SSH2/RSA`	`$HOME/.ssh/id_rsa`	`$HOME/.ssh/id_rsa.pub`
`Openssh/SSH2/DSA`	`$HOME/.ssh/id_dsa`	`$HOME/.ssh/id_dsa.pub`
`Openssh/SSH1/RSA`	`$HOME/.ssh/identity`	`$HOME/.ssh/identity.pub`
`Commercial/SSH2/RSA`	`$HOME/.ssh2/id_dss_1024_a`	`$HOME/.ssh2/ id_dss_1024_a .pub`

Sie können auch ein Paar aus öffentlichem/privatem Schlüssel mit OpenSSH Version 2 generieren. Informationen zum Generieren eines Schlüsselpaares aus öffentlichem und privatem Schlüssel unter Verwendung eines anderen SSH-Programms als OpenSSH oder einer anderen Version von OpenSSH finden Sie in der SSH-Dokumentation.

Vorgehensweise

So generieren Sie ein Schlüsselpaar aus öffentlichem und privatem Schlüssel unter Verwendung von OpenSSH, Version 2:

1. Melden Sie sich als Eigner des TADDM-Servers an.
2. Geben Sie den folgenden Befehl ein, um den SSH-Schlüssel zu generieren:

   ```
   $ ssh-keygen -t rsa
   ```

   Übernehmen Sie die Befehlsvoreinstellungen. TADDM unterstützt Schlüsselpaare mit oder ohne Kennphrase.
3. Fügen Sie auf jedem Zielsystem, an dem eine schlüsselbasierte Anmeldung möglich sein soll, den Inhalt der Datei `id_rsa.pub` in die Datei `$HOME/.ssh/authorized_keys` für den Service-Account ein.

Bestimmte SSH2-Implementierungen generieren die Schlüssel in einem anderen Verzeichnis als $HOME/.ssh. Wenn Ihre SSH-Implementierung die Schlüssel in einem anderen Verzeichnis oder unter einem anderen Namen generiert, kopieren, verknüpfen oder verschieben Sie die Datei mit dem privaten Schlüssel je nach Algorithmus nach *$HOME*/.ssh/id_rsa oder *$HOME*/.ssh/id_dsa.

Zugriffslisteneintrag für den Service-Account des Computersystems hinzufügen
Um die Kennwortauthentifizierung mit Secure Shell (SSH) zu konfigurieren, müssen Sie einen Zugriffslisteneintrag für den Service-Account des Computersystems, den Sie auf dem Zielsystem erstellt haben, hinzufügen.

Um einen Zugriffslisteneintrag für den Service-Account des Computersystems hinzuzufügen, führen Sie folgende Schritte aus:

1. Stellen Sie über die TADDM-Startseite sicher, dass alle Services in der Administratorkonsole gestartet wurden.
2. Starten Sie die Discovery Management Console.
3. Wählen Sie das Kontrollkästchen **Sichere Sitzung (SSL-Sitzung) aufbauen** aus, sodass die SSL-Sicherheitsoption verwendet wird. Bei Auswahl dieser Option werden alle Daten (einschließlich der Benutzernamen und Kennwörter in der Zugriffsliste) verschlüsselt, bevor sie zwischen der Discovery Management Console und dem TADDM-Server übertragen werden.
4. Fügen Sie einen Zugriffslisteneintrag für den Service-Account des Computersystems hinzu und geben Sie den Anmeldenamen und das Kennwort an.

System p und System i konfigurieren
Die Erkennung eines auf der IBM Power5-Technologie basierenden Systems (System p oder System i) und der zugehörigen logischen Partitionen erfolgt über eine Managementkonsole. TADDM unterstützt zwei Arten von Managementkonsolen: die Hardware Management Console (HMC) und den Integrated Virtualization Manager (IVM).

TADDM erkennt die Managementkonsole mit SSH. Der Erkennungsbereich muss die IP-Adresse der Managementkonsole enthalten und die Zugriffsliste einen Eintrag des Typs 'Computersystem' mit Angabe der entsprechenden Berechtigungsnachweise (Benutzername und Kennwort).

Neben den Benutzerberechtigungen muss der Erkennungsbenutzer in der Managementkonsole mit den folgenden Mindestberechtigungen definiert sein:

- Hardware Management Console (HMC)
  - Für eine HMC-Managementkonsole muss ein Benutzer auf Grundlage der **hmcoperator**-Rolle angelegt werden. Erstellen Sie beispielsweise eine neue Rolle namens *taddmViewOnly* auf Grundlage der hmcoperator-Rolle. Zudem müssen der neuen Rolle folgende Befehlszeilenaufgaben zugeordnet werden:

    Verwaltetes System
    Für die Verwendung der Befehle **lshwres** und **lssyscfg** erforderlich

    Logische Partition
    Für die Verwendung der Befehle **lshwres**, **lssyscfg** und **viosvrcmd** erforderlich.

    HMC-Konfiguration
    Für die Verwendung des Befehls **lshmc** erforderlich.
- Integrated Virtualization Manager (IVM)

  Für eine IVM-Managementkonsole muss ein Benutzer auf Grundlage der **View Only**-Rolle angelegt werden.

Für Erkennung der Ebene 3 konfigurieren
Neben den Voraussetzungen für eine Erkennung der Ebene 2 muss für die Erkennung der Ebene 3 Unterstützung für die Erkennung von Anwendungskonfigurations- und Hostdaten konfiguriert werden.

Web-Server und Anwendungsserver für die Erkennung konfigurieren

Sie müssen die Web-Server und Anwendungsserver in Ihrer Umgebung konfigurieren, die der TADDM-Server erkennen soll.

Dieser Abschnitt enthält die Schritte zur Konfiguration von Web-Servern und Anwendungsservern.

Für den Microsoft IIS-Server ist keine Konfiguration erforderlich. Es sind keine besonderen Zugriffsvoraussetzungen vorhanden. Das bereits auf dem Host errichtete Benutzerkonto verfügt über ausreichende Berechtigung.

Beim Apache-Web-Server muss der TADDM-Service-Account für das Hostsystem Leseberechtigungen für Apache-Konfigurationsdateien, zum Beispiel die Datei `httpd.conf`, besitzen.

Beim Oracle iPlanet-Web-Server muss der TADDM-Service-Account für das Hostsystem Leseberechtigungen für iPlanet-Konfigurationsdateien besitzen.

Stellen Sie für Lotus Domino-Server sicher, dass die Voraussetzungen erfüllt sind, die in Abschhnitt *IBM Lotus Domino-Serversensor* in den TADDM-*Referenzinformationen zu Sensoren* beschrieben sind.

Oracle-Anwendungsserver konfigurieren

Die Erkennung eines Oracle-Anwendungsservers verwendet JAR-Dateien, die im Oracle-Anwendungsserver enthalten sind. Diese JAR-Dateien sind nicht Bestandteil der TADDM-Serverinstallation.

Informationen zu diesem Vorgang

Die Datei `$COLLATION_HOME/etc/collation.properties` enthält eine Eigenschaft, mit der auf eine bereits vorhandene Oracle-Anwendungsserverinstallation verwiesen werden kann. Die Datei `$COLLATION_HOME/etc/collation.properties` enthält den folgenden Text:

```
# Location of the root directory for Oracle Application Server on
the Tivoli Application Dependency Discovery Manager
Server
# 1. An example is /home/oracle/product/10.1.3/OracleAS_1
# 2. A relative directory is relative to com.collation.home
# 3. This directory (and its subdirectories) must be accessible
     for the user under which the server runs, usually the collation user.
# 4. Ignore if you do not intend to discover an Oracle Application server.
```

Um auf einen bereits vorhandenen Oracle-Anwendungsserver zu verweisen, müssen Sie in der Datei `$COLLATION_HOME /etc/collation.properties` die folgende Zeile bearbeiten:

```
com.collation.oracleapp.root.dir=lib/oracleapp
```

In einer Oracle-Anwendungsserverinstallation sind die Verzeichnisse mit den erforderlichen JAR-Dateien Eigentum des `oracle`-Benutzers mit folgenden Berechtigungen: `rwx------`. Nur der Eigner (i. d. R. eine Oracle-Anwendung) kann also auf diese Verzeichnisse zugreifen. Wird der TADDM-Server unter dem Benutzer `oracle` ausgeführt, kann auf diese Verzeichnisse zugegriffen werden. Ist dies jedoch nicht der Fall, müssen Sie die Verzeichnisberechtigungen für die nachfolgenden Verzeichnisse in 711 ändern, damit sie allen Benutzern zugänglich sind:

- *OracleAppServerHome*
- *OracleAppServerHome*`/j2ee`
- *OracleAppServerHome*`/j2ee/home`
- *OracleAppServerHome*`/opmn`
- *OracleAppServerHome*`/opmn/lib`; dabei ist `/home/oracle/product/10.1.3/OracleAS_1` ein Beispiel für *OracleAppServerHome*.

Damit ein Oracle-Anwendungsserver erkannt werden kann, muss die Eigenschaft 'com.collation.platform.os.ignoreLoopbackProcesses' in der Datei `$COLLATION_HOME/etc/collation.properties` auf `true` gesetzt werden:

```
com.collation.platform.os.ignoreLoopbackProcesses=true
```

Vorgehensweise

Gehen Sie folgendermaßen vor, um die Zugriffsliste zu konfigurieren:

1. Erstellen Sie über die Discovery Management Console eine Erkennungsbereichsgruppe, die den Oracle-Anwendungsserver enthält, oder verwenden Sie einen bereits vorhandenen Bereich, in dem der Oracle-Anwendungsserver enthalten ist.
2. Klicken Sie zur Erstellung einer Zugriffsliste auf das Symbol **Zugriffsliste**.
3. Klicken Sie im Fenster 'Zugriffsliste' auf **Hinzufügen**.
4. Klicken Sie im Feld **Komponententyp** des Fensters 'Zugriffsdaten' auf **Anwendungsserver**.
5. Klicken Sie im Feld **Anbieter** auf **Oracle-Anwendungsserver**.
6. Geben Sie die Berechtigungsnachweise für den Oracle-Anwendungsserver ein.

VMware-Server konfigurieren

Bei einer ordnungsgemäßen Konfiguration der VMware-Server gibt der TADDM-Erkennungsprozess Informationen zu diesen Servern zurück.

Informationen zu diesem Vorgang

Um VMware-Server für die Erkennung zu konfigurieren, müssen Sie in der ESX-Konsole von WMware die Leseberechtigungen für den TADDM-Serviceaccount ohne Rootberechtigung setzen. Alternativ dazu können Sie den Rootbenutzer für die Erkennung verwenden. Weitere Informationen zu VMware-Server finden Sie in auf den Seiten der VMware-Community unter https://communities.vmware.com/welcome.

Für Erkennung eingerichtete Datenbank

Für Erkennungen in Ihrer Datenbank müssen Sie DB2-, Oracle- oder Sybase-Datenbankbenutzer für den TADDM-Server erstellen. Der TADDM-Server verwendet diese Datenbankbenutzer, um Informationen zu den Datenbanken zu erfassen, die auf fernen Hosts ausgeführt werden.

DB2-Benutzer erstellen

Für eine umfassendere Erkennung von DB2-Instanzen auf einem fernen Computer können Sie einen DB2-Benutzer erstellen.

Vorgehensweise

Gehen Sie folgendermaßen vor, um einen DB2-Benutzer zu erstellen:

1. Erstellen Sie einen Benutzer mit Zugriffsrechten auf folgende Komponenten:
   - Auf den TADDM-Server mit der DB2-Datenbank.
   - Auf alle Instanzen im TADDM-Server mit der DB2-Datenbank, die erkannt werden müssen.
2. Konfigurieren Sie diesen DB2-Benutzer so, dass er SSH-Zugriff auf das System hat, das den DB2-Datenbankserver betreibt.
3. Führen Sie in der TADDM-Serverzugriffsliste Folgendes aus, um den Benutzernamen und das Kennwort für den DB2-Benutzer hinzuzufügen:
   a) Klicken Sie in der Symbolleiste der Discovery Management Console auf **Erkennung** > **Zugriffsliste**.
      Das Teilfenster **Zugriffsliste** wird angezeigt.
   b) Klicken Sie auf **Hinzufügen**.
      Das Fenster **Zugriffsdaten** wird geöffnet.
   c) Gehen Sie im Fenster **Zugriffsdaten** wie folgt vor:
      1) Wählen Sie aus der Liste **Komponententyp Datenbank** aus.
      2) Wählen Sie in der Liste **Anbieter** die Option **DB2** aus.
      3) Geben Sie den Namen, den Benutzernamen und das Kennwort für den DB2-Benutzer ein.
   d) Klicken Sie auf **OK**, um die Angaben zu speichern.
      Das Teilfenster **Zugriffsliste** wird mit den neuen Informationen angezeigt.

Microsoft SQL Server-Benutzer erstellen
Für eine umfassendere Erkennung von Microsoft SQL Server-Instanzen auf einem fernen Computer können Sie einen Microsoft SQL Server-Benutzer erstellen.

Vorgehensweise

Gehen Sie folgendermaßen vor, um einen Microsoft SQL Server-Benutzer zu erstellen:

1. Erstellen Sie einen Microsoft SQL-Server-Benutzer mit der Rolle `db_datareader` und der Berechtigung `VIEW_ANY_DEFINITION`. Dies ist möglicherweise Aufgabe des Microsoft SQL Server-Administrators.
2. Führen Sie in der Discovery Management Console die folgenden Schritte aus, um der TADDM-Serverzugriffsliste den Benutzernamen und das Kennwort des Microsoft SQL Server-Benutzers hinzuzufügen:
   a) Klicken Sie in der Symbolleiste auf **Erkennung** > **Zugriffsliste**.
      Das Teilfenster **Zugriffsliste** wird angezeigt.
   b) Klicken Sie auf **Hinzufügen**.
      Das Fenster **Zugriffsdaten** wird geöffnet.
   c) Geben Sie im Fenster **Zugriffsdaten** die folgenden Informationen ein:
      1) Wählen Sie aus der Liste **Komponententyp Datenbank** aus.
      2) Wählen Sie in der Liste **Anbieter** die Option **Microsoft SQL Server** aus.
      3) Geben Sie den **Namen**, den **Benutzernamen** und das **Kennwort** ein.
   d) Klicken Sie auf **OK**, um die Angaben zu speichern.
      Die neuen Informationen werden jetzt im Teilfenster **Zugriffsliste** angezeigt.

Oracle-Benutzer erstellen
Für eine umfassendere Erkennung von Oracle-Instanzen auf fernen Computerhosts können Sie einen Oracle-Benutzer erstellen.

Vorgehensweise

Gehen Sie folgendermaßen vor, um einen Oracle-Benutzer zu erstellen:

1. Erstellen Sie einen Oracle-Benutzer mit SELECT_CATALOG_ROLE-Berechtigungen. Dies ist möglicherweise Aufgabe des Oracle-Administrators.

   Erstellen Sie beispielsweise den IBM Oracle-Benutzer mit folgendem Befehl:

   ```
   create user collation identified by collpassword;
   grant connect, select_catalog_role to collation;
   ```

2. Führen Sie in der Discovery Management Console die folgenden Schritte aus, um der TADDM-Serverzugriffsliste den Benutzernamen und das Kennwort des Oracle-Benutzers hinzuzufügen:
   a) Klicken Sie in der Symbolleiste auf **Erkennung** > **Zugriffsliste**.
      Das Teilfenster **Zugriffsliste** wird angezeigt.
   b) Klicken Sie auf **Hinzufügen**.
      Das Fenster **Zugriffsdaten** wird geöffnet.
   c) Gehen Sie im Fenster **Zugriffsdaten** wie folgt vor:
      1) Wählen Sie aus der Liste **Komponententyp Datenbank** aus.
      2) Wählen Sie in der Liste **Anbieter** die Option **Oracle** aus.
      3) Geben Sie den Namen, den Benutzernamen und das Kennwort für den Computer ein.
   d) Klicken Sie auf **OK**, um die Angaben zu speichern.
      Die neuen Informationen werden jetzt im Teilfenster **Zugriffsliste** angezeigt.

Sybase-Benutzer erstellen
Erstellen Sie zur vollständigen Erkennung von Sybase ASE auf fernen Computerhosts einen Sybase-Benutzer, dem Sie eine entsprechende Rolle zuweisen.

Vorgehensweise

Gehen Sie folgendermaßen vor, um einen Sybase-Benutzer zu erstellen:

1. Erstellen Sie mit dem folgenden Befehl einen Sybase-Benutzer, der Mitglied von 'sa-role' ist.

   ```
   sp_role "grant",sa_role,IBM
   ```

   Stellen Sie sicher, dass der Sybase IQ-Benutzer Mitglied von DBA ist. Ist der Sybase IQ-Benutzer nicht Mitglied von DBA, können keine Sybase IQ-datenbankspezifischen Informationen gefunden werden.
2. Führen Sie in der Discovery Management Console die folgenden Schritte aus, um der TADDM-Serverzugriffsliste den Benutzernamen und das Kennwort des Sybase-Benutzers hinzuzufügen:
   a) Klicken Sie zur Erstellung einer Zugriffsliste auf das Symbol **Zugriffsliste**.
   b) Klicken Sie im Fenster 'Zugriffsliste' auf **Hinzufügen**.
   c) Klicken Sie im Feld **Komponententyp** des Fensters 'Zugriffsdaten' auf **Datenbank**.
   d) Klicken Sie im Feld **Anbieter** auf **Datenbank**.
   e) Geben Sie die Berechtigungsnachweise (Benutzername und Kennwort) ein, um eine JDBC-Verbindung (Java Database Connectivity) zum Sybase-Server herzustellen.

Umgebung für die Erkennung von Windows-Systemen konfigurieren

Für die Erkennung von Windows-Computersystemen unterstützt TADDM sowohl die gatewaybasierte als auch die SSH-basierte Erkennung sowie die asynchrone und die scriptbasierte Erkennung.

Einzelheiten zur asynchronen Erkennung finden Sie im Abschnitt „Umgebung für eine asynchrone Erkennung konfigurieren” auf Seite 100. Einzelheiten zur scriptbasierten Erkennung finden Sie im Abschnitt „Konfiguration für die scriptbasierte Erkennung” auf Seite 104.

Bei der gatewaybasierten Erkennung muss ein dediziertes, über SSH zugängliches Windows-Computersystem als Gateway fungieren. Alle Erkennungsanforderungen werden durch das Gateway geleitet. Das Gateway verwendet die Windows Management Instrumentation (WMI) zur Erkennung der Windows-Zielcomputersysteme.

Fix Pack 2 Wenn Sie TADDM 7.3.0.2 oder höher statt WMI verwenden, können Sie auch die PowerShell-Sitzung verwenden, um die Windows-Zielcomputersysteme zu erkennen. Sie können TADDM so konfigurieren, dass die Kommunikation nur über die PowerShell-Sitzung möglich ist. Weitere Informationen siehe den Abschnitt *Konfiguration für Erkennung über eine Firewall ohne einen Anker* im TADDM-*Benutzerhandbuch* .

Die SSH-basierte Erkennung erfordert kein dediziertes Gateway-Computersystem. Stattdessen verwendet die Erkennung eine direkte SSH-Verbindung mit dem Windows-Zielcomputersystem.

Für gewöhnlich ist die gatewaybasierte Erkennung der SSH-basierten Erkennung vorzuziehen, da die Gateway- und WMI- oder PowerShell-Konfiguration einfacher ist als die SSH-Konfiguration. WMI ist standardmäßig auf allen Windows-Zielsystemen verfügbar, die von TADDM unterstützt werden. PowerShell wird nur für Ziele unterstützt, auf denen Windows Server 2008 und höher ausgeführt wird. Auf dem Gateway und den Zielsystemen muss PowerShell Version 2 oder höher installiert sein. Im Gegensatz zum Gateway-Computer ,der einen SSH-Server benötigt, bestehen für die Windows-Ziele keine besonderen Softwarevoraussetzungen. Die Erkennung unter Verwendung von SSH kann jedoch schneller sein, da im Erkennungsablauf kein Gateway involviert ist und kein WMI-Provider implementiert ist.

Für eine Direkterkennung wird auf allen Windows-Zielsystemen ein SSH-Server benötigt. Für eine direkte Erkennung mit SSH müssen Sie außerdem die Version 2 oder 3 von Microsoft .NET Framework auf jedem Windows-Zielsystem installieren. .NET Framework ist nicht standardmäßig auf Windows Server 2000 installiert.

Anmerkung: Fix Pack 2 Wenn Sie TADDM 7.3.0.2 oder höher verwenden, können Sie auch die .NET Framework-Versionen 4 oder 4.5 installieren.

Bei beiden Erkennungstypen wird das TADDM-Windows-Erkennungsprogramm TaddmTool.exe zur Durchführung der Erkennung verwendet. Bei der Erkennung unter Verwendung eines Gateways wird das TaddmTool-Programm während der Erkennungsinitialisierung im Gateway implementiert. Bei der Erkennung unter Verwendung von SSH wird das TaddmTool-Programm auf jedem Windows-Zielcomputersystem implementiert. Beim TaddmTool-Programm handelt es sich um eine .NET-Anwendung.

Standardmäßig ist TADDM nur für die gatewaybasierte Erkennung konfiguriert. Diese Konfiguration wird durch die folgenden zwei TADDM-Servereigenschaften gesteuert, die in den *Referenzinformationen zu Sensoren* für TADDM in Bezug auf den Sensor für Windows-Computersysteme beschrieben sind:

- `com.collation.AllowPrivateGateways=true`
- `com.collation.PreferWindowsSshOverGateway=false`

Standardmäßig ist TADDM für die Verwendung der WMI-Sitzung konfiguriert. Informationen dazu, wann die PowerShell-Sitzung verwendet werden soll und wie sie aktiviert wird, finden Sie im Abschnitt „PowerShell-Sitzung" auf Seite 121.

Unabhängig davon, ob Sie ein Windows-Gateway mit WMI verwenden oder eine direkte Verbindung zu SSH herstellen, werden immer dieselben Informationen abgerufen. In der folgenden Liste sind die Voraussetzungen für eine gatewaybasierte Erkennung und eine SSH-basierte Erkennung aufgeführt:

Voraussetzungen für die gatewaybasierte Erkennung mit WMI

1. Ein dediziertes Windows Server-Computersystem muss als Gateway fungieren. Die Betriebssystemvoraussetzungen für Gateway-Server sind mit den Windows-Betriebssystemvoraussetzungen für TADDM-Server identisch. Details zu unterstützten Windows-Betriebssystemen siehe den Abschnitt *TADDM-Server-Softwareanforderung* im TADDM-*Installationshandbuch*.
2. Das Gateway muss sich in derselben Firewallzone befinden wie die zu erkennenden Windows-Computer.
3. Sie müssen eine unterstützte Version eines SSH-Servers auf dem Gateway-Computersystem installieren.
4. Das Gateway verwendet eine ferne WMI zur Erkennung der einzelnen Windows-Ziele. Außerdem wird während der Initialisierung der Erkennung automatisch ein WMI-Provider auf jedem Windows-Zielcomputersystem implementiert. Mit dem WMI-Provider werden Daten nicht in die Kern-WMI aufgenommen. Aktivieren Sie WMI auf dem Windows-Zielcomputersystem, das erkannt werden soll. Standardmäßig ist WMI auf den meisten Windows 2000-Systemen und höheren Systemen aktiviert.

Fix Pack 2 Voraussetzungen für die gatewaybasierte Erkennung mit PowerShell

1. Ein dediziertes Windows Server-Computersystem muss als Gateway fungieren. Die Betriebssystemvoraussetzungen für Gateway-Server sind mit den Windows-Betriebssystemvoraussetzungen für TADDM-Server identisch. Details zu unterstützten Windows-Betriebssystemen siehe den Abschnitt *TADDM-Server-Softwareanforderung* im TADDM-*Installationshandbuch*.
2. Auf dem Gateway und den Zielsystemen müssen Sie PowerShell Version 2 oder höher installieren. Es werden nur Ziele unterstützt, auf denen Windows Server 2008 oder höher ausgeführt wird.
3. Sie müssen das Gateway mit folgendem Befehl konfigurieren:

   ```
   Set-Item WSMan:\localhost\Client\TrustedHosts * -Force
   ```

   Mit diesem Befehl wird die Liste **trustedHosts** festgelegt. Standardmäßig ist die Liste vorhanden, ist jedoch leer, sodass sie festgelegt werden muss, bevor die ferne Sitzung geöffnet wird. Mit dem Parameter `-Force` führt PowerShell den Befehl aus, ohne dass Sie für jeden Schritt zur Eingabe aufgefordert werden.
4. Sie müssen die Zielsysteme mit folgendem Befehl konfigurieren:

   ```
   Enable-PSRemoting -Force
   ```

Mit diesem Befehl wird der WinRM-Service gestartet, es wird festgelegt, dass er automatisch mit Ihrem System gestartet wird und es wird eine Firewallregel erstellt, um die eingehenden Verbindungen zuzulassen. Mit dem Parameter `-Force` führt PowerShell diese Aktionen aus, ohne dass Sie für jeden Schritt zur Eingabe aufgefordert werden.

Voraussetzungen für die SSH-basierte Erkennung

1. Sie müssen eine unterstützte Version eines SSH-Servers auf jedem Windows-Zielsystem installieren.
2. Sie müssen die Version 2 oder 3 von Microsoft .NET Framework auf jedem Windows Server-Zielsystem installieren.

   Anmerkung: Fix Pack 2 Wenn Sie TADDM 7.3.0.2 oder höher verwenden, können Sie auch die .NET Framework-Versionen 4 oder 4.5 installieren.

Weitere Informationen hierzu finden Sie im Abschnitt *Konfiguration für eine Windows-Erkennung ohne Administrator* in der TADDM *Sensor-Referenz*.

Bitvise WinSSHD konfigurieren

Bitvise WinSSHD ermöglicht Ihnen SSH-Zugriff auf Windows-Systeme.

Vorbereitende Schritte

Bei gatewaybasierter Erkennung muss Bitvise WinSSHD auf dem Gatewaysystem installiert sein. Bei direkter SSH-Erkennung muss Bitvise WinSSHD auf jedem Windows-System installiert sein.

Weitere Informationen zu den unterstützten Bitvise WinSSHD-Versionen finden Sie im TADDM-*Installationshandbuch* im Abschnitt *Windows-Gateways*.

Bitvise WinSSHD ist unter folgender Adresse verfügbar: http://www.bitvise.com/.

Informationen zu diesem Vorgang

In den folgenden Schritten wird beschrieben, wie Bitvise WinSSHD 5.22 konfiguriert wird. Die einzelnen Schritte können je nach Bitvise WinSSHD-Release variieren.

Vorgehensweise

1. Führen Sie folgende Schritte aus, um den SSH-Hostzugriff auf den TADDM-Server zu beschränken:
   a) Klicken Sie in der **WinSSHD-Steuerkonsole** auf **Open easy settings** (Einfache Einstellungen öffnen).
   b) Wählen Sie auf der Registerkarte **Server settings** (Servereinstellungen) für das Feld **Open Windows Firewall** (Windows-Firewall öffnen) **As set in Advanced WinSSHD settings** (Wie in erweiterten WinSSHD-Einstellungen festgelegt) aus.
   c) Klicken Sie auf **Save Changes** (Änderungen speichern).
   d) Klicken Sie in der **WinSSHD-Steuerkonsole** auf **Edit advanced settings** (Erweiterte Einstellungen bearbeiten).

      Das Fenster **Advanced WinSSDH Settings** (Erweiterte WinSSDH-Einstellungen) wird angezeigt.
   e) Klicken Sie auf **Einstellungen** > **Sitzung**.
   f) Setzen Sie den Wert der folgenden Elemente auf `0`:
      - IP blockieren - Fensterdauer
      - IP blockieren - Sperrzeitdauer
   g) Klicken Sie auf **OK**.
   h) Klicken Sie in der **WinSSHD-Steuerkonsole** auf **Edit advanced settings** (Erweiterte Einstellungen bearbeiten).

      Das Fenster **Advanced WinSSDH Settings** (Erweiterte WinSSDH-Einstellungen) wird angezeigt.
   i) Klicken Sie auf **Einstellungen** > **Zugriffssteuerung**.

j) Klicken Sie im rechten Teilfenster auf **IP-Regeln**.

k) Klicken Sie auf **Hinzufügen**.

l) Geben Sie die IP-Adresse des TADDM-Server ein.

m) Geben Sie im Feld **Number of significant bits** (Anzahl bedeutender Bits) 32 ein.

n) Geben Sie im Feld **Beschreibung** `TADDM-Server` ein.

o) Stellen Sie sicher, dass das Kontrollkästchen **Allow connect** (Verbindung zulassen) ausgewählt ist.

p) Klicken Sie auf **OK**.

q) Entfernen Sie den Eintrag `0.0.0.0/0` aus der Liste.

2. Führen Sie folgende Schritte aus, um eine virtuelle Gruppe und Benutzer zu erstellen und zu konfigurieren:

a) Klicken Sie in der **WinSSHD-Steuerkonsole** auf **Edit advanced settings** (Erweiterte Einstellungen bearbeiten).

Das Fenster **Advanced WinSSDH Settings** (Erweiterte WinSSDH-Einstellungen) wird angezeigt.

b) Klicken Sie auf **Einstellungen** > **Virtuelle Gruppen**.

c) Klicken Sie auf **Hinzufügen**, um eine neue Gruppe hinzuzufügen.

d) Geben Sie in den Feldern **Gruppe** und **Windows-Accountname** einen Namen ein.

e) Klicken Sie auf **OK**.

f) Klicken Sie auf **Einstellungen** > **Virtual Accounts (Virtuelle Accounts)**.

g) Klicken Sie auf **Hinzufügen**, um einen neuen Account hinzuzufügen.

h) Geben Sie im Feld **Virtual account name** (Virtueller Accountname) einen Namen ein.

i) Legen Sie mithilfe des Links für das Kennwort des virtuellen Accounts ein Kennwort fest.

j) Wählen Sie aus der Dropdown-Liste die virtuelle Gruppe aus, die Sie in einem vorherigen Schritt erstellt haben, und stellen Sie sicher, dass das Kontrollkästchen **Use group default Windows account** (Windows-Standardgruppenaccount verwenden) ausgewählt ist.

k) Klicken Sie auf **OK**.

3. Klicken Sie in der **WinSSHD-Steuerkonsole** auf **Start WinSSHD**.

Nächste Schritte

Wenn mehrere Windows-Server erkannt werden, erhalten Sie möglicherweise die folgende Nachricht:

```
A Working gateway cannot be found (Es wurde kein funktionierendes
Gateway gefunden)
```

Weitere Informationen zu möglicherweise hilfreichen zusätzlichen Konfigurationsschritten finden Sie im *Handbuch zur Fehlerbehebung* von TADDM im Abschnitt *Gateway problems* (Probleme mit dem Gateway).

Cygwin SSH-Dämon konfigurieren

Der Cygwin SSH-Dämon (sshd) ermöglicht Ihnen SSH-Zugriff auf Windows-Systeme.

Informationen zu diesem Vorgang

Bei gatewaybasierter Erkennung muss der Cygwin SSH-Dämon auf dem Gatewaysystem installiert sein, bei direkter SSH-Erkennung muss der Dämon auf jedem Windows-System installiert sein.

Weitere Informationen zu den unterstützten Versionen des Cygwin SSH-Dämons finden Sie im TADDM-*Installationshandbuch* im Abschnitt *Windows-Gateways*.

Wichtig: Für eine erfolgreiche Erkennung mithilfe von Cygwin SSH müssen die folgenden Voraussetzungen erfüllt sein:

- Anker und Gateways werden in der 64-Bit-Edition von Cygwin auf Windows Server 2012 x64 unterstützt.

- Der Erkennungsbenutzer und der Benutzer, der den Service startet, müssen identisch sein. Der Erkennungsbenutzer muss ein Mitglied der Administratorgruppe sein.

Cygwin ist unter folgender Adresse verfügbar: http://www.cygwin.com/.

Vorgehensweise

Gehen Sie folgendermaßen vor, um den Cygwin SSH-Dämon zu konfigurieren:

1. Starten Sie die Shell **cygwin bash**.
2. Erstellen Sie mithilfe des **cygwin mkpasswd**-Dienstprogramms ein `/etc/passwd` aus Ihren Systeminformationen.

   Mit dem Dienstprogramm **mkgroup** können Sie auch eine erste `/etc/`-Gruppe erstellen. Weitere ausführliche Informationen finden Sie im *Cygwin User's Guide*.

   Mit dem folgenden Befehl setzen Sie beispielsweise die Kennwortdatei `passwd` aus den lokalen Accounts Ihres Systems zusammen:

   ```
   mkpasswd -l > /etc/passwd
   ```
3. Führen Sie die Programmkonfiguration **ssh-host-config** aus.
4. Konfigurieren Sie SSH. Bestätigen Sie alle Fragen mit `Ja`.
5. Starten Sie den SSH-Server mit folgendem Befehl:

   ```
   net start sshd
   ```

Nächste Schritte

Für den Cygwin-Service (sshd) muss bei einem Zugriff auf den Gateway-Server ein Benutzeraccount der Verwaltungsdomäne verwendet werden. Dieser Benutzeraccount ist für einige Sensoren wie den Microsoft Exchange-Sensor erforderlich. Gehen Sie wie folgt vor:

1. Konfigurieren Sie den Domänenbenutzeraccount, indem Sie die folgenden Befehle ausführen:

   ```
   mkpasswd -u [domain_user] -d [domain] >> /etc/passwd
   mkgroup -d [domain] >> /etc/group
   ```
2. Starten Sie das Programm 'services.msc'. Überprüfen Sie die Anmeldeeigenschaften für den Cygwin-Service (sshd), der erstellt wurde. Prüfen Sie, ob der Service so konfiguriert ist, dass er über einen Benutzeraccount der Verwaltungsdomäne ausgeführt werden kann.
3. Die Cygwin-Konfiguration (sshd) und -Protokolldateien müssen dem Benutzeraccount der Verwaltungsdomäne zugeordnet sein, über den der Cygwin-Service (sshd) auf das Gateway zugreift. Führen Sie die folgenden Befehle aus:

   ```
   $ chown [domain_user] /var/log/sshd.log
   $ chown -R [domain_user] /var/empty
   $ chown [domain_user] /etc/ssh*
   ```
4. Der Domänenbenutzeraccount muss auf dem Gateway-Server über die Berechtigung verfügen,
   - die Speichergrößenbeschränkungen für einen Prozess anzupassen,
   - Tokenobjekte zu erstellen,
   - sich als Dienst anzumelden,
   - Token der Prozessebene zu ersetzen.

Wenn mehrere Windows-Server erkannt werden, erhalten Sie möglicherweise die folgende Nachricht:

```
A Working gateway cannot be found (Es wurde kein funktionierendes
Gateway gefunden)
```

Weitere Informationen zu möglicherweise hilfreichen zusätzlichen Konfigurationsschritten finden Sie im *Handbuch zur Fehlerbehebung* von TADDM im Abschnitt *Gateway problems* (Probleme mit dem Gateway).

RemotelyAnywhere konfigurieren
RemotelyAnywhere ermöglicht Ihnen SSH-Zugriff auf Windows-Systeme.

Informationen zu diesem Vorgang

Weitere Informationen zu den unterstützten Remotely Anywhere-Versionen finden Sie im TADDM-*Installationshandbuch* im Abschnitt *Windows-Gateways*.

Für eine gatewaybasierte Erkennung muss Remotely Anywhere auf dem Gatewaysystem installiert werden.

Für eine direkte SSH-Erkennung muss Remotely Anywhere auf jedem Windows-System installiert werden.

Sie können die Standardkonfigurationswerte in Remotely Anywhere verwenden. Weitere Informationen finden Sie unter http://remotelyanywhere.com/.

Tectia SSH Server konfigurieren
Über den Tectia SSH Server können Sie SSH-Zugriff auf Windows-Systeme bereitstellen.

Informationen zu diesem Vorgang

Informationen zu den unterstützten Versionen von Tectia SSH Server finden Sie im TADDM-*Installationshandbuch* im Abschnitt *Windows-Gateways*.

Für eine gatewaybasierte Erkennung muss Tectia SSH Server auf dem Gatewaysystem installiert werden.

Für eine direkte SSH-Erkennung muss Tectia SSH Server auf jedem Windows-System installiert werden.

Sie können die Standardkonfigurationswerte von Tectia SSH Server verwenden. Weitere Informationen finden Sie unter http://www.ssh.com.

OpenSSH-Server konfigurieren
Mit dem OpenSSH-Server können Sie SSH-Zugriff auf Windows-Systemen bereitstellen.

Informationen zu diesem Vorgang

Der OpenSSH-Server ist als installierbare Funktion Windows Server 2019 verfügbar.

Bei einer gatewaybasierten Erkennung muss der OpenSSH-Server auf dem Gatewaysystem installiert werden.

Bei der direkten SSH-Erkennung muss der OpenSSH-Server auf jedem Windows-System installiert sein.

Vorgehensweise

Informationen zum Konfigurieren von OpenSSH finden Sie hier: https://docs.microsoft.com/en-us/windows-server/administration/openssh/openssh_install_firstuse.

Abhängigkeit von Windows Management Instrumentation (WMI)
TADDM verwendet Windows Management Instrumentation (WMI) zur Erkennung von Windows-Computersystemen. Der TADDM kann so konfiguriert werden, dass ein Neustart des WMI-Service durchgeführt wird, wenn bei WMI ein Problem auftritt. Wenn der WMI-Service erneut gestartet wird, werden alle von WMI abhängigen Services, die vor dem Neustart aktiv waren, ebenfalls neu gestartet.

Folgende TADDM-Servereigenschaften steuern den WMI-Neustart.

Anmerkung: Der Standardwert für WMI-Neustarts lautet `false`. Wenn Sie die Werte der folgenden Eigenschaften auf `true` setzen, wird die Windows-Erkennung zwar möglicherweise zuverlässiger, aber es müssen auch die potenziell negativen Auswirkungen berücksichtigt werden, die auftreten, wenn der WMI-Service vorübergehend gestoppt und neu gestartet wird.

- `com.collation.RestartWmiOnAutoDeploy=false`
- `com.collation.RestartWmiOnAutoDeploy.1.2.3.4=false`

- `com.collation.RestartWmiOnFailure=false`
- `com.collation.RestartWmiOnFailure.1.2.3.4=false`

Weitere Informationen zu den TADDM-Servereigenschaften, die der Windows-Computersystemsensor verwendet, siehe den Abschnitt *Datei collation.properties konfigurieren* über den Windows-Computersystemsensor in den TADDM-*Referenzinformationen zu Sensoren*.

Fix Pack 2 **PowerShell-Sitzung**

Um Windows-Computersysteme zu erkennen, können Sie entweder die WMI- oder die PowerShell-Sitzung verwenden. Im Vergleich zur WMI-Sitzung sendet TADDM bei der PowerShell-Sitzung weniger Anforderungen für den Zugriff auf Zielsysteme, wodurch die Anzahl der Ereignisse, die protokolliert werden, reduziert wird. Die PowerShell-Sitzung kann nur mit den scriptbasierten Sensoren verwendet werden. Wenn Sie mit der Verwendung der PowerShell-Sitzung beginnen möchten, müssen Sie sie aktivieren, da sie standardmäßig inaktiviert ist.

Sie können beide Sitzungen gleichzeitig verwenden. Wenn Sie reguläre und scriptbasierte Erkennungen ausführen, können Sie die WMI-Sitzung nicht inaktivieren, da die reguläre Erkennung ohne sie fehlschlägt. Sie können die Verwendung der PowerShell-Sitzung jedoch priorisieren.

Wichtig: Wenn Sie nur reguläre Erkennungen ausführen, wird die PowerShell-Sitzung nicht unterstützt.

Sie können die Verwendung und Priorisierung der PowerShell-Sitzung mithilfe der folgenden Eigenschaften steuern:

- `com.collation.PowerShellAccessEnabled=false`
- `com.collation.WmiAccessEnabled=true`
- `com.collation.PreferPowerShellOverWMI=true`
- `com.collation.PowerShellPorts=5985,5986`
- `com.ibm.cdb.session.ps.useSSL=false`
- `com.ibm.cdb.session.ps.allowDNS=true`
- `com.ibm.cdb.session.ps.fallbackToIP=true`
- `com.collation.PowerShellTimeoutFudge=10000`
- **Fix Pack 3** `com.ibm.cdb.session.ps.urlPrefix=wsman`

Um die PowerShell-Sitzung zu aktivieren, setzen Sie die Eigenschaft `com.collation.PowerShellAccessEnabled` auf `true`. Die PowerShell-Sitzung wird gegenüber der WMI-Sitzung standardmäßig bevorzugt.

Weitere Informationen zu diesen Eigenschaften finden Sie im Abschnitt *Dateieinträge in `collation.properties` konfigurieren* für den Windows-Computersystemsensor in den *Referenzinformationen zu Sensoren* für TADDM.

Anmerkung: In einem sehr speziellen Fall, wenn Sie Ihre Firewall so konfiguriert haben, dass die Kommunikation nur über PowerShell-Sitzungen möglich ist, müssen Sie PowerShell-Ports öffnen und die Ping-Sensoreigenschaft konfigurieren. Weitere Informationen siehe den Abschnitt *Konfiguration für Erkennung über eine Firewall ohne einen Anker* im TADDM-*Benutzerhandbuch*.

Beispielszenarios

Je nachdem, wie Sie Ihre Windows-Zielsysteme erkennen, können Sie die vorhergehenden Eigenschaften auf die folgenden Arten konfigurieren.

- Sie verwenden nur die Sensoren, die die scriptbasierte Erkennung unterstützen. In diesem Fall können Sie die PowerShell-Sitzung aktivieren, indem Sie die Eigenschaft `com.collation.PowerShellAccessEnabled` auf `true` setzen, und die WMI-Sitzung inaktivieren, indem Sie die Eigenschaft `com.collation.WmiAccessEnabled` auf `false` setzen. Wenn PowerShell jedoch nicht verfügbar ist, schlagen die Sitzung und die Erkennung fehl.
- Sie verwenden Sensoren, die die scriptbasierte und die reguläre Erkennung unterstützen. Inaktivieren Sie in diesem Fall die WMI-Sitzung nicht, da sonst die reguläre Erkennung fehlschlägt. Aktivieren Sie die

PowerShell-Sitzung, indem Sie die Eigenschaft `com.collation.PowerShellAccessEnabled` auf `true` setzen. Um die PowerShell-Sitzung einzurichten, wann immer es möglich ist, ändern Sie den Standardwert der Eigenschaft `com.collation.PreferPowerShellOverWMI` nicht. In diesem Fall erstellt TADDM eine Hybridsitzung, die sowohl PowerShell- als auch WMI-Funktionen verwenden kann. Die WMI-Sitzung wird nur verwendet, wenn die PowerShell-Sitzung keine Tasks ausführen kann, die von den regulären Sensoren angefordert werden.

Fix Pack 3 Für Erkennung von Platzhaltern konfigurieren

Sie können TADDM so konfigurieren, dass es Platzhalter für nicht erkannte Abhängigkeiten in Ihrer Infrastruktur erstellt.

Platzhalter sind Objekte, die zwar Teil einer Infrastruktur sind, aber in TADDM mit den Standardeinstellungen nicht dargestellt werden. Für diese fehlende Darstellung kann es verschiedene Gründe geben: eine Seite der Verbindung wird nicht erkannt, es gibt keinen Sensor für diese Art von Objekt oder es ist keine benutzerdefinierte Serverschablone dafür erstellt.

Platzhalter gehören zur Klasse SSoftwareServer. Für sie sind die Attribute `hierarchyDomain` und `hierarchyType` gesetzt. In der folgenden Tabelle sind die Werte der Attribute angegeben:

Tabelle 36. Werte der Attribute `hierarchyDomain` und `hierarchyType`

Verbindungsseite	**`hierachyDomain`-Attributwert**	**`hierarchyType`-Attributwert**
Lokal	`app.placeholder.client.local`	Name des Befehls, der die Verbindung aufbaut, z. B. `Java`
Remote	`app.placeholder.server.remote`	`Unbekannt`

Bei Verwendung dieser Werte können Sie unerwünschte Beziehungen in der Traversierungskonfiguration von Geschäftsanwendungen filtern. Details siehe Abschnitt *Traversierungskonfiguration* im TADDM *Benutzerhandbuch*.

Wird ein Platzhalter erstellt und anschließend der äquivalente App-Server durch einen Sensor oder eine benutzerdefinierte Serverschablone erstellt, führt der PlaceholderCleanupAgent den Platzhalter mit dem erkannten App-Server zusammen.

Anmerkung: Sie können Platzhalter in TADDM 7.3.0.2 erstellen, allerdings nur mit Einschränkungen. Deshalb wird empfohlen, Platzhalter in TADDM 7.3.0.3 und höher zu verwenden. Die Migration von in FP2 erstellten Platzhaltern nach FP3 wird nicht unterstützt.

Erstellung von Platzhaltern aktivieren

Um die Erstellung von Platzhaltern zu aktivieren, müssen Sie folgende Eigenschaft zur Datei `collation.properties` hinzufügen:

```
com.ibm.cdb.topomgr.topobuilder.agents.ConnectionDependencyAgent2
dependencyPlaceholders=true
```

Der Standardwert ist `false`.

Wenn Sie diese Eigenschaft zum ersten Mal auf `true` setzen, müssen Sie TADDM erneut starten, um erweiterte Attribute für LogicalConnection- und SoftwareServer-Klassen zu aktivieren. Die erweiterten Attribute sind für eine ordnungsgemäße Ausführung dieser Funktion notwendig.

Wenn die oben genannte Eigenschaft auf `true` gesetzt ist, müssen die folgenden Eigenschaften nicht explizit in der Datei `collation.properties` festgelegt werden. Vielmehr werden die standardmäßig codierten Werte verwendet.

```
com.ibm.taddm.dependencyPlaceholders.create.localClient.to.remoteServer
=true
```

Der Standardwert ist `true`.

```
com.ibm.taddm.dependencyPlaceholders.create.remoteClient.to.localServer
=false
```

Der Standardwert ist `false`.

Anmerkung: Das Verhalten von Platzhaltern kann geändert werden, indem diese Eigenschaften in `collation.properties` festgelegt werden.

Wichtig: Wenn Sie die Erstellung von Platzhaltern aktivieren, werden Ihre Geschäftsanwendungen möglicherweise deutlich größer und der Erstellungsprozess kann länger dauern. Um dies zu verhindern, können Sie unerwünschte Beziehungen in der Traversierungskonfiguration von Geschäftsanwendungen filtern.

Platzhalter anzeigen

Die Platzhalter können im Teilfenster **Bestandsübersicht** angezeigt werden, nachdem Sie den Filter auf `Platzhalter` gesetzt haben. Platzhalter für nicht erkannte Abhängigkeiten finden sich auf der Registerkarte **Softwareserver**.

Benutzerdefinierte Serverschablonen erstellen

Sie können mithilfe von Platzhaltern wie folgt benutzerdefinierte Serverschablonen erstellen:

- Indem Sie Informationen zu Platzhaltern verwenden, die von dem Tool `bizappscli` generiert werden. Details finden Sie im Abschnitt *Aktionen zur Analyse des Inhalts von Geschäftsanwendungen* im TADDM *Benutzerhandbuch*.
- Indem Sie die Befehlszeileninformationen verwenden, die auf der Registerkarte **Laufzeit** im Fenster **Details** für Platzhalter des Typs `app.placeholder.*.local` angezeigt werden.

Weitere Informationen zu benutzerdefinierten Serverschablonen finden Sie im Abschnitt *Benutzerdefinierte Serverschablonen erstellen und verwalten* im TADDM *Benutzerhandbuch*.

Fix Pack 2 Anwendungsserver der Ebene 3 ohne Berechtigungsnachweise erstellen

Wenn Basisinformationen der Ebene 3 zu Ihren Infrastrukturelementen erkannt werden sollen, müssen Sie nicht Berechtigungsnachweise in der Zugriffsliste angeben. Sie können Anwendungsserver mithilfe interner Sensorschablonen erstellen. Diese Schablonen können von CustomAppServerTopoAgent oder während eines Erkennungslaufs von einem angepassten Serverschablonensensor verarbeitet werden.

Informationen zu diesem Vorgang

Indem Sie Anwendungsserver ohne Berechtigungsnachweise erstellen, können nur Basisinformationen zu Ihrer Infrastruktur erkannt werden, z. B., welche Art von Software installiert ist. Wählen Sie diesen Modus, wenn Sie keine Berechtigungsnachweise für die Erkennung der Ebene 3 bereitstellen möchten, aber Basisinformationen zu Ihrer Infrastruktur erkannt werden sollen.

Es gibt zwei Methoden zum Erstellen von Anwendungsservern der Ebene 3. Sie können einen angepassten Serverschablonensensor ausführen oder CustomAppServerTopoAgent aktivieren.

Prozedur

- Führen Sie einen angepassten Serverschablonensensor aus.

  Gehen Sie wie folgt vor:

  1. Setzen Sie in der Datei `collation.properties` die Eigenschaft `com.collation.internaltemplatesenabled` auf `true`. Diese Eigenschaft aktiviert interne Schablonen von Sensoren der Ebene 3. Der Standardwert ist `false`.
  2. Führen Sie die Erkennung unter Verwendung eines Profils aus, das keinen Sensor enthält, der normalerweise die Informationen erkennen würde, die mithilfe des angepassten Serverschablonensensors erkannt werden sollen. Wenn beispielsweise Basisinformationen für den DB2-Server erkannt werden sollen, wählen Sie die Profilerkennung der Ebene 2 oder ein eigenes Profil, das kei-

nen IBM DB2-Sensor enthält, aus. Wenn das Profil einen IBM DB2-Sensor enthält, wird dieser Sensor anstelle des angepassten Serverschablonensensors ausgeführt.

- Führen Sie CustomAppServerTopoAgent aus.

  CustomAppServerTopoAgent verwendet Laufzeitprozesse, die zuvor vom generischen Serversensor erkannt wurden. Sie können den Agenten manuell ausführen oder so einstellen, dass er automatisch ausgeführt wird. Gehen Sie wie folgt vor:

  1. Setzen Sie sowohl für den manuellen als auch den automatischen Modus des Agenten in der Datei `collation.properties` die Eigenschaft `com.collation.internaltemplatesenabled` auf `true`. Diese Eigenschaft aktiviert interne Schablonen von Sensoren der Ebene 3. Der Standardwert ist `false`.
  2. Führen Sie folgenden Befehl aus, um CustomAppServerTopoAgent manuell zu starten:

     ```
     COLLATION_HOME/support/bin/runtopobuild.sh -a CustomAppServerTopoAgent
     ```

  3. Wenn Sie den Agenten für automatische Ausführungen konfigurieren möchten, legen Sie für die Eigenschaft `com.ibm.cdb.topobuilder.groupinterval.discovery=` in der Datei `collation.properties` einen Wert fest.
     Diese Eigenschaft gibt an, wie oft der Agent ausgeführt wird. Standardmäßig ist kein Wert angegeben, d. h., der Agent ist inaktiviert. Geben Sie zum Aktivieren des Agenten einen Wert in Stunden an, z. B. `com.ibm.cdb.topobuilder.groupinterval.discovery=4`.

- Optional: Wählen Sie Schablonen aus, die von der Verarbeitung ausgeschlossen werden sollen.

  Wenn Sie nur einige interne Schablonen von Sensoren der Ebene 3 aktivieren möchten, können Sie dies über folgende Eigenschaft steuern:

  ```
  com.collation.discovery.ignoreTemplateList
  ```

  Diese Eigenschaft gibt eine Liste mit internen Schablonen an, die nicht verarbeitet werden sollen. Der Wert dieser Eigenschaft ist eine durch Semikolons getrennte Liste mit Schablonennamen, z. B. `com.collation.discovery.ignoreTemplateList=DB2Unix;MSSQL`. Sie können den Namen einer internen Schablone im Datenmanagementportal im Feld **Objektname** finden, das sich auf der Registerkarte **Allgemein** im Fenster **Details** von **Andere Datenbankserver** befindet.

Positions-Tagging konfigurieren

Das Positions-Tagging zeigt an, wo jedes Konfigurationselement erstellt wurde. Es wird das positionsbezogene Filtern von Konfigurationselementen in BIRT-Berichten und API-Abfragen ermöglicht.

Wenn Sie das Positions-Tagging aktivieren, enthält jedes erkannte Objekt, das in der Erkennungsdatenbank gespeichert wird, das Attribut **`locationTag`** (Zeichenfolge). Objekte wie beispielsweise Beziehungen, Aggregationsobjekte und Vererbungsobjekte, die von Topologieagenten erstellt werden, enthalten unter bestimmten Bedingungen Daten zum Positions-Tagging:

- Eine Eins-zu-eins-Beziehung (z. B. als Abhängigkeit oder Netzverbindung) enthält einen Positions-Tag, wenn die Position für beide verbundene Objekte verwendet wird.
- Ein Aggregationsobjekt (z. B. ein Cluster) enthält einen Positions-Tag, wenn die Position für alle zusammengefassten Objekte verwendet wird.

  Anmerkung: Bei benutzerdefinierten Objektgruppen wird das Attribut **`locationTag`** nur gesetzt, wenn die locationTag-*Werte* aller Core-CIs der benutzerdefinierten Objektgruppe identisch sind. Sobald eine Core-CI mit einem anderen locationTag-Wert hinzukommt, wird das Attribut **`locationTag`** dieser benutzerdefinierten Objektgruppe gelöscht.
- Ein einfaches Objekt enthält den Positions-Tag aus dem Objekt, auf dem es basiert.

In allen anderen Fällen enthalten Objekte, die von Topologieagenten erstellt werden, keinen Wert für einen Positions-Tag.

Zur Aktivierung des Positions-Tagging legen Sie in der Datei `collation.properties` die folgende Eigenschaft fest:

```
com.ibm.cdb.locationTaggingEnabled=true
```

Die Werte für das Positions-Tagging können statisch (für einen bestimmten Server oder Anker) oder dynamisch (für eine bestimmte Erkennung oder den Import eines bestimmten IdML-Buches) sein. Der Wert eines Positions-Tags ist auf 192 Zeichen beschränkt. Wenn der angegebene Positions-Tag 192 Zeichen überschreitet, wird er auf die erforderliche Länge gekürzt.

Beschränkungen
Wenn Sie ein Verzeichnis der Ebene 1 ausführen, werden die bereits in der Datenbank enthaltenen Konfigurationselemente nicht aktualisiert. Folglich werden Positionstags nur den neu erkannten Objekten zugeordnet.

Statisches Positions-Tagging
Beim statischen Positions-Tagging wird das Attribut **`locationTag`** allen Objekten zugeordnet, die mithilfe des Imports des IdML-Buchs auf Basis der statischen Konfiguration des TADDM- oder Ankerservers erkannt oder geladen wurden.

TADDM-Server

Für die Konfiguration des Werts für den Positions-Tag von Konfigurationselementen, die auf einem TADDM-Server erstellt werden, geben Sie in der Datei `collation.properties` folgende Eigenschaft an:

```
com.ibm.cdb.locationTag=location
```

Dabei ist **`location`** der Wert des Positions-Tags, den Sie verwenden möchten.

Anker

Für die Konfiguration des Werts für den Positions-Tag von Konfigurationselementen, die auf einem Anker erstellt werden, konfigurieren Sie das Attribut **anchor_location_n** in der Datei `$COLLATION_HOME/etc/anchor.properties`. Die folgenden Beispieleinträge aus der Datei `anchor.properties` zeigen an, wie die Positionsinformationen für Anker festgelegt werden:

```
anchor_host_1=192.168.1.13
anchor_scope_1=FIRST_SCOPE
anchor_zone_1=FIRST_ZONE
anchor_location_1=FIRST_LOCATION
anchor_host_2=192.168.2.22
anchor_scope_2=SECOND_SCOPE
anchor_location_2=SECOND_LOCATION
Port=8497
```

Wenn für einen Anker kein Positions-Tag angegeben ist, wird die Position jedes Konfigurationselements, das im Anker erstellt wurde, auf die Position gesetzt, die für den TADDM-Server angegeben ist, mit dem die Konfigurationselemente verbunden sind.

Wenn für den Anker oder den TADDM-Server kein Wert für den Positions-Tag angegeben ist, werden für dieses Konfigurationselement keine Positionsinformationen festgelegt.

Dynamisches Positions-Tagging
Durch das dynamische Positions-Tagging wird das Attribut **`locationTag`** mit einem Wert festgelegt, der für eine bestimmte Erkennung oder den Import eines IdML-Buches angegeben wird.

Erkennung

Um während der Erkennung einen Wert für das Positions-Tagging anzugeben, starten Sie die Erkennung aus der Befehlszeile und geben den Positions-Tag mit der optionalen Option **-l** oder **-myLocation** an, wie im folgenden Beispiel gezeigt wird:

```
api.sh -u administrator -p collation discover start -n discovery1 -p myProfile -l myLocation myScope
```

Dabei ist **locationTag** der Wert des Positions-Tags, den Sie verwenden möchten. Der von Ihnen angegebene Wert überschreibt den Wert eines statischen Positions-Tags für Objekte, die während dieser bestimmten Erkennung erstellt wurden.

Anmerkung: Wenn das Positions-Tagging in der Datei `collation.properties` nicht aktiviert ist, wird durch die Angabe eines Positions-Tags während einer Erkennungsanfrage eine Ausnahmebedingung für die Erkennung verursacht.

Import eines IdML-Buchs

Um während des Imports eines IdML-Buchs einen Wert für das Positions-Tagging anzugeben, geben Sie den Positions-Tag mit der optionalen Option -l an, wie im folgenden Beispiel gezeigt wird:

```
loadidml.sh -f idml_book.xml -l locationTag
```

Dabei ist **locationTag** der Wert des Positions-Tags, den Sie verwenden möchten. Wenn Sie mehrere IdML-Bücher mit unterschiedlichen Positions-Tags importieren möchten, muss jedes Buch separat geladen werden.

Zugriffsliste

Sie können Zugriffslisteneinträge erstellen, denen ein Positions-Tag zugeordnet ist.

Das Attribut für den Positions-Tag ist verbindlich, kann aber später geändert werden. Die Berechtigungsnachweise werden nach Position gefiltert, weshalb nur die Zugriffseinträge für bestimmte Positionen verwendet werden. Dadurch wird das Risiko des Ausspionierens eines Kennworts von anderen Kunden oder aus anderen Positionen begrenzt. Wenn Sie die Erkennung ohne Positions-Tag ausführen, werden keine der gekennzeichneten Berechtigungsnachweise verwendet.

Wenn der Positions-Tag (locationTag) bei einem neuen Zugriffseintrag auf den Stern (*) gesetzt wird, wird dieser Tag bei der Erkennung während der Einrichtung einer Sitzung mit dem Endpunkt als letzter Zugriffseintrag ausprobiert.

Der Stern (*) ist der Standardwert. Er kann mit folgendem Parameter geändert werden:

```
com.ibm.cdb.locationTag.global=GLOBAL
```

In diesem Fall ist der Zugriffseintrag mit dem Tag GLOBAL der letzte Eintrag, der bei einer Erkennung ausprobiert wird. Der vorangegangene Positions-Tag wird nur für die Zugriffsliste verwendet und hat keine Auswirkung auf die Positions-Tags, die den während der Erkennung gefundenen CIs zugewiesen werden.

BIRT-Berichte

BIRT-Berichte (BIRT = Business Intelligence and Reporting Tools) können gefiltert werden, um die Daten zur Position bestimmter Kunden zu generieren.

Wenn das Positions-Tagging aktiviert ist, befindet sich das Textfeld im Fenster mit den BIRT-Berichten unter der Liste der Berichte. Sie können einen BIRT-Bericht für einen beliebigen Positions-Tag ausführen, damit die Daten angezeigt werden, die nur zu dieser Position gehören.

Standardberichte können Positions-Tags nicht verarbeiten. Wenn Sie die BIRT-Berichte verwenden müssen, müssen diese für die Unterstützung der Filterfunktion nach Positions-Tag manuell aktualisiert werden.

Wartung und Optimierung

Für eine optimale TADDM-Leistung können weitere Konfigurationsschritte und fortlaufende Wartungsaufgaben durchgeführt werden.

Optimierung der Parameter des Dienstprogramms zum Laden von Massendaten

Sie können das Verhalten des Massenladeprogramms anpassen, indem Sie bestimmte Parameter während der Ausführung angeben oder die Datei `bulkload.properties` konfigurieren.

Das Laden von Daten mithilfe des Massenladeprogramms erfolgt in drei unterschiedlichen Phasen:

1. Analyse der Objekte und Beziehungen zur Bestimmung der Diagramme in den Daten.

   In der Regel 1 - 5 % der Ausführungszeit
2. Erstellung von Modellobjekten und Diagrammen.

   In der Regel 2 - 5 % der Ausführungszeit
3. Übergeben der Daten an den API-Server.

   In der Regel 90 - 99 % der Ausführungszeit

Es stehen zwei Optionen zum Laden von Daten zur Verfügung:

- Daten können Dokument für Dokument geladen werden. Dies ist der Standardmodus. Für die folgenden Dateien müssen Sie Datensätze nacheinander laden:
  - Fehlerhafte Dateien.
  - Dateien mit erweiterten Attributen.
- Daten können als Massendaten geladen werden. Dies wird als Diagrammerstellung bezeichnet, da ein ganzes Diagramm geladen wird und nicht nur ein Datensatz.

  Das Laden von Massendaten mit der Diagrammerstellungsoption ist schneller als das sukzessive Laden von Datensätzen. (Details hierzu finden Sie in den Messwerten des Dienstprogramms zum Laden von Massendaten). Im Folgenden finden Sie ein Beispiel für die Diagrammerstellungsoption; dabei ist '-g=buffer' und Datenblöcke werden an den API-Server übergeben.

  ```
  ./loadidml.sh –g –f /home/configia/testfiles/sample.xml
  ```

  Mithilfe der folgenden Parameter in der Datei `bulkload.properties` kann die Leistung beim Laden von Massendaten verbessert werden:

  ```
  com.ibm.cdb.bulk.cachesize=2000
  ```

  Der Parameter `cachesize` steuert die Anzahl der Objekte, die beim Laden von Massendaten mit der Diagrammerstellungsoption in einer einzelnen Schreiboperation verarbeitet werden. Durch Erhöhen des Wertes für die Cachegröße wird die Leistung verbessert, jedoch erhöht sich dadurch auch die Gefahr, dass auf dem Client oder dem Server nicht ausreichend Speicherplatz vorhanden ist. Daher sollte die Anzahl nur geändert werden, wenn es explizite Hinweise gibt, dass die Verarbeitung einer Datei mit einem größeren Cache eine Leistungsverbesserung bewirkt. Der Standardwert für die Cachegröße ist 2000 und der Höchstwert für die Cachegröße ist 40000.

  ```
  com.ibm.cdb.bulk.allocpoolsize=1024
  ```

  Dieser Wert gibt den Maximalwert des Speichers an, der dem Prozess des Massenladeprogramms zugeordnet werden kann. Es handelt sich dabei um einen Xmx-Wert, der an die Java-Hauptklasse des Massenladeprogramms übergeben wird. Geben Sie den Wert in Megabytes an.

  Stellen Sie sicher, dass bei einer Java Virtual Machine keine Speicherknappheit auftritt. Hierfür müssen Sie Threadspeicherauszüge der TADDM-Prozesse erfassen und überprüfen. Erhöhen Sie gegebenenfalls die Speicherkapazität.

Tipp: Tests, die für das ITNMIP-Buch ausgeführt wurden, geben an, dass die Leistung optimal ist, wenn Sie die Prozesseigenschaften und -parameter des Dienstprogramms zum Laden von Massendaten auf die folgenden Werte setzen:

```
com.ibm.cdb.bulk.cachesize=4000
com.ibm.cdb.bulk.allocpoolsize=4096
value-Xms768M|-Xmx1512M|-DTaddm.xmx64=6g|
```

Es ist auch wichtig, dass Sie den Befehl **RUNSTATS** während des Prozesses zum Laden von Massendaten häufig ausführen.

Datenbankpflege

Um eine gleichbleibend optimale Systemleistung zu erreichen, müssen Sie für eine regelmäßige Wartung und Optimierung der TADDM-Datenbank sorgen.

Standardmäßige Datenbankkonfiguration

Die mit TADDM bereitgestellten Standarddatenbankkonfigurationen sind zur Prüfung der Erfolgschancen, zur Prüfung der Technologie sowie für kleine Pilotimplementierungen von TADDM ausreichend.

Optimierungsrichtlinien für DB2- und Oracle-Datenbanken

Die folgenden Optimierungsrichtlinien gelten für DB2- und Oracle-Datenbanken:

1. Begrenzen Sie die Anzahl der für die Datenbank verfügbaren physischen Plattenlaufwerke nicht ausschließlich auf der Grundlage der Speicherkapazität.
2. Die folgenden Komponenten sollten sich nach Möglichkeit auf verschiedenen Plattenlaufwerken oder -einheiten befinden:
   - Anwendungsdaten (wie Tabellen und Indizes)
   - Datenbankprotokolle
   - Temporärer Datenbankspeicherplatz: wird für Sortierungs- und Verknüpfungsvorgänge verwendet
3. Verwenden Sie für die Protokolldateien die schnellsten Platten.
4. Aktivieren Sie asynchrone Ein-/Ausgabe auf Betriebssystemebene.

Weitere Informationen zur Optimierung von DB2- und Oracle-Datenbanken finden Sie in *Database Performance Tuning on AIX* unter http://www.redbooks.ibm.com/redbooks/pdfs/sg245511.pdf.

Weitere Informationen zur Optimierung von DB2-Datenbanken finden Sie in *Relational Database Design and Performance Tuning for DB2 Database Servers* unter http://www-01.ibm.com/support/docview.wss?uid=tss1wp100764 sowie in den unter *DB2 UDB Version 8 Product Manuals* unter http://www.ibm.com/support/docview.wss?rs=71&uid=swg27009554 aufgeführten Dokumenten.

Alte Datenbanksätze löschen

Mit der Zeit wächst die Anzahl der Datensätze in den Tabellen. Daher müssen Sie möglicherweise je nach verfügbarem Speicherplatz Daten ab und zu manuell entfernen, damit die Tabellen eine gewisse Größe nicht übersteigen. Nach dem Löschen der Tabelle CHANGE_HISTORY_TABLE können Sie die entsprechenden Einträge aus der Tabelle CHANGE_CAUSE_TABLE entfernen. Durch das Löschen alter Datensätze aus der Tabelle ALIASES_JN können Sie außerdem die Leistung und den Bedienungskomfort des Tools zur Datenintegrität verbessern.

Datensätze aus CHANGE_HISTORY_TABLE und CHANGE_CAUSE_TABLE löschen

Sie können alte Datensätze entfernen, um die Leistung zu verbessern und eine kleinere Größe der Tabellen beizubehalten. Nach dem Entfernen der Datensätze aus der Tabelle CHANGE_HISTORY_TABLE können Sie die entsprechenden Einträge sicher aus der Tabelle CHANGE_CAUSE_TABLE entfernen.

Entfernen Sie mithilfe von SQL-Abfragen alte Daten manuell aus der Tabelle CHANGE_HISTORY_TABLE, um Speicherplatz in TADDM-Datenbanken freizugeben. Der folgende Befehl ist ein Beispiel für eine solche SQL-Abfrage, in der die Ganzzahl 1225515600000 für das Datum, 1. November 2008, steht, ausgedrückt in dem Format, das auch durch die Java-Methode 'System.currentTimeMillis() zurückgegeben wird, oder

durch eine Zahl, die der in Millisekunden gemessenen Differenz zwischen der aktuellen Uhrzeit und Mitternacht des 1. Januar 1970 UTC entspricht:

```
DELETE FROM CHANGE_HISTORY_TABLE
WHERE PERSIST_TIME < 1225515600000 (die Java-Zeitmarke)
```

Verwenden Sie den folgenden Code, um ein Datum in eine Java-Zeitmarke zu konvertieren:

```
import java.util.*;
import java.text.*;
import java.sql.Timestamp;

public class DateToString {

  public static void main(String args[]) {
  try {
     String str = args[0];
     SimpleDateFormat formatter = new SimpleDateFormat("dd/MM/yyyy");
     Date date = formatter.parse(str);

    long msec = date.getTime();

    System.out.println("Date is " +date);
    System.out.println("MillSeconds is " +msec);

    } catch (ParseException e)
    {System.out.println("Exception :"+e);    }

  }
}
```

Führen Sie den Code wie folgt aus:

```
java DateToString 1/11/2008
Date is Sat Nov 01 00:00:00 EST 2008
MillSeconds is 1225515600000
```

Verwenden Sie die daraus hervorgehende Java-Zeitmarke in der SQL-Abfrage.

Wenn die Datenbank CHANGE_HISTORY_TABLE eine außergewöhnliche Anzahl an Datensätzen enthält, müssen Sie möglicherweise inkrementelle Löschvorgänge ausführen (das Löschen von jeweils einer Untergruppe an Datensätzen), um das Anhäufen von Transaktionsprotokollen in der Datenbank zu vermeiden.

Nach dem Löschen der Tabelle CHANGE_HISTORY_TABLE können Sie die entsprechenden Einträge sicher aus der Tabelle CHANGE_CAUSE_TABLE entfernen. Bei der Tabelle CHANGE_CAUSE_TABLE handelt es sich um eine Linktabelle, mit der die Weitergabe geändert wird. Wenn Sie beispielsweise dem Betriebssystem eine neue Softwarekomponente hinzufügen, verknüpft die Tabelle diese Änderung mit dem Computersystem, auf dem das Betriebssystem ausgeführt wird. Sie können Datensätze in der Tabelle CHANGE_CAUSE_TABLE mit folgendem Befehl entfernen:

```
delete from change_cause_table where cause_id not in (select id from change_history_table)
```

Zeitrahmen für das Entfernen von Daten

Um das Datenbankwachstum im Laufe der Zeit zu begrenzen, können Sie die Größe der von TADDM gespeicherten Änderungsprotokolldaten verwalten. Wenn Sie den optimalen Zeitrahmen zum Entfernen von Daten aus der Änderungsprotokolltabelle bestimmen, sollten Sie sich überlegen, wofür Sie die Änderungsprotokolldaten verwenden und ob die Änderungsprotokolldaten von anderen Anwendungen verwendet werden.

Wenn die Änderungsprotokolldaten von einer anderen Anwendung verwendet werden, müssen Sie sicherstellen, dass Sie Anwendungssynchronisationen häufiger als die Wochenanzahl der Daten des Änderungsprotokolls ausführen, die in der Tabelle CHANGE_HISTORY_TABLE gepflegt werden.

Die folgenden Beispiele zeigen einige typische Szenarios:

- Wenn Sie die Daten des Änderungsprotokolls zur Fehlerbestimmung heranziehen und Probleme untersuchen möchten, die vor fünf Wochen auftraten, speichern Sie mindestens fünf Wochen alte Daten in der Tabelle CHANGE_HISTORY_TABLE.
- Wenn Sie Tivoli Business Service Manager (TBSM) wöchentlich synchronisieren, pflegen Sie Daten des Änderungsprotokolls, die mehr als eine Woche alt sind, in der TADDM-Tabelle des Änderungsprotokolls.

Beachten Sie unbedingt, dass sich die Zeit, die für eine vollständige Synchronisation benötigt wird, in Synchronisationsserverimplementierungen durch eine große Menge an Änderungsprotokolldaten verlängert.

Datenpflege in einer Synchronisationsserverimplementierung

In einer Domänenserverimplementierung können Sie Entscheidungen bezüglich der Datenpflege ausschließlich auf Basis des Datenbedarfs für die Domäne treffen. In einer Synchronisationsserverimplementierung müssen Sie jedoch die Entfernung der Daten des Änderungsprotokolls zwischen den einzelnen Domänenserverdatenbanken und der Synchronisationsserverdatenbank koordinieren. Außerdem müssen Sie die Daten aus all diesen Datenbanken entfernen.

Verwenden Sie in einer Synchronisationsserverimplementierung die folgenden Richtlinien für die Datenpflege:

- Bewahren Sie die Daten des Änderungsprotokolls auf Domänenebene für einen Zeitraum auf, der länger als die Zeitspanne zwischen den einzelnen terminierten Synchronisationen der Domänenserverdatenbanken mit der Synchronisationsserverdatenbank ist. Wenn die Synchronisation beispielsweise wöchentlich stattfindet, bewahren Sie mindestens zwei Wochen alte Änderungsprotokolldaten in jeder Domänenserverdatenbank auf.
- Entfernen Sie zuerst Daten aus einer Domänenserverdatenbank. Entfernen Sie anschließend Daten aus der Synchronisationsserverdatenbank.
- Es hat sich bewährt, in allen TADDM-Datenbanken die Änderungsprotokolldaten für dieselbe Anzahl an Wochen aufzubewahren. Der Zeitraum, für den die Änderungsprotokolldaten in der Synchronisationsserverdatenbank aufbewahrt werden, kann jedoch von dem Zeitraum abweichen, für den solche Daten in den Domänenserverdatenbanken gespeichert werden.
- Nachdem Sie einen Zeitrahmen für die Datenentfernung ermittelt haben, der die speziellen Anforderungen Ihrer Umgebung erfüllt, sollten Sie die Daten am besten direkt nach einer Synchronisation zwischen den Domänenserverdatenbanken und der Synchronisationsserverdatenbank entfernen.

Datensätze aus der Tabelle ALIASES_JN löschen

Wenn alte Datensätze aus der Tabelle ALIASES_JN gelöscht werden, kann dadurch die Leistung und Benutzerfreundlichkeit des Datenintegritätstools verbessert werden, außerdem wird zusätzlicher Speicherbereich in der Datenbank freigegeben.

Informationen zu diesem Vorgang

Die Tabelle ALIASES_JN enthält den Verlauf der Änderungen an der Tabelle ALIASES. Für das Tool zur Datenintegrität ist es erforderlich, dass die zusammengestellten Daten ein mögliches übergeordnetes Zusammenführen von Konfigurationselementen in der Datenbank finden. Im Laufe der Zeit wächst die Anzahl der Datensätze in der Tabelle ALIASES_JN zu einer beträchtlichen Größe an. Die Größe dieser Tabelle beeinträchtigt die Leistung und die Benutzerfreundlichkeit des Tools zur Datenintegrität und erhöht den Speicherbedarf in der TADDM-Datenbank.

Der Topologieagent `AliasesJnTableCleanup` nimmt die Bereinigung der Tabelle ALIASES_JN vor.

Standardmäßig werden alle Zeilen entfernt, die älter als 30 Tage sind. Durch Konfiguration der folgenden Eigenschaft in der Datei `collation.properties` können Sie das Alter ändern, ab dem die Datensätze gelöscht werden sollen:

```
com.ibm.cdb.topomgr.topobuilder.agents.AliasesJnTableCleanupAgent.removeOlderThanDays=30
```

Wenn Sie für die Eigenschaft den Wert '-1' festlegen, ist der Agent inaktiviert. Wenn Sie einen zu niedrigen Wert für das Alter angeben, kann es passieren, dass das Datenüberprüfungstool mit der Option für übergeordnetes Zusammenführen keine vollständigen Ergebnisse liefert.

Standardmäßig wird der Agent maximal 1800 Sekunden (30 Minuten) ausgeführt. Falls in diesem Zeitraum nicht alle veralteten Zeilen entfernt werden können, wird bei der nächsten Ausführung des Agenten versucht, die verbleibenden Zeilen zu löschen. Durch Konfiguration der folgenden Eigenschaft in der Datei `collation.properties` können Sie den Zeitlimitwert für den Agenten einstellen:

```
com.ibm.cdb.topomgr.topobuilder.agents.AliasesJnTableCleanupAgent.timeout=1800
```

DB2-Datenbankpflege
Um eine annehmbare Leistung zu gewährleisten, muss die TADDM-DB2-Datenbank regelmäßig gepflegt werden.

Informationen zu diesem Vorgang

Folgende DB2-Dienstprogramme sind verfügbar:

REORG
Nach zahlreichen Änderungen der Tabellendaten durch Insertion, Löschen und Aktualisieren von Spalten variabler Länge können sich logisch sequenzielle Daten auf Seiten nichtsequenzieller physischer Daten befinden. Deshalb muss der Datenbankmanager für den Datenzugriff zusätzliche Leseoperationen durchführen. Reorganisieren Sie die DB2-Tabellen mit dem Dienstprogramm **REORG**, um die Fragmentierung aufzuheben und Speicherbereich freizugeben. Verwenden Sie bei Bedarf das Dienstprogramm **REORG**, wenn **RUNSTATS** mehr Zeit erfordert als üblich oder wenn der DB2-Befehl **REORGCHK** die Notwendigkeit anzeigt. Fahren Sie den TADDM-Server herunter, wenn Sie das Dienstprogramm **REORG** ausführen, da Anwendungen bei einer Offline-Tabellen- oder Indexreorganisation (Daten-Defragmentierung) zwar auf die Daten in Tabellen zugreifen, aber keine Aktualisierungen durchführen können. Da der TopologyBuilder häufig ausgeführt wird, auch ohne Erkennung, verursachen solche Sperren möglicherweise unvorhersehbare Ergebnisse innerhalb der Anwendung.

RUNSTATS (manuelle Statistikerfassung)
Das DB2-Optimierungsprogramm verwendet Informationen und statistische Daten im DB2-Katalog, um auf Grundlage der bereitgestellten Abfrage den optimalen Zugriff auf die Datenbank zu ermitteln. Statistische Informationen werden für bestimmte Tabellen und Indizes in der lokalen Datenbank erfasst, wenn Sie das Dienstprogramm **RUNSTATS** ausführen. Wenn eine beträchtliche Anzahl an Tabellenzeilen hinzugefügt oder entfernt wird oder Daten in Spalten, für die Sie statistische Daten sammeln, aktualisiert werden, muss der Befehl **RUNSTATS** für die Aktualisierung der statistischen Daten verwendet werden. Um eine optimale Leistung zu erzielen, führen Sie die **RUNSTATS**-Task wöchentlich oder bei hoher Datenbankaktivität täglich aus. Wenn die statistischen Daten nicht in ausreichendem Umfang aktualisiert werden, kann sich die Leistung innerhalb von TADDM erheblich verschlechtern. Das Dienstprogramm **RUNSTATS** kann ausgeführt werden, solange der TADDM-Server aktiv ist. TADDM erfordert ein bestimmtes **RUNSTATS**-Format, das an späterer Stelle beschrieben wird. Außerdem muss die DB2-Option **AUTO_RUNSTATS** inaktiviert sein.

AUTO_RUNSTATS (automatische Statistikerfassung)
Sie können die automatische Statistikerfassung (auch bekannt als **`auto-runstats`**) aktivieren, damit DB2 entscheiden kann, ob die TADDM-Datenbankstatistik aktualisiert werden muss. Das Dienstprogramm **RUNSTATS** wird im Hintergrund ausgeführt und die Datenbankstatistik ist stets aktuell.

Um die automatische Statistikerfassung zu aktivieren, müssen Sie die Parameter `AUTO_MAINT`, `AUTO_TBL_MAINT` und `AUTO_RUNSTATS` auf `ON` setzen. Führen Sie folgenden Befehl aus:

```
CONNECT TO <DB-Alias>
UPDATE DB CONFIG USING AUTO_MAINT ON AUTO_TBL_MAINT ON AUTO_RUNSTATS ON
```

Dabei steht *<DB-Alias>* für den Namen Ihrer Datenbank.

Einschränkung: Sie können dieses Dienstprogramm nur verwenden, wenn Sie den DB2-APAR IT05733 installiert haben und der Parameter `DB2_SELECTIVITY=DSCC` festgelegt ist. Der DB2-APAR IT05733 ist in den folgenden und höheren Releases von DB2 enthalten:

- 9.7 Fixpack 11
- 10.1 Fixpack 6

- 10.5 Fixpack 7

Wenn Sie den Parameter DB2_SELECTIVITY=DSCC in DB2 Version 10.*x* festlegen möchten, führen Sie folgenden Befehl aus:

```
db2set -immediate DB2_SELECTIVITY=DSCC
```

Anmerkung: In DB2 9.7 wird der Parameter `-immediate` nicht unterstützt. Wenn Sie den Parameter DB2_SELECTIVITY=DSCC in dieser Version festlegen möchten, führen Sie den Befehl **db2set DB2_SELECTIVITY=DSCC** aus und starten Sie DB2 erneut.

Anmerkung: Wenn der TADDM-Benutzer ein Upgrade der Version von DB2 in einer TADDM-Installation durchführt, muss die kompatible Version des Treibers ebenfalls aktualisiert werden. Sie können Ihren DBA bitten, Ihnen die Datei `db2jcc.jar` vom TADDM-DB2-Server zur Verfügung zu stellen, oder die für Ihre jeweilige DB2-Version passende Datei unter folgendem Link herunterladen:http://www-01.ibm.com/support/docview.wss?uid=swg21363866. Sobald Sie die Datei erhalten haben, stoppen Sie TADDM, kopieren Sie die Datei in das Verzeichnis `dist/lib/jdbc/`, vergewissern Sie sich, dass die Berechtigungseinstellungen korrekt sind, so dass der TADDM-Benutzer die Datei lesen kann, und starten Sie TADDM erneut. Wiederholen Sie diesen Schritt auf allen TADDM-Servern in Ihrer Umgebung.

DB2 HEALTH MONITOR (Statusüberwachung)

Es ist üblich, die DB2-Statusüberwachung gegen die TADDM-Datenbank auszuführen, um proaktiv zu überwachen, ob sich Bedingungen verändert haben, sodass **RUNSTATS** oder **REORG** oder eine andere Optimierung erforderlich ist. Die Statusüberwachung kann einen Datenbankadministrator über mögliche Probleme beim Systemzustand benachrichtigen. Die Statusüberwachung erkennt proaktiv Probleme, die möglicherweise zu Hardwarefehlern oder zu inakzeptabler Systemleistung oder -funktion führen können. Mithilfe der proaktiven Statusüberwachung können Sie sich mit einem Problem befassen, bevor es zu einem Fehler führt, der die Systemleistung beeinträchtigt.

DB2 PERFORMANCE ANALYSIS SUITE (Leistungsanalyse-Suite)

Wenn ein DB2-Problem vermutet wird, kann das Performance Analyst-Tool sehr rasch einen DB2-Snapshot analysieren, der während des Problems erstellt wird, und Aktionen vorschlagen. Sie können dieses Tool unter https://www.ibm.com/developerworks/community/groups/community/perfanalyst herunterladen.

Gehen Sie folgendermaßen vor, um einen DB2-Snapshot für TADDM zu erstellen:

1. Verbinden Sie Ihre TADDM-Datenbank vom DB2-Server aus und führen Sie folgenden Befehl aus:

   ```
   db2 -tf updmon.sql
   ```

   wobei die Datei `updmon.sql` die folgenden Einträge enthält:

   ```
   UPDATE MONITOR SWITCHES USING BUFFERPOOL ON ;
   UPDATE MONITOR SWITCHES USING LOCK       ON ;
   UPDATE MONITOR SWITCHES USING SORT       ON ;
   UPDATE MONITOR SWITCHES USING STATEMENT  ON ;
   UPDATE MONITOR SWITCHES USING TABLE      ON ;
   UPDATE MONITOR SWITCHES USING UOW        ON ;
   UPDATE MONITOR SWITCHES USING TIMESTAMP  ON ;
   RESET MONITOR ALL
   ```

2. Nachdem Schritt 1 abgeschlossen ist, führen Sie den Befehl "DB2 get monitor switches" aus, um zu überprüfen, ob alle Schalter eingestellt sind. Sie müssen alle den Status ON (EIN) aufweisen.
3. Führen Sie den Prozess aus, bei dem Sie Leistungsprobleme haben.
4. Führen Sie, während der verlangsamte Prozess ausgeführt wird, von DB2 aus den folgenden Befehl in geeigneten Intervallen aus:

   ```
   db2 get snapshot for all on <DB-Name> > <DB-Name>-dbsnap.out
   ```

   Führen Sie diesen Befehl in dem Fenster aus, das Sie für die Ausführung des Befehls in Schritt 1 verwendet haben. Dieser Befehl kann nicht mithilfe eines Scripts ausgeführt werden.

5. Erstellen Sie die Snapshots jedes Mal unter Verwendung einer anderen Ausgabedatei mit Zeitmarke. Verwenden Sie solche Intervalle, dass drei oder vier Snapshots während des Prozesses erstellt werden, aber die Zeit zwischen den Ausführungen 1 Stunde nicht überschreitet.

Sobald ein Snapshot erfasst wurde, analysieren Sie ihn mit dem Performance Analyst-Tool; beginnen Sie dabei immer mit dem letzten Snapshot. Beispiel: Hohe CPU- und hohe durchschnittliche Ausführungszeit auf der Registerkarte 'Bericht' bei einer Abfrage, die häufig ausgeführt wird, zeigt im Allgemeinen ein Optimierungsproblem an, das mithilfe des Dienstprogramms **RUNSTATS** gelöst werden kann. Ein hoher Überlaufprozentsatz auf der Registerkarte 'Tabellen' kann anzeigen, dass eine Ausführung des Dienstprogramms **REORG** notwendig ist. Überprüfen Sie die Registerkarte 'Pufferpool', um sicherzustellen, dass hier keine Alerts vorhanden sind. Ein zu kleiner Pufferpool führt möglicherweise zu schwacher Leistung.

Vorbereitende Schritte

Nach größeren Wartungsmaßnahmen, die eine Schemaänderung bedingen, z. B. nach Anwendung eines Fixpacks, muss die Datei `TADDM_table_statistics.sql` auf dem TADDM-Speicherserver generiert werden. Diese Datei wird für die regelmäßig auszuführenden **RUNSTATS**-Tasks zur Datenbankpflege benötigt. TADDM erfordert aufgrund einer DB2-Einschränkung bei der Behandlung von Spalten mit langen gemeinsamen Präfixen wie etwa Klassennamen, die innerhalb von TADDM sehr häufig verwendet werden, ein spezielles Format zur Aktualisierung von Datenbankstatistiken . Verwenden Sie deshalb nicht die DB2-Option **AUTO_RUNSTATS**, sondern die **RUNSTATS**-Syntax, die Sie mithilfe der folgenden Schritte erstellen können. Falls Sie jedoch den DB2-APAR IT05733 installiert haben und der Parameter `DB2_SELECTIVITY=DSCC` festgelegt ist, können Sie die Option **AUTO_RUNSTATS** verwenden.

Anmerkung: Die folgenden Anweisungen beziehen sich auf die Betriebssysteme Linux und UNIX. Verwenden Sie für die Datenbankpflege auf dem Betriebssystem Windows anstelle des Scripts `.sh` das entsprechende Script `.bat`.

Gehen Sie zum Generieren der Datei `TADDM_table_stats.sql` wie folgt vor:

1. Führen Sie folgenden Befehl aus:

```
cd $COLLATION_HOME/bin
```

2. Führen Sie den folgenden Befehl aus, wobei *tmpdir* für ein Verzeichnis steht, in dem diese Datei erstellt werden kann:

```
./gen_db_stats.jy > temporäres_Verzeichnis/TADDM_table_stats.sql
```

   In einer Streaming-Server-Implementierung wird dieser Befehl auf dem primären Speicherserver ausgeführt.

3. Kopieren Sie die Datei auf den Datenbankserver oder stellen Sie sie Ihrem Datenbankadministrator (DBA) zur Verfügung, der einen Lauf gegen die TADDM-Datenbank ausführt, wie in Schritt 2 in der Prozedur gezeigt. Aktualisieren Sie die Datenbankstatistik mindestens wöchentlich oder, falls größere Änderungen an Tabellen durchgeführt werden, auch öfter.

Vorgehensweise

Gehen Sie wie folgt vor, um Wartungsmaßnahmen an einer DB2-Datenbank vorzunehmen:

1. Gehen Sie wie folgt vor, um das Dienstprogramm **REORG** zu verwenden:
   a) Stellen Sie auf dem Datenbankserver die folgende SQL-Abfrage, mit der die **REORG TABLE**-Befehle REORG TABLE generiert werden, in eine Datei:

```
select  'reorg table '||CAST(RTRIM(creator) AS VARCHAR(40))||'.
 "'||substr(name,1,60)||'" ; ' from sysibm.systables where creator
 = 'dbuser' and type = 'T' and name not in ('CHANGE_SEQ_ID')
 order by 1;
```

   dbuser ist dabei der Wert aus `com.collation.db.user=`.

Anmerkung: Achten Sie darauf, dass die Groß-/Kleinschreibung von *DB-Benutzer* der Schreibweise des Werts in der Spalte `creator` der Datenbanktabelle `sysibm.systables` entspricht.

b) Stoppen Sie den TADDM-Server.

c) Stellen Sie in einer DB2-Befehlszeile eine Verbindung zur Datenbank her und führen Sie die folgenden Befehle aus:

```
db2 –x –tf temp.sql > cmdbreorg.sql
db2 –tvf cmdbreorg.sql > cmdbreorg.out
```

d) Stellen Sie sicher, dass das Dienstprogramm **REORG** erfolgreich war, indem Sie die Datei `cmdbreorg.out` auf Fehler überprüfen.

e) Starten Sie den TADDM-Server.

2. Gehen Sie wie folgt vor, um das Dienstprogramm **RUNSTATS** zu verwenden: Automatisieren Sie den Prozess, sodass er mindestens einmal wöchentlich ausgeführt wird.

a) Führen Sie auf dem Datenbankserver den TADDM-spezifischen Befehl **RUNSTATS** aus, indem Sie die Ausgabe verwenden, die Sie zuvor erstellt haben:

```
db2 -tvf temporäres_Verzeichnis/TADDM_table_stats.sql  > table_stats.out
```

b) Stellen Sie sicher, dass das Dienstprogramm **RUNSTATS** erfolgreich war, indem Sie die Datei `table_stats.out` auf Fehler überprüfen.

Oracle-Datenbankpflege

Diese Pflege- und Optimierungsrichtlinien gelten für Oracle-Datenbanken.

1. Führen Sie für die Datenbanktabellen das Paket 'dbms_stats' aus. Oracle verwendet ein kostenbasiertes Optimierungsprogramm. Für dieses Optimierungsprogramm sind Daten für die Erstellung des Zugriffsplans erforderlich; diese Daten werden vom Paket 'dbms_stats' generiert. Oracle-Datenbanken sind auf Daten zu den Tabellen und Indizes angewiesen. Ohne diese Daten muss das Optimierungsprogramm auf Schätzungen zurückgreifen.

   Für eine optimale Leistung der Oracle-Datenbanken ist die erneute Indexerstellung und die Ausführung des Pakets 'dbms_stats' unbedingt erforderlich. Nach dem Füllen der Datenbank sollte dieser Vorgang in geplanten regelmäßigen Abständen, z. B. wöchentlich, durchgeführt werden.

   - **`REBUILD INDEX`**: Nach zahlreichen Änderungen der Tabellendaten durch Einfüge-, Lösch- und Aktualisierungsvorgänge können sich logisch sequenzielle Daten auf Seiten nichtsequenzieller physischer Daten befinden, sodass der Datenbankmanager zusätzliche Leseoperationen durchführen muss, um auf Daten zugreifen zu können. Erstellen Sie die Indizes neu, um die SQL-Leistung zu verbessern.

     a. Generieren Sie die **`REBUILD INDEX`**-Befehle, indem Sie für die Oracle-Datenbank die folgende SQL-Anweisung ausführen; dabei wird für *dbuser* der Wert aus `com.collation.db.user=` übernommen:

     ```
     select 'alter index dbuser.'||index_name||' rebuild tablespace '
     ||tablespace_name||';' from dba_indexes where owner = 'dbuser'
     and index_type not in ('LOB');
     ```

     Dadurch werden alle **`ALTER INDEX`**-Befehle generiert, die Sie ausführen müssen.

     b. Führen Sie die Befehle in SQLPLUS oder einem vergleichbaren Tool aus. Die erneute Erstellung der Indizes auf einer großen Datenbank dauert 15 - 20 Minuten.

2. **`DBMS_STATS`**: Verwenden Sie das Oracle-Managementsystem für relationale Datenbanken zur Erfassung vieler verschiedener Statistikarten, um dadurch zu einer Leistungsverbesserung beizutragen. Das Optimierungsprogramm verwendet Informationen und statistische Daten im Wörterbuch, um auf Grundlage der bereitgestellten Abfrage den optimalen Zugriff auf die Datenbank zu ermitteln. Statistische Informationen werden für bestimmte Tabellen und Indizes in der lokalen Datenbank erfasst, wenn Sie den Befehl **`DBMS_STATS`** ausführen. Wenn eine beträchtliche Anzahl an Tabellenzeilen hinzugefügt oder entfernt wird oder Daten in Spalten, für die Sie statistische Daten sammeln, aktualisiert werden, führen Sie den Befehl **`DBMS_STATS`** erneut aus, um die statistischen Daten zu aktualisieren.

- Das Programm gen_db_stats.jy im Verzeichnis $COLLATION_HOME/bin gibt für eine Oracle- oder DB2-Datenbank die Datenbankbefehle aus, mit denen die Statistikdaten für die TADDM-Tabellen aktualisiert werden. Das folgende Beispiel zeigt, wie das Programm verwendet wird:

  a. cd $COLLATION_HOME/bin

  b. Führen Sie diese SQL-Anweisung aus; dabei steht *tmpdir* für ein Verzeichnis, in dem diese Datei erstellt wird:

  ```
  ./gen_db_stats.jy > tmpdir/TADDM_table_stats.sql
  ```

  In einer Streaming-Server-Implementierung wird diese Anweisung auf dem primären Speicherserver ausgeführt.

  c. Kopieren Sie nach Abschluss dieses Vorgangs die Datei auf den Datenbankserver und führen Sie folgenden Befehl aus:

  – Geben Sie @ und dann den Dateinamen ein, um eine Scriptdatei in SQLPlus auszuführen: SQL > @{Datei}

  d. Führen Sie die Befehle in SQLPLUS oder einem vergleichbaren Tool aus.

3. Pufferpool: Ein Pufferpool oder Puffercachespeicher ist eine Speicherstruktur für jede Instanz innerhalb des Oracle-SGA (SGA = System Global Area). Dieser Puffercachespeicher wird zur Zwischenspeicherung von Datenblöcken im Speicher verwendet. Auf Daten im Speicher kann erheblich schneller zugegriffen werden als auf die Daten der Platte. Das Ziel der Blockpufferoptimierung ist es, häufig verwendete Datenblöcke effizient im Puffercachespeicher (SGA) zwischenzuspeichern und schnelleren Zugriff auf Daten zu ermöglichen. Die Blockpufferoptimierung ist eine wichtige Task jedes Oracle-Optimierungsvorhabens und ist Teil der laufenden Optimierung und Überwachung von Produktionsdatenbanken. Oracle verwaltet für jede Instanz einen eigenen Puffercachespeicher im SGA. Ein Puffercachespeicher ausreichender Größe kann normalerweise eine Cachetrefferquote von über 90 Prozent erzielen; das bedeutet, neun von zehn Anfragen werden erfüllt, ohne dass die Platte verwendet werden muss. Die Cachetrefferquote eines zu kleinen Puffercachespeichers ist gering, was eine größere Anzahl an Ein-/Ausgaben der physischen Platte zur Folge hat. Bei einem zu großen Puffercachespeicher sind Teile des Puffercachespeichers nicht ausgelastet und Speicherressourcen werden verschwendet.

Tabelle 37. Richtlinien zur Pufferpoolgröße (db_cache_size)

Anzahl der KE	**Richtlinien zur Pufferpoolgröße**
< 500.000	38000
500.000 - 1.000.000	60000
> 1.000.000	95000

4. Sie können den Maximalwert der geöffneten Cursor verdoppeln, wenn die vollständige Ausführung der Erkennung oder des Ladens von Massendaten zu lange dauert und NRS den folgenden Fehler enthält:

```
com.ibm.tivoli.namereconciliation.service.NrsService
getAliases(masterGuid)
SEVERE: NOTE ^*** SQL State = 60000.   SQL Code  = 604.   SQL Message =
ORA-00604: error occurred at recursive SQL level 1
ORA-01000: maximum open cursors exceeded
ORA-01000: maximum open cursors exceeded
```

5. Überprüfen Sie, ob die Versionen Ihres Oracle-JDBC-Treibers und des Oracle-Servers identisch sind. Ersetzen Sie den Oracle-JDBC-Treiber gegebenenfalls an den folgenden Orten.

   Anmerkung: Dies ist nur erforderlich, wenn der BIRT Report Viewer aktiviert ist.

   - TADDM 7.3.0 - $COLLATION_HOME/deploy-tomcat/birt-viewer/WEB-INF/platform/plugins/org.eclipse.birt.report.data.oda.jdbc_2.2.1.r22x_v20070919/drivers/
   - TADDM 7.3.0.1 und höher - $COLLATION_HOME/apps/birt-viewer/WEB-INF/platform/plugins/org.eclipse.birt.report.data.oda.jdbc_2.2.1.r22x_v20070919/drivers/
   - $COLLATION_HOME/lib/jdbc/

Datenbankkommunikation
Wenn die Datenbank nicht verfügbar ist, versucht der Speicherserver die Verbindung herzustellen.

Besteht zwischen Speicherserver und Datenbank keine Verbindung, so wartet der Speicherserver die in `com.ibm.cdb.db.timeout` angegebene Zeit und versucht dann, eine Verbindung herzustellen. Die Anzahl der Wiederholungsversuche bei der Herstellung der Datenbankverbindung wird durch `com.ibm.cdb.db.max.retries` festgelegt.

Weitere Informationen zu den Datenbankeigenschaften finden Sie im Abschnitt Datenbankeigenschaften.

Optimierung der Erkennungsleistung

Sie können die Eigenschaften `com.collation.discover.dwcount`, `com.collation.discover.observer.topopumpcount` und `com.ibm.cdb.discover.observer.topopump.threshold` in der Datei `collation.properties` aktualisieren, um Einfluss auf die Erkennungsrate und die Rate zu nehmen, mit der die Ergebnisse der Erkennung in der TADDM-Datenbank gespeichert werden. Außerdem können Sie damit auch die Anzahl der Threads begrenzen, die für das Speichern von Daten verantwortlich sind.

Einzelheiten zu diesen Eigenschaften finden Sie im Abschnitt „Leistungseigenschaften” auf Seite 86.

Wenn Sie die Werte der Eigenschaften `com.collation.discover.dwcount` oder `com.collation.discover.observer.topopumpcount` erhöhen, müssen Sie möglicherweise auch die Größe des installierten Speichers erhöhen, indem Sie die Einstellung für die maximale Größe des Heapspeichers für die folgenden Java Virtual Machines (JVMs) erhöhen:

Für die Eigenschaft `dwcount`:

- In einer Streaming-Serverimplementierung:
  - Discover
  - DiscoverService
- In einer Domänenserverimplementierung:
  - Discover

Für die Eigenschaft `topopumpcount`:

- In einer Streaming-Serverimplementierung:
  - StorageService
- In einer Domänenserverimplementierung:
  - Topology

Weitere Informationen finden Sie im Abschnitt „Java Virtual Machine (JVM): Optimierung von IBM Parametern” auf Seite 138.

Weitere Informationen zur Optimierung der Erkennungsleistung finden Sie im Dokument *Tuning Discovery Performance* unter http://www.ibm.com/software/brandcatalog/ismlibrary/.

Optimierung der Erkennungsrate
Das Attribut für die Erkennungsrate bietet die besten Möglichkeiten für eine Optimierung. Die Anzahl der Erkennungs-Worker-Threads ist die Eigenschaft mit der größten Auswirkung auf die Leistung. Sie können mit den Sensoren, die sich in Bearbeitung befinden, auch die Leistung überwachen oder diese durch die Angabe der Größe des Sitzungspools verbessern.

Bei einem Erkennungs-Worker-Thread handelt es sich um einen Thread, der Sensoren ausführt. Die folgende Eigenschaft gibt die maximale Anzahl der Erkennungs-Worker-Threads an:

```
com.collation.discover.dwcount=32
```

Falls der Server über genügend freie Kapazitäten verfügt, können Sie diese Anzahl erhöhen und somit weitere Sensoren parallel ausführen.

Sensoren, die sich in Bearbeitung befinden

Zur Überwachung der Leistung können Sie die Sensoren betrachten, die sich in Bearbeitung befinden. Ein Sensor in Bearbeitung kann sich in einer der drei Ausführungsphasen befinden:

started (gestartet)
Ein Sensor in dieser Phase ermittelt ein oder mehrere Konfigurationselemente.

discovered (ermittelt)
Ein Sensor in dieser Phase hat die Ermittlung eines oder mehrerer Konfigurationselemente abgeschlossen, wartet aber auf die Ergebnisse, um sie im Datenspeicher zu speichern.

storing (speichern)
Ein Sensor in dieser Phase speichert die Ergebnisse der Ermittlung in der Datenbank.

Um die Sensoren, die sich in Bearbeitung befinden, nach Ausführungsphase zu sortieren, klicken Sie auf die Spalte 'Description' (Beschreibung).

Durch die Beobachtung eines Erkennungsverlaufs und den Vergleich der Anzahl der Sensoren, die sich im Startstadium der Bearbeitung befinden, mit denen, die sich im Erkennungs- und im Speicherstadium in Bearbeitung befinden, können Sie bewerten, ob die Attributerkennung in einer bestimmten Umgebung schneller oder langsamer ist als die Attributspeicherung. Wie nach allen Änderungen der Datei `collation.properties` müssen Sie den Server erneut starten, damit die Änderungen wirksam werden.

Beispiele:

Sensoren in Bearbeitung: STARTED, DISCOVERED, STORING.

Wenn die Anzahl von (DISCOVERED + STORING) kleiner ist als die von STARTED, kann dies darauf hinweisen, dass es sich bei der Erkennung um den Leistungsengpass handelt.

Wenn die Anzahl von (DISCOVERED + STORING) die Anzahl von STARTED übersteigt, kann dies darauf hinweisen, dass es sich beim Speichern um den Leistungsengpass handelt.

Größe von Sitzungs- und Gateway-Pools

Um die Attribute eines bestimmten Konfigurationselements zu ermitteln, muss eine SSH- oder WMI-Sitzung zwischen dem Sensor und dem zugehörigen Hostcomputer aufgebaut sein. Zur Verbesserung der Leistung werden diese Sitzungen zusammengefasst und zwischengespeichert. Die standardmäßige Poolgröße ist in den meisten Fällen ausreichend. Ist dies nicht der Fall, kann die Erkennungsrate dadurch eingeschränkt werden. Zur Überwachung dieser Bedingung können Sie die folgende Eigenschaft auf `true` setzen:

```
com.collation.platform.session.ExtraDebugging=false
```

Sie müssen den Erkennungsserver erneut starten, damit diese Änderung wirksam wird. Nach der Ausführung der Erkennung können Sie in den DiscoverManager-Protokollen nach Problemen bei der Bereitschaftszeit suchen, die sich auf die Sitzungspools beziehen. Durchsuchen Sie dazu die Protokolle für `pool lock`. Im Anschluss finden Sie ein Beispiel für Leistungseinbußen, die durch einen Sitzungspoolkonflikt verursacht wurden:

```
2006-08-04 16:11:50,733 DiscoverManager [DiscoverWorker-34]
WindowsComputerSystemAgent(192.168.16.181)
INFO session.SessionClientPool -
Session client [3x ssh2:/admlxz@151.179.84.85]#9612508
waited 158.682 seconds for pool lock
```

Sie können die Poolgröße erhöhen, wenn die Bereitschaftszeit für eine Sitzung zu lang ist. Dazu gibt es zwei Möglichkeiten. Sie können die Poolgröße global für Sitzungen pro Host ändern, indem Sie folgenden Eigenschaft in der Datei `collation.properties` bearbeiten:

```
com.collation.platform.session.PoolSize=3
```

Es ist jedoch unwahrscheinlich, dass der Konflikt die Sitzungen der meisten oder aller Hosts in der Umgebung betrifft. Der Konflikt ist vermutlich auf eine kleine Anzahl größerer Host beschränkt, die von vielen Sensoren verwendet werden. Der Erkennungsserver verwendet eine bereichsorientierte Eigenschaft, d. h.

viele der Eigenschaften in der Datei `collation.properties` verwenden einen Wert für allgemeine Ziele und einen anderen Wert für bestimmte Ziele. Sie können diese Eigenschaft anpassen, indem Sie eine IP-Adresse oder einen Namen für den Erkennungsserverbereich hinzufügen, wie dies im folgenden Beispiel gezeigt wird:

```
com.collation.platform.session.PoolSize.10.10.250.1=20
```

In diesem Fall hat die Poolgröße für 10.10.250.1 den Wert 20, aber für alle anderen Hosts den Wert 3. Sie können die Protokollnachrichten beispielsweise in den DiscoverManager-Protokollen anzeigen und ermitteln, für welche Hosts die standardmäßige Poolgröße für die Sitzung nicht ausreicht und anschließend die erforderlichen Änderungen in der Datei `collation.properties` vornehmen.

Die Poolgröße des Gateways ist eine zugehörige Einstellung. Dadurch wird die Anzahl der Sitzungen festgelegt, die zwischen dem Erkennungsserver und dem Windows-Gateway zulässig sind. Sie können die Anzahl durch die Bearbeitung der folgenden Eigenschaft angeben:

```
com.collation.platform.session.GatewayPoolSize=10
```

Falls sich Ihre Umgebung hauptsächlich aus Windows-Computersystemen zusammensetzt, passen Sie diese Eigenschaft nach oben an, damit sie der Anzahl der Erkennungs-Worker-Threads entspricht.

Optimierung des Speichers
Der Speicher ist der zweite wichtige Bereich, der optimiert werden kann. Wenn die Anzahl der Sensoren im Speicherstadium ungefähr dem Wert der Eigenschaft entspricht, mit der die Anzahl der parallelen Speicher-Threads angegeben wird, führt die Speicherung der Erkennungsergebnisse zu dem Leistungsengpass. Um die Leistung zu verbessern, können Sie auch die Anzahl der Threads begrenzen, die für das Speichern der Daten verantwortlich sind.

Die folgende Eigenschaft gibt die Anzahl der parallelen Speicher-Threads an. Sie ist eine der Haupteinstellungen zur Steuerung der Erkennungsspeicherleistung:

```
com.collation.discover.observer.topopumpcount
```

Um die Speicherleistung bei der Ausführung von Topologieagenten zu verbessern, können Sie die Anzahl der Threads begrenzen, die für das Speichern der Daten während einer Erkennung verantwortlich sind. In diesem Fall nimmt eine Erkennung weniger Zeit in Anspruch. Um den Grenzwert für die ausgeführten Threads anzugeben, bearbeiten Sie die folgenden Eigenschaften in der Datei `collation.properties`:

com.ibm.cdb.discover.observer.topopump.threshold
Diese Eigenschaft gibt die Anzahl der zu begrenzenden Speicher-Threads an.

com.ibm.cdb.discover.observer.topopump.threshold.<Agentengruppenname>
Diese Eigenschaft gibt die Anzahl der zu begrenzenden Speicher-Threads an, wenn die angegebene Agentengruppe ausgeführt wird.

In der folgenden Tabelle ist dargestellt, in welchem Ausmaß die Eigenschaft `com.ibm.cdb.discover.observer.topopump.threshold` die Erkennungsleistung verbessern kann. Die Berechnungen betreffen eine Datenbank mit 76.000 Konfigurationselementen.

Wert der Schellenwerteigenschaft	Prozentsatz der zeitlichen Verbesserung
0,2	55
0,5	33
0,7	13
1	0

Java Virtual Machine (JVM): Optimierung von IBM Parametern

Sie können JVM-Parameter (Java Virtual Machine) setzen, mit denen die Fragmentierung des Java-Heapspeichers verringert und die Leistung erhöht werden kann.

Mit steigender Anzahl der verarbeiteten Objekte kann eine Fragmentierung des JavaHeapspeichers auftreten. Durch die Festlegung einer Reihe von Parametern können Sie zu einer Verringerung der Fragmentierung im Heapspeicher beitragen.

- Ein kCluster ist ein Speicherbereich, der ausschließlich für Klassenblocks verwendet wird. Er kann bis zu 1280 Einträge aufnehmen. Jeder Klassenblock ist 256 Bytes lang. Dieser Standardwert ist in der Regel zu klein und kann zu einer Fragmentierung des Heapspeichers führen. Legen Sie den kCluster-Parameter **-Xk** wie folgt fest, um die Fragmentierung des Heapspeichers zu reduzieren. Hierbei handelt es sich um Anfangswerte, die gegebenenfalls in Ihrer Umgebung optimiert werden müssen. Durch eine Analyse des Heapspeicherauszugs lässt sich die optimale Größe am besten bestimmen.
  - Topology: -Xk8300
  - EventsCore: -Xk3500
  - DiscoverAdmin: -Xk3200
  - Proxy: -Xk5700
  - Discover: -Xk3700

  Implementieren Sie diese Änderungen in die Datei `collation.properties`, indem Sie Einträge im Abschnitt 'JVM Vendor Specific Settings' (Anbieterspezifische JVM-Einstellungen) hinzufügen. Fügen Sie beispielsweise folgende Zeile hinzu, um diese Änderungen für den Topologieserver zu implementieren:

  ```
  com.collation.Topology.jvmargs.ibm=-Xk8300
  ```

- Bei Fragmentierungsproblemen besteht zudem die Möglichkeit, einen gewissen Speicherplatz speziell für große Objekte zu reservieren; > 64K. Verwenden Sie den Parameter **`-Xloratio`**. Beispiele dafür sind:
  - **`-Xloratio0.2`**

  Durch diesen Befehl werden x Prozent des aktiven Java-Heapspeichers (nicht x Prozent von -Xmx, sondern x Prozent der derzeitigen Größe des Java-Heapspeichers) ausschließlich für große Objekte (≥64 KB) reserviert. Bei einer Änderung sollte auch -Xmx geändert werden, um sicherzustellen, dass die Größe des Bereichs für kleine Objekte nicht verringert wird. Durch eine Analyse des Heapspeicherauszugs lässt sich die optimale Einstellung für diesen Parameter am besten ermitteln.

Eine Reihe weiterer Parameter, die die Java-Leistung beeinflussen, kann festgelegt werden. Bearbeiten Sie eine der folgenden Dateien, um für eine bereits vorhandene JVM-Option einen anderen Wert zu wählen:

- Für einen Domänenserver in TADDM 7.3.0 die Datei `$COLLATION_HOME/deploy-tomcat/ROOT/WEB-INF/cmdb-context.xml`.
- Für einen Domänenserver in TADDM 7.3.0.1 und höher die Datei `$COLLATION_HOME/apps/ROOT/WEB-INF/cmdb-context.xml`.
- Für einen Synchronisationsserver in TADDM 7.3.0 die Datei `$COLLATION_HOME/deploy-tomcat/ROOT/WEB-INF/ecmdb-context.xml`.
- Für einen Synchronisationsserver in TADDM 7.3.0.1 und höher die Datei `$COLLATION_HOME/apps/ROOT/WEB-INF/ecmdb-context.xml`.
- Für einen Erkennungsserver in TADDM 7.3.0 die Datei `$COLLATION_HOME/deploy-tomcat/ROOT/WEB-INF/discovery-server-context.xml`.
- Für einen Erkennungsserver in TADDM 7.3.0.1 und höher die Datei `$COLLATION_HOME/apps/ROOT/WEB-INF/discovery-server-context.xml`.
- Für einen Speicherserver in TADDM 7.3.0 die Datei `$COLLATION_HOME/deploy-tomcat/ROOT/WEB-INF/storage-server-context.xml`.
- Für einen Speicherserver in TADDM 7.3.0.1 und höher die Datei `$COLLATION_HOME/apps/ROOT/WEB-INF/storage-server-context.xml`.

Soll eine dieser Dateien bearbeitet werden, um die Einstellungen eines TADDM-Service zu ändern, müssen Sie zunächst in der Datei nach diesem Service suchen. Hier ein Beispiel für den ersten Teil einer Servicedefinition in der XML-Datei:

```
<bean id="Discover"
    class="com.collation.platform.service.ServiceLifecycle" init-method="start"
    destroy-method="stop">
    <property name="serviceName">
        <value>Discover</value>
    </property>
```

Die JVM-Argumente werden durch einige Elemente und Attribute in der Definition gesteuert. Beispiele dafür sind:

```
<property name="jvmArgs">
    <value>-Xms8M;-Xmx512M;
    -Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
    </value>
</property>
```

Die JVM-Argumente können als eine durch Semikola getrennte Liste im folgenden Element festgelegt werden:

```
<property name="jvmArgs"><value>
```

Sie können auch die JVM-Eigenschaften ändern, die sich in der Datei `collation.properties` befinden. Diese Eigenschaften können eines der folgenden Formate haben:

com.collation.JVM.jvmargs.VENDOR
Eine derartige Eigenschaft wird den Werten hinzugefügt, die aus der Datei `*-config.xml` gelesen werden.

com.collation.jvmargs.VENDOR
Eine derartige Eigenschaft wird allen TADDM-JVMs hinzugefügt.

com.collation.JVM.jvmargs
Eine derartige Eigenschaft überschreibt alle Werte, die in der Datei `*-config.xml` angegeben sind.

Dabei gilt Folgendes:

- JVM ist `Proxy`, `Topology`, `EventsCore`, `ExcmdbCore`, `DiscoverAdmin`, `StorageService`, `DiscoveryService`
- VENDOR ist `ibm` oder `sun`

Optimierung von JVM-Eigenschaften (Java Virtual Machine)

In der Datei `collation.properties` hängen die für die TADDM Discovery Management Console geltenden Standardwerte der JVM-Eigenschaften (Java Virtual Machine) von der Anzahl der Serverentsprechungen (SEs) in Ihrer Umgebung ab.

Standardwerte der JVM-Eigenschaften für die Discovery Management Console

- Kleine Umgebung (weniger als 1000 SEs):
  - com.collation.gui.initial.heap.size=128m
  - com.collation.gui.max.heap.size=512m
- Mittlere Umgebung (1000 – 2500 SEs):
  - com.collation.gui.initial.heap.size=256m
  - com.collation.gui.max.heap.size=768m
- Große Umgebung (2500 – 5000 SEs):
  - com.collation.gui.initial.heap.size=512m

– com.collation.gui.max.heap.size=1024m

Netzoptimierung

Nach der Implementierung eines Systems sollte das Netz überwacht werden, um sicherzustellen, dass die Auslastung der Bandbreite 50 % nicht übersteigt.

Das Netz kann die Gesamtleistung der Anwendung beeinflussen und wirkt sich in der Regel auf die Leistung aus, wenn in den folgenden Situationen eine Verzögerung auftritt:

- Zwischen dem Senden einer Anforderung von einem Clientsystem an den Server und dem Empfang dieser Anforderung auf dem Server.
- Zwischen dem Zurücksenden von Daten vom Serversystem an das Clientsystem und dem Empfang dieser Daten im Clientsystem

DNS-Optimierung

In TADDM spielt die Leistung der implementierten DNS-Infrastruktur eine gewisse Rolle. Selbst wenn die DNS-Leistung für andere Anwendungen ausreichend ist, müssen möglicherweise einige Konfigurationsschritte ausgeführt werden, um die Leistung für TADDM zu optimieren.

TADDM führt sehr viele DNS-Suchabfragen aus, um aussagekräftige Anzeigenamen für Komponenten und Ereignisse auflösen zu können. Im Gegensatz zu den meisten anderen Anwendungen verwendet TADDM vorwiegend umgekehrte Adressauflösungen (IP-Adressen werden zu Namen zugeordnet) anstelle von vorwärtsgerichteten Adressauflösungen (Namen werden zu IP-Adressen zugeordnet).

Aufgrund dieses Verwendungsmusters können sich Probleme im Zusammenhang mit der DNS-Leistung stärker auf die TADDM-Leistung auswirken als auf andere Anwendungen. Eine DNS-Reaktionszeit von 500 Millisekunden hat beispielsweise auf eine Standardanwendung wahrscheinlich nur sehr geringe Auswirkungen, während sie für TADDM aufgrund der vielen von TADDM ausgeführten DNS-Abfragen beträchtliche Leistungsprobleme bedeuten kann. Da andere Anwendungen außerdem nur vorwärtsgerichtete Adressauflösungen ausführen, wirkt sich ein Leistungsproblem bei umgekehrten Adressauflösungen zwar auf TADDM aus, nicht jedoch auf die meisten Anwendungen.

Grundsätzlich sollten Leistungsprobleme bei der DNS-Infrastruktur gelöst werden, da alle Nutzer der DNS-Services davon profitieren. Ist dies nicht möglich, lässt sich die Beeinträchtigung von TADDM durch DNS-Leistungsprobleme auf mehrere Arten mindern:

- Stellen Sie sicher, dass die Delegierung in in-addr.arpa für umgekehrte Adressauflösungen ordnungsgemäß konfiguriert ist. Delegierungsprobleme können zu langen Pausen oder Blockierungen bei umgekehrten Adressauflösungen führen, da der TADDM-Server versucht, nicht vorhandene Server zu erreichen. Diese Art von Konfigurationsproblem wirkt sich nur auf Anwendungen aus, die umgekehrte Adressauflösungen ausführen (beispielsweise TADDM).
- Richten Sie mindestens einen Caching/Forwarding-DNS-Server auf einem TADDM-Serversystem ein und konfigurieren Sie die TADDM-Server so, dass sie den betreffenden DNS-Server für die Suche verwenden. Dadurch können DNS-Suchvorgänge auf Basis der TTL-Regeln für die Zonen in der lokalen TADDM-Umgebung zwischengespeichert werden. Da dieser Servertyp statusunabhängig ist, muss er nur in äußerst geringem Maß verwaltet werden und verursacht wenig Systemaufwand.
- Richten Sie mindestens einen DNS-Slave-Server auf einem TADDM-Serversystem ein und konfigurieren Sie die TADDM-Server so, dass sie den betreffenden DNS-Server für die Suche verwenden. Dadurch können ohne Kommunikation mit der weitläufigeren DNS-Infrastruktur DNS-Suchvorgänge innerhalb der lokalen TADDM-Umgebung ausgeführt werden. Da ein DNS-Slave-Server seinen Status automatisch pflegt, muss er nur in äußerst geringem Maß verwaltet werden und verursacht wenig Systemaufwand.
- Verwenden Sie anstelle von DNS ein Alternativverfahren für die Suche, zum Beispiel die Datei `hosts`. (Diese Lösung kann einen beträchtlichen Verwaltungsaufwand bedeuten.)

Anmerkung: Ändern Sie die standardmäßigen DNS-Cacheparameter in der Datei `java.security` nicht. Die Parameter für das Caching können sich zwar auf die DNS-Leistung auswirken, die Änderungen an dieser Konfigurationsdatei bleiben bei der Ausführung von TADDM-Wartungskorrekturen jedoch nicht erhalten. Verwenden Sie stattdessen zur Optimierung der DNS-Leistung eines der in diesem Abschnitt beschriebenen Verfahren.

Optimierung des Synchronisationsservers

Die Leistung des Synchronisationsservers hängt in hohem Maße von der Datenbankverarbeitung und damit von der Datenbankpflege und -optimierung ab. Sollten Leistungsprobleme bei der Synchronisationsverarbeitung auftreten, lesen Sie die Informationen zur Datenbankoptimierung und dabei vor allem die Angaben zu den Pufferpooleinstellungen für DB2-Datenbanken, zu den Puffercacheeinstellungen für Oracle-Datenbanken sowie die Informationen zur Datenbankpflege.

Aktualisieren Sie für den Synchronisationsserver die DB2-Datenbankkonfiguration, indem Sie den folgenden Befehl eingeben:

```
UPDATE DATABASE CONFIG FOR TADDM USING
   UTIL_HEAP_SZ 5000
   LOGBUFSZ 1024
   LOCKLIST 20000
   SORTHEAP 2048
   PCKCACHESZ AUTOMATIC
;
```

Optimierung von Windows-Systemen

Über eine Optimierung der Windows-Systeme können Sie den TADDM-Services mehr Speicher zuordnen.

Führen Sie die folgenden Schritte aus:

- Die Systempagingdatei darf sich nicht auf demselben Laufwerk wie das Betriebssystem befinden; sie sollte nach Möglichkeit auf einem eigenen Plattenlaufwerk untergebracht sein.
- Konfigurieren Sie den Datenbank- und Anwendungsserver entsprechend für eine Maximierung der Daten für Netzanwendungen.

Berichtswesen

Mit externen Berichtsanzeigefunktionen, JSP-Berichtsanzeigefunktionen oder dem BIRT-Berichtssystem können Sie Berichte erstellen und dem Datenmanagementportal angepasste Berichte hinzufügen.

Externe Berichtsanzeigefunktionen

Mit einer externen Berichtsanzeigefunktion können Sie ein externes Programm ausführen, das einen Bericht generiert. Das externe Programm nutzt die TADDM-API über eine Befehlszeile für den Zugriff auf Daten. Der Bericht wird anschließend in der Benutzerschnittstelle angezeigt.

Logik der externen Berichtsanzeige erstellen

Ein externer Bericht kann in jedem ausführbaren Programm, zum Beispiel in einem Perl-Script, einem Shell-Script oder einem Java-Programm, implementiert werden. Der generierte Bericht wird nur im Datenmanagementportal angezeigt, wenn das externe Programm über die Standardausgabe eine gültige HTML-Datei ausgibt.

Informationen zu diesem Vorgang

Bei einer typischen Implementierung einer externen Berichtsanzeige wird für die Abfrage des TADDM-APIs und zur Ausgabe der XML-Abfrageergebnisse in einer temporären Datei ein Shell-Script verwendet. Danach startet das Shell-Script einen XSLT-Prozessor, der die Abfrageergebnisse in eine HTML-Ausgabe umsetzt, die wiederum an STDOUT ausgegeben wird.

Wichtig: Externe Berichtsanzeigen, welche die TADDM-API verwenden, müssen Berechtigungsnachweise an das Befehlszeilenprogramm übergeben (Script `api.sh` unter Linux und UNIX, Datei `api.bat` unter Windows). Da es sich bei den Berechtigungsnachweisen um Befehlszeilenparameter für das Script `api.sh` und `api.bat` handelt, können sie für andere Benutzer des Systems in Verarbeitungslisten sichtbar sein. Um die Offenlegung sensibler Kennwörter zu verhindern, empfiehlt sich die Einrichtung eines Dummy-Accounts, das Lesezugriff auf die Objekte besitzt, die in den extern generierten Berichten enthalten sein sollen.

Folgendes Beispiel zeigt eine einfache Implementierung eines externen Berichts mittels eines Bourne-Shell-Scripts. Kopieren Sie die folgenden Inhalte in eine neue Datei, `$COLLATION_HOME/sdk/bin/`

appServers.sh. Der Benutzer, unter dem der TADDM-Server ausgeführt wird, muss Lese- und Schreibzugriff für diese Datei erhalten:

```
#!/bin/sh
# Set environment variables for called scripts
export COLLATION_HOME=/opt/ibm/taddm/dist

# Invoke the query via API and output to $COLLATION_HOME/sdk/bin/appServers.xml
# NOTE: Change ‘restrictedUser' and ‘restrictedPassword' to your dummy account
#       credentials.
sh $COLLATION_HOME/sdk/bin/api.sh -l log -H localhost -u restrictedUser -p
restrictedPassword \ find AppServer > $COLLATION_HOME/sdk/bin/appServers.xml

# Invoke the XSLT processor
sh $COLLATION_HOME/sdk/bin/xslt.sh -XSL $COLLATION_HOME/sdk/bin/appServers.xsl
```

Nachfolgend finden Sie ein Beispiel für das Style-Sheet appServers.xsl, mit dem die über das Shell-Script generierte Datei appServers.xml umgewandelt wird. Der Bericht zeigt die Namen der Anwendungsserver und ihre Produktversionen an. Kopieren Sie die Inhalte in eine neue Datei, $COLLATION_HOME/sdk/bin/appServers.xsl. Der Benutzer, unter dem der TADDM-Server ausgeführt wird, muss Lesezugriff für diese Datei erhalten.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version = '1.0' xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:coll="urn:www-collation-com:1.0" xmlns:xhtml="http://www.w3.org/1999/xhtml">
        <xsl:variable name="nl">
        <xsl:text>
        </xsl:text>
        </xsl:variable>

        <xsl:variable name="pageheadertext">
                Simple Application Server report
        </xsl:variable>

        <xsl:variable name="pagefootertext">
                End Simple Application Server report
        </xsl:variable>


        <xsl:template match="/">
        <html>
                <head>
                <link rel="stylesheet" type="text/css" media="all"
                             href="styles.css" />
                </head>
        <body>
                <h3>
                        <xsl:value-of select="$pageheadertext"/>
                </h3>
    <table border="1" width="100%">
                 <tr>
                     <th>Product Version/th>
                     <th>Name/th>
                        </tr>

      <xsl:apply-templates select="document('appServers.xml')/coll:results"/>
    </table>
                <xsl:value-of select="$nl"/>
        </body></html>
        </xsl:template>

        <xsl:template match="coll:AppServer">
                        <tr>
                          <td>xsl:value-of select="coll:productVersion"/></td>
                          <td>xsl:value-of select="coll:displayName"/></td>
                        </tr>
        </xsl:template>
</xsl:stylesheet>
```

Führen Sie das Script appServer.sh zum Testen der Berichtslogik über eine Befehlszeile aus. Die gültige HTML-Ausgabe wird angezeigt.

Externe Berichtsanzeige zum Datenmanagementportal hinzufügen
Berichte werden dem Datenmanagementportal mittels einer Änderung der Datei reports.xml hinzugefügt. Die Datei reports.xml befindet sich im Verzeichnis $COLLATION_HOME/etc/cdm/xml/.

Vorgehensweise

Gehen Sie wie folgt vor, um dem Datenmanagementportal die externe Berichtsanzeige hinzuzufügen:

1. Öffnen Sie die Datei $COLLATION_HOME/etc/cdm/xml/reports.xml in einem Texteditor.
2. Geben Sie in der Datei reports.xml den Berichtsdeskriptor, die Berichtsgruppe, den Berichtsnamen und ein externes Script für den Bericht an. Im folgenden Beispiel wird gezeigt, wie ein externer Bericht mit dem Namen Application Servers (Anwendungsserver) erstellt wird, der sich in der Gruppe Inventory Reports (Bestandsberichte) befindet und die Datei sdk/bin/appServers.sh angibt:

```
<bean class="com.collation.cdm.reports.viewer.ExternalReportViewer" id="AppServers1">
    <property name="reportGroup"><value>Inventory Reports</value></property>
    <property name="reportName"><value>Application Servers</value></property>
    <property name="script"><value>sdk/bin/appServers.sh</value></property>
</bean>
```

3. Speichern Sie die Datei $COLLATION_HOME/etc/cdm/xml/reports.xml.
4. Der Bericht wird nun im Datenmanagementportal angezeigt.

JSP-Berichtsanzeigefunktionen

Eine JSP-Berichtsanzeigefunktion bietet zusätzliche Flexibilität und Sicherheit für Benutzer, die sich mit der JSP-Erstellung (JSP - Java Server Pages) auskennen. Die Berichtslogik, einschließlich aller API-Zugriffe, wird in eine JSP-Seite gestellt, die dann vom Datenmanagementportal wiedergegeben wird. Bei Verwendung von JSP-Berichtsanzeigefunktionen werden Sicherheitsberechtigungsnachweise automatisch vom angemeldeten Benutzer übernommen.

Logik der JSP-Berichtsanzeige erstellen

Die Logik einer JSP-Berichtsanzeige befindet sich in einer JSP-Datei, die vom Datenmanagementportal aufgerufen wird. Bei einer typischen Implementierung eines JSP-Berichts wird für die Abfrage des TADDM-APIs eine Java-Helper-Klasse 'TMSDataHelper' verwendet. Die Ergebnisse der Abfrage sind Objekte, die mit Java-Methoden bearbeitet werden können. Weitere Informationen zur TADDM-API und zum Modell finden Sie in der SDK-Dokumentation unter $COLLATION_HOME/sdk/doc.

Informationen zu diesem Vorgang

Folgendes Beispiel zeigt eine einfache Implementierung einer JSP-Berichtsanzeige. Kopieren Sie die folgenden Inhalte in eine neue Datei, $COLLATION_HOME/deploy-tomcat/reports.war/WEB-INF/view/custom.jsp (bei Verwendung von TADDM 7.3.0) bzw. $COLLATION_HOME/apps/reports.war/WEB-INF/view/custom.jsp (bei Verwendung von TADDM 7.3.0.1 und höher), und sorgen Sie dafür, dass der Benutzer, der den TADDM-Server betreibt, die Datei lesen und ausführen kann.

Nachfolgend finden Sie ein Beispiel für das Style-Sheet appServers.xsl, mit dem die über das Shell-Script generierte Datei appServers.xml umgewandelt wird. Der Bericht zeigt die Namen der Anwendungsserver und ihre Produktversionen an. Kopieren Sie die Inhalte in eine neue Datei, $COLLATION_HOME/sdk/bin/appServers.xsl. Der Benutzer, unter dem der TADDM-Server ausgeführt wird, muss Lesezugriff für diese Datei erhalten.

```
<%@ page language="java" %>
<%@ page import="com.collation.cdm.common.util.TMSDataHelper" %>
<%@ page import="java.lang.StringBuffer" %>
<%@ page import="com.collation.cdm.reports.util.ReportsParser" %>
<%@ page import="com.collation.cdm.common.util.TMSReportingTransformer" %>
<%@ page import="com.collation.platform.model.AttributeNotSetException" %>
<%@ page import="com.collation.platform.model.ModelObject" %>
<%@ page import="com.collation.platform.model.topology.sys.ComputerSystem" %>
<%@ page import="com.collation.platform.model.topology.process.BusinessProcess" %>
<%@ page import="com.collation.platform.model.topology.process.Activity" %>
<%@ taglib prefix="x" uri="http://java.sun.com/jstl/xml" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ page import="com.collation.platform.util.Props" %>
<%@ page import="java.util.ArrayList"%>
<%@ page import="com.collation.cdm.common.messages.CdmLocalizedMessages"%>
<%@ taglib uri="/WEB-INF/struts-bean.tld" prefix="bean" %>
<%@ taglib uri="/WEB-INF/struts-html.tld" prefix="html" %>
<%
java.util.Locale locale =
com.collation.cdm.common.util.CDMUtil.checkLocale(request.getLocale());
```

```
    if (null == session.getAttribute(org.apache.struts.Globals.LOCALE_KEY)) {
session.setAttribute(org.apache.struts.Globals.LOCALE_KEY, locale);
    }
%>
<%
//TMSDataHelper is a utility class for running MQL queries against the DB
TMSDataHelper tms = new TMSDataHelper(locale);

//Perform a query for all ComputerSystems
ModelObject dataIn[] = tms.doModelObjectQuery("SELECT * FROM ComputerSystem",null);

  //Build an HTML report based on the API output
  StringBuffer output = new StringBuffer();
  output.append("<p>");
  output.append("<table border=\"1\">");
                int c = 0;
                int s = dataIn.length;
                while (cs) {
                        ComputerSystem tmo = (ComputerSystem)dataIn[c];
                        String csName = null;
                        String csLabel = null;
                        if (tmo.hasName()) {
                                try {
                                        csName = tmo.getName();
                                } catch (AttributeNotSetException e) {
                                        csName = "unknown";
                                }
                        }
                        if (tmo.hasSignature()) {
                                try {
                                        csLabel = tmo.getSignature();
                                } catch (AttributeNotSetException e) {
                                        csLabel = "";
                                }
                        }
                        output.append("<tr><td colspan=\"2\" bgcolor=\"#9999FF\">");
                        output.append("ComputerSystem" + "<br>");
                        output.append(" Name: " + csName + "<br>");
                        output.append("</td><td>");
                        output.append("Signature: " + csLabel);
                        output.append("</td></tr>");
                        c++;
                }
output.append("</table>");
String bpstring = output.toString();
%>
<html>
<body>
<h1>Sample JSP Report/h1>
<%=bpstring%>
</body>
</html>
```

JSP-Berichtsanzeige zum Datenmanagementportal hinzufügen

Berichte werden dem Datenmanagementportal mittels einer Änderung der Datei `reports.xml` hinzugefügt. Die Datei `reports.xml` befindet sich im Verzeichnis `$COLLATION_HOME/etc/cdm/xml/`.

Vorgehensweise

Gehen Sie wie folgt vor, um dem Datenmanagementportal die JSP-Berichtsanzeige hinzuzufügen:

1. Öffnen Sie die Datei `$COLLATION_HOME/etc/cdm/xml/reports.xml` in einem Texteditor.
2. Geben Sie in der Datei `reports.xml` den Berichtsdeskriptor, die Berichtsgruppe, den Berichtsnamen und ein externes Script für den Bericht an. Im folgenden Beispiel wird gezeigt, wie ein externer Bericht mit dem Namen `Custom Report` (Angepasster Bericht) erstellt wird, der sich in der Gruppe `Inventory Reports` (Bestandsberichte) befindet und das Script `/WEB-INF/view/custom.jsp` angibt:

   ```
   <bean class="com.collation.cdm.reports.viewer.JSPReportViewer" id="CustomReport">
       <property name="reportGroup"><value>Inventory Reports</value></property>
       <property name="reportName"><value>Custom Report</value></property>
       <property name="script"><value>/WEB-INF/view/custom.jsp</value></property>
   </bean>
   ```
3. Speichern Sie die Datei `$COLLATION_HOME/etc/cdm/xml/reports.xml`.
4. Der Bericht müsste jetzt im Datenmanagementportal angezeigt werden.

Berichterstellung mit Tivoli Common Reporting

Da die Anzeige der BIRT-Berichte im BIRT Report Viewer als nicht sicher gilt, wurde sie standardmäßig inaktiviert. Sie können die BIRT-Berichte für TADDM jedoch in Tivoli Common Reporting importieren. Dieses Reporting-Programm ermöglicht die produktübergreifende Berichterstellung, also auch die Anzeige der TADDM-Daten. Sie können die Funktionen von Tivoli Common Reporting, zum Beispiel die Berichtsplanung, verwenden oder Tivoli Common Reporting als zentrales Repository für Berichte nutzen.

Bei einigen Tasks richten sich die auszuführenden Schritte nach der von Ihnen verwendeten Version von Tivoli Common Reporting.

Fix Pack 1 Wenn Sie TADDM 7.3 Fixpack 1 oder höher haben, lesen Sie auch die Informationen zu bewährten Verfahren unter The enhanced Cognos model in TADDM 7.3 FP*x*.

Übersicht über Tivoli Common Reporting

Das Tool Tivoli Common Reporting ist eine Berichtsfunktion, die im Lieferumfang bestimmter Tivoli-Produkte enthalten ist und eine zentrale Methode zur Anzeige und Verwaltung von Berichten mit einer konsistenten Darstellung und Funktionsweise für mehrere Produkte bereitstellt.

Tivoli Common Reporting enthält einen Datenspeicher zum Speichern und Zusammenfassen von Berichten sowie Schnittstellen zum Verwalten, Ausführen, Planen und Anzeigen von Berichten. Tivoli Common Reporting verwendet sowohl die Cognos- als auch die BIRT-Laufzeitengine.

Wichtig: Tivoli Common Reporting wird auf dem IBM Jazz for Service Management-Installationsdatenträger bereitgestellt. Wenn Sie nicht planen, IBM Jazz for Service Management zu installieren, können Sie die integrierten BIRT-Berichtsfunktionen verwenden.

Wenn auf Ihrem System Tivoli Common Reporting bereits installiert ist, können Sie optional die vordefinierten TADDM-Berichte importieren, die mit Tivoli Common Reporting kompatibel sind. Sie können anschließend Tivoli Common Reporting als zentrales Repository für Tivoli-Produktberichte verwenden. Sie können auch erweiterte Berichtsoptionen verwenden, einschließlich der produktübergreifenden Berichterstellung, der rollenbasierten Sicherheit und der Berichtsplanung.

Unterstützte Versionen dieses Produkts finden Sie im Abschnitt „Unterstützte Versionen” auf Seite 177.

Anmerkung: Wenn Sie TADDM mit IBM Tivoli Change and Configuration Management Database (CCMDB) oder IBM SmartCloud Control Desk verwenden, finden Sie in der Dokumentation zu CCMDB oder IBM SmartCloud Control Desk Informationen darüber, welche Versionen von Tivoli Common Reporting unterstützt werden.

Weitere Informationen zu Tivoli Common Reporting finden Sie im Abschnitt https://www.ibm.com/developerworks/community/groups/service/html/communityview?communityUuid=9caf63c9-15a1-4a03-96b3-8fc700f3a364.

Tivoli Common Reporting und IBM Cognos Framework Manager installieren

Sie müssen Tivoli Common Reporting und IBM Cognos Framework Manager installieren.

Vorgehensweise

Führen Sie zur Installation von Tivoli Common Reporting und IBM Cognos Framework Manager die folgenden Schritte aus:

1. Installieren Sie Tivoli Common Reporting mit den vorgegebenen Standardeinstellungen.

   Wenn Sie eine Oracle-Datenbank verwenden, müssen Sie Tivoli Common Reporting 2.1 oder 3.1 verwenden.
2. Installieren Sie das IBM Cognos Framework Manager-Paket aus dem Ordner `CognosModeling`. Übernehmen Sie dabei die vorgegebenen Standardeinstellungen.
3. Falls verfügbar, installieren Sie das Sicherheitspatch aus dem Ordner `CognosModelingFix`. Übernehmen Sie dabei die vorgegebenen Standardeinstellungen.

Datenbankclient installieren und konfigurieren

Wenn Sie Tivoli Common Reporting auf einem anderen Computer als dem TADDM-Datenbankserver installiert haben, müssen Sie einen Datenbankclient installieren, um eine Verbindung zur Datenbank her-

stellen zu können. Je nach Typ der TADDM-Datenbank müssen Sie einen DB2- oder einen Oracle-Datenbankclient installieren. Wenn Tivoli Common Reporting auf dem TADDM-Server installiert ist, brauchen Sie keinen Datenbankclient zu installieren.

Vorgehensweise

Führen Sie folgende Tasks aus:

- Wenn Sie den DB2-Datenbankclient verwenden möchten, führen Sie die folgenden Schritte aus:
  a. Installieren Sie den DB2-Client auf dem Computer, auf dem TCR installiert ist. Übernehmen Sie dabei die vorgegebenen Standardeinstellungen.
  b. Stellen Sie sicher, dass die TADDM-Datenbank katalogisiert wurde. Nur dann kann Tivoli Common Reporting über den DB2-Client eine Verbindung mit dem DB2-Server herstellen.
- Wenn Sie den Oracle-Datenbankclient verwenden möchten, führen Sie zu dessen Installation und Konfiguration die folgenden Schritte in den Assistenten 'Oracle Universal Installer' und 'Oracle Net Configuration Assistant' aus:
  a. Wählen Sie auf der Seite **Select Installation Type** (Installationstyp auswählen) des Assistenten 'Oracle Universal Installer' **Administrator** als Installationsmodus aus.
  b. Geben Sie auf der Seite **Specify Home Details** (Ausgangsdetails angeben) den Namen der Installation und den Pfad des Verzeichnisses an, in dem Sie das Produkt installieren möchten.
  c. Überprüfen Sie auf der Seite **Product-Specific Prerequisite Checks** (Produktspezifische Prüfungen der Voraussetzungen), ob die Voraussetzungen für die Installation und Konfiguration erfüllt sind. Setzen Sie die Installation erst fort, wenn alle Prüfungen mit dem Status **Succeeded** (Erfolgreich) abgeschlossen wurden.
  d. Inaktivieren Sie auf der Seite **Welcome** (Willkommen) des Assistenten 'Oracle Net Configuration Assistant' das Kontrollkästchen **Perform typical configuration** (Standardkonfiguration ausführen).
  e. Wählen Sie auf der Seite **Naming Methods Configuration, Select Naming Method** (Konfiguration der Benennungsmethode - Benennungsmethode auswählen) die Option **Local Naming** (Lokale Benennung) als Benennungsmethode aus.
  f. Geben Sie auf der Seite **Net Service Name Configuration, Service Name** (Konfiguration des Netz-Servicenamens - Servicename) den Servicenamen des fernen Oracle-Datenbankservers ein, zum Beispiel ORCL.
  g. Wählen Sie auf der Seite **Net Service Name Configuration, Select Protocols** (Konfiguration des Netz-Servicenamens - Protokolle auswählen) **TCP** als Verbindungsprotokoll für die Datenbank aus.
  h. Geben Sie auf der Seite **Net Service Name Configuration, TCP/IP Protocol** (Konfiguration des Netz-Servicenamens - TCP/IP-Protokoll) den Hostnamen des Computers ein, auf dem die Datenbank ausgeführt wird. Wählen Sie **Use the standard port number of 1521** (Standardportnummer 1521 verwenden) aus.
  i. Wählen Sie auf der Seite **Net Service Name Configuration, Test** (Konfiguration des Netz-Servicenamens - Test) die Option **Yes, perform a test** (Ja, Test ausführen) aus.

     Wenn Benutzername und Kennwort Ihrer Datenbank korrekt eingegeben wurden, wird folgender Text angezeigt:

     ```
     Connecting... Test successful.
     ```

     Falls keine Verbindung mit der Datenbank zustande kommt, müssen Sie vermutlich Ihre Anmeldedaten ändern. Zum Ändern der Anmeldedaten für die Datenbank klicken Sie auf **Change Login** (Anmeldung ändern) und geben Sie einen gültigen Datenbankbenutzernamen sowie das zugehörige Kennwort ein.

j. Bestätigen Sie auf der Seite **Net Service Name Configuration, Net Service Name** (Konfiguration des Netz-Servicenamens - Netz-Servicename) den Standardservicenamen (dies sollte der zuvor angegebene Servicename sein).

k. Erstellen Sie eine Windows-Systemvariable namens `TNS_ADMIN` und setzen Sie den Wert auf den vollständigen Pfad des Ordners, der die Datei `tnsnames.ora` enthält. Während der Installation wird die Datei `tnsnames.ora` im Ordner `%ORACLE_HOME%/client_1/NETWORK/ADMIN` erstellt, zum Beispiel in `C:/oracle/product/10.2.0/client_1/NETWORK/ADMIN`.

l. Legen Sie die Variable `TNS_ADMIN` im Script `startTCRserver.sh/bat` so fest, dass sie auf die Position der Datei `tnsnames.ora` verweist, zum Beispiel auf `%ORACLE_HOME%/client_1/NETWORK/ADMIN`.

m. Starten Sie den Computer erneut, damit die neue Systemvariable zur Verfügung steht.

IBM Cognos Framework Manager konfigurieren

Sie müssen die Eigenschaften von IBM Cognos 10 Framework Manager mit den korrekten Werten aktualisieren.

Informationen zu diesem Vorgang

Anmerkung: Die folgende Prozedur bezieht sich auf die Konfiguration von IBM Cognos 10 Framework Manager for Tivoli Common Reporting 3.1. Sie ist jedoch ebenso gültig für IBM Cognos 8 Framework Manager for Tivoli Common Reporting 2.1.

Bei der Installation von Tivoli Common Reporting wird das Programm IBM Cognos Configuration installiert, wobei einige Eigenschaften aktualisiert werden. Bei der Installation von IBM Cognos 10 Framework Manager wird hingegen eine andere Version des IBM Cognos Configuration-Programms installiert, wobei nicht alle Eigenschaften aktualisiert werden. Daher müssen Sie die Werte einiger Eigenschaften manuell aus der Tivoli Common Reporting-Version von IBM Cognos Configuration in die IBM Cognos 10 Framework Manager-Version von IBM Cognos Configuration kopieren.

Vorgehensweise

Führen Sie zur Konfiguration von IBM Cognos 10 Framework Manager die folgenden Schritte aus:

1. Öffnen Sie die bei der Installation von Tivoli Common Reporting installierte Version von IBM Cognos Configuration. Klicken Sie dazu auf **Start** > **Programme** > **Tivoli Common Reporting 3.1** > **IBM Cognos Configuration**.
2. Öffnen Sie die Version von IBM Cognos Configuration, die von IBM Cognos 10 installiert wurde. Klicken Sie dazu auf **Start** > **Programme** > **IBM Cognos 10** > **IBM Cognos Configuration**.
3. Klicken Sie in beiden Versionen von IBM Cognos Configuration auf **Local Configuration (Lokale Konfiguration)** > **Environment (Umgebung)**.
4. Kopieren Sie den Wert der Eigenschaft **Gateway URI** (Gateway-URI) aus der IBM Cognos Configuration-Version von Tivoli Common Reporting in die Eigenschaft **Gateway URI** der IBM Cognos Configuration-Version von IBM Cognos 10. Die URI-Syntax lautet wie folgt: `http://tcrhost:16310/tarf/servlet/dispatch`.
5. Kopieren Sie den Wert der Eigenschaft **External dispatcher URI** (Externe Dispatcher-URI) aus der IBM Cognos Configuration-Version von Tivoli Common Reporting in die Eigenschaft **Dispatcher URI for external applications** (Dispatcher-URI für externe Anwendungen) der IBM Cognos Configuration-Version von IBM Cognos 10. Die URI-Syntax lautet wie folgt: `http://tcrhost:16310/tarf/servlet/dispatch`.
6. Speichern Sie die Änderungen, die Sie an IBM Cognos Version 10 von IBM Cognos Configuration durchgeführt haben.

Fix Pack 1 **TADDM-Modell generieren**

Sie können das TADDM-Modell so generieren, dass es die Momentaufnahme des TADDM-Datenbankinhalts, einschließlich Definitionen aller erweiterten Attribute, enthält. Wenn Sie keine erweiterten Attribute verwenden, können Sie diese Prozedur überspringen und die vorgefertigte TADDM-Cognos-Modelldatei `$COLLATION_HOME/etc/reporting/tcr/model.xml` verwenden.

Vorbereitende Schritte
Das generierte TADDM-Modell schließt alle Common Data Model-Klassen, die von TADDM unterstützt werden, und Definitionen erweiterter Attribute, die in der TADDM-Datenbank gespeichert sind, ein. Sie können das TADDM-Modell im Tivoli Common Reporting-Server veröffentlichen und in Cognos-Berichten verwenden. Das TADDM-Modell kann viele Male generiert werden. Nach jeder neuen Generierung enthält das Modell den aktualisierten TADDM-Datenbankinhalt.

Hinweise:

- Wenn Definitionen erweiterter Attribute aus der TADDM-Datenbank entfernt werden, nachdem das TADDM-Modell im Tivoli Common Reporting-Server veröffentlicht wurde, funktionieren die Cognos-Berichte, die die Definitionen verwenden, möglicherweise nicht mehr.
- Wenn Sie das Betriebssystem Windows verwenden, ändern Sie die Erweiterung der in der folgenden Prozedur verwendeten Scripts von `.sh` in `.bat`.

Vorgehensweise

1. Öffnen Sie auf dem TADDM-Server das Verzeichnis `$COLLATION_HOME/bin`.
2. Aktualisieren Sie wie folgt die Ansichten für erweiterte Attribute:
   a) Wenn Sie Ansichten für erweiterte Attribute erstellt haben, entfernen Sie sie durch Ausführung des folgenden Befehls:
   ```
   ./extattr_views.sh remove
   ```
   b) Generieren Sie SQL-Scripts mit Definitionen von Ansichten für erweiterte Attribute, indem Sie folgenden Befehl ausführen:
   ```
   ./extattr_views.sh scripts
   ```
   c) Erstellen Sie die Ansichten für erweiterte Attribute mit Verwendung der generierten SQL-Scripts, indem Sie folgenden Befehl ausführen:
   ```
   ./extattr_views.sh create
   ```
3. Führen Sie folgenden Befehl aus, um die Cognos-Modelldatei zu generieren:
   ```
   ./genCognosModel.sh
   ```
   Das generierte TADDM-Modell wird in der Datei `model.xml` gespeichert und im Verzeichnis `$COLLATION_HOME/etc/reporting/tcr`abgelegt. . Die Protokollnachrichten des Befehls stehen in der Datei `$COLLATION_HOME/log/genCognosModel.log`.

Nächste Schritte
Sie können das generierte TADDM-Modell mithilfe von IBM Cognos Framework Manager im Tivoli Common Reporting-Server veröffentlichen. Weitere Informationen finden Sie im Abschnitt „Modell mit IBM Cognos Framework Manager veröffentlichen” auf Seite 153.

Weitere Informationen zu Ansichten für erweiterte Attribute finden Sie im Abschnitt *Extended attributes views* (Ansichten für erweiterte Attribute) im TADDM *SDK Developer's Guide*.

Modell- und Musterberichte in Tivoli Common Reporting importieren
Die TADDM-Musterberichte können in Tivoli Common Reporting Version 2.1 und 3.1 importiert werden.

Informationen zu diesem Vorgang

Diese Prozedur bezieht sich auf **Tivoli Common Reporting Version 2.1**.

Vorgehensweise

Führen Sie zum Importieren der Modell- und Musterberichte in Tivoli Common Reporting 2.1 die folgenden Schritte aus:

1. Kopieren Sie das Paket `$COLLATION_HOME/etc/reporting/TADDMPackage.zip` vom TADDM-Server in den Ordner `TCRComponent/cognos/deployment` des Tivoli Common Reporting-Servers.
2. Öffnen Sie die Tivoli Common Reporting-Homepage.
3. Klicken Sie auf **Berichterstellung** > **Common Reporting**.
4. Klicken Sie im **Startmenü** auf **Administration**. Das Fenster **Administration** wird angezeigt.
5. Klicken Sie auf die Registerkarte **Configuration** (Konfiguration).
6. Klicken Sie auf das Symbol für **Neuer Import**.

   Der Assistent **New Import** (Neuer Import) wird geöffnet.
7. Wählen Sie in der Liste der verfügbaren Pakete das Paket **TADDMPackage** aus. Klicken Sie auf **Weiter**.
8. Geben Sie im Feld **Description** (Beschreibung) eine Beschreibung des Pakets ein. Klicken Sie auf **Weiter**.
9. Aktivieren Sie das Kontrollkästchen neben dem Namen des Pakets.
10. Klicken Sie im Bereich **Optionen** auf **The owner from the source** (Eigentümer aus Quelle) und **New and existing entries** (Neue und vorhandene Einträge). Wählen Sie im Menü **Recording level** (Aufzeichnungsstufe) die Option **Basic** (Basis) aus. Klicken Sie auf **Weiter**.
11. Klicken Sie auf **Save and run once** (Speichern und einmal ausführen). Klicken Sie auf **Weiter**.

Informationen zu diesem Vorgang

Diese Prozedur bezieht sich auf **Tivoli Common Reporting Version 3.1**.

Vorgehensweise

Führen Sie zum Importieren der Modell- und Musterberichte in Tivoli Common Reporting 3.1 die folgenden Schritte aus:

1. Kopieren Sie das Paket `$COLLATION_HOME/etc/reporting/TADDMPackage.zip` vom TADDM-Server in den Ordner `reporting/cognos/deployment` der JazzSM-Installation.
2. Öffnen Sie die Tivoli Common Reporting-Homepage.
3. Klicken Sie auf **Berichterstellung** > **Common Reporting**.
4. Klicken Sie im **Startmenü** auf **IBM Cognos Administration**. Das Fenster **Administration** wird angezeigt.
5. Klicken Sie auf die Registerkarte **Configuration** (Konfiguration).
6. Wechseln Sie zu **Content Administration** (Inhaltsadministration). Klicken Sie auf das Symbol für **New Import** (Neuer Import).

   Der Assistent **New Import** wird geöffnet.
7. Wählen Sie in der Liste der verfügbaren Pakete das Paket **TADDMPackage** aus. Klicken Sie auf **Weiter**.
8. Geben Sie im Feld **Description** (Beschreibung) eine Beschreibung des Pakets ein. Klicken Sie auf **Weiter**.
9. Aktivieren Sie das Kontrollkästchen neben dem Namen des Pakets. Klicken Sie auf **Weiter**.
10. Klicken Sie im Bereich **Entry ownership** (Eigentumsrecht für Einträge) auf **The owner from the source** (Eigentümer aus Quelle) und **New and existing entries** (Neue und vorhandene Einträge). Wählen Sie im Menü **Recording level** (Aufzeichnungsstufe) im Bereich **Deployment record** (Implementierungsdatensatz) die Option **Basic** (Basis) aus. Klicken Sie auf **Weiter**.
11. Stellen Sie sicher, dass die angegebenen Werte korrekt sind. Klicken Sie auf **Weiter**.
12. Klicken Sie auf **Save and run once** (Speichern und einmal ausführen). Klicken Sie auf **Fertigstellen**.
13. Klicken Sie auf **Run** (Ausführen).

Datenansichten im TADDM-Modell

Aus der TADDM-Datenmodelldatei `model.xml` können Sie Berichte generieren.

Das Datenmodell setzt sich aus mehreren Namensbereichen zusammen. Ein Namensbereich ist ein logischer Container, der eindeutige Namen enthält. Jeder Namensbereich enthält Abfragesubjekte, Abfrageelemente und Objekte. Folgende Namensbereiche stehen nach dem Import der TADDM-Datei `model.xml` zur Verfügung:

Fix Pack 1 **CDM-Namensbereiche**

Diese Ansichten enthalten die Abfragesubjekte für fast alle Common Data Model-Klassen, einschließlich erkennungsbezogener Klassen, nach ihren Paketnamen in mehrere Namensbereiche aufgeteilt. Paketnamen werden alphabetisch sortiert. Die Simplified Model-Klassen sind an dem Präfix `simple` im Namen des Namensbereichs erkennbar. Mithilfe dieser Daten können Sie Berichte generieren, die unterschiedliche Typen von CDM-Objekten enthalten.

Die Abfragesubjekte in CDM-Namensbereichen enthalten vordefinierte Beziehungen, die über die `Parent`-Attribute hergestellt werden. Beispielsweise hat die Klasse `app.j2ee.J2EEDomain` das `Servers`-Attribut mit dem Typ `app.j2ee.J2EEServer[]`. Und die Klasse `app.j2ee.J2EEServer` hat das `Parent`-Attribut mit dem Typ `app.j2ee.J2EEDomain`. Deshalb gibt es zwischen allen kompatiblen Paaren von CDM-Klassen vordefinierte Beziehungen, z. B.:

- `app.j2ee.J2EEDomain [0..1] - [0..n] app.j2ee.J2EEServer`
- `app.j2ee.J2EEDomain [0..1] - [0..n] app.j2ee.jboss.JBossServer`
- `app.j2ee.J2EEDomain [0..1] - [0..n] app.j2ee.weblogic.WebLogicServer`
- `app.j2ee.jboss.JBossDomain [0..1] - [0..n] app.j2ee.jboss.JBossServer`
- `app.j2ee.websphere.WebSphereCell [0..1] - [0..n] app.j2ee.websphere.WebSphereServer`

In TADDM 7.3.0.1 werden einige Abfragesubjekte in CDM-Namensbereichen für nicht persistente Attribute des Feldgruppentyps definiert. In TADDM 7.3.0.2 werden Abfragesubjekte für alle Attribute des Feldgruppentyps definiert. Ihre Namen haben folgendes Format: "*[Name der Klasse zur Deklaration des Feldgruppenattributs]-->[Name des Feldgruppenattributs]*". Beispielsweise hat die Klasse `simple.SGroup` das `GroupMembers`-Attribut mit dem Typ `ModelObject[]`, sodass das Abfragesubjekt `"SGroup-->GroupMembers"` lautet. Diese Abfragesubjekte enthalten vordefinierte Beziehungen zwischen den beschriebenen Feldgruppenattributen und allen CDM-Klassen, die diese Attribute enthalten. Beispielsweise sind für das genannte `GroupMembers`-Attribut unter anderem folgende Beziehungen definiert:

- `simple.SGroup [1..1] - [0..n] simple."SGroup-->GroupMembers"`
- `simple.SBaseCollection [1..1] - [0..n] simple."SGroup-->GroupMembers"`
- `app.biztalk.BizTalkGroup [1..1] - [0..n] simple."SGroup-->GroupMembers"`
- `app.hacmp.HACMPResourceGroup [1..1] - [0..n] simple."SGroup-->GroupMembers"`

Um Attribute des Feldgruppentyps verwenden zu können, müssen Sie eine Beziehung zwischen einem Attribut des Feldgruppentyps und der erforderlichen CDM-Klasse definieren, indem Sie dessen `PK_C`-Attribut oder, im Falle eines nicht persistenten Attributs des Feldgruppentyps (`ModelObject[]`), dessen `Guid`-Attribut verwenden. Beispiele dafür sind:

- Fix Pack 2 Um einen Cognos-Bericht zu erstellen, der `sys.zOS.ZReportFile`-Objekte als `ZReportfiles` der `sys.ComputerSystem`-Objekte anzeigt, müssen Sie in IBM Cognos Report Studio eine Verknüpfung zwischen folgenden Spalten definieren:

```
sys."ComputerSystem-->ZReportfiles".PK__ZReportfiles_C
 [0..n]-[0..1] sys.zOS.ZReportFile.PK_C
```

- Um einen Cognos-Bericht zu erstellen, der `app.AppServer`-Objekte als `GroupMembers` der `simple.SBaseCollection`-Objekte anzeigt, müssen Sie in IBM Cognos Report Studio eine Verknüpfung zwischen folgenden Spalten definieren:

```
 simple."SGroup-->GroupMembers".GroupMembersGuids [0..n]-
[0..1] app.AppServer.Guid
```

Fix Pack 2 In vielen Fällen müssen Verknüpfungen für Attribute des Feldgruppentyps nicht manuell erstellt werden, da die entsprechenden `Parent`-Attribute der abhängigen Objekte vorhanden sind. Das Cognos-Modell enthält Beziehungen für sie. Sie müssen beispielsweise keine Verknüpfungen manuell erstellen, um einen Bericht zu erstellen, der `sys.FileSystem`-Objekte als `FileSystems` der `sys.ComputerSystem`-Objekte anzeigt, da die`sys.FileSystem`-Objekte über das `Parent`-Attribut verfügen, das auf die `sys.ComputerSystem`-Objekte verweist.

Fix Pack 3 In TADDM 7.3.0.3 und höher enthält das Cognos-Modell Abfrageelemente des Typs `Data Time` für alle Attribute des Zeitmarkentyps. Zum Beispiel enthält das Abfragesubjekt `sys.aix.AixUnitaryComputerSystem` folgende Abfrageelemente:

- `LastStoredTime` des Typs `Int64`, das auf die Spalte LASTSTOREDTIME_C in der Bausteinansicht verweist. Beispielwert in der Spalte: `1445417251307`.
- `LastStoredTimeT` des Typs `Data Time`, das auf die Spalte LASTSTOREDTIME_T in der Bausteinansicht verweist. Beispielwert in der Spalte: `Oct 21, 2015 10:47:31 AM`.

Das Abfrageelement `LastStoredTimeT` entspricht funktional dem Abfrageelement `LastStoredTime`, nur dass es im Format der koordinierten Weltzeit (UTC) statt als UNIX-Epoche (lange Ganzzahl) angegeben wird. Die Abfrageelemente, die das Suffix T enthalten, sind die Zeitmarkenäquivalente des ursprünglichen langen ganzzahligen Attributs.

Namensbereich 'WebSphere'

Anmerkung: Fix Pack 1 Namensbereich 'WebSphere' wird in TADDM 7.3.0.1 und höher nicht mehr verwendet.

Diese Ansicht enthält die primären Abfragesubjekte für eine WebSphere-Umgebung. Mithilfe dieser Daten können Sie WebSphere-spezifische Berichte generieren, z. B. Eigenschaftenlisten oder JVM-Einstellungen von WebSphere-Servern. Dieses Abfragesubjekt `WebSphere-Server` ist mit dem Abfragesubjekt `Anwendungsserver` aus dem gemeinsamen Namensbereich verknüpft. Die Abfragesubjekte `WebSphere-Cluster` und `WebSphere-Zelle` sind mit den Abfragesubjekten `Anwendungsservercluster` und `J2EE-Domäne` aus dem gemeinsamen Namensbereich verknüpft.

Gemeinsamer Namensbereich

Anmerkung: Fix Pack 1 Gemeinsamer Namensbereich wird in TADDM 7.3.0.1 und höher nicht mehr verwendet.

Diese Ansicht enthält Abfragesubjekte, die als Schlüsselklassen gelten und zur Überbrückung bei Datenverknüpfungen zwischen verschiedenen Namensbereichen verwendet werden können. Der gemeinsame Namensbereich enthält Informationen zu Computersystemen und Datensammlungsklassen. Anhand dieser Daten können Bestandsberichte erstellt werden.

Namensbereich 'Geschäftsanwendungen'

Anmerkung: Fix Pack 1 Namensbereich 'Geschäftsanwendungen' wird in TADDM 7.3.0.1 und höher nicht mehr verwendet.

Diese Ansicht enthält die Abfragesubjekte `Anwendung` und `Funktionale Gruppe` für eine Geschäftsanwendung. Das Abfragesubjekt 'Funktionale Gruppe' ist über das Abfragesubjekt `Datensammlung` mit dem gemeinsamen Namensbereich verknüpft. Anhand dieser Daten können Berichte erstellt werden, in denen die Geschäftsanwendungen und ihre Elemente aufgeführt sind.

Namensbereich 'Datenbank'

Anmerkung: Fix Pack 1 Namensbereich 'Datenbank' wird in TADDM 7.3.0.1 und höher nicht mehr verwendet.

Diese Ansicht enthält Abfragesubjekte für die Datenbank und Datenbankserver. Mit dem Abfragesubjekt `Alle Datenbanken` können Sie eher allgemeine Datenbankberichte anstatt herstellerspezifischer Datenbankberichte erstellen. Der Datenbankinhalt ist über das Abfragesubjekt `Anwendungsserver` mit dem gemeinsamen Namensbereich verknüpft.

Namensbereich 'Abhängigkeiten und Beziehungen'

Anmerkung: Fix Pack 1 Namensbereich 'Abhängigkeiten und Beziehungen' wird in TADDM 7.3.0.1 und höher nicht mehr verwendet.

Diese Ansicht enthält Abfragesubjekte, die generierte Beziehungen und Abhängigkeiten darstellen, z. B. IP-Abhängigkeiten oder Switch-to-Device-Beziehungen. Mit dem allgemeinen Abfragesubjekt `Beziehung (unverknüpft)` können Sie manuelle Verknüpfungen beim Generieren eines Berichts oder einer Abfrage erstellen. Das Abfragesubjekt `SwitchToDevice` verbindet Wechsel zu Computersystemobjekten im gemeinsamen Namensbereich.

Es gibt drei Abfragesubjekte zur Serveraffinität. Das Abfragesubjekt `Server` zeigt die Union-Verknüpfung aller Computersysteme, Anwendungsserver und Serviceobjekte in der Datenbank an. Das Abfragesubjekt `Affinität (zielverknüpft)` verbindet jede Affinitätsbeziehung mit ihrem jeweiligen Ziel im Abfragesubjekt `Server`. Das Abfragesubjekt `Affinität (quellenverknüpft)` verbindet jede Affinitätsbeziehung mit ihrer jeweiligen Quelle im Abfragesubjekt`Server`. Der Serverinhalt ist über die Abfragesubjekte 'Computersystem', 'Anwendungsserver' und 'Service' mit dem gemeinsamen Namensbereich verknüpft. Mithilfe dieser Daten können Sie einen allgemeinen Bericht generieren, der die Beziehung zwischen Konfigurationselementen im Netz anzeigt.

Modell mit IBM Cognos Framework Manager veröffentlichen

Wenn Sie dem TADDM-Datenmodell Objekte hinzufügen möchten (Datei `model.xml`), müssen Sie die Datei bearbeiten und anschließend mit IBM Cognos 10 Framework Manager importieren.

Informationen zu diesem Vorgang

Die folgende Prozedur bezieht sich auf IBM Cognos 10 Framework Manager. Sie ist jedoch ebenso gültig für IBM Cognos 8 Framework Manager.

Vorgehensweise

Führen Sie zum Importieren des Datenmodells mit IBM Cognos 10 Framework Manager die folgenden Schritte aus:

1. Starten Sie IBM Cognos 10 Framework Manager.
2. Erstellen Sie ein neues Projekt.
3. Geben Sie auf Aufforderung die Berechtigungsnachweise für den Tivoli Common Reporting-Server ein.

   Eventuell müssen Sie diese Berechtigungsnachweise auch mehrmals eingeben.
4. Schließen Sie IBM Cognos 10 Framework Manager.
5. Kopieren Sie die folgende Datei vom TADDM-Server in den Projektordner von Cognos Framework:

   ```
   $COLLATION_HOME/etc/reporting/tcr/model.xml
   ```

   Die vorhandene `model.xml`-Datei im Cognos Framework-Projektordner muss dabei überschrieben werden.
6. Starten Sie IBM Cognos 10 Framework Manager und öffnen Sie das zuvor erstellte Projekt.
7. Klicken Sie im **Project Viewer** auf **Datenquellen** > ***Content-Manager-Datenquelle***.
8. Wenn Sie eine DB2-Datenbank mit einem anderen Namen als dem Namen verwenden, der in der Cognos-Datenbankquelle definiert ist, ersetzen Sie den Inhalt des Feldes **Schema** durch den DB2-Instanznamen, der für die TADDM-Datenbank verwendet wird.
9. Speichern Sie das Projekt.
10. Klicken Sie im **Project Viewer** auf **Pakete**.
11. Klicken Sie mit der rechten Maustaste auf den Paketnamen und wählen Sie **Publish Packages** (Pakete veröffentlichen) aus. Die Prüfung und Veröffentlichung des TADDM-Cognos-Modells kann mehrere Minuten dauern.

Datenquelle in Tivoli Common Reporting konfigurieren

Die Datenquelle können Sie in Tivoli Common Reporting konfigurieren.

Vorbereitende Schritte

Stellen Sie sicher, dass eine der folgenden Situationen zutreffend ist:

- Die TADDM-Datenbank ist lokal katalogisiert.
- Tivoli Common Reporting wird auf dem Server mit der TADDM-Datenbank ausgeführt.

Wenn Sie eine DB2-Datenbank verwenden, muss der Schemaname dem Namen der DB2-Instanz entsprechen. Der Schemaname gibt den Namen der DB2-Datenbank an, mit der der Zugriff auf die angegebene Datenbank autorisiert wird. Der Name der DB2-Instanz wurde bei der Installation von TADDM festgelegt. Der in der TADDM-Datei `model.xml` definierte Standardname der Instanz lautet `DB2INST1`. Bei Bedarf können Sie den Schemanamen ändern.

Wenn Sie eine Oracle-Datenbank verwenden, muss der Schemaname leer sein.

Vorgehensweise

Führen Sie zur Konfiguration der Datenquelle in Tivoli Common Reporting die folgenden Schritte aus:

1. Öffnen Sie die Tivoli Common Reporting-Homepage.
2. Klicken Sie auf **Berichterstellung** > **Common Reporting**.
3. Wählen Sie im Menü **Launch** (Start) abhängig von der verwendeten Version von Tivoli Common Reporting einen der folgenden Menüpunkte aus:
   - Version 2.1 - **Administration**.
   - Version 3.1 - **IBM Cognos Administration**.

   Das Fenster **Administration** wird angezeigt.
4. Klicken Sie auf die Registerkarte **Configuration** (Konfiguration).
5. Klicken Sie auf das Symbol für **Neue Datenquelle**.

   Daraufhin wird der Assistent **New Data Source** (Neue Datenquelle) angezeigt.
6. Geben Sie im Feld **Name** den Namen `CMDBTCR` ein.

   Dieser Name wird im Datenmodell referenziert; die neue Datenquelle muss daher den gleichen Namen erhalten.
7. Wählen Sie im Menü **Type** (Typ) den Typ der verwendeten Datenbank aus.
8. Führen Sie einen der folgenden Schritte aus:
   - Wenn Sie eine DB2-Datenbank verwenden, geben Sie im Feld **DB2 database name** (Name der DB2-Datenbank) den TADDM-Datenbanknamen bzw. den Aliasnamen der katalogisierten TADDM-Datenbank ein.
   - Wenn Sie eine Oracle-Datenbank verwenden, geben Sie im Feld **SQL*Net connect string** (SQL*Net-Verbindungszeichenfolge) den Servicenamen der Oracle-Datenbank ein, zum Beispiel `ORCL`. Der Servicename der Oracle-Datenbank wurde bei der Konfiguration des Oracle-Datenbankclients festgelegt. Den Namen finden Sie in der Datei `%TNS_ADMIN%/tnsnames.ora`. Suchen Sie die folgende Zeichenfolge:

     ```
     SERVICE_NAME =
     ```
9. Geben Sie im Bereich **Signon** (Anmeldung) den Benutzernamen und das Kennwort des Datenbankbenutzers ein.
10. Klicken Sie zum Testen der Datenbankverbindung auf **Test**.

    Auf der **Ergebnisseite** des Assistenten **New Data Source** (Neue Datenquelle) wird der Status des Tests angezeigt.

TADDM-Berichtspaket in Tivoli Common Reporting importieren

Um die vordefinierten TADDM-Berichte in Tivoli Common Reporting zu importieren, können Sie das TADDM-Berichtspaket importieren.

Vorbereitende Schritte
Zuerst muss die Funktion Tivoli Common Reporting auf Ihrem System installiert sein. Tivoli Common Reporting wird mit einigen Tivoli-Produkten bereitgestellt, ist jedoch derzeit nicht im Lieferumfang von TADDM enthalten.

Informationen zu diesem Vorgang
Ein Tivoli Common Reporting-*Berichtspaket* ist eine ZIP-Datei, die einen oder mehrere Berichte oder Berichtsentwürfe zusammen mit ihren erforderlichen Ressourcen in einem Format enthält, das von Tivoli Common Reporting verwendet werden kann. Die vordefinierten BIRT-Berichte für TADDM werden in einem Berichtspaket bereitgestellt, das Sie in Tivoli Common Reporting importieren können.

Zu einigen BIRT-Berichten sind verschiedene Berichtsversionen verfügbar, wobei es darauf ankommt, auf welchem Server der Bericht ausgeführt wird. So wird beispielsweise der Bericht TADDM_SNAPSHOT_CHANGE auf dem Domänen- oder Speicherserver und der Bericht TADDM_SNAPSHOT_SYNC_CHANGE auf dem Synchronisationsserver verwendet. Im Allgemeinen ist immer nur die entsprechende Berichtsversion verfügbar, nach dem Import von BIRT-Berichten in Tivoli Common Reporting kann es jedoch sein, dass beide Versionen zur Verfügung stehen. Achten Sie darauf, nur die entsprechende Berichtsversion für den jeweiligen Server zu verwenden, auf dem der Bericht ausgeführt werden soll.

Nach dem Import von BIRT-Berichten in Tivoli Common Reporting kann es Berichte mit dem Text "Drillthrough only" (Nur Drillthrough) im Berichtsnamen geben. Die Ausführung dieser Berichte ist im Rahmen einer Drilloperation für ausgewählte Daten in einem anderen Bericht vorgesehen, diese Berichte dürfen nicht separat ausgeführt werden.

Der Bericht 'Serveraffinität nach Bereich' kann nicht in Tivoli Common Reporting importiert werden.

Weitere Informationen zum Importieren von Berichtspaketen finden Sie in der Dokumentation zu Tivoli Common Reporting.

Vorgehensweise

Gehen Sie für den Import von TADDM-Berichten wie folgt vor:

1. Führen Sie bei Verwendung von Tivoli Common Reporting 1.3 die folgenden Schritte aus:
   a) Wechseln Sie im Berichtsnavigationsfenster von Tivoli Common Reporting zur Registerkarte **Navigation**.
   b) Klicken Sie mit der rechten Maustaste auf den Stammknoten der Navigationsstruktur (Report Sets (Berichtsgruppen)).
   c) Klicken Sie auf **Import Report Package** (Berichtspaket importieren).
   d) Geben Sie im Fenster 'Import Report Package' die Speicherposition der Berichtspaketdatei `TADDM-Reports.zip` an.
      Diese Datei befindet sich im Verzeichnis `$COLLATION_HOME/etc/reporting`.
   e) Erweitern Sie den Bereich **Advanced Options** (Erweiterte Optionen) und führen Sie folgende Schritte aus:
      1) Markieren Sie das Kontrollkästchen **Overwrite** (Überschreiben). Dadurch wird sichergestellt, dass alle zuvor installierten Kopien der Berichte überschrieben werden.
      2) Geben Sie im Feld **Security Set** (Sicherheitsgruppe) den Namen der Sicherheitsgruppe ein, in welche der Inhalt des Berichtspakets importiert werden soll.
   f) Klicken Sie auf **Import** (Importieren).
      Das TADDM-Berichtspaket wird in den Tivoli Common Reporting-Datenspeicher importiert.
2. Führen Sie bei Verwendung von Tivoli Common Reporting 2.1 folgende Schritte aus:
   a) Öffnen Sie eine Befehlszeile und wechseln Sie in das Verzeichnis `TIP-Installationsverzeichnis/tipv2Components/TCRComponent/bin`.
   b) Führen Sie den Importbefehl aus:

```
trcmd -user Benutzer-ID -password Kennwort -import -bulk Paketdatei
```

Dabei steht *Paketdatei* für den Pfad zur Berichtspaketdatei TADDMReports.zip, die aus dem Verzeichnis $COLLATION_HOME/etc/reporting auf dem TADDM-Server auf den Tivoli Common Reporting-Server kopiert wird.

c) Das TADDM-Berichtspaket wird in den Tivoli Common Reporting-Datenspeicher importiert.

3. Führen Sie bei Verwendung von Tivoli Common Reporting 3.1 die folgenden Schritte aus:

a) Öffnen Sie eine Befehlszeile und wechseln Sie in das Verzeichnis JazzSM_install_dir/reporting/bin.

b) Führen Sie den Importbefehl aus:

```
trcmd -user Benutzer-ID -password Kennwort -import -bulk Paketdatei
```

Dabei steht *Paketdatei* für den Pfad zur Berichtspaketdatei TADDMReports.zip, die aus dem Verzeichnis $COLLATION_HOME/etc/reporting auf dem TADDM-Server auf den Tivoli Common Reporting-Server kopiert wird.

c) Das TADDM-Berichtspaket wird in den Tivoli Common Reporting-Datenspeicher importiert.

Nächste Schritte

Nachdem Sie die TADDM-Berichte importiert haben, müssen Sie die JDBC-Datenquelle für die einzelnen Berichte neu konfigurieren.

TADDM-BIRT-Berichte in Tivoli Common Reporting konfigurieren

Nachdem Sie die TADDM-Berichte in Tivoli Common Reporting importiert haben, müssen Sie die JDBC-Datenquelle konfigurieren, die von den einzelnen Berichten verwendet wird.

Vorbereitende Schritte

Vergewissern Sie sich vor der Konfiguration des JDBC-Zugriffs, dass die entsprechenden JDBC-Treiberdateien im Treiberverzeichnis von Tivoli Common Reporting installiert sind. Für Tivoli Common Reporting 1.3 befinden sich diese Dateien im folgenden Verzeichnis:

```
TCR-Installationsverzeichnis/products/tcr/lib/birt-runtime-2_2_1/ReportEngine/plugins/
  org.eclipse.birt.report.data.oda.jdbc_2.2.1.r22x_v20070919/drivers
```

Für Tivoli Common Reporting 2.1 befinden sich diese Dateien im folgenden Verzeichnis:

```
TIP-Installationsverzeichnis/tip21Components/TCRComponent/lib/birt-runtime-2_2_2/ReportEngine/
plugins/
org.eclipse.birt.report.data.oda.jdbc_2.2.2.r22x_v20071206/drivers
```

Für Tivoli Common Reporting 3.1 befinden sich diese Dateien im folgenden Verzeichnis:

```
JazzSM-Installationsverzeichnis/reporting/lib/birt-runtime-2_2_2/ReportEngine/plugins/
org.eclipse.birt.report.data.oda.jdbc_2.2.2.r22x_v20071206/drivers
```

Wenn Sie eine Oracle-Datenbank verwenden, muss die Datei ojdbc14.jar oder ojdbc5.jar im Verzeichnis enthalten sein.

Informationen zu diesem Vorgang

Importierte Berichte werden anfänglich für die Verwendung einer Standarddatenquelle konfiguriert. Sie müssen die Datenquelleneigenschaften der einzelnen TADDM-Berichte so ändern, dass die Datenbank verwendet wird, in dem die Erkennungsdaten gespeichert sind. Die TADDM-Berichte verwenden keine gemeinsame Datenquelle. Führen Sie daher die folgenden Schritte aus, um Datenquelleneigenschaften aller TADDM-Berichte zu konfigurieren.

Vorgehensweise

Gehen Sie wie folgt vor, um JDBC-Datenquellen für Tivoli Common Reporting zu konfigurieren:

1. Führen Sie bei Verwendung von Tivoli Common Reporting 1.3 die folgenden Schritte aus:

a) Klicken Sie in der Tivoli Common Reporting-Tabelle **Berichte** mit der rechten Maustaste auf den TADDM-Bericht, den Sie konfigurieren möchten.

b) Klicken Sie im Kontextmenü auf **Data Sources** (Datenquellen).

c) Geben Sie im Fenster **Report Data Sources** (Berichtsdatenquellen) den JDBC-Treiber, die URL, die Benutzer-ID und das Kennwort ein.

Die korrekten Werte für diese Einstellungen finden Sie in der Datei `collation.properties` im Verzeichnis `$COLLATION_HOME/etc`.

d) Wiederholen Sie die vorherigen Schritte für jeden zu konfigurierenden TADDM-Bericht.

2. Führen Sie bei Verwendung von Tivoli Common Reporting 2.1 oder 3.1 folgende Schritte aus:

a) Öffnen Sie eine Befehlszeile und wechseln Sie in das Verzeichnis `tip_install_dir/tip21Components/TCRComponent/bin` für Tivoli Common Reporting 2.1 oder in das Verzeichnis `JazzSM_install_dir/reporting/bin` für Tivoli Common Reporting 3.1.

b) Wenn Sie alle JDBC-Quellen für alle Berichte konfigurieren möchten, müssen Sie den Befehl **modify** in einer Zeile eingeben:

Wichtig: Die folgenden Befehle enthalten den Namen des Verzeichnisses, das BIRT-Berichte enthält, `'IBM Tivoli Products'`. Dieser Name gilt für TADDM 7.3.0.1 und höher. Wenn Sie TADDM 7.3.0 verwenden, ersetzen Sie diesen Namen durch `'Tivoli Products'`.

```
trcmd -user Benutzer-ID -password Kennwort -modify
-datasources -reports -reportname "/content/package[@name='IBM Tivoli
Products']/folder[@name='TADDM Reports']//report" -setdatasource
odaDriverClass=Treiberklasse odaURL=jdbcUrl
odaUser=Datenbankbenutzer odaPassword=Datenbankkennwort
```

Wenn Sie beispielsweise eine DB2-Datenbank verwenden, geben Sie den folgenden Befehl in einer Zeile ein:

```
trcmd -user tipadmin -password tipadmin -modify -datasources -reports
-reportname "/content/package[@name='IBM Tivoli Products']/folder[@name=
'TADDM Reports']//report" -setdatasource
odaDriverClass=com.ibm.db2.jcc.DB2Driver
odaURL=jdbc:db2://100.101.102.103:50000/SAMPLEDB
odaUser=db2inst1 odaPassword=db2inst1
```

Wenn Sie beispielsweise eine Oracle-Datenbank verwenden, geben Sie den folgenden Befehl in einer Zeile ein:

```
trcmd -user tipadmin -password tipadmin -modify -datasources -reports
-reportname "/content/package[@name='IBM Tivoli Products']/folder[@name=
'TADDM Reports']//report" -setdatasource
odaDriverClass=oracle.jdbc.driver.OracleDriver
odaURL=jdbc:oracle:thin:@192.168.0.1:1521:orcl
odaUser=taddm_dev odaPassword=taddm_dev
```

TADDM-Berichte überprüfen

Sie können überprüfen, ob TADDM-Berichte korrekt in Tivoli Common Reporting angezeigt werden.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um zu überprüfen, ob TADDM-Berichte in Tivoli Common Reporting korrekt angezeigt werden:

1. Öffnen Sie die Tivoli Common Reporting-Homepage.
2. Klicken Sie auf **Berichterstellung** > **Common Reporting**.
3. Stellen Sie sicher, dass die Ordner **TADDM** und **Tivoli Products** (Tivoli-Produkte) angezeigt werden.
4. Klicken Sie auf **TADDM**.
5. Klicken Sie auf das Symbol **Ausführen**, um einen der Berichte auszuführen.
   Der Bericht wird angezeigt.
6. Überprüfen Sie, ob der Bericht korrekt und vollständig angezeigt wird.

7. Kehren Sie über den Navigationspfad zu **Allgemein zugängliche Ordner** zurück.
8. Klicken Sie auf **Tivoli-Produkte** > **TADDM-Berichte**. Klicken Sie auf das Symbol **Ausführen**, um einen der Berichte auszuführen.

   Der Bericht wird angezeigt.
9. Überprüfen Sie, ob der Bericht korrekt und vollständig angezeigt wird.

Berichterstellung mit BIRT

Mit der Berichtsfunktion Business Intelligence and Reporting Tools (BIRT) können Sie vordefinierte und angepasste Berichte auf Basis der Daten aus der TADDM-Datenbank ausführen.

Übersicht über die BIRT-Berichterstellung

Zusätzlich zu den integrierten Berichten, die über das Datenmanagementportal verfügbar sind, können Sie Berichte auf Basis des Open-Source-Systems Business Intelligence and Reporting Tools (BIRT) entwerfen, entwickeln und installieren.

Wichtig: Die Anzeige der BIRT-Berichte im BIRT Report Viewer (BIRT-Laufzeitengine) des Data Managemenet Portal ist nicht sicher und daher inaktiviert. Vielmehr wird empfohlen, die BIRT-Berichte nach dem Import der TADDM-Berichte in TCR in Tivoli Common Reporting (TCR) anzuzeigen.

Wenn Sie sich der Risiken des BIRT Report Viewer bewusst sind und diese eingehen möchten, können Sie den Viewer wiederherstellen und ihn, wie in den folgenden Abschnitten beschrieben, verwenden.

TADDM enthält die Open-Source-BIRT-Laufzeitengine als integrierte Komponente. Außerdem enthält TADDM Hunderte von vordefinierten Datenbankansichten und vordefinierte Berichte. Zusätzlich zu den vordefinierten Berichten können Sie auch das BIRT Designer-Tool zum Erstellen neuer Berichte für die Verwendung mit der TADDM-BIRT-Laufzeitengine verwenden. Diese Berichte können JDBC-Datenquellen verwenden, die Daten mit vordefinierten Datenbankansichten extrahieren.

Die Datenmanagementportal-Schnittstelle bietet Möglichkeiten für die Verwaltung dieser BIRT-Berichte. Sie können neue Berichte hinzufügen, ausgewählte Berichte herunterladen, hochgeladene Berichte löschen und Berichte ausführen. Die vordefinierten Berichte sind ebenfalls gepackt, sodass sie zusammen mit dem Tool Tivoli Common Reporting verwendet werden können.

Business Intelligence and Reporting Tools (BIRT)

Business Intelligence and Reporting Tools (BIRT) ist ein auf Eclipse basierendes Open-Source-System zum Entwerfen, Entwickeln und Ausführen von Berichten. Sie können BIRT-Berichte für TADDM entwickeln und so entwerfen, dass sie JDBC-Datenbankquellen und SQL-Abfragen von vordefinierten Datenbankansichten verwenden.

Wichtig: BIRT-Berichte dürfen keine Daten verwenden, die direkt den TADDM-Datenbanktabellen entnommen wurden. Entwerfen Sie Ihre Berichte stattdessen immer so, dass sie eine JDBC-Datenquelle und die TADDM-Datenbankansichten verwenden, die im TADDM *SDK Developer's Guide* dokumentiert sind.

Das BIRT-System enthält folgende zwei Hauptkomponenten:

- Den BIRT Designer, ein grafisches Tool für die Gestaltung und Entwicklung neuer Berichte
- Die BIRT-Laufzeitengine, die Unterstützung für die Ausführung von Berichten und die Wiedergabe der veröffentlichten Berichtsausgabe bereitstellt

TADDM enthält die BIRT-Laufzeitengine, die Sie für die Ausführung vordefinierter Berichte verwenden können. Wenn Sie Ihre BIRT-Berichte selbst erstellen möchten, müssen Sie das BIRT Designer-Tool herunterladen, das der Version der BIRT-Laufzeitengine entspricht, die im Lieferumfang von TADDM enthalten ist (derzeit Version 2.2.1).

Weitere Informationen zum BIRT-Projekt, u. a. zum Herunterladen des BIRT Designer-Tools, finden Sie auf folgender Website: http://www.eclipse.org/birt.

Zugehörige Tasks

„BIRT Report Viewer wiederherstellen” auf Seite 173

Wenn Sie sich der Sicherheitsrisiken des BIRT Report Viewer bewusst sind und diese eingehen möchten, können Sie den Viewer wiederherstellen.

Vordefinierte BIRT-Berichte
Die vordefinierten BIRT-Berichte, die im Lieferumfang von TADDM enthalten sind, stellen Informationen zu erkannten Systemen, Betriebssystemen und Serverprozessen bereit.

Anwendungsserver - Bestandsbericht
Der Bestandsbericht für Anwendungsserver enthält alle Anwendungsserver, die von TADDM erkannt wurden. Wenn Sie den Bericht ausführen, können Sie einen Parameterwert angeben, der den Bericht auf Anwendungsserver eines bestimmten Typs begrenzt. Der Bericht gruppiert die erkannten Anwendungsserver nach System und listet sie nach dem vollständig qualifizierten Hostnamen auf.

Die Daten für diesen Bericht werden der Datenbankansicht CM_APP_SERVERS_PER_HOST_V entnommen.

Computersystem-Bestandsbericht
Der Computersystem-Bestandsbericht enthält alle Computersysteme in der TADDM-Datenbank, denen IP-Adressen zugeordnet wurden. Die Computersysteme werden nach dem vollständig qualifizierten Hostnamen aufgelistet. Für diesen Bericht gibt es keine Parameter.

Dieser Bericht ist so konzipiert, dass er in eine Datei mit durch Kommas getrennten Werten exportiert werden kann, die in eine Tabellenkalkulationsanwendung importiert werden kann. Wenn ein System keine IP-Adresse hat, ist es in dem Bericht nicht enthalten. Es ist möglich, dass derselbe Computersystemname mehrmals im Bericht aufgelistet ist, einmal für jede eindeutige IP-Adresse (einschließlich der Loopback-Adresse 127.0.0.1).

Die Daten für diesen Bericht werden der Datenbankansicht CM_COMPUTER_SYSTEMS_V entnommen.

Computersystem - Bestandsbericht nach Betriebssystemtyp
Der Bestandsbericht nach Betriebssystemtyp für Computersysteme enthält alle erkannten Computersysteme, deren Betriebssysteme ebenfalls erkannt wurden. Für diesen Bericht gibt es keine Parameter.

Dieser Bericht ist so konzipiert, dass er in eine Datei mit durch Kommas getrennten Werten exportiert werden kann, die in eine Tabellenkalkulationsanwendung importiert werden kann. Es ist möglich, dass derselbe Computersystemname mehrmals im Bericht aufgelistet ist, einmal für jede eindeutige IP-Adresse (einschließlich der Loopback-Adresse 127.0.0.1). Um in diesem Bericht enthalten zu sein, muss ein Betriebssystem einem System in der TADDM-Datenbank zugeordnet sein. Ebenso ist ein System, für das in der TADDM-Datenbank kein Betriebssystem definiert ist, nicht enthalten.

Klicken Sie auf den Namen eines Systems im Bericht, um einen Drillthrough-Bestandsdetailbericht für dieses System zu öffnen.

Die Daten für diesen Bericht werden folgenden Datenbankansichten entnommen:

- DP_UNITARY_COMP_GENERAL_V
- DP_UNITARY_COMP_OS_V
- DP_UNITARY_COMP_IP_INTERFACE_V
- BB_OPERATINGSYSTEM62_V

ITNM-IP-Bericht
Dieser Bericht enthält Informationen zu den installierten Instanzen des Network Manager-Produkts und listet alle Network Manager-Ressourcen auf, die eine Beziehung zu einem Computersystem aufweisen.

Der Network Manager-Bestandsbericht ist in der Konsole des TADDM-Domänenmanagers verfügbar. Der Bericht enthält folgende Abschnitte:

Serverübersicht
Hier finden Sie Informationen zu den installierten Instanzen des Network Manager-Produkts einschließlich der installierten Network Manager-Versionen, der Host-Adressen der Server, auf denen Network Manager installiert ist, und der URLs für den Zugriff auf die grafische Benutzeroberfläche von Network Manager.

Ressourcenübersicht
Hier finden Sie eine Auflistung aller Network Manager-Ressourcen mit einer Beziehung zu einem Computersystem. Dabei sind auch Informationen zur IP-Adresse, zum Hersteller, zum Ressourcentyp (z. B. Router) sowie zur eindeutigen Kennung in der Network Manager-Datenbank angegeben.

Kurzer Computersystem-Bestandsbericht

Im kurzen Computersystem-Bestandsbericht können Sie die IP-Adressen anzeigen, die mit einem Erkennungsprofil der Ebene 1 erkannt wurden. Für jede IP-Adresse wird im Bericht auch der zugehörige Computersystemname sowie der Betriebssystem- oder Steuerungssoftwarename angezeigt (wenn diese Informationen erkannt wurden).

Der kurze Computersystem-Bestandsbericht ist zwar für die Verwendung nach einer Erkennung der Ebene 1 konzipiert, Sie können ihn jedoch auch nach einer Erkennung der Ebene 3 verwenden. Andere Berichte wie der Computersystem-Bestandsbericht enthalten jedoch nach einer Erkennung mit Berechtigungsnachweisen detailliertere Informationen.

Fibre Channel-Netzbericht

Der Fibre Channel-Netzbericht zeigt Fibre Channel-Verbindungen zwischen einem ausgewählten Fibre Channel-Switch und anderen Computersystemen an.

Geben Sie zum Ausführen des Berichts den weltweiten Namen (WWN) des Fibre Channel-Switch an, um die Fibre Channel-Verbindungen zwischen diesem Switch und anderen Computersystemen anzuzeigen. Geben Sie den Namen im Fenster **Parameter** ein oder wählen Sie ihn aus der Dropdown-Liste der erkannten Fibre Channel-Switches aus.

Der Bericht zeigt folgende Informationen zu jedem verbundenen Computersystem an:

- Computersystem (Anzeigename; WWN im Fall von Fibre Channel-Switches)
- Hersteller
- Modell
- Seriennummer

Klicken Sie im Bericht auf den Anzeigenamen eines Computersystems, um einen weiteren Fibre Channel-Netzbericht zu öffnen. Dieser Bericht zeigt Fibre Channel-Verbindungen zwischen dem ausgewählten Computersystem und anderen Computersystemen an.

Hostbusadapter-Bestandsbericht

Der Hostbusadapter-Bestandsbericht zeigt eine Liste aller erkannten Hostbusadapter und der Computersysteme an, auf denen sie installiert sind.

Zu jedem erkannten Hostbusadapter werden folgende Informationen im Bericht angezeigt:

Hostbusadaptername
Der Name des Hostbusadapters.

Vollständig qualifizierter Domänenname
Der vollständig qualifizierte Domänenname des Computersystems, auf dem der Hostbusadapter installiert ist.

Host verwendet Speicherbereiche
Ein boolescher Wert, der anzeigt, ob das Host-Computersystem Speicherdatenträger in einem Speicherbereich verwendet.

Bestandsübersicht

Die Bestandsübersicht enthält ein Kreisdiagramm mit den Betriebssystemen, die auf erkannten Computersystemen installiert sind, basierend auf den von TADDM erkannten Bereichen. Jedes Segment des Diagramms stellt einen Betriebssystemtyp dar und gibt die Gesamtzahl der erkannten Server an, die dieses Betriebssystem ausführen. Für diesen Bericht gibt es keine Parameter.

Klicken Sie auf ein Segment des Diagramms, um einen Drillthrough-Computersystem-Bestandsbericht für den ausgewählten Betriebssystemtyp zu öffnen.

Die Daten für diesen Bericht werden der Datenbankansicht BB_OPERATINGSYSTEM62_V entnommen.

Überwachungsabdeckungsberichte

Die Überwachungsabdeckungsberichte enthalten Details zu verschiedenen Komponenten in Ihrer Umgebung. Sie können einen Bericht für Betriebssysteme, Datenbanken, Microsoft-Anwendungen, VMware-Server und System p-Komponenten in Ihrer Umgebung erstellen. Diese Komponenten werden durch Agenten von IBM Tivoli Monitoring 6.1 oder höher überwacht. Sie können diesen Bericht im Teilefenster 'BIRT-Berichte' des Datenmanagementportals ausführen.

Tabelle 38 auf Seite 161 listet die verfügbaren Abdeckungsberichte auf. Die Überwachungsabdeckung für Betriebssystemberichte kann durch den IBM® Tivoli® Monitoring Scope-Sensor ausgefüllt werden. Zum Ausfüllen aller anderen Berichte ist hingegen der IBM® Tivoli® Monitoring-Erkennungsbibliotheksadapter (DLA; Discovery Library Adapter) erforderlich.

Die Berichte bestehen aus drei Abschnitten:

Abdeckung nach Typ

In diesem Abschnitt wird die Anzahl der überwachten, unüberwachten und gesamten Instanzen gruppiert nach Berichtstyp angezeigt. Das Fenster **Abdeckungsdetails** enthält eine grafische Darstellung der folgenden Statistikdaten:

- Abdeckung gesamt
- Abdeckung nach Plattform

Abdeckungsdetails

In diesem Abschnitt werden der vollständig qualifizierte Domänenname, der Name des verwalteten Systems und der Überwachungsstatus gruppiert nach Berichtstyp angezeigt. Der Überwachungsstatus wird zusammen mit Angaben zur Agentenversion, falls überwacht, aufgeführt. Klicken Sie auf den Namen des verwalteten System, um das Fenster **Agentendetails** zu öffnen.

Agentendetails

Dieser Abschnitt zeigt ausführliche Informationen zum Agenten und dem Betriebssystem an, das darauf aufgeführt wird. Die angezeigten Informationen richten sich danach, ob der Agent überwacht oder unüberwacht ist. Informationen zu Affinität und Quellentoken sind gemeinsam mit einem Launch-in-Context-Link zur Tivoli Enterprise Portal-Ansicht von IBM Tivoli Monitoring enthalten.

Der Management-Software-System-Abschnitt listet den Bestand der installierten IBM Tivoli Monitoring-Agenten auf und enthält einen Launch-in-Context-Link zu Arbeitsbereichen in IBM Tivoli Monitoring. Die Überwachungsabdeckungszusammenfassung stellt eine Liste der überwachten und unüberwachten Systeme bereit, die dem Benutzer die Überwachung und Verwaltung der Überwachungsagenten erleichtert.

Bei einer Level-1-Erkennung kann zum Ausfüllen der Überwachungsabdeckung von Betriebssystemberichten der IBM Tivoli Monitoring Scope-Sensor verwendet werden. Zum Ausfüllen aller anderen Berichte muss der IBM Tivoli Monitoring-Erkennungsbibliotheksadapter (DLA) verwendet werden. Informationen zum IBM Tivoli Monitoring-DLA finden Sie im TADDM-*Administratorhandbuch*.

Tabelle 38 auf Seite 161 listet die verfügbaren Abdeckungsberichte auf.

Tabelle 38. Überwachungsabdeckungsberichte

Berichtsname	**Beschreibung**
Überwachungsabdeckung für Betriebssysteme	Dieser Bericht zeigt die Betriebssystemdetails Ihrer Umgebung.
Überwachungsabdeckung für Datenbanken	Dieser Bericht zeigt Details zur DB2-Instanz und zum SQL-Server Ihrer Umgebung.
Überwachungsabdeckung für Microsoft-Anwendungen	Dieser Bericht zeigt Details zu Active Directory sowie zum Cluster-Server, Exchange-Server, Host-Integrationsserver, Internet Information Services-Server und zur aktivierten Rolle des Hyper-V-Servers.
Überwachungsabdeckung für VMware	Dieser Bericht zeigt Details zu VMware ESX-Servern und VMware Virtual Center-Servern.

Tabelle 38. Überwachungsabdeckungsberichte (Forts.)	
Berichtsname	**Beschreibung**
Überwachungsabdeckung für System p	Dieser Bericht zeigt Details zu System p, zur Hardware Management Console, zum virtuellen E/A-Server und zu logischen AIX-Partitionen.

Sensorberichte

Die vordefinierten Sensorberichte sortieren die zu Sensorkennzahlen zusammengestellten Informationen.

In Tabelle 39 auf Seite 162 sind die verfügbaren vordefinierten Sensorberichte aufgelistet.

Tabelle 39. Vordefinierte Sensorberichte	
Berichtsname	**Beschreibung**
TADDM_SENSORS_WEEKLY_METRICS_ALL TADDM_SENSORS_WEEKLY_METRICS	Dieser Bericht zeigt die wöchentliche Erfolgsrate in Prozent für Sensoren, die in einem Erkennungsprofil der Ebene 1, Ebene 2 oder Ebene 3 aktiviert sind. Die folgenden Informationen werden angezeigt: • Datum • Ebene 1 (L1) % Erfolg • Ebene 2 (L2) % Erfolg • Ebene 3 (L3) % Erfolg • L1, L2 % Erfolg • Alle % Erfolg Der zweite Bericht, "TADDM_SENSORS_WEEKLY_METRICS", enthält dieselben Informationen, stellt sie jedoch in einem Balkendiagramm dar.
TADDM_SENSORS_SUMMARY_TOTAL TADDM_SENSORS_SUMMARY	Dieser Bericht zeigt die Gesamtzahl der Sensoren, die erfolgreich ausgeführt wurden. Die folgenden Informationen werden angezeigt: • Ebene • Ausführungen mit gespeicherten Konfigurationselementen • Erfolge • Fehlschläge Zusätzlich zeigt eine Zusammenfassung die Ebenen der Erkennungsprofile sowie die Erfolgs- und Fehlerraten insgesamt für jede Ebene in Prozent an. Der Bericht "TADDM_SENSORS_SUMMARY" zeigt die Erfolgs- bzw. Fehlerrate für einzelne Sensoren während einer Erkennung in Prozent an. Die folgenden Informationen werden angezeigt: • Ebene • Sensor • Verlegungsreservierungen • Erfolge • Fehlschläge • % Erfolg • % Fehler

Tabelle 39. Vordefinierte Sensorberichte (Forts.)

Berichtsname	Beschreibung
TADDM_SENSORS_SERVER_SCANS_IP	Dieser Bericht zeigt den Status an, nachdem ein Server durch Angabe seiner IP-Adresse gescannt wurde. Die folgenden Informationen werden angezeigt: • Woche • Status Am Anfang des Berichts stehen zusammenfassende Informationen zur IP-Adresse, zum Hostnamen, zum vollständig qualifizierten Domänennamen sowie zum ersten und letzten Scandatum und zum ersten und letzten Status.
TADDM_SENSORS_SERVER_SCANS_ HOSTNAME	Dieser Bericht zeigt den Status an, nachdem ein Server durch Angabe des Hostnamens gescannt wurde. Die folgenden Informationen werden angezeigt: • Woche • Status Am Anfang des Berichts stehen zusammenfassende Informationen zum Hostnamen, zur IP-Adresse, zum vollständig qualifizierten Domänennamen sowie zum ersten und letzten Scandatum und zum ersten und letzten Status.
TADDM_SENSORS_MONTHLY_COVERAGE	Dieser Bericht zeigt ein Balkendiagramm mit der monatlichen Abdeckung des Sitzungssensors. Er enthält Informationen zur Anzahl der Scanläufe und zur Anzahl der erfolgreichen bzw. nicht erfolgreichen Scans. Der Sitzungssensor baut eine Sitzung zwischen dem TADDM-Server und dem Zielcomputersystem auf.
TADDM_SENSORS_METRICS_LEVEL_1_ AND_2 TADDM_SENSORS METRICS_LEVEL3	Dieser Bericht enthält ein Balkendiagramm, das für eine angegebene Woche die Erfolgsrate für einzelne Sensoren in Prozent anzeigt, wenn eine Erkennung auf Ebene 1 und Ebene 2 durchgeführt wird. Das Balkendiagramm für den Bericht "TADDM_SENSORS METRICS_LEVEL3" zeigt die Kennzahlen für einzelne Sensoren, wenn eine Erkennung der Ebene 3 durchgeführt wird.
TADDM_SENSORS_FAILED_LEVELS_1_2_3 TADDM_SENSORS_FAILED_LEVEL	Dieser Bericht enthält ein Kreisdiagramm für eine bestimmte Woche, basierend auf Fehlern beim Ausführen einer Erkennung auf Ebene 1, Ebene 2 oder Ebene 3. Die einzelnen Segmente des Diagramms stellen jeweils Sitzungsprobleme, Sensorprobleme, Verbindungsprobleme und andere Probleme dar. Das Kreisdiagramm für den Bericht "TADDM_SENSORS_FAILED_LEVEL" zeigt die Kennzahlen für eine angegebene Erkennungsebene an.

Tabelle 39. Vordefinierte Sensorberichte (Forts.)	
Berichtsname	**Beschreibung**
TADDM_SENSORS_EVENTS_SENSOR_IP TADDM_SENSORS_EVENTS_SENSOR TADDM_SENSORS_EVENTS_IP TADDM_SENSORS_DONE_EVENTS_RUN	Dieser Bericht zeigt die Ereignisdaten für einen angegebenen Sensor und eine angegebene IP-Adresse an. Die folgenden Informationen werden angezeigt: • Datum • Sensordetails • Schweregrad • Beschreibung Der Bericht "TADDM_SENSORS_EVENTS_SENSOR" enthält dieselben Informationen, zeigt jedoch die Ereignisdaten für einen angegebenen Sensor an. Der Bericht "TADDM_SENSORS_EVENTS_IP" enthält dieselben Informationen, zeigt jedoch die Ereignisdaten für eine angegebene IP-Adresse an. Der Bericht "TADDM_SENSORS_DONE_EVENTS_RUN" enthält dieselben Informationen, zeigt jedoch die Ereignisdaten für einen angegebenen Erkennungslauf an.

Serveraffinität nach Bereich

Im Bericht 'Serveraffinität nach Bereich' werden Beziehungen zwischen Servern angezeigt, die der Quelle und dem Ziel der einzelnen Beziehungen entsprechend angeordnet sind. In der ersten Tabelle werden alle Server im angegebenen Bereich angezeigt, die Quellen von Beziehungen darstellen, sowie die Verbindungen von diesen Servern zu anderen Servern. In der zweiten Tabelle werden alle Server im angegebenen Bereich angezeigt, die Ziele von Beziehungen darstellen, sowie die Verbindungen von anderen Servern zu diesen Servern.

Der Bericht 'Serveraffinität nach Bereich' steht nur in Domänenserverimplementierungen zur Verfügung.

Um ein Diagramm anzuzeigen, in dem die Kommunikation zwischen Servern dargestellt ist, klicken Sie auf **Affinitätsdiagramm starten**. Im Diagramm werden transaktionsorientierte Abhängigkeiten und Serviceabhängigkeiten zwischen Computersystemen angezeigt, wobei Abhängigkeiten durch Verbindungslinien zwischen Systemen dargestellt werden. Das Diagramm enthält alle Verbindungslinien für Abhängigkeiten, die mindestens ein System im Erkennungsbereich enthalten, wobei Systeme, die Mitglieder des Bereichs sind, gelb hervorgehoben sind.

Im Affinitätsdiagramm angezeigte Links können entweder transaktionsorientierte Beziehungen oder Servicebeziehungen darstellen. Die Richtung einer Verbindungslinie gibt an, welches System die Quelle und welches System das Ziel der Abhängigkeitsbeziehung ist. Die Quellen- und Zielobjekte können je nach Beziehung verschiedenen Typen zugeordnet werden:

- Computersystem
- Anwendungsserver
- Service

Verbindungslinien im Diagramm werden immer zwischen Computersystemen gezogen. Bei einer Beziehung, die einen Anwendungsserver oder Service einschließt, verläuft die Verbindungslinie zum Hostcomputersystem hin. Um weitere Informationen zu einer Abhängigkeitsbeziehung anzuzeigen (einschließlich Quelle, Ziel, Befehlsnamen und beteiligter Portnummer), bewegen Sie Ihren Mauszeiger zu der Verbindungslinie im Diagramm.

Der Bericht 'Serveraffinität nach Bereich' kann nicht in Tivoli Common Reporting importiert werden.

Momentaufnahmeberichte

Vordefinierte Momentaufnahmeberichte fassen die in einer oder mehreren Momentaufnahmen erfassten Informationen zusammen.

Eine Momentaufnahme ist eine Kopie der erkannten Computerinformationen, die zu einem bestimmten Zeitpunkt aufgenommen wurde. Weitere Informationen zum Erstellen von Momentaufnahmen finden Sie im Abschnitt „Momentaufnahmetool verwenden" auf Seite 173.

Der jeweilige Berichtsname richtet sich nach dem Server, den Sie ausführen und mit dem BIRT-Berichte angezeigt werden. Wenn Sie das Datenmanagementportal im Domänen- oder Speicherserver verwenden, führen Sie den Standardbericht aus, z. B. TADDM_SNAPSHOT_CHANGE. Wenn Sie das Datenmanagementportal im Synchronisationsserver verwenden, führen Sie den Bericht mit "SYNC" im Namen aus, z. B. TADDM_SNAPSHOT_SYNC_CHANGE. Bei den folgenden Berichten handelt es sich um Ausnahmen, die auf allen Servern denselben Namen haben:

- TADDM_SNAPSHOT_FRAME
- TADDM_SNAPSHOT_HOST

Wenn Sie einen BIRT-Bericht in Tivoli Common Reporting importieren, wird ein geänderter Berichtsname angezeigt. Der Bericht TADDM_SNAPSHOT_SYNC_SESSION_FAILED wird beispielsweise als "TADDM: Details about failed sessions (Enterprise)" (TADDM: Details zu fehlgeschlagenen Sitzungen (Unternehmen) angezeigt.

In Tabelle 40 auf Seite 165 sind die verfügbaren vordefinierten Momentaufnahmeberichte aufgelistet.

Tabelle 40. Vordefinierte Momentaufnahmeberichte

Berichtsname	**Beschreibung**
TADDM_SNAPSHOT_FRAME	Zeigt die folgenden detaillierten Informationen zu erkannten Servern an: • Rahmenname • Seriennummer • Hersteller • Modell • CPU-Typ • CPU-Geschwindigkeit • Anzahl der CPUs • Hauptspeicher • Position • Unterstützungsbereich • Letzte Erkennung
TADDM_SNAPSHOT_HOST	Zeigt die folgenden detaillierten Informationen zu physischen und virtuellen Servern an: • Rahmenname • Systemname • IP-Adresse • Betriebssystemtyp • Hosttyp • Name des verwalteten Systems • Letzte Erkennung

Tabelle 40. Vordefinierte Momentaufnahmeberichte (Forts.)

Berichtsname	**Beschreibung**
TADDM_SNAPSHOT_SESSION_FAILED TADDM_SNAPSHOT_SYNC_SESSION _FAILED	Zeigt Namens- und IP-Adresseninformationen zu erkannten Servern an, für die TADDM aufgrund fehlgeschlagener Sitzungen keine L2-Informationen abrufen konnte.
TADDM_SNAPSHOT_CHANGE TADDM_SNAPSHOT_SYNC_CHANGE	Vergleicht zwei Momentaufnahmen, die zu verschiedenen Zeitpunkten erstellt wurden. Für jeden Server, der in der Zeit zwischen den beiden Momentaufnahmen hinzugefügt oder entfernt wurde, zeigt er folgende Informationen an: • Name • IP-Adresse • virtuell Er zeigt auch Informationen zur Änderung des Verhältnisses von physischen zu virtuellen Servern in der Zeit zwischen den beiden Momentaufnahmen an.
TADDM_SNAPSHOT_DISCOVERY_ERROR TADDM_SNAPSHOT_SYNC _DISCOVERY_ERROR	Zeigt Informationen zu Fehlern an, die während Erkennungen generiert wurden.
TADDM_SNAPSHOT_FQDN_OS _CHANGES TADDM_SNAPSHOT_SYNC_FQDN _OS_CHANGES	Zeigt Informationen zu Servern mit in der Zeit zwischen zwei Momentaufnahmen geändertem vollständig qualifizierten Domänennamen oder geänderten Betriebssysteminformationen an.
TADDM_SNAPSHOT_REFERENCE TADDM_SNAPSHOT_SYNC_REFERENCE	Vergleicht eine Momentaufnahme mit einer Referenzliste. Er zeigt Informationen zu den Servern an, die sich in der Referenzliste, jedoch nicht in der Momentaufnahme befinden, und zu den Servern, die sich in der Momentaufnahme, jedoch nicht in der Referenzliste befinden.
TADDM_SNAPSHOT_RECONCILIATION _SUMMARY TADDM_SNAPSHOT_SYNC _RECONCILIATION_SUMMARY	Fordert Sie auf, eine Momentaufnahme zu erstellen, und zeigt folgende Übersichtsdaten zu erkannten Servern an: • Ausgangshostname • Ausgangs-IP-Adresse • TADDM-Hostname • TADDM-IP-Adresse • Status • Fehlerursache

Tabelle 40. Vordefinierte Momentaufnahmeberichte (Forts.)	
Berichtsname	**Beschreibung**
TADDM_SNAPSHOT_RECONCILIATION _DETAIL TADDM_SNAPSHOT_SYNC _RECONCILIATI-ON_DETAIL	Fordert Sie auf, eine Momentaufnahme zu erstellen, und zeigt die folgenden detaillierten Informationen zu erkannten Servern an: • Ausgangshostname • Ausgangs-IP-Adresse • TADDM-Hostname • TADDM-IP-Adresse • Status • Fehlerursache • Fehlerbeschreibung • Bereichsname • gefilterte Ausnahme • TADDM-Rahmen • TADDM-Hostname • vollständig qualifizierter TADDM-Domänenname • TADDM-Name • TADDM-Anzeigename • TADDM-JDO-Klasse • TADDM - abgeleitetes Betriebssystem • TADDM-Betriebssystemname • TADDM-IP-Adresse • TADDM-Seriennummer • TADDM-Hersteller • TADDM-Modell • TADDM-Hosttyp • TADDM - virtuell • TADDM-Typ • TADDM-Erkennungsdatum

Bericht über Speicherbereiche nach Host

Der Bericht über Speicherbereiche nach Host enthält eine Liste der Speicherdatenträger und Speicherbereiche, die von einem angegebenen Computersystem verwendet werden.

Beim Ausführen des Berichts werden Sie aufgefordert, den Hostnamen des Computersystems anzugeben, dessen Speicherinformationen angezeigt werden sollen. Geben Sie den Hostnamen im Fenster **Parameter** ein, oder wählen Sie ihn aus der Dropdown-Liste aus.

Folgende Informationen werden im Bericht angezeigt:

- Speicherdatenträger
- Speicherbereich
- Hersteller
- Modell
- Seriennummer
- Verfügbare Kapazität

- Zugeordnete Kapazität

Speicherbereich-Konsumentenbericht

Der Speicherbereich-Konsumentenbericht enthält eine Liste der Computersysteme und Anwendungsserver, die einen bestimmten Speicherbereich verwenden.

Beim Ausführen des Berichts werden Sie aufgefordert, den Namen eines Speicherbereichs anzugeben. Geben Sie den Namen des Speicherbereichs im Fenster **Parameter** ein, oder wählen Sie ihn aus der Dropdown-Liste aus.

Der Bericht wird in Form der folgenden drei Tabellen angezeigt:

Computersysteme, die den Speicherbereich *Speicherbereich_Name* verwenden
In dieser Tabelle sind alle erkannten Computersysteme aufgeführt, die den angegebenen Speicherbereich verwenden.

Anwendungsserver, die den Speicherbereich *Speicherbereichsname* verwenden
In dieser Tabelle sind alle erkannten Anwendungsserver aufgeführt, die den angegebenen Speicherbereich verwenden.

Geschäftsanwendungen, die den Speicherbereich *Speicherbereichsname* verwenden
In dieser Tabelle sind alle erkannten Geschäftsanwendungen aufgeführt, die den angegebenen Speicherbereich verwenden.

Systemverbindungs-Topologiebericht

Der Systemverbindungs-Topologiebericht stellt einen Textbericht für Computersysteme mit Netzverbindungen zu oder von anderen Computersystemen dar. Wenn Sie den Bericht ausführen, müssen Sie das Konfigurationselement eingeben, für das der Bericht ausgeführt werden soll, und Sie müssen angeben, ob es sich um ein Computersystem oder eine Geschäftsanwendung handelt.

Wenn der Bericht für ein Computersystem ausgeführt wird, werden alle Computersysteme mit Netzverbindungen zu oder von dem ausgewählten Computersystem zusammen mit Messwerten für die einzelnen Netzverbindungen in einer Tabelle angezeigt. Wenn der Bericht für eine Geschäftsanwendung ausgeführt wird, werden alle Computersysteme mit Netzverbindungen zu oder von der ausgewählten Geschäftsanwendung in einer Tabelle angezeigt.

Sie können die Systemverbindungstopologie für jedes Computersystem anzeigen, indem Sie im Bericht auf den Namen des Systems klicken.

Serverauslastung - Bericht der Stundenspitzenwerte

In diesem Bericht werden Stundenspitzenwerte für die Systemauslastung für Systeme im angegebenen Bereich am angegebenen Datum angezeigt.

Die Auslastungsmesswerte umfassen folgende Informationen:

- Stundenwert für die 95-prozentige CPU-Auslastung
- Stundenspitzenwert für die Speicherauslastung in Prozent
- Stundenspitzenwert für die Netzbandbreitenauslastung
- Stundenspitzenwert für die Platten-E/A-Auslastung

Systemauslastungsbericht

In diesem Bericht werden die Konfiguration des generischen Serverbetriebssystems und zugehörige Auslastungsinformationen angezeigt.

Die Konfigurationsdaten für das Serverbetriebssystem umfassen folgende Informationen:

- CPU
- Hauptspeicher
- Dateisystem

Es handelt sich um die neuesten Konfigurationsdaten, die TADDM zur Verfügung stehen. Die Serverauslastungsdaten umfassen folgende Informationen:

- CPU
- Hauptspeicher
- Netz
- Platte

Unbekannte Server - Bericht

Der Bericht für unbekannte Server enthält alle erkannten Serverprozesse, die von TADDM nicht erkannt wurden. Der Bericht gruppiert die erkannten Serverprozesse nach System und listet sie nach dem vollständig qualifizierten Hostnamen auf. Für diesen Bericht gibt es keine Parameter.

Unbekannte Server werden nach einer Erkennung von einem Topologieerstellungsagenten ermittelt. Der Topologieerstellungsagent wird in durch die konfigurierte Frequenz vorgegebenen regelmäßigen Abständen im Hintergrund ausgeführt, es kann also sein, dass unbekannte Server nicht unmittelbar nach Abschluss einer Erkennung erkannt werden. Standardmäßig wird der Topologieerstellungsagent alle vier Stunden ausgeführt.

Wenn Sie also den Bericht zu unbekannten Servern vor Beendigung des Agenten zur Topologieerstellung ausführen, werden in dem Bericht unter Umständen nicht alle unbekannten Server aufgeführt.

Folgende Informationen werden im Bericht angezeigt:

Name
Der Name des Computers, auf dem der unbekannte Serverprozess ausgeführt wird.

Kontext-IP
Die IP-Adresse des Computers, auf dem der unbekannte Serverprozess ausgeführt wird.

PID
Die Prozess-ID des unbekannten Serverprozesses.

PPID
Die Prozess-ID des übergeordneten Prozesses des unbekannten Serverprozesses.

Befehlszeile
Der Befehl, mit dem der unbekannte Serverprozess ausgeführt wird.

Die Daten für diesen Bericht werden der Datenbankansicht BB_RUNTIMEPROCESS15_V entnommen.

BIRT-Bericht ausführen

BIRT-Berichte können über den Abschnitt 'Analyse' des Datenmanagementportals ausgeführt werden.

Informationen zu diesem Vorgang

Wichtig: Die Ausführung eines BIRT-Berichts im Datenmanagementportal ist nur möglich, wenn der BIRT Report Viewer aktiviert ist. BIRT Report Viewer ist jedoch aus Sicherheitsgründen inaktiviert. Alternativ können die BIRT-Berichte nach dem Import der TADDM-Berichte in TCR auch in Tivoli Common Reporting (TCR) angezeigt werden. Wenn Sie sich der Risiken des BIRT Report Viewer bewusst sind und diese eingehen möchten, können Sie den BIRT Report Viewer wiederherstellen.

Vorgehensweise

Gehen Sie folgendermaßen vor, um einen BIRT-Bericht auszuführen:

1. Klicken Sie im Teilfenster 'Funktionen' auf **Analyse**.
2. Klicken Sie im Abschnitt 'Analyse' auf **BIRT-Berichte**.
   Die Liste **TADDM BIRT-Berichte** wird mit allen verfügbaren BIRT-Berichten angezeigt.
3. Klicken Sie in der Liste **TADDM BIRT-Berichte** auf den Bericht, den Sie ausführen möchten, um ihn zu markieren.
4. Optional: Geben Sie den Tagwert für die Position an.
   Der Wert `com.ibm.cdb.locationTaggingEnabled` in der Datei `COLLATION_HOME/etc/collation.propertes` muss auf `true` gesetzt werden. Es werden nur die Berichtsdaten für diesen bestimmten Positionstag angezeigt.

Anmerkung: Die in TADDM enthaltenen BIRT-Berichte unterstützen derzeit keine Positionsfilterung ohne zusätzliche Anpassung.

5. Klicken Sie auf **Bericht ausführen**.

   Wenn der Bericht Parameter hat, werden Sie anschließend aufgefordert, die Parameterwerte anzugeben. Wenn Sie mit der Angabe der Parameterwerte fertig sind, klicken Sie auf **OK**.

Ergebnisse

Der formatierte Bericht wird im Fenster 'BIRT Report Viewer' (BIRT-Berichtsanzeigefunktion) angezeigt. Klicken Sie auf die Symbole am oberen Rand des Berichts, um im Bericht vor- und zurückzublättern bzw. den Bericht zu drucken oder in eine Datei zu exportieren. Um einen Drillthrough-Bericht zu öffnen, in dem weitere Details zu einer Untergruppe der Berichtsdaten angezeigt werden, klicken Sie auf einen Link im Bericht.

Anmerkung: Exportierte Berichte im DOC-Format sind mit Microsoft Word 2003 oder höher kompatibel.

BIRT-Bericht von der Befehlszeilenschnittstelle aus ausführen

BIRT-Berichte können von der Befehlszeilenschnittstelle des TADDM-Servers aus ausgeführt werden.

Vorgehensweise

Gehen Sie wie folgt vor, um einen BIRT-Bericht von der Befehlszeilenschnittstelle aus auszuführen:

1. Öffnen Sie eine Eingabeaufforderung und navigieren Sie abhängig von der verwendeten TADDM-Version zu einem der folgenden Verzeichnisse:
   - 7.3.0: `$COLLATION_HOME/deploy-tomcat/birt-viewer/WEB-INF/resources`
   - 7.3.0.1 und höher: `$COLLATION_HOME/apps/birt-viewer/WEB-INF/resources`
2. Definieren Sie die Variable `BIRT_HOME`. Führen Sie eine der folgenden Aktionen aus:
   - Führen Sie unter Linux abhängig von der verwendeten TADDM-Version einen der folgenden Befehle aus:
     - 7.3.0:

       ```
       export BIRT_HOME=$COLLATION_HOME/deploy-tomcat/birt-viewer
       ```

     - 7.3.0.1 und höher:

       ```
       export BIRT_HOME=$COLLATION_HOME/apps/birt-viewer
       ```

   - Führen Sie unter Windows abhängig von der verwendeten TADDM-Version einen der folgenden Befehle aus:
     - 7.3.0:

       ```
       set BIRT_HOME=%COLLATION_HOME%/deploy-tomcat/birt-viewer
       ```

     - 7.3.0.1 und höher:

       ```
       set BIRT_HOME=%COLLATION_HOME%/apps/birt-viewer
       ```

3. Führen Sie den BIRT-Bericht aus. Führen Sie eine der folgenden Aktionen aus:
   - Führen Sie unter Linux folgenden Befehl aus:

     ```
     ./genReport.sh -f Format -o Ausgabe -F Parameter Bericht
     ```

   - Führen Sie den folgenden Befehl unter Windows aus:

     ```
     genReport.bat -f Format -o Ausgabe -F Parameter Bericht
     ```

   Mit dem Programm **genReport** werden folgende Befehlszeilenoptionen verwendet:

Format
Das Ausgabeformat der Berichtsdatei. Gültige Werte sind PDF und HTML.

Ausgabe
Der Pfad der zu erstellenden Berichtsdatei. Beispiele: `/home/cognos/utilization.pdf` unter Linux oder `C:\data\utilization.pdf` unter Windows.

Parameter

(Optional) Der Pfad zu einer Eigenschaftendatei, in der jede Eigenschaft einen erforderlichen Berichtsparameter darstellt. Beispiele: `/home/cognos/utilization.properties` unter Linux oder `C:\data\utilization.properties` unter Windows.

Im Folgenden sehen Sie ein Beispiel für den Inhalt einer Eigenschaftendatei:

```
scope=All Windows Machines
metric=ALL
operator=N/A
value1=N/A
value2=N/A
appdeps=N/A
```

Es ist darauf zu achten, dass Leerzeichen in einem Parameternamen mit dem Backslash-Zeichen versehen werden. Lautet der Parametername beispielsweise `Snapshot ID Parameter`, muss der entsprechende Eintrag in der Eigenschaftendatei wie folgt lauten:

```
Snapshot\ ID\ Parameter=my_id
```

Bericht
Der Pfad des auszuführenden Berichts. Dabei ist dem Namen die Zeichenfolge "compiled" angehängt. Beispiele dafür sind:

- Unter Linux und TADDM 7.3.0: `$COLLATION_HOME/deploy-tomcat/birt-viewer/WEB-INF/report/taddm_server_utilization.rptdesigncompiled`.
- Unter Linux und TADDM 7.3.0.1 und höher: `$COLLATION_HOME/apps/birt-viewer/WEB-INF/report/taddm_server_utilization.rptdesigncompiled`.
- Unter Windows und TADDM 7.3.0: `%COLLATION_HOME%\deploy-tomcat\birt-viewer\WEB-INF\report\taddm_server_utilization.rptdesigncompiled`.
- Unter Windows und TADDM 7.3.0.1 und höher: `%COLLATION_HOME%\apps\birt-viewer\WEB-INF\report\taddm_server_utilization.rptdesigncompiled`

Ergebnisse

Anmerkung: Mit dem Befehl **genReport** werden keine Drillthrough-Berichte generiert. Daher funktionieren Links in dem generierten Bericht nicht.

BIRT-Bericht importieren
Durch den Import von BIRT-Berichtsentwürfen können im Datenmanagementportal angepasste Berichte hinzugefügt werden.

Vorbereitende Schritte
Um einen angepassten Bericht hinzuzufügen, müssen Sie den Bericht zuerst mithilfe des BIRT Designer-Tools entwerfen und entwickeln. Der Berichtsentwurf muss in einer Datei `.rptdesign` gespeichert werden, auf die vom Clientsystem aus ein Zugriff möglich ist.

Anmerkung: Fix Pack 3 In TADDM 7.3.0.3 und höher haben Spalten in Datenbankansichten für erweiterte Attribute bestimmte Datentypen, z. B. VARCHAR. In den früheren TADDM-Releases gab es für die Spalten nur den Typ CLOB. Deshalb kann es sein, dass nach einem Upgrade auf Fixpack 3 die BIRT-Berichte, die erweiterte Attribute nutzen, nicht mehr funktionieren. Zum Beispiel können Fehler generiert werden, wenn Spalten mit erweiterten Attributen nicht in einen bestimmten Datentyp, z. B. VARCHAR, umgesetzt werden.

Vorgehensweise

Gehen Sie für den Import eines BIRT-Berichts wie folgt vor:

1. Klicken Sie im Teilfenster 'Funktionen' des Datenmanagementportals auf **Analyse**.
2. Klicken Sie im Abschnitt 'Analyse' auf **BIRT-Berichte**.

   Die Liste **TADDM BIRT-Berichte** wird mit allen verfügbaren BIRT-Berichten angezeigt.
3. Klicken Sie auf **Neu**.
4. Geben Sie die Details zu dem neuen Bericht an, einschließlich des Namens, der Beschreibung und der Speicherposition der Berichtsentwurfsdatei, wenn Sie dazu aufgefordert werden.

   Anhand des Namens und der Beschreibung kann der Bericht in der Liste **TADDM BIRT-Berichte** ermittelt werden.
5. Klicken Sie auf **OK**.

Ergebnisse

Der Berichtsentwurf wird auf den Server hochgeladen, der neue Bericht ist nun über das Datenmanagementportal verfügbar.

Anmerkung: Wenn der Bericht bereits auf dem Server vorhanden ist, schlägt der Import fehl. Dies kann auch der Fall sein, wenn der vorhandene Bericht im Datenmanagementportal nicht sichtbar ist. (So wird beispielsweise der Serveraffinitätsbericht auf dem Synchronisationsserver nicht unterstützt und daher im Datenmanagementportal nicht angezeigt, selbst wenn er auf dem Server vorhanden ist.)

BIRT-Bericht löschen

BIRT-Berichte können über das Datenmanagementportal vom Server gelöscht werden.

Vorbereitende Schritte

Wenn ein Bericht vom Server gelöscht wird, wird die von dem Bericht verwendete Datei `.rptdesign` aus dem Berichtsverzeichnis des Servers entfernt. Falls der Berichtsentwurf für die zukünftige Verwendung gespeichert werden soll, ist vor Löschen des Berichts sicherzustellen, dass eine Sicherungskopie der Datei `.rptdesign` vorhanden ist.

Vorgehensweise

Gehen Sie folgendermaßen vor, um einen BIRT-Bericht zu löschen:

1. Klicken Sie im Teilfenster 'Funktionen' auf **Analyse**.
2. Klicken Sie im Abschnitt 'Analyse' auf **BIRT-Berichte**.

   Die Liste **TADDM BIRT-Berichte** wird mit allen verfügbaren BIRT-Berichten angezeigt.
3. Wählen Sie den Bericht aus, der gelöscht werden soll.
4. Klicken Sie auf **Löschen**.
5. Klicken Sie auf **Aktualisieren**, um die Liste **TADDM BIRT-Berichte** zu aktualisieren.

Ergebnisse

Der ausgewählte Bericht wird vom Server gelöscht und wird nicht mehr in der Liste **TADDM BIRT-Berichte** im Datenmanagementportal angezeigt. Außerdem wird die Datei `.rptdesign` für den Bericht aus dem Berichtsverzeichnis des TADDM-Servers gelöscht.

BIRT-Berichtsentwurf exportieren

BIRT-Berichtsentwürfe können über das Datenmanagementportal aus dem Server exportiert werden.

Informationen zu diesem Vorgang

Sie können einen Berichtsentwurf exportieren, wenn Sie einen vorhandenen Bericht als Basis für einen neuen angepassten Bericht verwenden möchten oder wenn Sie den Berichtsentwurf in einen anderen Server importieren möchten.

Vorgehensweise

Gehen Sie folgendermaßen vor, um einen BIRT-Berichtsentwurf zu exportieren:

1. Klicken Sie im Teilfenster 'Funktionen' auf **Analyse**.
2. Klicken Sie im Abschnitt 'Analyse' auf **BIRT-Berichte**.
   Die Liste **TADDM BIRT-Berichte** wird mit allen verfügbaren BIRT-Berichten angezeigt.
3. Wählen Sie den Bericht aus, der exportiert werden soll.
4. Klicken Sie auf **Herunterladen**.
5. Geben Sie an, dass Sie die Datei speichern möchten, und geben Sie eine Speicherposition an, wenn Sie von Ihrem Browser dazu aufgefordert werden.

Ergebnisse
Der von dem ausgewählten Bericht verwendete Entwurf wird an der angegebenen Speicherposition als RPTDESIGN-Datei gespeichert. Sie können diese Datei mit dem BIRT Designer-Tool öffnen und ändern.

BIRT Report Viewer wiederherstellen
Wenn Sie sich der Sicherheitsrisiken des BIRT Report Viewer bewusst sind und diese eingehen möchten, können Sie den Viewer wiederherstellen.

Vorgehensweise

1. Setzen Sie die Eigenschaft `com.ibm.taddm.birtviewer.enabled` in der Datei `collation.properties` auf `true`:

   ```
   com.ibm.taddm.birtviewer.enabled=true
   ```

2. Starten Sie den TADDM-Server erneut.

   Anmerkung: Nach dem TADDM-Server-Upgrade ist dieses Flag standardmäßig auf `false` gesetzt.

Momentaufnahmetool verwenden
Mit dem Momentaufnahmetool können Sie eine Kopie der Computersysteminformationen, der Erkennungsereignisse und der Serveranwendungen, die zum Zeitpunkt der Momentaufnahme ausgeführt werden, erstellen.

Mit dem Momentaufnahmetool können Sie auch Informationen laden, die für den Datenabgleich verwendet werden. Beispiele:

- eine Liste mit erwarteten Servern, auch Referenzliste genannt
- eine Liste mit ausgeschlossenen Servern

Mithilfe von Berichten können Sie die Informationen abfragen, die vom Momentaufnahmetool erfasst wurden. Beispiele:

- die Server, die hinzugefügt oder entfernt wurden
- das Verhältnis von physischen zu virtuellen Servern
- die Server, die nicht vollständig erkannt werden konnten, da keine SSH-Sitzung erfolgreich aufgebaut wurde
- die Differenz zwischen der Liste der erkannten Server und der Liste der erwarteten Server

Einschränkung: Machen Sie Momentaufnahmen, sobald die Erkennung und die Topologieagenten ihre Ausführung abgeschlossen haben. Wenn Sie eine Momentaufnahme machen, bevor die Topologieagenten die Verarbeitung der erkannten Informationen abgeschlossen haben, kann es vorkommen, dass einige Momentaufnahmeberichte (zum Beispiel der Snapshot-Bericht über fehlgeschlagene Sitzungen) nicht vollständig sind.

Die Befehlssyntax für 'snapshot.sh'

Mithilfe des Befehls `snapshot.sh` können Sie eine Momentaufnahme des Systems und der zugehörigen Ereignisse und Server erstellen. Der Befehl `snapshot_.sh` befindet sich im Verzeichnis `$COLLATION_HOME/bin`.

Sie können den Befehl `snapshot.sh` auf dem TADDM-Server ausführen. In einer Streaming-Server-Implementierung müssen Sie den Befehl `snapshot.sh` auf dem primären Speicherserver ausführen.

Befehlssyntax

`snapshot.sh` *Aktion* [*Aktionsparameter*]

Parameter

addexclude *Dateiname* [*Ausschlussliste*]
Fügt die Ausschlussliste zur Datei hinzu oder ersetzt eine vorhandene Instanz der Liste in der Datei.

addref *Dateiname* [*Referenzliste*]
Fügt die Referenzliste zur Datei hinzu oder ersetzt eine vorhandene Instanz der Liste in der Datei.

clear
Löscht alle Momentaufnahmedaten und Tabellen.

compare [*Momentaufnahme_A Momentaufnahme_B*]
Zeigt die Differenz zwischen den letzten beiden Momentaufnahmen oder Momentaufnahme_A und Momentaufnahme_B auf Basis des Hostnamens an.

compareref [*Momentaufnahme_A Referenzliste*]
Zeigt die Differenz zwischen der Momentaufnahme und der Referenzliste an.

comparesig [*Momentaufnahme_A Momentaufnahme_B*]
Zeigt die Differenz zwischen den letzten beiden Momentaufnahmen,Momentaufnahme_A und Momentaufnahme_B, auf Basis der Signatur für Änderungen im Hostnamen oder Betriebssystem an.

compsys
Zeigt die Computersysteme an.

detail [*Momentaufnahme_A*]
Zeigt alle Details der Computersysteme in der letzten Momentaufnahme oder Momentaufnahme_A an.

detailos [*Momentaufnahme_A*]
Zeigt Betriebssysteminformationen der Computersysteme in der letzten Momentaufnahme oder Momentaufnahme_A an.

help
Zeigt eine ausführliche Hilfe zur Syntax des Befehls **snapshot.api** an.

list [*Momentaufnahme_A*]
Zeigt die letzte Momentaufnahme oder Momentaufnahme_A an.

listall [*Standard*]
Zeigt alle Momentaufnahmen an.

listexclude [*Ausschlussliste*]
Zeigt die letzte Ausschlussliste oder die namentlich angegebene Liste an.

listref [*Referenzliste*]
Zeigt die letzte Referenzliste oder die namentlich angegebene Liste an.

listallexclude
Zeigt alle Ausschlusslisten an.

listallref
Zeigt alle Referenzlisten an.

nosession [*Momentaufnahme_A*]
Zeigt Computersysteme, die keine Sitzung per Hosting bereitstellen konnten, in der letzten Momentaufnahme oder Momentaufnahme_A an.

remove *Momentaufnahme_A* [*Typ*]
Entfernt Momentaufnahme A oder alle Momentaufnahmen des angegebenen Typs

removeexclude *Ausschlussliste*
Entfernt die namentlich angegebene Ausschlussliste.

removeref *Referenzliste*
Entfernt die namentlich angegebene Referenzliste.

session [*Momentaufnahme_A*]
Zeigt die Computersysteme, die eine Sitzung per Hosting bereitgestellt haben, in der letzten Momentaufnahme oder Momentaufnahme_A an.

sensorerror [*Momentaufnahme_A*]
Zeigt alle Sensorfehler in der letzten Momentaufnahme oder Momentaufnahme_A.

take [*Typ*] [*Beschreibung*]
Erstellt eine Momentaufnahme einschließlich Typ- und Beschreibungsinformationen, falls angegeben.

Mit dem Momentaufnahmetool die Anzahl der physischen Server reduzieren

Sie können das Momentaufnahmetool verwenden, wenn Sie durch die Ausführung virtueller Server viele physische Server durch eine geringere Anzahl physische Server ersetzen.

Vorgehensweise

Gehen Sie folgendermaßen vor, um hilfreiche Informationen abzurufen, wenn Sie versuchen, die Anzahl der verwendeten physischen Server zu reduzieren:

1. Führen Sie eine Erkennung aller bekannten Systeme durch.
2. Erstellen Sie mit dem Momentaufnahmetool eine Momentaufnahme:

   ```
   snapshot.sh take
   ```

   Sie können auch Typ- und Beschreibungsangaben zur Momentaufnahme hinzufügen:

   ```
   snapshot.sh take Typ Beschreibung
   ```

3. Führen Sie im Datenmanagementportal den Bericht TADDM_SNAPSHOT_SESSION_FAILED aus.

   Der Bericht gibt Informationen zu den Systemen zurück, die nicht erkannt wurden, da keine SSH-Sitzung aufgebaut werden konnte.
4. Stellen Sie sicher, dass mit allen Systemen SSH-Sitzungen aufgebaut werden können. Möglicherweise müssen die Authentifizierungsdetails in TADDM aktualisiert werden.
5. Führen Sie nur eine Erkennung der Systeme durch, auf die bei der ersten Erkennung nicht zugegriffen wurde, um sicherzustellen, dass alle Verbindungsprobleme gelöst wurden.
6. Führen Sie nach einer angemessenen Zeitspanne, z. B. nach einem Monat, eine Erkennung aller bekannten Systeme durch.
7. Führen Sie im Datenmanagementportal den Bericht TADDM_SNAPSHOT_CHANGE aus.

   Der Bericht gibt Informationen zu den neuen Systemen zurück, die seit der letzten Momentaufnahme sichtbar sind, zu den Systemen, die nicht mehr vorhanden sind und zum Verhältnis der physischen Server zu den virtuellen Servern in Prozent.

Momentaufnahmetool für den Abgleich der erwarteten und tatsächlichen Systemlisten verwenden

Mit dem Momentaufnahmetool und den vordefinierten Berichten können Sie überprüfen, ob die Liste der im Netz verfügbaren Server der Liste der erwarteten Server entspricht.

Vorgehensweise

Gehen Sie für den Abgleich der erwarteten und tatsächlichen Systeme wie folgt vor:

1. Bereiten Sie eine Referenzliste vor, die die Liste der erwarteten Server enthält.

   Die Referenzliste ist eine Textdatei im CSV-Format (Comma-Separated Values, durch Kommas getrennte Werte) mit folgenden Feldern:

- Hostname
- IP-Adresse
- Rahmen
- Betriebssystem
- Hosttyp
- Kommentare
- Unterstützungsbereich
- Position

Weitere Informationen zur Syntax der Referenzdatei erhalten Sie, wenn Sie den Befehl **snapshot.sh** mit dem Parameter 'help' ausführen:

```
snapshot.sh help
```

2. Bereiten Sie bei Bedarf eine Ausschlussliste vor, die die Liste der Server enthält, die im Abgleichprozess ignoriert werden sollen.

   Die Ausschlussliste ist eine Textdatei im CSV-Format mit folgenden Feldern:

   - Hostname
   - Ausschlusstyp

   Weitere Informationen zur Syntax der Ausschlussdatei erhalten Sie, wenn Sie den Befehl **snapshot.sh** mit dem Parameter 'help' ausführen:

   ```
   snapshot.sh help
   ```

3. Erstellen Sie mit dem Momentaufnahmetool eine Momentaufnahme:

   ```
   snapshot.sh take
   ```

   Sie können auch Typ- und Beschreibungsangaben zur Momentaufnahme hinzufügen:

   ```
   snapshot take Typ Beschreibung
   ```

4. Führen Sie im Datenmanagementportal einen der folgenden BIRT-Berichte aus:

   - TADDM_SNAPSHOT_RECONCILIATION_SUMMARY
   - TADDM_SNAPSHOT_RECONCILIATION_DETAIL

Momentaufnahmeberichte in einer Umgebung mit Synchronisationsserver verwenden

In einer Synchronisationsserver-Implementierung kann zum Zusammenstellen von Informationen die Unternehmensversion der vordefinierten Momentaufnahmeberichte ausgeführt werden.

Vorgehensweise

Gehen Sie wie folgt vor, um in einer Synchronisationsserver-Implementierung vordefinierte Momentaufnahmeberichte auszuführen:

1. Richten Sie die Momentaufnahmetabelle ein, sofern dies noch nicht geschehen ist. Gehen Sie hierzu folgendermaßen vor:

   a) Führen Sie auf den einzelnen TADDM-Servern den Befehl snapshot.sh ohne Parameter aus.

   b) Starten Sie TADDM in den einzelnen Domänen- und Synchronisationsservern erneut.

   Mit diesem Verfahren werden die Momentaufnahmetabellen erstellt, sofern sie noch nicht vorhanden sind. Die Momentaufnahmetabellen müssen für jede TADDM-Umgebung jeweils nur einmal eingerichtet werden.

2. Führen Sie in jeder TADDM-Domäne eine Erkennung durch und erstellen Sie gegebenenfalls Momentaufnahmen an den einzelnen Domänen.

3. Führen Sie auf dem Synchronisationsserver eine Synchronisation durch. Stellen Sie sicher, dass Sie alle Domänen einschließen.
4. Erstellen Sie eine Momentaufnahme des Unternehmens. Führen Sie auf dem Synchronisationsserver den folgenden Befehl aus:

```
snapshot.sh take
```

5. Führen Sie die Berichte in jeder Domäne aus. Verwenden Sie die normale Version der einzelnen Momentaufnahmeberichte, z. B. TADDM_SNAPSHOT_CHANGE.
6. Führen Sie die Berichte auf dem Synchronisationsserver aus. Verwenden Sie die Unternehmensversion der einzelnen Momentaufnahmeberichte, z. B. TADDM_SNAPSHOT_SYNC_CHANGE.

Kombinierter Einsatz von TADDM mit anderen Tivoli-Produkten

Um erweiterte Funktionalität zur Verwaltung Ihrer IT-Umgebung zu erhalten, können Sie den IBM Tivoli Application Dependency Discovery Manager (TADDM) zusammen mit anderen Tivoli-Produkten einsetzen, einschließlich IBM Tivoli Business Service Manager, IBM Tivoli Monitoring und Ereignismanagementsystemen wie IBM Tivoli Netcool/OMNIbus.

Unterstützte Versionen

Folgender Tabelle können Sie entnehmen, welche Versionen der Produkte, mit denen TADDM integriert werden kann, unterstützt werden.

Folgende Tabelle zeigt die unterstützten Versionen der Produkte, mit denen TADDM integriert werden kann.

Tabelle 41. Unterstützte Produktversionen

Produktname	**Unterstützte Version**
Kontextmenüservice und Datenintegrationsservice (CMS/DIS)	
IBM Control Desk (ICD)	• 7.6
IBM SmartCloud Control Desk (SCCD)	• 7.5.1 - verwenden Sie die neuste verfügbare Fixpackstufe
IBM Tivoli Business Service Manager (TBSM)	• 4.2.1 - verwenden Sie die neuste verfügbare Fixpackstufe • 6.1.0 - verwenden Sie die neuste verfügbare Fixpackstufe • 6.1.1 - verwenden Sie die neuste verfügbare Fixpackstufe
IBM Tivoli Change And Configuration Management Database (CCMDB)	• 7.2.1
IBM Tivoli Integration Composer (ITIC)	• 7.5.1 - verwenden Sie die neuste verfügbare Fixpackstufe
IBM Tivoli Monitoring (ITM)	• 6.2.1 • 6.2.2 - FP3 • 6.2.3 • 6.3

Tabelle 41. Unterstützte Produktversionen (Forts.)	
Produktname	**Unterstützte Version**
IBM Tivoli Netcool/OMNIbus	• 7.3 • 7.4 • Fix Pack 1 8.*x* - mit TADDM 7.3.0.1 und höher unterstützt
IBM Tivoli Network Manager IP (ITNMIP)	• 3.9 • 4.1
Jazz for Service Management (JazzSM)	• 1.1
Tivoli Common Reporting (TCR)	• 1.3 • 2.1.1 • 3.1
Tivoli Directory Integrator (TDI)	• 7.0 • 7.1 • 7.1.1
Tivoli Netcool/IMPACT	• 7.1
Tivoli Workload Scheduler (TWS)	• 8.5.1 • 8.6

Weitere Informationen zu Produkten, die Sie mit TADDM integrieren können, finden Sie in der zugehörigen Dokumentation:

- Informationen zu Context Menu Service and Data Integration Service (CMS/DIS) finden Sie im Abschnitt *Konfiguration für Context Menu Service and Data Integration Service durchführen* im TADDM-*Installationshandbuch*.
- IBM Control Desk (ICD)
- IBM SmartCloud Control Desk (SCCD)
- IBM Tivoli Business Service Manager (TBSM)
- IBM Tivoli Change and Configuration Management Database (CCMDB)
- IBM Tivoli Integration Composer (ITIC)
- IBM Tivoli Monitoring (ITM)
- IBM Tivoli Netcool/OMNIbus
- IBM Tivoli Network Manager IP (ITNMIP)
- Jazz for Service Management (JazzSM)
- Tivoli Common Reporting (TCR)
- Tivoli Directory Integrator (TDI)
- Tivoli Netcool/Impact
- Tivoli Workload Scheduler (TWS)

TADDM über OSLC Automation mit IBM Tivoli Monitoring integrieren

TADDM kann mithilfe von OSLC Automation mit IBM Tivoli Monitoring integriert werden. Wenn Sie TADDM mit IBM Tivoli Monitoring 6.3 integrieren möchten, sollten Sie OSLC Automation verwenden. Das Integra-

tionsverfahren unter Verwendung des IBM Tivoli Monitoring Scope-Sensors ist veraltet und steht in künftigen Releases nicht mehr zur Verfügung.

TADDM nutzt die IBM Tivoli Monitoring-Infrastruktur auf zweierlei Weise:

- TADDM ruft die Liste mit IBM Tivoli Monitoring-Endpunkten über OSLC Automation Session vom Tivoli Enterprise Portal Server ab.
- TADDM führt auf den Zielsystemen für die Sensoren von Erkennungen der Ebene 2 und 3 CLI-Befehle aus und erfasst die Ausgaben dieser Befehle.

Bei Installationsproblemen finden Sie im *Handbuch zur Fehlerbehebung* von TADDM weitere Informationen im Abschnitt *ITM OSLC Execute Automation Service Provider problems* (Probleme mit ITM OSLC Execute Automation Service Provider).

Voraussetzungen:

Fix Pack 5

Unter Windows 7 und höher ist Folgendes erforderlich:

1. PowerShell Version 2+
2. TEMS SOAP URL
3. Stellen Sie sicher, dass Sie sowohl zum TEMS als auch zum TEPS eine Verbindung herstellen können.

In der folgenden Tabelle werden die Schritte beschrieben, die Sie für eine erfolgreiche Aktivierung der Integration von TADDM mit IBM Tivoli Monitoring über OSLC Automation ausführen müssen.

Tabelle 42. TADDM über OSLC Automation mit IBM Tivoli Monitoring integrieren

Schritt	**Details**
ITM Tivoli Enterprise Monitoring Server (TEMS) und ITM TEPS-Hosts konfigurieren	„ITM Tivoli Enterprise Monitoring Server (TEMS) und ITM TEPS-Hosts konfigurieren” auf Seite 183
OSLC Execute Automation Service Provider unter IBM Tivoli Monitoring installieren **Anmerkung:** Stellen Sie sicher, dass alle im Abschnitt „Voraussetzungen für die Installation von OSLC Execute Automation Service Provider unter IBM Tivoli Monitoring” auf Seite 182 angegebenen Voraussetzungen erfüllt sind.	„OSLC Execute Automation-Service-Provider unter IBM Tivoli Monitoring installieren” auf Seite 186
TADDM für Verwendung von OSLC Execute Automation Service Provider konfigurieren	„TADDM für den OSLC Execute Automation-Service-Provider konfigurieren” auf Seite 192
TADDM für Erkennung konfigurieren: • Konfigurieren Sie Automation-Eigenschaften in der Datei `collation.properties`. • Erstellen Sie in der Zugriffsliste einen neuen Zugriffslisteneintrag des Typs `<"Integration">"OSLC Automation"`.	„Erkennung mit OSLC Automation Session konfigurieren” auf Seite 107

Nach Ausführung dieser Schritte können Sie eine Erkennung mithilfe von ITM OSLC Execute Automation Service Provider durchführen.

Zugehörige Konzepte

„TADDM durch OSLC Automation mit anderen Produkten integrieren” auf Seite 191

TADDM kann durch Open Services for Lifecycle Collaboration (OSCL) Automation mit anderen Produkten integriert werden. TADDM stellt eine Verbindung mit dem OSLC Execute Automation-Service-Provider her, der Daten zu den Infrastrukturen anderer Produkte bereitstellt, die von TADDM mit OSLC Automation Session erkannt werden können.

ITM OSLC Execute Automation-Service-Provider
Der ITM OSLC Execute Automation-Service-Provider wird für den Import von IP-Adressdaten der von IBM Tivoli Monitoring verwalteten Endpunkte in TADDM und zur Erkennung der IBM Tivoli Monitoring-Endpunkte mit OSLC Automation Session verwendet.

Abbildung 1 zeigt eine mit dem ITM OSLC Execute Automation-Service-Provider verbundene TADDM-Instanz, in der die Daten zu der von ITM verwalteten Infrastruktur mittels KT1-Befehlen erfasst werden.

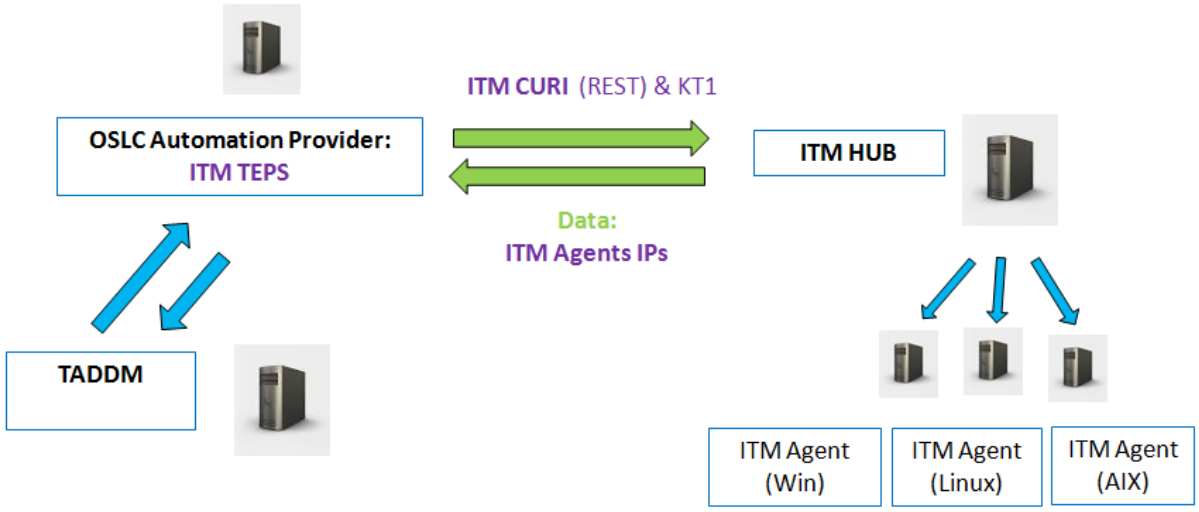

Abbildung 4. Mit dem ITM OSLC Execute Automation-Service-Provider verbundene TADDM-Instanz, in der die Daten zu der von ITM verwalteten Infrastruktur mittels KT1-Befehlen erfasst werden

TADDM erhält das Ziel für den OSLC Execute Automation-Service-Provider über die Jazz SM Registry Services oder aus der Datei `collation.properties`. Abbildung 2 zeigt eine TADDM-Instanz, die die Adresse des OSLC Execute Automation-Service-Provider über die Jazz SM Registry Services erhält.

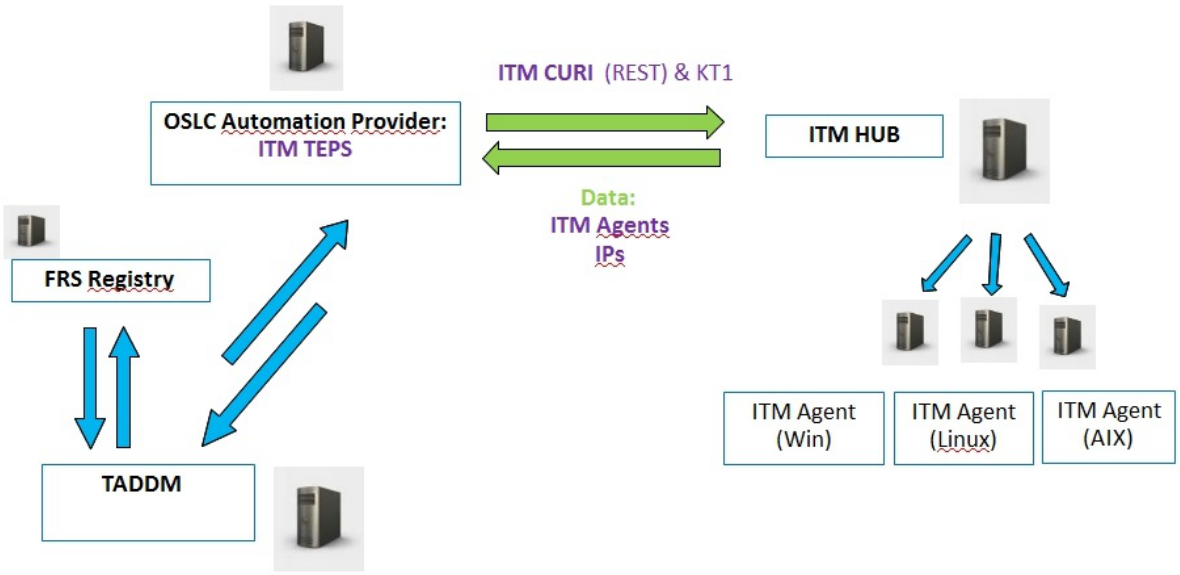

Abbildung 5. TADDM-Instanz, die die Adresse des OSLC Execute Automation-Service-Provider über die Jazz SM Registry Services erhält

TADDM kann mit mehreren OSLC Execute Automation-Service-Providern direkt verbunden werden, aber auch mit einer einzelnen Instanz der Jazz SM Registry Services, in der mehrere Provider registriert sein können. Abbildung 3 zeigt eine TADDM-Instanz, die die Adressen der auf mehreren ITM TEPS (Portalservern) implementierten OSLC Execute Automation-Service-Providern aus Jazz SM Registry Services herunterlädt.

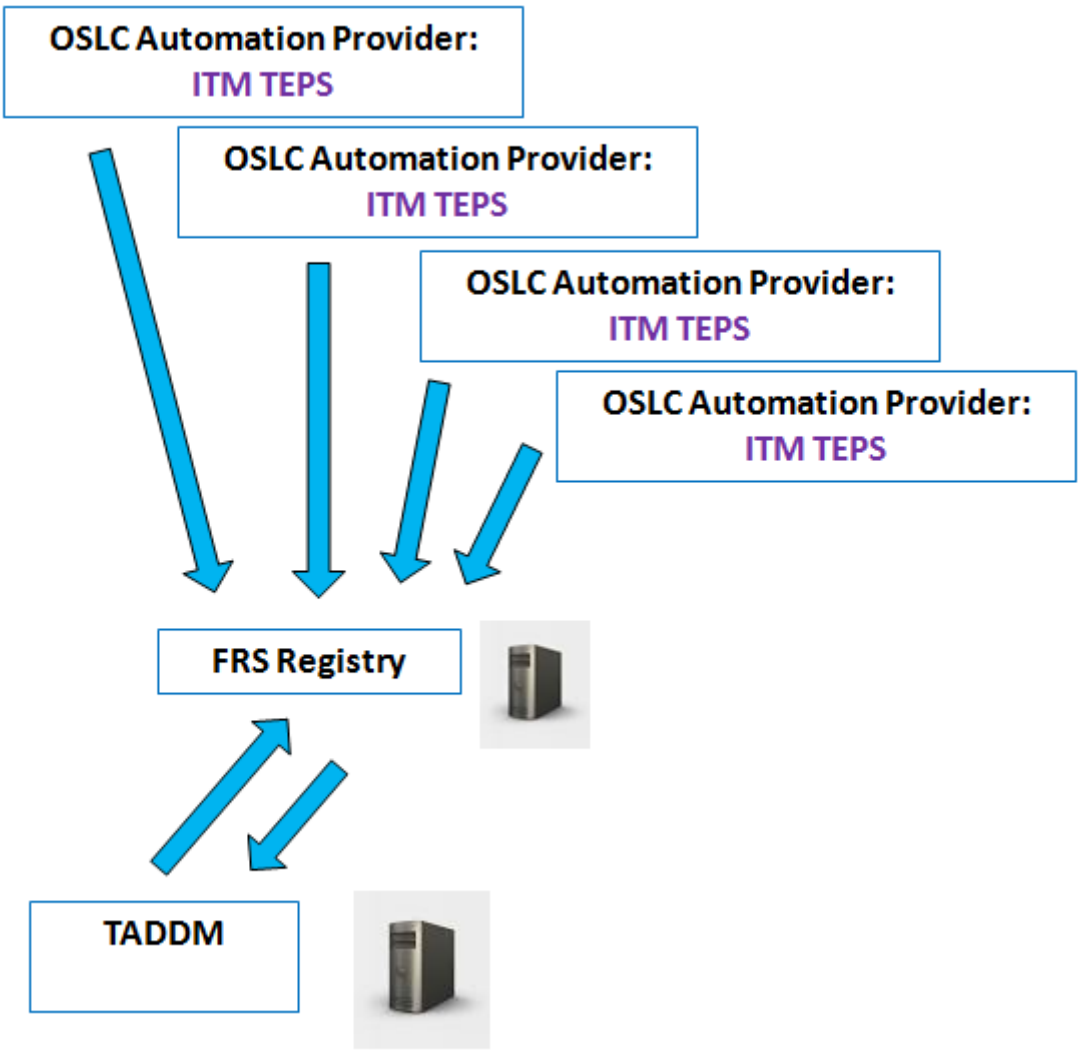

Abbildung 6. TADDM-Instanz, die die Adressen der auf mehreren ITM TEPS (Portalservern) implementierten OSLC Execute Automation-Service-Providern aus Jazz SM Registry Services herunterlädt

Zugehörige Konzepte

„OSLC Execute Automation-Service-Provider" auf Seite 191
Der OSLC Execute Automation-Service-Provider stellt TADDM Daten zu den IP-Adressen von Endpunkten bereit, die von anderen Produkten verwaltet werden. Die Daten werden zur Erkennung von Endpunkten mit OSLC Automation Session verwendet.

ITM OSLC Execute Automation-Service-Provider installieren

Für den Import von IP-Adressdaten der von IBM Tivoli Monitoring (ITM) verwalteten Endpunkte in TADDM und zur Ausführung von Erkennungen müssen Sie den OSLC Execute Automation-Service-Provider unter IBM Tivoli Monitoring installieren.

Bei Installationsproblemen finden Sie im *Handbuch zur Fehlerbehebung* von TADDM weitere Informationen im Abschnitt *ITM OSLC Execute Automation Service Provider problems* (Probleme mit dem ITM OSLC Execute Automation-Service-Provider).

Voraussetzungen für die Installation von OSLC Execute Automation Service Provider unter IBM Tivoli Monitoring

Vor der Installation eines OSLC Execute Automation-Service-Providers unter IBM Tivoli Monitoring (ITM) müssen Sie Ihre Umgebung so konfigurieren, dass alle Voraussetzungen erfüllt sind.

Der ITM OSLC Execute Automation-Service-Provider muss auf dem Host von ITM Tivoli Enterprise Portal Server (TEPS) installiert werden. Die unterstützte Version von IBM Tivoli Monitoring ist IBM Tivoli Monitoring 6.3.

ITM Tivoli Enterprise Monitoring Server (TEMS) und ITM TEPS-Hosts konfigurieren

Fix Pack 5 Schritt 1 - TEMS und TEPS neu konfigurieren

Die Konfiguration von TEMS und TEPS erfolgt am besten über die MTEMS-GUI (Manage Tivoli Enterprise Monitoring Services).

Bei einem Windows-Betriebssystem müssen Sie den ITM-Prozess **`kinconfig .exe`** starten, um die MTEMS-GUI aufzurufen.

Bei Unix/Linux können Sie die MTEMS-GUI über den CLI-Befehl "**`./itmcmd manage`**" starten.

Für beide ITM-Komponenten (TEMS und TEPS) ist jeweils wie folgt vorzugehen:

- Markieren Sie die Komponente auf der MTEMS-GUI.
- Klicken Sie mit der rechten Maustaste und wählen Sie "Reconfigure" (Neu konfigurieren).

Hierdurch gelangen Sie in das Konfigurationsfenster für den TEMS bzw. TEPS.

Geben Sie alle TEMS-Parameter korrekt ein (TEMS-Typ, TEMS-Name und -Protokoll bei den Grundeinstellungen für die Konfiguration und Hostname/IP-Adresse sowie Port bei den erweiterten Einstellungen) und klicken Sie dann auf "**OK**".

Wählen Sie im Konfigurationsfenster des TEPS die Option "**`Enable the dashboard data provider`**" (Dashboarddatenprovider aktivieren) aus.

Fix Pack 5 Schritt 2 - TADDM-Scripts ausführen und Konfiguration vornehmen

Dieser Schritt beinhaltet die Ausführung zweier zentraler TADDM-Scripts, mit deren Hilfe die Integration von TADDM in ITM konfiguriert wird.

1. Konfigurieren der Datei 'provider.properties'

ITM TEMS-Host

Aktivieren Sie die KT1-Befehle (**`tacmd get/put/execute`**) durch Ausführung eines der folgenden Scripts auf dem ITM TEMS-Host:

- Unter Linux:

```
TADDM_CD_ISO/itm-discovery-support/configure_tems.sh <-i <ITM_HOME>>
 [-t <TEMP-DIR>
```

- Unter Windows:

```
TADDM_CD_ISO/itm-discovery-support/configure_tems.ps1 <-i <ITM_HOME>>
 [-t <TEMP-DIR>
```

Dabei ist *<ITM_HOME>* das Installationsverzeichnis von ITM TEPS, zum Beispiel `/opt/IBM/ITM`, und *<TEMP-DIR>* ist das Zielverzeichnis für temporäre Dateien. Der Standardwert des Parameters *<TEMP-DIR>* ist `/var/log/automation_provider`.

Fix Pack 5 Wenn dies abgeschlossen ist, sollten Ihnen am Ende der Scriptausführung die folgenden Zeilen angezeigt werden.

```
INFO: ITM TEPS wird gestoppt...
INFO: ITM TEPS wurde gestoppt.
INFO: ITM TEPS wird gestartet...
INFO: ITM TEPS wurde gestartet.
INFO: Es wird geprüft, ob TEPS aktiv ist...
INFO: Es wird geprüft, ob OSLC Automation Provider installiert ist...
INFO: Installation von OSLC Automation Provider wurde erfolgreich abgeschlossen.
```

Anmerkung: Verwenden Sie für eine Erstbenutzer-ID-Validierung (**`tacmd get/put/execute`**) zunächst **http:1920** oder **https:3661** und dann **ip.pipe:1920** oder **ip.spipe:3660** für KT1-Arbeit von ei-

nem Automation-Provider für das erkannte Ziel. Diese Protokolle müssen in ITM aktiviert werden, um die Erkennung abzuschließen.

ITM TEPS-Host
Überprüfen Sie, ob der ITM-Dashboarddatenprovider installiert oder aktiviert ist. Falls nicht, installieren bzw. aktivieren Sie ihn optional. Informationen hierzu finden Sie in der IBM Tivoli Monitoring-Dokumentation im Abschnitt Verifying the dashboard data provider is enabled (Überprüfen, ob der ITM-Dashboarddatenprovider aktiviert ist).

Wichtig: Stellen Sie bei Verwendung eines 64-Bit Windows-Betriebssystems sicher, dass die Systemvariable PATH (bzw. der Pfadeintrag in der Datei `kfwenv`) auf das 64-Bit-Verzeichnis TMAITM6 verweist. Falls es nicht vorhanden ist, fügen Sie es manuell hinzu. Wenn ITM beispielsweise im Verzeichnis `C:\IBM\ITM\` installiert ist, müssen Sie `C:\IBM\ITM\TMAITM6_x64` entweder in der Systempfadumgebungsvariablen oder in der PATH-Anweisung in der Datei `C:\IBM\ITM\CNPS\kfwenv` angeben.

Fix Pack 5

2. Ausführen des Scripts `automation_provider` vom ITM TEPS-Host aus

Führen Sie das Script **`automation_provider`** auf Ihrem Server aus:

- Unter Linux:

```
automation_provider.sh install  -t /tmp/log  -i /opt/IBM/ITM -c
/tmp/provider.properties
```

- Unter Windows:

```
automation_provider.ps1 install  -t /tmp/log  -i /opt/IBM/ITM -c
/tmp/provider.properties
```

Ein wichtiger Punkt, der hierbei zu beachten ist: Selbst wenn bereits vor dem Ausführen des Scripts 'automation_provider' eine Datei 'provider.properties' erstellt wurde, wird diese ignoriert und im Konfigurationsverzeichnis des ITM TEPS ($ITM_HOME/iw/profiles/ITMProfile/installedApps/ITMCell/itmautomationprovider.ear/itmautomationprovider.war/WEB-INF/provider.properties) wird eine Standarddatei 'provider.properties' erstellt.

Diese Datei müssen Sie anschließend suchen, die Änderungen an den Parametern manuell vornehmen und dann den TEPS neu starten, damit die Datei wirksam wird.

Erforderliche Dateien

In dem Verzeichnis, aus dem Sie das Installationsscript ausführen, müssen die folgenden Dateien vorhanden sein:

- Das Installationsscript:
  - Unter Linux und AIX: `TADDM_CD_ISO/itm-discovery-support/automation_provider.sh` und seine Untermodule im Verzeichnis `TADDM_CD_ISO/itm-discovery-support/mod/sh/`.
  - Unter Windows: `TADDM_CD_ISO/itm-discovery-support/automation_provider.ps1` und seine Untermodule im Verzeichnis `TADDM_CD_ISO/itm-discovery-support/mod/ps/`.
- `itmautomationprovider.ear` - das Paket mit dem ITM OSLC Execute Automation-Service-Provider. Die genaue Speicherposition der Datei ist `TADDM_CD_ISO/itm-discovery-support/ear/itmautomationprovider.ear`.
- `provider.properties` - die Beispielkonfigurationsdatei für den ITM OSLC Execute Automation-Service-Provider. Die Datei kann manuell konfiguriert und dem Installationsscript als Parameter übergeben werden. Wenn keine Konfigurationsdatei übergeben wird, müssen die erforderlichen Parameter während der Installation eingegeben werden. Die genaue Speicherposition der Datei ist `TADDM_CD_ISO/itm-discovery-support/template_provider.properties`.
- KT1-Unterstützungsbibliotheken für das jeweilige Betriebssystem und seine Architektur (32- oder 64-Bit).

– Unter Linux:

```
TADDM_CD_ISO/itm-discovery-support/linux32
TADDM_CD_ISO/itm-discovery-support/linux64
```

– Unter AIX:

```
TADDM_CD_ISO/itm-discovery-support/aix32
TADDM_CD_ISO/itm-discovery-support/aix64
```

– Unter Linux on IBM System Z (zLinux):

```
TADDM_CD_ISO/itm-discovery-support/linuxz32
TADDM_CD_ISO/itm-discovery-support/linuxz64
```

– Unter Windows:

```
TADDM_CD_ISO/itm-discovery-support/win32
TADDM_CD_ISO/itm-discovery-support/win64
```

Datei `provider.properties` konfigurieren

Konfigurieren Sie optional die Datei `provider.properties`, indem Sie folgende Parameter setzen:

- `com.ibm.automationprovider.registration.host=`*`http://localhost:15210`* - aktiviert Verbindung mit ITM. Der Wert gibt eine öffentliche URL für TEPS an. Der Standardwert dieses Parameters ist `http://localhost:15210`.

  Anmerkung: Ändern Sie den Wert von 'localhost' in den Hostnamen oder die IP-Adresse des TEPS-Servers.

- `com.ibm.automationprovider.itm.curi.url=`*`http://localhost:15210`* - gibt die URL-Adresse des ITM CURI (REST)-Providers an. Der Standardwert ist `http://localhost:15210`.

  Anmerkung: Ändern Sie in diesem Fall den Wert von 'localhost' in den Hostnamen oder die IP-Adresse Ihres TEPS-Servers.

- `com.ibm.automationprovider.itm.soap.url=`*`http://localhost:1920///cms/soap`* - gibt die URL-Adresse von ITM SOAP an. Der Standardwert ist `http://localhost:1920///cms/soap`.

  Anmerkung: Ändern Sie in diesem Fall den Wert von 'localhost' in den Hostnamen oder die IP-Adresse Ihres Hub-TEPS-Servers, der sich auf demselben Host wie Ihr TEPS befinden kann oder auch nicht (in den meisten Produktionsumgebungen befinden sich TEMS und TEPS auf separaten Servern).

Anmerkung: Bei einer vom Standard abweichenden Konfiguration von ITM CURI oder ITM SOAP bzw., wenn für ITM TEPS die SSL-Sicherheit aktiviert ist (oder wenn beides der Fall ist), müssen Sie für die Parameter `com.ibm.automationprovider.itm.curi.url` und `com.ibm.automationprovider.itm.soap.url` die richtigen URLs eingeben.

Die Werte der Parameter in der Datei `provider.properties` haben Vorrang vor den über die Befehlszeile festgelegten Parameterwerten.

Fix Pack 5 Wenn Sie vorhaben, eine Erkennung auf dem RTEMS durchzuführen, vergewissern Sie sich, dass in der Umgebungsdatei der Parameter "KT1_TEMS_SECURE=YES" aktiviert ist.

OSLC Execute Automation-Service-Provider in JAZZ SM Registry Services (FRS) registrieren

Die OSLC Execute Automation-Service-Provider können Sie optional in JAZZ SM Registry Services (FRS) registrieren. Wählen Sie eine der folgenden Methoden aus:

- Fügen Sie der Datei `provider.properties` folgende Parameter hinzu:
  – `com.ibm.automationprovider.frs.url` - gibt die FRS URL-Adresse für die Registrierung des OSLC Execute Automation-Service-Providers an. Beachten Sie, dass die vollständige URL-Adresse der Objektgruppe erforderlich ist, zum Beispiel `http://9.122.100.100:9083/oslc/pr/collection`.

– `com.ibm.automationprovider.frs.user` - gibt den Benutzernamen für die Verbindung mit FRS an.
– `com.ibm.automationprovider.frs.password` - gibt das Kennwort für die Verbindung mit FRS an.
– `com.ibm.automationprovider.registration.initialdelay=5000` - gibt die Zeit zwischen dem Start des OSLC Execute Automation-Service-Providers und dem ersten Registrierungsversuch bei FRS an. Der Standardwert ist `5000` (in Millisekunden). Zur Inaktivierung der Registrierung setzen Sie diesen Parameter auf `-1`.

- Wenn Sie bei der Befehlszeilenausführung des Scripts die Option `-f` eingeben (z. B. `./automation_provider.sh -f`), werden Sie bei der Installation des Providers nach den erforderlichen Parametern gefragt.

OSLC Execute Automation-Service-Provider unter IBM Tivoli Monitoring installieren

Um einen OSLC Execute Automation-Service-Provider unter IBM Tivoli Monitoring (ITM) zu installieren, müssen Sie das Script `automation_provider` ausführen. Der OSLC Execute Automation-Service-Provider kann im interaktiven und im nicht interaktiven Modus installiert werden.

Prozedur

Zur Installation eines OSLC Execute Automation-Service-Providers führen Sie das folgende `automation_provider`-Script vom ITM TEPS-Host aus:

– Unter Linux:

```
TADDM_CD_ISO/itm-discovery-support/automation_provider.sh install
[-i <ITM-AUSGANGSVERZEICHNIS>] [-t <TEMPORÄRES_VERZEICHNIS>] [[-c <KONFIGURATIONSDATEI> | [-
h <TEPS-IP>]
 [-p <TEPS-PORT>]] [-f]
```

– Unter Windows:

```
TADDM_CD_ISO/itm-discovery-support/automation_provider.ps1 install
[-i <ITM-AUSGANGSVERZEICHNIS>] [-t <TEMPORÄRES_VERZEICHNIS>] [[-c <KONFIGURATIONSDATEI> | [-
h <TEPS-IP>]
 [-p <TEPS-PORT>]] [-f]
```

Dabei gilt:

-i *<ITM-AUSGANGSVERZEICHNIS>*
Das Installationsverzeichnis von ITM TEPS, zum Beispiel `/opt/IBM/ITM`.

-t *<TEMPORÄRES_VERZEICHNIS>*
Das Zielverzeichnis für temporäre Dateien. Der Standardwert ist `/var/log/automation_provider`.

-h *<TEPS-IP>*
Die IP-Adresse des ITM TEPS-Host.

-p *<TEPS-PORT>*
Der HTTP-Port von ITM TEPS.

-c *<KONFIGURATIONSDATEI>*
Das Ziel mit der Datei `provider.properties`, die die Konfiguration des OSLC Execute Automation-Service-Providers enthält.

-f
Ein Flag, das, wenn es gesetzt ist, bei der Installation die Parameter für die Registrierung der OSLC Execute Automation-Service-Provider in Jazz SM Registry Services anfordert.

Wichtig: Alle Parameter des Installationsscripts sind optional und können in beliebiger Reihenfolge angegeben werden.

Beispiele:

```
automation_provider.sh install -t /tmp/log -i /opt/IBM/ITM -h 9.100.100.200 -p 15210
automation_provider.ps1 install -i /opt/IBM/ITM
```

- OSLC Execute Automation-Service-Provider können im nicht interaktiven Modus installiert werden. Gehen Sie wie folgt vor:
  a) Konfigurieren Sie die Datei `provider.properties`. Informationen hierzu finden Sie im Abschnitt „Datei provider.properties konfigurieren" auf Seite 185.
  b) Führen Sie das folgende `automation_provider`-Script vom ITM TEPS-Host aus:
     – Unter Linux:

```
automation_provider.sh install  -t /tmp/log
-i /opt/IBM/ITM -c /tmp/provider.properties
```

     – Unter Windows:

```
automation_provider.ps1 install  -t /tmp/log
-i /opt/IBM/ITM -c /tmp/provider.properties
```

  Anmerkung: Bei einer vom Standard abweichenden Konfiguration von ITM CURI oder ITM SOAP bzw., wenn für ITM TEPS die SSL-Sicherheit aktiviert ist (oder wenn beides der Fall ist), sollten Sie OSLC Execute Automation-Service-Provider im nicht interaktiven Modus installieren. Stellen Sie in diesem Fall sicher, dass Sie für die Eigenschaften `com.ibm.automationprovider.itm.curi.url` und `com.ibm.automationprovider.itm.soap.url` die richtigen URLs eingeben.
- OSLC Execute Automation-Service-Provider können auch im interaktiven Modus installiert werden. Hierbei müssen die erforderlichen Parameter bei der Installation eingegeben werden, wie im Abschnitt „Datei provider.properties konfigurieren" auf Seite 185 beschrieben.

Installation des OSLC Execute Automation-Service-Providers prüfen

Sie können nun prüfen, ob die Installation des OSLC Execute Automation-Service-Providers unter IBM Tivoli Monitoring erfolgreich ausgeführt wurde.

Vorgehensweise

1. Stellen Sie mit folgendem Befehl sicher, dass ITM TEMS über einen aktiven Windows-, Linux- oder UX-Agenten verfügt:

```
/opt/IBM/ITM /bin/tacmd login -u admin -p password -s localhost
/opt/IBM/ITM /bin/tacmd listSystems
```

2. Stellen Sie sicher, dass jeder ITM TEMS über einen Automationsplan verfügt. Die Pläne müssen die IP-Adressen der ITM-Endpunkte enthalten. Öffnen Sie die folgenden Webadressen in Ihrem Web-Browser:

```
http://<ITM_TEPS>:<ITM_PORT>/itmautomationprovider
http://<ITM_TEPS>:<ITM_PORT>/itmautomationprovider/services/plans
```

   Beispiel

```
http://9.100.200.100:15210/itmautomationprovider/services/plans
```

Status des ITM OSLC Execute Automation-Service-Providers prüfen

Sie können den Status der ITM OSLC Execute Automation-Service-Provider-Installation prüfen.

Prozedur

- Führen Sie das folgende `automation_provider`-Script aus:
  – Unter Linux:

```
automation_provider.sh status [i- <ITM-AUSGANGSVERZEICHNIS>] [t- <TEMP-VERZEICHNIS>]
```

– Unter Windows:

```
automation_provider.ps1 status [i- <ITM-AUSGANGSVERZEICHNIS>] [t- <TEMP-VERZEICHNIS>]
```

Dabei gilt:

i- *<ITM-AUSGANGSVERZEICHNIS>*
Das Installationsverzeichnis von ITM TEPS, zum Beispiel `/opt/IBM/ITM`.

t- *<TEMP-VERZEICHNIS>*
Das Zielverzeichnis für temporäre Dateien. Der Standardwert ist `/var/log/automation_provider`.

Wichtig: Alle Parameter des Scripts sind optional und können in beliebiger Reihenfolge angegeben werden.

Beispiele

```
automation_provider.sh status
automation_provider.ps1 status -i /opt/IBM/ITM
automation_provider.sh status -t /tmp/log -i /opt/IBM/ITM
```

OSLC Execute Automation-Service-Provider deinstallieren

Der ITM OSLC Execute Automation-Service-Provider kann mit dem Script `automation_provider` deinstalliert werden.

Prozedur

- Führen Sie das folgende `automation_provider`-Script aus:
  – Unter Linux:

```
automation_provider.sh uninstall [i- <ITM-AUSGANGSVERZEICHNIS>] [t- <TEMP-VERZEICHNIS>]
```

  – Unter Windows:

```
automation_provider.ps1 uninstall [i- <ITM-AUSGANGSVERZEICHNIS>] [t- <TEMP-VERZEICHNIS>]
```

  Dabei gilt:

  i- *<ITM-AUSGANGSVERZEICHNIS>*
  Das Installationsverzeichnis von ITM TEPS, zum Beispiel `/opt/IBM/ITM`.

  t- *<TEMP-VERZEICHNIS>*
  Das Zielverzeichnis für temporäre Dateien. Der Standardwert ist `/var/log/automation_provider`.

  Wichtig: Alle Parameter des Scripts sind optional und können in beliebiger Reihenfolge angegeben werden.

  Beispiele

```
automation_provider.sh uninstall
automation_provider.ps1 uninstall -i /opt/IBM/ITM
automation_provider.sh uninstall -t /tmp/log -i /opt/IBM/ITM
```

Erkennung des ITM OSLC Execute Automation-Service-Providers konfigurieren

Wenn Sie ITM OSLC Execute Automation Service Provider verwenden, können Sie den Erkennungsprozess durch Einstellung der folgenden Eigenschaften konfigurieren.

com.collation.discover.dwcount=32
Der Standardwert ist 32.

Diese Eigenschaft ist eine TADDM-Server-Eigenschaft, die die Anzahl der Worker-Threads der Erkennung definiert.

Die besten Ergebnisse erzielen Sie, wenn Sie die Eigenschaften `com.collation.discover.dwcount` und KT1_RPC_THREADS auf denselben Wert setzen.

com.ibm.automationprovider.kt1.concurenttasks.limit=100
Der Standardwert ist 100.

Diese Eigenschaft ist eine ITM OSLC Execute Automation Service Provider-Eigenschaft, die in der Datei `provider.properties` bearbeitet werden kann. Sie definiert die Anzahl der gleichzeitigen Anforderungen, die der Provider für TEMS ausgibt. Darüber hinausgehende Anforderungen werden auf der Provider-Ebene in die Warteschlange gestellt.

Anmerkung: Ändern Sie den Wert dieser Eigenschaft nur, wenn eine zusätzliche Regulierung zwischen TADDM und TEMS erforderlich ist oder wenn mehr als 100 KT1-Arbeitsthreads eingestellt wurden.

KT1_RPC_THREADS=10
Der Standardwert ist 10.

Dies ist eine ITM TEMS-Eigenschaft, die in der Datei `ITM_HOME/config/kbbenv.ini` bearbeitet werden kann. Sie definiert die Anzahl von Arbeitsthreads, die auf KT1-Anforderungen antworten.

Die besten Ergebnisse erzielen Sie, wenn Sie die Eigenschaften KT1_RPC_THREADS und `com.collation.discover.dwcount` auf denselben Wert setzen.

Fix Pack 6 Fehlerbehebung bei der Erkennung von OSLC-Automation

In diesem Abschnitt werden die bewähren Verfahren bei der TADDM ITM OSLC-Erkennung im Hinblick auf die Leistung aufgeführt. Für die Erkennung mehrerer Ziele mit einer auf TADDM ITM OSLC basierten Erkennung beachten Sie die folgenden Punkte:

1. Öffnen Sie einen Fall mit ITM-Unterstützung für die folgenden Programmkorrekturen, falls diese nicht in Ihrem ITM-Release verfügbar sind.

APAR IJ02368 ITM OSLC TADDM discovery performance
Der Abschluss von TADDM-Erkennungen über ITM dauert zu lang und TACMD ist möglicherweise blockiert. Es ist außerdem möglich, den Zeitlimitwert über die Umgebungsvariable KT1_SOAP_EXECUTE_TIMEOUT=20 zu überschreiben, die in der Datei TEPS `cq.ini` definiert ist.

Ein Anforderungszeitlimit von 20 Sekunden ist für die meisten Anforderungen ausreichend. Die Verwendung einer Erkennungsrate von 300 Sekunden (Standardeinstellung) beschränkt eine Umgebung, in der selbst eine kleine Anzahl von Anforderungen zu einer Zeitlimitüberschreitung führt.

APAR IJ02662 TADDM discoveries via ITM monopolies the hub's soap thread and take too long
Diese APAR unterstützt das Zwischenspeichern des Knotenstatus und beseitigt einen Großteil dieser Anforderungen. Das Zwischenspeichern wird über die folgende Umgebungsvariable ermöglicht, wobei 120 für die Anzahl der Sekunden steht, die ein Eintrag im Zwischenspeicher verbleibt.

`KT1_NODE_STATUS_CACHE_TTL=120` definiert in der Datei TEPS `cq.ini`.

Die Programmkorrektur APAR IJ01062 optimiert Anforderungen an die Tabelle mit der Knotenliste, die in IJ02662 enthalten ist.

Gehen Sie zur Installation folgendermaßen vor:

- Ersetzen Sie einige der von TEPS verwendeten Bibliotheken
- Definieren Sie zwei neue Umgebungsvariablen
- Starten Sie TEPS erneut

Auf der Hub-Seite gibt es keine Änderungen.

2. Öffnen Sie einen Fall mit TADDM-Unterstützung, um Programmkorrekturen zur Leistung für den Automation-Provider für APAR IJ12778 abzurufen. Aktuell umfasst die Programmkorrektur die folgenden Dateien: `Execute.class. June14 KT1Wrapper.class.may23`

Erstellen Sie eine Sicherungskopie und ersetzen Sie die Dateien in TEPS. Entfernen Sie das Datumssuffix, bestätigen Sie die korrekten Berechtigungen und starten Sie TEPS erneut. In den Übungssystemen befinden sich die Dateien in den folgenden Verzeichnissen:

```
/opt/IBM/ITM/lx8266/iw/profiles/ITMProfile/installedApps/ITMCell
/itmautomationprovider.ear/itmautomationprovider.war/WEB-INF/cla
sses/com/ibm/cdb/integration/actions/Execute.class
and KT1Wrapper.class is here;
/opt/IBM/ITM/lx8266/iw/profiles/ITMProfile/installedApps/ITMCell
/itmautomationprovider.ear/itmautomationprovider.war/WEB-INF/cla
sses/com/collation/platform/session/
and restart TEPS
```

Es ist wichtig, dass alle ITM- und TADDM-Patches in einem Prozess angewendet werden. Bei allen APARs handelt es sich um TEPS-Fixes, genauer gesagt, es handelt sich um Fixes für die KT1-Komponente, auf die TEPS für die Ausgabe von Erkennungsanforderungen für den Hub angewiesen ist. Nach dem erneuten Start von TEPS überprüfen Sie nach 10 Minuten, ob die Pläne verfügbar sind. Ist dies nicht der Fall, überprüfen Sie das TEPS-Protokoll 'SystemOut.log' auf Fehler.

Wenn sie verfügbar sind, führen Sie den OSLCAutomationAgent auf TADDM aus, um die Datenbank mit den neuen Inhalten zu aktualisieren:

```
dist/support/bin/runtopobuild.sh -w -a OSLCAutomationAgent--forceScopeSetRefresh true
```

Hier finden Sie weitere bewähre Verfahren aus früheren Ausgaben, die Sie in Betracht ziehen können.

Vorschläge für 'collation.properties':

- Wenn Sie keine direkte Erkennung von Servern ausführen (Erkennung ohne ITM), setzen Sie diese Eigenschaft auf 'true', da das Caching für die Authentifizierung während der OSLC-Erkennung zu Problemen führen kann, wenn zu viele Threads auf einmal die gleiche Authentifizierung verwenden:

  ```
  com.ibm.cdb.security.auth.cache.disabled=false
  ```

  Anmerkung: In APAR IJ01289 ist der Standardwert mit der neuen **ITM only**-Eigenschaft `com.ibm.cdb.security.auth.cache.itm.disabled` 'true'. Überspringen Sie daher bei FP5 diesen Schritt.
- Wenn Sie das vorkonfigurierte Level-3-Erkennungsprofil verwenden und erwarten, dass dieses OSLC verwendet, setzen Sie den Wert auf 'true', da standardmäßig 'false' festgelegt ist:

  ```
  com.ibm.cdb.session.prefer.OSLCAutomation.Level_3_Discovery=false
  ```
- Wenn Agenten zwischen TEMS verschoben werden können, setzen Sie diese Eigenschaft im PSS auf 'true', um sicherzustellen, dass der Cache jedes Mal vollständig aktualisiert wird:

  ```
  com.ibm.cdb.topobuilder.integration.oslc.automation.scope.always refresh=true
  ```

Sie können `runtopobuild` vor der Erkennung manuell ausführen, um sicherzustellen, dass die Agenten ordnungsgemäß zugeordnet werden, oder, wenn diese nicht so häufig verschoben werden, behalten Sie es so bei. Standardmäßig sollte der Agent mehrmals am Tag ausgeführt werden (siehe Eigenschaft 'com.ibm.cdb.topobuilder.groupinterval.integration'). Das Auftreten von Fehlern wie beispielsweise "CTJTP1404E Request failed with the following error returned from the Automation Provider: Could not execute remote command automation plan: Specified ip address 1.2.3.4 is not allowed for given AutomationPlan" zeigt an, dass der TADDM-Cache veraltet ist und der Agent ausgeführt werden muss.

Stellen Sie sicher, dass die Anzahl der TEMS-SOAP-Threads höher als der Wert der Eigenschaft 'dwcount' ist (2 x dwcount wird als optimal betrachtet).

Der Standardwert für 'dwcount' in `dist/etc/collation.properties` beträgt 32.

Die Eigenschaft 'dwcount' steht für die Anzahl der Sensoren, die gleichzeitig ausgeführt werden können. Der Wert dieser Eigenschaft basiert auf Ihrer Kapazität von TADDM und ITM. Der Standardwert ist normalerweise ausreichend. Wenn der Wert von 'dwcount' höher ist als der Standardwert, sollten Sie die maximale Hauptspeicherkapazität für den TADDM-Erkennungsservice erhöhen und JVMs über '-Xmx jvmarg' erkennen.

Informationen zum Festlegen der Anzahl der SOAP-Threads auf 2 x 'dwcount' finden Sie hier: https://www.ibm.com/developerworks/community/blogs/0587adbc-8477-431f8c689226adea11ed/entry/tacmd_commands_running_slow_and_the_numebr_of_SOAP_processes?lang=en

Weitere ITM-Änderungen:

```
KT1_RPC_THREADS=10
```

```
Der Standardwert ist 10.
```

Dies ist eine ITM TEMS-Eigenschaft, die in der Datei `ITM_HOME/config/kbbenv.ini` bearbeitet werden kann. Sie definiert die Anzahl von Arbeitsthreads, die auf KT1-Anforderungen antworten.

Die besten Ergebnisse erzielen Sie, wenn Sie die Eigenschaften KT1_RPC_THREADS und `com.collation.discover.dwcount` auf denselben Wert setzen.

Im normalen Betrieb sollten die folgenden Eigenschaften in `provider.properties` auf 'false' gesetzt sein, da andernfalls die Ausgabedateien beibehalten werden, bis sie manuell gelöscht werden:

```
com.ibm.automationprovider.temp.remote.keepoutputs=false
com.ibm.automationprovider.temp.local.keepfiles=false
com.ibm.automationprovider.temp.remote.keepscripts=false
```

TADDM durch OSLC Automation mit anderen Produkten integrieren

TADDM kann durch Open Services for Lifecycle Collaboration (OSCL) Automation mit anderen Produkten integriert werden. TADDM stellt eine Verbindung mit dem OSLC Execute Automation-Service-Provider her, der Daten zu den Infrastrukturen anderer Produkte bereitstellt, die von TADDM mit OSLC Automation Session erkannt werden können.

Die Erkennung mit dem OSLC Execute Automation-Service-Provider ist ein generischer Prozess, der jedoch erweitert werden kann, so dass er auch andere Produkte mit eigenen OSLC Execute Automation-Service-Providern erkennt. Während der Erkennung wird ein Port pro Host für einen OSLC Execute Automation-Service-Provider bzw. für Jazz SM Registry Services geöffnet. Dadurch kann die Sicherheit besser gewährleistet werden.

In der folgenden Tabelle finden Sie Themen mit weiteren Informationen zur Erkennung via OSLC.

Tabelle 43. Themen mit weiteren Informationen zur Erkennung via OSLC.

Informationen	**Speicherort**
Umgebung für die Erkennung konfigurieren	„Erkennung mit OSLC Automation Session konfigurieren” auf Seite 107
TADDM-Servereigenschaften	„Eigenschaften für die Erkennung mit OSLC Automation Session” auf Seite 79
Sensoren mit Unterstützung für die Erkennung mit OSLC Automation Session	Siehe auch *Sensoren mit Unterstützung für die Erkennung mit OSLC Automation Session* in der *Sensorreferenz* zu TADDM.

OSLC Execute Automation-Service-Provider
Der OSLC Execute Automation-Service-Provider stellt TADDM Daten zu den IP-Adressen von Endpunkten bereit, die von anderen Produkten verwaltet werden. Die Daten werden zur Erkennung von Endpunkten mit OSLC Automation Session verwendet.

TADDM kann das Ziel für den OSLC Execute Automation-Service-Provider über die Jazz SM Registry Services oder aus der Datei `collation.properties` erhalten.

TADDM kann mit mehreren OSLC Execute Automation-Service-Providern direkt verbunden werden, aber auch mit einer einzelnen Instanz der Jazz SM Registry Services, in der mehrere OSLC Execute Automation-Service-Provider registriert sein können. Jeder OSLC Execute Automation-Service-Provider speichert

die Informationen einer Instanz eines bestimmten Produkts, mit dem TADDM integriert ist (z. B. ITM HUB).

Zugehörige Verweise

„ITM OSLC Execute Automation-Service-Provider" auf Seite 180
Der ITM OSLC Execute Automation-Service-Provider wird für den Import von IP-Adressdaten der von IBM Tivoli Monitoring verwalteten Endpunkte in TADDM und zur Erkennung der IBM Tivoli Monitoring-Endpunkte mit OSLC Automation Session verwendet.

TADDM für den OSLC Execute Automation-Service-Provider konfigurieren

Zur Ausführung der Erkennung mit OSLC Automation Session muss TADDM für die Verwendung des OSLC Execute Automation-Service-Providers konfiguriert sein.

Vorgehensweise

Zur Konfiguration von TADDM für die Verwendung des OSLC Execute Automation-Service-Providers führen Sie die folgenden Schritte aus:

1. Stellen Sie sicher, dass der OSLC Execute Automation-Service-Provider installiert ist und ausgeführt wird.
2. Stellen Sie eine Verbindung zwischen TADDM und dem OSLC Execute Automation-Service-Provider her. Diese Verbindung kann direkt oder über Jazz for Service Management Registry Services erfolgen. Bei Verwendung mehrerer OSLC Execute Automation-Service-Provider können auch beide Methoden kombiniert werden.
   - Für die direkte Verbindung zwischen TADDM und einem OSLC Execute Automation-Service-Provider fügen Sie die OSLC Execute Automation-Service-Provider in der Datei `collation.properties` zur Eigenschaft `com.ibm.cdb.topobuilder.integration.oslc.automationprovider` hinzu.
   - Für die Verbindung zwischen TADDM und einem OSLC Execute Automation-Service-Provider über Jazz for Service Management Registry Services müssen Sie das TADDM Jazz SM Registry Services-Lookup für die OSLC Execute Automation-Service-Provider aktivieren. Gehen Sie wie folgt vor:
     a. Stellen Sie sicher, dass die Jazz SM Registry Services ausgeführt werden.
     b. Stellen Sie sicher, dass die OSLC Execute Automation-Service-Provider mit den Jazz SM Registry Services verbunden sind.
     c. Stellen Sie in der Datei `collation.properties` über eine der folgenden Eigenschaften die Adresse der Jazz SM Registry Services bereit:

```
com.ibm.cdb.topobuilder.integration.oslc.frsurl
com.ibm.cdb.topobuilder.integration.oslc.automation.frsurl
```

3. Starten Sie den TADDM-Server erneut.

Ergebnisse

Nach der Konfiguration von TADDM können Sie die Erkennung mit OSLC Automation Session ausführen.

Zugehörige Verweise

„Eigenschaften für die Erkennung mit OSLC Automation Session" auf Seite 79
Diese Eigenschaften gelten für die Erkennung mit OSLC Automation Session.

Befehlszeilenschnittstelle für OSLCAutomationAgent

OSLCAutomationAgent wird für die Erfassung der Daten von OSLC Execute Automation-Service-Providern verwendet. Mit den entsprechenden Befehlen können Sie den Agenten manuell ausführen und die von ihm erstellten Bereichsgruppen aktualisieren.

Die Adressen der OSLC Execute Automation-Service-Provider können in der Datei `collation.properties` konfiguriert oder aus Jazz SM Registry Services heruntergeladen werden (oder auch beides). Der Agent stellt eine Verbindung zu jedem OSLC Execute Automation-Service-Provider her, um die Liste der mit TADDM kompatiblen Automationspläne abzurufen. Der Automationsplan enthält die IP-Adressen, die der Agent zum Zwischenspeichern und Erstellen der Erkennungsbereichsgruppen verwendet. Wenn

TADDM beispielsweise mit IBM Tivoli Monitoring integriert ist, enthält der Automationsplan die IP-Adressen der von IBM Tivoli Monitoring verwalteten ITM TEMS-Server und -Endpunkte (Agenten). OSLCAutomationAgent speichert und erstellt die Bereichsgruppen mit den IP-Adressen der IBM Tivoli Monitoring-Agenten. Jede ITM TEMS-Instanz hat eine eigene Bereichsgruppe.

OSLCAutomationAgent wird regelmäßig in der Gruppe 'Integration Agents' ausgeführt.

An OSLCAutomationAgent können die folgenden Befehle ausgeführt werden.

- Zur manuellen Ausführung des Agenten:

```
/taddm/dist/support/bin/runtopobuild.sh -a OSLCAutomationAgent
```

- Zur Aktualisierung der Bereichsgruppen:

```
/taddm/dist/support/bin/runtopobuild.sh -a OSLCAutomationAgent -s true
```

  Anmerkung: Die Bereichsgruppen werden nur aktualisiert, wenn sich der Automationsplan des ITM-Automationsproviders geändert hat. Erzwingen können Sie die Aktualisierung der Bereichsgruppen mit dem folgenden Befehl:

```
/taddm/dist/support/bin/runtopobuild.sh -a OSLCAutomationAgent
--forceScopeSetRefresh true
```

  Die Bereichsgruppen werden in der Discovery Management Console in der Anzeige **Scopes** (Bereiche) aufgeführt.

- Zur Anzeige der zwischengespeicherten Bereichsgruppen:

```
/taddm/dist/support/bin/runtopobuild.sh -a OSLCAutomationAgent -d true
/taddm/dist/support/bin/runtopobuild.sh -a OSLCAutomationAgent
 --displayCache true
```

  Die Bereichsgruppen werden in den folgenden Protokolldateien von TADDM angezeigt:

  – <COLLATION_HOME>/dist/log/services/TopologyBuilder.log

  – <COLLATION_HOME>/dist/log/agents/OSLCAutomationAgent.log

  Folgendes Beispiel zeigt die Ausgabe in der Datei <COLLATION_HOME>/dist/log/agents/OSLCAutomationAgent.log:

```
2014-07-22 11:42:54,660 TopologyBuilder [pool-1-thread-1]  DEBUG
oslc.OSLCAutomationAgent - OSLCAutomationAgent:displaying cache
2014-07-22 11:42:54,669 TopologyBuilder [pool-1-thread-1]  INFO
 oslc.OSLCAutomationAgent - <AGENT_IP_2> http://9.120.100.100:15210/
itmautomationprovider/services/plans/2 1406009933764
2014-07-22 11:42:54,669 TopologyBuilder [pool-1-thread-1]  INFO
 oslc.OSLCAutomationAgent - <AGENT_IP_2> http:///9.120.100.100:15210/
itmautomationprovider/services/plans/2 1406009933764
2014-07-22 11:42:54,675 TopologyBuilder [pool-1-thread-1]  DEBUG
 oslc.OSLCAutomationAgent - OSLCAutomationAgent:cache end
```

Zugehörige Konzepte

„Übersicht über den Topologieerstellungsprozess” auf Seite 14

TADDM führt den Topologieerstellungsprozess in regelmäßigen Abständen aus. Bis der Topologieerstellungsprozess nach einer Erkennung oder nach einer Operation des Dienstprogramms zum Laden von Massendaten abgeschlossen ist, können in der TADDM-Datenbank nicht abgeglichene Objekte vorhanden sein und die Topologiebeziehungen können unvollständig sein.

TADDM mit IBM Tivoli Monitoring integrieren (altes Verfahren)

Je nach den Tasks, die Sie in Ihrer IT-Umgebung ausführen müssen, können Sie die Integrationsfunktionen verwenden, die zwischen IBM Tivoli Application Dependency Discovery Manager (TADDM) und IBM Tivoli Monitoring verfügbar sind. Sie können TADDM mithilfe des IBM Tivoli Monitoring Scope-Sensors mit IBM Tivoli Monitoring integrieren.

Neues Integrationsverfahren

Wichtig: Ab TADDM Version 7.3.0 sollte die Integration mit IBM Tivoli Monitoring 6.3 mithilfe von OSLC Automation erfolgen. Das Integrationsverfahren unter Verwendung des IBM Tivoli Monitoring Scope-Sensors ist veraltet und steht in künftigen Releases nicht mehr zur Verfügung.

Weitere Informationen zur TADDM-Integration mit IBM Tivoli Monitoring über OSLC Automation finden Sie im Abschnitt „TADDM über OSLC Automation mit IBM Tivoli Monitoring integrieren" auf Seite 178; Informationen zu Sensoren, die die Erkennung mithilfe von OSLC Automation unterstützen, finden Sie in der TADDM-*Sensorreferenz* im Abschnitt *Sensoren mit Unterstützung für die Erkennung mit OSLC Automation Session*.

Altes Integrationsverfahren

Alle folgenden Abschnitte beziehen sich auf das alte Integrationsverfahren. Sie können es zwar weiter verwenden, sollten aber beachten, dass dieses Verfahren veraltet ist und in künftigen Releases nicht mehr verfügbar sein wird.

In Tabelle 1 werden einige der Tasks, die Sie möglicherweise ausführen möchten, zusammen mit den Integrationsfunktionen aufgeführt, die Sie verwenden sollten. Die restlichen Abschnitte bieten einen Überblick über diese Integrationsfunktionen.

Tabelle 44. Benutzertasks mit den zugehörigen zu verwendenden Integrationsfunktionen

Task	Verwendbare Integrationsfunktion
Einen Einblick in die Verfügbarkeit erhalten Sie, indem Sie die Betriebssystemeinstellungen, Anwendungseinstellungen und Änderungsprotokolle der Systeme anzeigen, die durch IBM Tivoli Monitoring überwacht werden.	• „Erkennung mithilfe von IBM Tivoli Monitoring" auf Seite 194 • „Kontextbezogen aufrufen" auf Seite 196
Stellen Sie sicher, dass die Verfügbarkeit der durch TADDM erkannten Betriebssysteme überwacht wird.	• „Erkennung mithilfe von IBM Tivoli Monitoring" auf Seite 194 • „Überwachungsabdeckungsberichte" auf Seite 196
Zeigen Sie die Verfügbarkeit und Leistung von durch TADDM erkannten Systemen an.	• „IBM Tivoli Monitoring-DLA" auf Seite 195 • „Überwachungsabdeckungsberichte" auf Seite 196
Überwachen Sie eine Geschäftsanwendung in Bezug auf Konfigurationsänderungen.	• „Erkennung mithilfe von IBM Tivoli Monitoring" auf Seite 194 • „Änderungsereignisse" auf Seite 196 • „Kontextbezogen aufrufen" auf Seite 196

Erkennung mithilfe von IBM Tivoli Monitoring

TADDM kann mithilfe einer IBM Tivoli Monitoring-Infrastruktur der Version 6.2.1 oder höher Erkennungen der Ebene 1 und 2 und einige Erkennungen der Ebene 3 durchführen. TADDM verwendet bei der Erkennung von Konfigurationselementen in der IBM Tivoli Monitoring-Umgebung nicht die Berechtigungsnachweise für jeden Computer, der durch den Portalserver überwacht wird, sondern ausschließlich die Berechtigungsnachweise für Ihren Tivoli Enterprise Portal Server.

TADDM nutzt die Tivoli Monitoring-Infrastruktur folgendermaßen auf zweierlei Weise:

- TADDM ruft die Liste der Tivoli Monitoring-Endpunkte vom Tivoli Enterprise Portal Server ab, um einerseits Informationen für grundlegende Erkennungen der Ebene 1 zu erstellen und andererseits Bereiche für genauere Erkennungen der Ebene 2 und 3 zu erstellen.

- TADDM verwendet die Infrastruktur von Tivoli Monitoring, um in der Befehlszeilenschnittstelle (CLI) auf Zielsystemen Befehle für Erkennungen der Ebene 2 und 3 an die Sensoren auszugeben und ebenso um die Ausgabe dieser Befehle zu erfassen.

Diese Funktionalität bietet folgende Vorteile:

- Schnelle Implementierung von TADDM in bereits vorhandenen Tivoli Monitoring-Umgebungen
- TADDM-Ankerserver und -Gateway-Server sind nicht erforderlich
- Die Definition von festgelegten Bereichen, die die zu durchsuchenden Computer enthalten, ist nicht erforderlich. Es wird ausschließlich ein Bereich mit einem einzigen Eintrag für den Tivoli Enterprise Portal Server benötigt.
- Die Definition einer Zugriffsliste (Betriebssystem-Berechtigungsnachweise) für Erkennungsziele ist nicht erforderlich.
- Es wird ausschließlich ein einziger Zugriffslisteneintrag für die Anmeldung an der GUI des Tivoli Enterprise Portal Servers benötigt.

Tabelle 45. Artikel mit weiteren Informationen zur Erkennung mithilfe von IBM Tivoli Monitoring

Informationen	**Position der Informationen**
Erkennung mit IBM Tivoli Monitoring konfigurieren	„Erkennung mit IBM Tivoli Monitoring konfigurieren (altes Verfahren)" auf Seite 105
TADDM-Servereigenschaften für die Erkennung mit IBM Tivoli Monitoring	„Eigenschaften für Erkennung mit IBM Tivoli Monitoring (altes Verfahren)" auf Seite 77
• Sensoren mit Unterstützung für die Erkennung mit IBM Tivoli Monitoring • Informationen zum IBM Tivoli Monitoring Scope-Sensor sowie zur Konfiguration des Sensors und zur Fehlerbehebung bei allen bekannten Problemen, die bei der Implementierung oder Verwendung des Sensors auftreten können	TADDM *Referenzinformationen zu Sensoren*

IBM Tivoli Monitoring-DLA

Der DLA (Discovery Library Adapter = Erkennungsbibliotheksadapter) von IBM Tivoli Monitoring extrahiert Konfigurationsdaten von Tivoli Monitoring über die Computersysteme und Datenbanken, die durch Tivoli Monitoring überwacht werden. Die Ausgabe des DLA ist eine formatierte XML-Datei, die diese Komponenten sowie ihre Beziehungen enthält. Die Ausgabe des DLA enthält zudem Daten, die Tivoli Monitoring-Agenten darstellen, sowie Daten, die zum Starten von Verfügbarkeitsansichten über TADDM verwendet werden. Detaillierte Informationen zum Laden der DLA-exportierten Daten in TADDM siehe den Abschnitt *Das Massenladeprogramm* im TADDM-*Benutzerhandbuch*.

Gehen Sie zur Ausführung des Discovery Library Adapter (DLA) wie folgt vor:

1. Generieren Sie den DLA in ITM, wie im Abschnitt *Using the Tivoli Management Services Discovery Library Adapter* unter http://www-01.ibm.com/support/knowledgecenter/SSTFXA_6.2.2.1/com.ibm.itm.doc_6.2.2fp1/discoverylibraryadapter_tms.htm?lang=en angegeben.
2. Kopieren Sie die DLA-Ausgabedatei auf den TADDM-Host.
3. Laden Sie den DLA mit dem Massenladeprogramm aus ITM in TADDM. Verwenden Sie folgenden Befehl:

```
$COLLATION_HOME/bin/loadidml.sh -u Benutzer -p Kennwort -f Pfad_zu_DLA
```

Wenn Sie neue Tivoli Monitoring-Agenten installieren, können sie dem DLA für Tivoli Monitoring zusätzliche Unterstützung bieten. Die Agenten liefern Informationen für die Überwachungsabdeckungsberichte. Nur für die Überwachungsabdeckungsberichte von Betriebssystemen ist kein DLA erforderlich.

Wenn Sie Agenten installieren, müssen Sie auch die Anwendungsunterstützung für diese Agenten aktivieren, um sicherzustellen, dass die Agenten die vom DLA erstellte Ausgabe empfangen. Nicht alle Agenten unterstützen Tivoli Monitoring DLA.

Informationen zur Konfiguration der Anwendungsunterstützung für nicht standardgemäße Agenten finden Sie in der jeweiligen Dokumentation. Informationen dazu, ob ein Agent den DLA für Tivoli Monitoring unterstützt, finden Sie in der Dokumentation zum Agenten für IBM Tivoli Composite Application Manager.

Überwachungsabdeckungsberichte

Die Überwachungsabdeckungsberichte enthalten Details zu verschiedenen Komponenten in Ihrer Umgebung. Sie können einen Bericht für Betriebssysteme, Datenbanken, Microsoft-Anwendungen, VMware-Server und System p-Komponenten in Ihrer Umgebung erstellen. Diese Komponenten werden Agenten für IBM Tivoli Monitoring 6.1 oder höher überwacht.

Weitere Informationen zu Überwachungsabdeckungsberichten finden Sie im TADDM-*Benutzerhandbuch*.

Änderungsereignisse

Sie können TADDM so konfigurieren, dass IBM Tivoli Monitoring bei der Feststellung von Änderungen einer erkannten Ressource benachrichtigt wird.

Tabelle 46. Abschnitte mit weiteren Informationen zu Änderungsereignissen

Informationen	Position der Informationen
• TADDM für das Senden von Änderungsereignissen konfigurieren • IBM Tivoli Monitoring-Datenanbieter konfigurieren • Änderungsereignisse für ein Business-System konfigurieren	„Änderungsereignisse an externe Systeme senden” auf Seite 204

Kontextbezogen aufrufen

Mithilfe von Launch-in-Context können Sie TADDM-Daten in den Tivoli Enterprise Portal-Ansichten von IBM Tivoli Monitoring anzeigen.

Durch die Konfiguration von Topologieansichten zur Anzeige im Tivoli Enterprise Portal können Sie die physische Infrastruktur und die Business-System-Topologien innerhalb der Verfügbarkeitsansichten von Tivoli Enterprise Portal anzeigen.

Tabelle 47. Abschnitte, die weitere Informationen zu Launch-in-Context enthalten

Informationen	Position der Informationen
Für die Anzeige von Topologieansichten erforderliche URLs	„Konfiguration für den Start im Kontext” auf Seite 201
Anweisungen zur Konfiguration von Launch-in-Context zur Anzeige der Betriebssystemeinstellungen, Anwendungseinstellungen und Änderungsprotokolle für eingehende Änderungsereignisse	„Detaillinks in Konfigurationsänderungs-Ereignisberichten in IBM Tivoli Monitoring erstellen” auf Seite 213

Konfigurationselemente für den Kontextmenüservice und den Datenintegrationsservice registrieren

Wenn der Kontextmenüservice (Context Menu Service, CMS) und der Datenintegrationsservice (Data Integration Service, DIS) eingesetzt werden, um flexible produktübergreifende Startpunkte zu erhalten, müssen Sie TADDM-Konfigurationselemente (Configuration Items, CIs) in der CMS/DIS-Datenbank speichern.

Vorbereitende Schritte

Vor Verwendung des Context Menu Service (CMS) und des Data Integration Service (DIS) müssen Sie die CMS/DIS-Datenbank einrichten.

Informationen zu diesem Vorgang

TADDM-Konfigurationselemente werden auf folgende Weise in der CMS/DIS-Datenbank registriert:

- Durch eine Erstregistrierung mithilfe des CMS/DIS-Registrierungsscripts.
- Durch regelmäßige automatische Aktualisierungen durch den CMSDISAgent-Agenten für die Topologieerstellung.

Erstregistrierung durchführen

Um eine Erstregistrierung der TADDM-Konfigurationselemente (Configuration Items, CIs) in der CMS/DIS-Datenbank (Context Menu Service, Data Integration Service) durchzuführen, müssen Sie das Script **run_cms_dis_registration** manuell ausführen. Der CMSDISAgent-Agent für die Topologieerstellung nimmt erst nach Abschluss der Erstregistrierung eine automatische Aktualisierung der Konfigurationselement-Registrierung vor.

Informationen zu diesem Vorgang

Bei Verwendung einer Streaming-Server-Implementierung wird das Registrierungsscript auf dem primären Speicherserver ausgeführt. Bei Verwendung einer Synchronisationsserverimplementierung muss das Script auf dem Synchronisationsserver ausgeführt werden.

Vorgehensweise

So führen Sie eine Erstregistrierung der TADDM-Konfigurationselemente durch:

1. Wechseln Sie in einer Eingabeaufforderung in das Verzeichnis `$COLLATION_HOME/bin`.
2. Führen Sie das Script **run_cms_dis_registration** für Ihr Betriebssystem aus:
   - Linux- und UNIX-Systeme:

     ```
     ./run_cms_dis_registration.sh [ register [GUID] |
                                     clean [GUID [Klassentyp]] |
                                     re-register-all | register-menu |
                                     help ]
     ```

   - Unter Windows-Systemen:

     ```
     run_cms_dis_registration.bat [ register [GUID] |
                                     clean [GUID [Klassentyp]] |
                                     re-register-all | register-menu |
                                     help ]
     ```

   Dabei gilt Folgendes:

   register [*GUID*]

   Die TADDM-Daten werden in der CMS/DIS-Datenbank registriert. Sie haben auch die Möglichkeit, die global eindeutige ID (GUID) eines Modellobjekts anzugeben, das registriert werden soll.

   Bei der ersten Ausführung, bei der das Script unter Angabe der Option `register` und ohne Angabe einer GUID ausgeführt wird, werden alle TADDM-Daten in der Datenbank registriert und alle Startpunkte im Kontextmenüservice (CMS) registriert. Bei jeder weiteren Ausführung des Scripts

unter Angabe dieser Option werden nur die Änderungen an TADDM-Daten registriert, die seit der letzten Ausführung vorgenommen wurden. Diese Option ist die Standardoption.

Bei Angabe einer GUID wird nur das über die GUID angegebene Modellobjekt registriert.

Anmerkung: Die Erstregistrierung aller TADDM-Daten kann einige Zeit in Anspruch nehmen.

clean [*GUID* [*Klassentyp*]]

Die Registrierung von TADDM-Daten in der Datenbank wird zurückgenommen.

Wird keine GUID angeben, wird die Registrierung aller TADDM-Daten zurückgenommen. Bei Angabe einer GUID wird nur die Registrierung des über die GUID angegebenen Modellobjekts zurückgenommen. Steht das über die GUID angegebene Modellobjekt nicht mehr in TADDM zur Verfügung, müssen Sie auch den Modellobjekttyp angeben.

re-register-all
Die Registrierung aller TADDM-Daten und Startpunkte wird zurückgenommen und anschließend die Erstregistrierung wiederholt. Diese Option entspricht einer Ausführung des Scripts unter Angabe der Option `clean` und anschließend unter Angabe der Option `register`.

register-menu
Aktualisiert nur die Menüdefinitionen, die in der Datenbank des Kontextmenüservice registriert sind. Verwenden Sie diese Option, wenn die TADDM-Daten registriert sind, Sie jedoch nur die Menüdefinitionen aktualisieren möchten.

help
Es werden Hilfeinformationen zum Script angezeigt.

Beispiel

- Mit dem folgenden Beispiel werden bei einer ersten Ausführung alle TADDM-Daten im Kontextmenüservice und Datenintegrationsservice registriert; bei jeder weiteren Ausführung werden alle Änderungen registriert, die seit der letzten Ausführung vorgenommen wurden:

```
./run_cms_dis_registration.sh
```

- Mit diesem Beispiel wird nur das über die GUID angegebene Modellobjekt registriert:

```
./run_cms_dis_registration.sh register 3950DF835FA0337A829D864415CC1384
```

- Mit diesem Beispiel wird die Registrierung aller TADDM-Daten aufgehoben:

```
./run_cms_dis_registration.sh clean
```

- Mit diesem Beispiel wird das über die GUID und den Modellobjekttyp angegebene Objekt entfernt:

```
./run_cms_dis_registration.sh clean 3950DF835FA0337A829D864415CC1384
  LinuxUnitaryComputerSystem
```

- Mit diesem Beispiel werden alle registrierten TADDM-Daten entfernt und anschließend die Registrierung erneut vorgenommen:

```
./run_cms_dis_registration.sh re-register-all
```

Nächste Schritte

Soll das Registrierungsscript später erneut ausgeführt werden, müssen Sie zunächst den CMSDISAgent-Agenten für die Topologieerstellung inaktivieren, um Deltaaktualisierungen zu stoppen. Der Agent wird inaktiviert, indem in der Datei `$COLLATION_HOME/etc/collation.properties` die folgende Eigenschaft gesetzt wird:

```
com.ibm.cdb.DisCmsIntegration.enabled=false
```

Nach Abschluss des Scripts müssen Sie den Agenten wieder aktivieren, indem Sie diese Eigenschaft auf `true` setzen.

CMSDISAgent konfigurieren
Der CMSDISAgent wird in regelmäßigen Abständen als ein Agent für die Topologieerstellung ausgeführt; er aktualisiert die Registrierung von TADDM-Konfigurationselementen (Configuration Items, CIs) in der CMS/DIS-Datenbank (Context Menü Service, Data Integration Service), indem neue oder aktualisierte CIs registriert werden bzw. die Registrierung gelöschter CIs zurückgenommen wird.

Informationen zu diesem Vorgang
Bei einer Aktivierung wird der CMDDISAgent nach der Erstregistrierung der TADDM-CIs mithilfe des Scripts **run_cms_dis_registration** ausgeführt. Über eine Änderung der Agentenkonfiguration können Sie die Ausführungsweise des Agenten ändern.

Prozedur

- Der CMSDISAgent kann in der Datei `$COLLATION_HOME/etc/collation.properties` über die folgende Eigenschaft aktiviert bzw. inaktiviert werden:

  ```
  com.ibm.cdb.DisCmsIntegration.enabled=Wert
  ```

  Dabei kann *Wert* auf `true` oder `false` gesetzt werden. Bei Angabe von `true` wird der Agent nach Abschluss der Erstregistrierung in regelmäßigen Abständen ausgeführt. (Diese Eigenschaft hat keine Auswirkung auf die Ausführung des Scripts **run_cms_dis_registration** - dieses Script kann jederzeit ausgeführt werden.)
- Sie können die CIs angeben, die in der Datenbank registriert werden sollen, indem Sie im Verzeichnis `$COLLATION_HOME/etc/cmsdis` die folgenden Dateien entsprechend ändern:

  classtype-changehistory.list
  Enthält eine Liste der CI-Modellobjekttypen, für die TADDM das Launch-in-Context-Feature für den Änderungsprotokollbericht unterstützt.

  classtype-detailPanel.list
  Enthält eine Liste der CI-Modellobjekttypen, für die TADDM das Launch-in-Context-Feature für die Anzeige mit Detailangaben unterstützt.

  Sie können alle Modellobjekttypen entfernen, die nicht für andere Produkte benötigt werden, um einen Launch-in-Context für TADDM durchzuführen. Allerdings dürfen diesen Dateien keine Typen hinzugefügt werden, da TADDM das Launch-in-Context-Feature für zusätzliche Typen möglicherweise nicht unterstützt.

  Nachdem Sie die Dateien mit den Klassentypenlisten geändert haben, müssen Sie den Agenten inaktivieren und das Script **run_cms_dis_registration** unter Angabe der Option `re-register-all` erneut ausführen.

Erkennungsbibliotheksspeicher erstellen

Der Speicher einer Erkennungsbibliothek ist ein Verzeichnis oder ein Ordner auf einem Computer im Rechenzentrum. In diesen Speicher schreiben alle Erkennungsbibliotheksadapter die XML-Dateien mit Ressourcendaten. XML-Datendateien, die mithilfe des Dienstprogramms zum Laden von Massendaten in ein TADDM-System geladen werden sollen, werden im Erkennungsbibliotheksspeicher gespeichert. Um das Massenladeprogramm verwenden zu können, müssen Sie einen Erkennungsbibliotheksspeicher erstellen.

Vorbereitende Schritte

Ein Erkennungsbibliotheksadapter (Discovery Library Adapter, DLA) ist ein Softwareprogramm, das Daten aus einer Quellenanwendung extrahiert, zum Beispiel aus IBM Tivoli Monitoring oder IBM Tivoli Business Service Manager.

Jeder Erkennungsbibliotheksadapter erstellt XML-Dateien mit Ressourcendaten im XML-Format IdML (Identity Markup Language). Eine XML-Datei im IdML-Format wird allgemein als *Buch* bezeichnet. Eine

Sammlung von Tivoli-Büchern, die die TADDM-Datenbank mit Daten aus anderen Tivoli-Produkten laden können, finden Sie unter http://www.ibm.com/software/brandcatalog/ismlibrary/.

Jedes Produkt hat spezifische Erkennungsbibliotheksadapter (DLA), da jedes Produkt eine andere Methode für den Zugriff auf Ressourcen in einer Umgebung verwendet. Die Konfiguration und Installation eines DLAs richtet sich nach der jeweiligen Anwendung. Für gewöhnlich wird ein DLA auf einem System installiert, das Zugriff auf die Daten einer bestimmten Anwendung hat. So ist der DLA für IBM Tivoli Monitoring auf einem Computer installiert, der Zugriff auf die Systemdatenbank zum unternehmensweiten Management von IBM Tivoli Monitoring hat. Alle DLAs werden über die Befehlszeilenschnittstelle ausgeführt und können über jedes beliebige Terminplanungsprogramm des Unternehmens (zum Beispiel cron) zeitgesteuert ausgeführt werden.

Sie können einen DLA erstellen, der Daten aus vorhandenen Produkten oder Datenbanken Ihrer Umgebung extrahiert.

Weitere Informationen zum Erstellen eines Erkennungsbibliotheksadapters (DLA), zur IdML-Spezifikation sowie zum Erkennungsbibliotheksspeicher finden Sie im TADDM-Handbuch *Discovery Library Adapter Developer's Guide* (Entwicklerhandbuch für Erkennungsbibliotheksadapter).

Informationen zu diesem Vorgang

In der Regel befindet sich der Erkennungsbibliotheksspeicher auf dem TADDM-Server. Wenn Sie den Erkennungsbibliotheksspeicher nicht auf dem TADDM-Server installieren möchten, müssen Sie sicherstellen, dass das TADDM-Massenladeprogramm, das auf dem TADDM-Server ausgeführt wird, auf den Erkennungsbibliotheksspeicher zugreifen kann. Auf dem Hostcomputer des Erkennungsbibliotheksspeichers können auch andere Anwendungen ausgeführt werden.

Vorgehensweise

Gehen Sie folgendermaßen vor, um den Erkennungsbibliotheksspeicher zu erstellen:

1. Erstellen Sie zur Speicherung der XML-Dateien auf einem Computer ein Verzeichnis mit einem eindeutigen Namen (zum Beispiel `c:\IBM\DLFS`). Für jeden verwendeten Erkennungsbibliotheksadapter können Sie im primären Erkennungsbibliotheksspeicher Unterverzeichnisse erstellen.
2. Konfigurieren Sie einen FTP-Server (File Transfer Protocol) mit mindestens einer Benutzer-ID. Diese Benutzer-ID muss für das Verzeichnis, in dem sich die XML-Dateien der Erkennungsbibliothek befinden, über Schreib- und Lesezugriff sowie über die Berechtigung zum Umbenennen verfügen. Wenn Sie die XML-Dateien nicht mithilfe von FTP an den Erkennungsbibliotheksspeicher weiterleiten, stellen Sie sicher, dass das verwendete Tool und die zur Ausführung des Tools verwendete Benutzer-ID über Schreibzugriff auf das Verzeichnis des Erkennungsbibliotheksspeichers verfügen.
3. Stellen Sie sicher, dass die verschiedenen Erkennungsbibliotheksadapter Zugriff auf den Namen des Systems (Hostname) haben, das als Host für den Erkennungsbibliotheksspeicher dient. Die meisten Erkennungsbibliotheksadapter kopieren XML-Dateien in den Erkennungsbibliotheksspeicher.
4. Vergewissern Sie sich, dass alle DLAs die Benutzer-ID und das Kennwort kennen, um eine Verbindung zum FTP-Server herstellen zu können.
5. Wenn der Erkennungsbibliotheksadapter nicht FTP verwendet, kopieren Sie die XML-Dateien (Bücher), auf die das Massenladeprogramm zugreifen soll, in das gemeinsam verwendete Verzeichnis. Das Massenladeprogramm muss auf das gemeinsam genutzte Verzeichnis zugreifen können.

   Die Autoren der Bücher und der Administrator sind nicht dafür zuständig, die Bücher im Erkennungsbibliotheksspeicher zu speichern. Definieren Sie stattdessen beispielsweise einen Cron-Job, um die erstellten IdML-Bücher über FTP in den Erkennungsbibliotheksspeicher zu senden.

Nächste Schritte

Wenn Sie einen Erkennungsbibliotheksspeicher erstellen und eine TADDM-Datenbank für die Bücher des Erkennungsbibliotheksadapters einrichten möchten, kann ein lokales Laufwerk auf dem Domänenserver als Netzerkennungsbibliotheksspeicher verwendet werden. Dieses Verzeichnis muss auf dem Domänenserver, auf den die Daten geladen werden, in der Datei `$COLLATION_HOME/etc/bulkload.properties` definiert werden. Bei mehreren Domänenservern müssen Sie das richtige Massenladeprogramm für

den Zugriff auf das entsprechende gemeinsam genutzte Verzeichnis konfigurieren. Das Massenladeprogramm löscht keine XML-Dateien aus dem Erkennungsbibliotheksspeicher. Die Dateien müssen im Erkennungsbibliotheksspeicher verwaltet werden. Stellen Sie sicher, dass auf dem Server ausreichend Plattenspeicher für die Dateien in dem Verzeichnis zur Verfügung steht. Werden diesem Verzeichnis häufig neue XML-Dateien hinzugefügt, muss es regelmäßig bereinigt werden.

Bei einer Synchronisationsserverimplementierung müssen Sie eine der folgenden Optionen auswählen:

- Wenn die Ressourcen, auf die in einem Buch verwiesen wird, in den Bereichsdefinitionen enthalten sind, die auf einem Domänenserver definiert sind, laden Sie dieses Buch auf den entsprechenden Domänenserver.
- Wenn die Ressourcen, auf die in einem Buch verwiesen wird, ***nicht*** in den Bereichsdefinitionen enthalten sind, die auf einem Domänenserver definiert sind, laden Sie alle Bücher auf den Synchronisationsserver.

Konfiguration für den Start im Kontext

Detailliertere Informationen zu Komponenten in Ihrer Umgebung erhalten Sie, indem Sie TADDM-Ansichten über andere Tivoli-Anwendungen starten. Um Ihre Anwendung für den Start von TADDM-Ansichten im Kontext konfigurieren zu können, müssen Sie eine URL angeben.

Ansichten, die über andere Tivoli-Anwendungen gestartet werden können

Ansichten des Datenmanagementportals können über andere Tivoli-Anwendungen gestartet werden. Zudem können auch die Details und der Änderungsprotokollbericht für ein angegebenes Konfigurationselement (CI) gestartet werden.

In den Ansichten des Datenmanagementportals finden Sie weitere Informationen zu den folgenden Komponentengruppierungen:

- Geschäftsanwendungen
- Geschäftsservices
- Objektgruppen

Wenn der TADDM-Server und die Anwendung, über die TADDM gestartet wird, nicht für eine einmalige Anmeldung konfiguriert sind, wird ein Anmeldungsfenster angezeigt. Bevor Sie zusätzliche Informationen im Datenmanagementportal anzeigen können, müssen Sie einen Benutzernamen und ein Kennwort eingeben.

URL zum Starten der TADDM-Ansichten angeben

Wenn Sie TADDM-Ansichten im Kontext aus anderen Tivoli-Anwendungen starten möchten, müssen Sie eine URL angeben.

Das URL-Format für den Start im Kontext lautet wie folgt:

```
Protokoll://TADDMHostname:TADDMPort/KontextRoot/?Abfragezeichenfolge
```

In der folgenden Liste sind die gültigen Werte für die einzelnen Variablen im URL-Format beschrieben:

Protokoll
Das zu verwendende Web-Protokoll. Gültige Werte sind `http` oder `https`.

TADDMHostname
Der Hostname des TADDM-Servers, auf dem der Start ausgeführt wird.

TADDMPort
Die Portnummer des TADDM-Servers, auf dem der Start ausgeführt wird. Der Standardwert ist `9430`.

KontextRoot
Die folgenden Werte sind gültig:

`cdm/servlet/LICServlet`
Der relative Pfad zum Java-Servlet, das im Apache Tomcat-Server für TADDM 7.3.0 und im WAS Liberty Profile-Server für TADDM 7.3.0.1 und höher implementiert ist.

cdm/queryHomePage.do
Der relative Pfad der Abfragehomepage beim Start aus IBM Tivoli Monitoring mit einmaliger Anmeldung und Angabe eines Suchbegriffs.

Abfragezeichenfolge
Enthält durch Trennzeichen begrenzte Name/Wert-Paar-Parameter. Das Format für ein Name/Wert-Paar lautet `Name=Wert`. Trennen Sie Namen und Werte mithilfe von = und Name/Wert-Paare mithilfe von &.

In der folgenden Liste sind die gültigen Name/Wert-Paare beschrieben, die in der Variablen *Abfragezeichenfolge* verwendet werden können:

Ansicht
Gibt an, dass ein Änderungsprotokoll angezeigt werden soll.

Der einzige gültige Wert ist `changehistory`.

days_previous
Gibt den Zeitraum (die Anzahl der vergangenen Tage) an, für den das Änderungsprotokoll eines bestimmten Konfigurationselements angezeigt werden soll.

Der gültige Wert ist eine positive Ganzzahl.

hoursback
Gibt den Zeitraum (die Anzahl der vergangenen Stunden) an, für den das Änderungsprotokoll eines bestimmten Konfigurationselements angezeigt werden soll.

Der gültige Wert ist eine positive Ganzzahl.

GUID
Gibt die GUID (GUID = Globally Unique Identifier) für ein Konfigurationselement an.

Für den Domänenserver und den Synchronisationsserver werden in Tabelle 48 auf Seite 203 die gültigen Werte für den Parameter `graph` aufgelistet und es wird angegeben, ob der Parameter `guid` bei dem entsprechenden Diagrammwert optional oder erforderlich ist.

Wenn für den Parameter `graph` einer der folgenden Werte angegeben ist, ist der Parameter `guid` optional:

- `businessapplications`
- `applicationinfrastructure`
- `physicalinfrastructure`

Wenn für den Parameter `graph` ein beliebiger anderer Topologiediagrammtyp angegeben ist, ist der Parameter `guid` erforderlich.

Der gültige Wert ist eine gültige Zeichenfolge einer GUID, wie im folgenden Beispiel veranschaulicht:

```
BA2842345F693855A3165A4B5F0D8BDE
```

Für jede URL-Anforderung zum Start im Kontext sollte nur eine GUID angegeben werden.

graph
Gibt an, welcher Typ von Topologiediagramm gestartet werden soll.

Wenn Sie zudem ein Konfigurationselement angeben, indem Sie die GUID des Elements im Parameter `guid` bereitstellen, wird das gewünschte Konfigurationselement ausgewählt, wenn es im Topologiediagramm, das in diesem `graph`-Parameter angegeben ist, gefunden wird.

Für den Domänenserver und den Synchronisationsserver werden in Tabelle 48 auf Seite 203 die gültigen Werte für den Parameter `graph` aufgelistet und es wird angegeben, ob der Parameter `guid` bei dem entsprechenden Diagrammwert optional oder erforderlich ist.

Tabelle 48. Gültige Diagrammwerte und ihre Beziehung zum Parameter guid		
	Gültiger Wert	**Ist der Parameter guid bei diesem Diagrammwert optional oder erforderlich?**
Domänenserver	businessapplications	Optional
	applicationinfrastructure	Optional
	physicalinfrastructure	Optional
	Für benutzerdefinierte Objektgruppenobjekte: • ba_infrastructure	Erforderlich
Synchronisationsserver	businessapplications	Optional
	physicalinfrastructure	Optional
	Für benutzerdefinierte Objektgruppenobjekte: • ba_infrastructure	Erforderlich

Anmerkung: Alle anderen Diagrammtypen früherer TADDM-Versionen, die zur Wiedergabe bestimmter Gruppenentitäten durch Angabe einer GUID verwendet wurden, sind veraltet. Aus Kompatibilitätsgründen werden jedoch Anforderungen mit einem alten Diagrammtyp (mit Angabe einer GUID) in den neuen Topologietyp umgesetzt.

username
Gibt den Benutzernamen an, der für die Anmeldung am TADDM verwendet wird.

password
Gibt das Kennwort an, das für die Anmeldung am TADDM verwendet wird.

launchsource
Der einzige gültige Wert ist ITM. Wird immer mit dem Name/Wert-Paar searchtext=*Suchbegriff* verwendet.

Die Suche ist auf die Konfigurationselemente vom Typ ComputerSystem und TMSAgent begrenzt, die in der Konfigurationsdatei $COLLATION_HOME/etc/cdm/xml/itm_query_components.xml aufgelistet sind.

Von den Ergebnissen der Abfragehomepage aus können Sie für jedes aufgelistete Konfigurationselement die folgenden Ansichten starten:

- Teilfenster 'Änderungsprotokoll'
- Teilfenster 'Details'
- Datenmanagementportal mit dem Teilfenster 'Details'

searchtext
Gibt den Suchbegriff an. Wird immer mit dem Name/Wert-Paar launchsource=ITM verwendet.

Beispiele zur Vorgehensweise bei der Angabe der URL

Die folgenden Beispiele veranschaulichen die Vorgehensweise bei der Angabe der URL zum Start der TADDM-Ansichten:

URL zum Start des Datenmanagementportals ohne separate Eingabe der Berechtigungsinformationen

```
http://home.taddm.com:9430/cdm/servlet/LICServlet?username=administrator
&password=adminpwd&guid=BA2842345F693855A3165A4B5F0D8BDE
```

Wenn Sie über eine gesicherte Verbindung verfügen, dürfen Sie Berechtigungsnachweise nur als Teil der URL zum Start im Kontext verwenden, da Benutzername und Kennwort nicht verschlüsselt sind.

URL zum Start des Fensters Query Home Page (Abfragehomepage) für IBM Tivoli Monitoring bei Verwendung der einmaligen Anmeldung und bei der Suche nach einem Konfigurationselement, das dem Suchbegriff entspricht

```
http://home.taddm.com:9430/cdm/queryHomePage.do?launchsource=itm&search
text=127.0.0.1
```

URL für die Anzeige der Topologie einer benutzerdefinierten Objektgruppe durch Angabe des Parameters guid

```
http://home.taddm.com:9430/cdm/servlet/LICServlet?username=administrator
&password=adminpwd&graph=ba_infrastructure&guid=BA2842345F693855A3165A4B5F0D8BDE
```

Änderungsereignisse an externe Systeme senden

Sie können TADDM so konfigurieren, dass ein externes System, das Ereignisse handhabt, bei der Feststellung von Änderungen einer erkannten Ressource benachrichtigt wird.

Um Änderungsereignisse von TADDM senden zu können, müssen Sie über mindestens ein installiertes Ereignisbehandlungssystem verfügen:

- IBM Tivoli Monitoring 6.2.1 Fixpack 2 oder höher
- IBM TivoliNetcool/OMNIbus, einschließlich der Event Integration Facility (EIF)-Sonde

Unterstützte Versionen dieses Produkts finden Sie im Abschnitt „Unterstützte Versionen” auf Seite 177.

Nach Abschluss einer Erkennung überprüft TADDM Elemente, die durch externe Ereignisbehandlungssysteme überwacht werden, auf Änderungen. Bei Feststellung derartiger Änderungen werden diese über EIF direkt an IBM Tivoli Netcool/OMNIbus und mithilfe von Universal Agent an IBM Tivoli Monitoring gesendet.

Der Universal Agent wandelt die empfangenen Benachrichtigungen in asynchrone Ereignisse um und leitet die Daten an die Komponente des IBM Tivoli Enterprise Monitoring Server von IBM Tivoli Monitoring weiter. Der IBM Tivoli Monitoring Server speichert die Ereignisse und verwendet sie zur Bewertung von Situationen. Dann werden die Ereignisse zur Anzeige an IBM Tivoli Enterprise Portal übermittelt.

IBM Tivoli Netcool/OMNIbus-Server verarbeiten empfangene Ereignisse in Übereinstimmung mit ihren internen Regeln und zeigen sie an.

Um das Senden von Änderungsereignissen von TADDM an externe Ereignisbehandlungssysteme einzurichten, müssen Änderungsereignisse in TADDM aktiviert und gegebenenfalls alle externen Empfänger für die Verarbeitung eingehender Ereignisse konfiguriert sein.

TADDM für das Senden von Änderungsereignissen konfigurieren

Um Änderungsereignisse senden zu können, muss TADDM mit Angaben zu den Ereignisbehandlungssystemen konfiguriert werden, an die Änderungsereignisse gesendet werden sollen.

Informationen zu diesem Vorgang

Nehmen Sie je nach TADDM-Implementierung die Änderungen auf den folgenden TADDM-Servern vor:

- In einer Domänenserverimplementierung müssen die Änderungen auf dem Domänenserver erfolgen.
- In einer Synchronisationsserverimplementierung müssen die Änderungen auf dem Synchronisationsserver erfolgen.
- In einer Streaming-Server-Implementierung müssen die Änderungen auf dem primären Speicherserver erfolgen.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um das Senden von Änderungsereignisinformationen zu aktivieren:

1. Um Änderungsereignisse zu aktivieren, muss in der Datei $COLLATION_HOME/etc/collation.properties die folgende Eigenschaft festgelegt sein: com.ibm.cdb.omp.changeevent.enabled=true
2. Bearbeiten Sie die Datei $COLLATION_HOME/etc/EventConfig.xml, um zu konfigurieren, bei welchen Ressourcen Änderungen protokolliert und an welche Ereignisbehandlungssysteme die Ereignisse gesendet werden sollen.

   Informationen zu dem Format, in dem Angaben in der Datei EventConfig.xml eingetragen werden müssen, finden Sie im Abschnitt „Konfiguration des TADDM OMP-Änderungsereignismoduls" auf Seite 205.

   Bei einem Upgrade von TADDM wird die Datei EventConfig.xml aus dem vorherigen TADDM-Release beibehalten; damit wird sichergestellt, dass keine der von Ihnen konfigurierten angepassten Einstellungen verloren gehen. Informationen zu den neuen Features und ihrer Verwendung finden Sie in der Datei $COLLATION_HOME/etc/EventConfigDefault.xml. Die Datei EventConfigDefault.xml dient nur zu Referenzzwecken. Wenn eines der neuen Features verwendet werden soll, müssen Sie die Datei EventConfig.xml entsprechend den jeweiligen Beispielen in EventConfigDefault.xml aktualisieren.
3. Wenn Sie ein IBMTivoliNetcool/OMNIbus-System für die Ereignisverarbeitung in der Datei EventConfig.xml angegeben haben, müssen Sie eine entsprechende EIF-Eigenschaftendatei für den Systemtyp erstellen. Gehen Sie hierzu folgendermaßen vor:
   a) Erstellen Sie die Eigenschaftendatei $COLLATION_HOME/etc/omnibus.eif.properties.
   b) Passen Sie die Datei omnibus.eif.properties an. Weitere Informationen zum Anpassen einer EIF-Eigenschaftendatei finden Sie in der IBM Tivoli Netcool/OMNIbus-Dokumentation unter http://www-01.ibm.com/support/knowledgecenter/SSSHTQ_7.4.0/com.ibm.netcool_OMNIbus.doc_7.4.0/omnibus/wip/install/task/omn_con_ext_configuringtaddmevents.html?lang=en im Abschnitt *Configuring support for TADDM events in your integrated environment* (Unterstützung für TADDM-Ereignisse in Ihrer integrierten Umgebung konfigurieren).

Konfiguration des TADDM OMP-Änderungsereignismoduls

Um Änderungsereignisse senden zu können, müssen Sie in der Datei EventConfig.xml zunächst Ereignislistener und Ereignisempfänger festlegen.

Ereignislistener

Einen Listener definieren Sie durch die Bereitstellung der für eine TADDM-Abfrage erforderlichen Kriterien. Die sich daraus ergebenden, durch die Abfrage ausgewählten Objekte werden nach jeder Erkennung auf Änderungen überprüft. Sie können auch mehrere Listener einrichten. Damit eine Ereignisweiterleitung stattfinden kann, muss sowohl ein Listener- als auch ein zugehöriger Empfängerblock definiert sein.

Gehen Sie zur Definition eines Listener nach folgender Syntax vor:

```
<listener object="[OBJECT_TYPE]"
          enabled="true|false">
          sendCauses="true|false"
          sendOriginGuid="true|false">
   <alert recipient="[RECIPIENT_SYSTEM_NAME]"/>
   <attribute name="[ATTRIBUTE_NAME]" operator="[OPERATOR]">
     <value>
     [ATTRIBUTE_VALUE]
     </value>
   </attribute>
   <causeFilter object="[CAUSEFILTER_OBJECT_TYPE]"
                sendOriginGuid="true|false"/>

 </listener>
```

Dabei gilt:

[OBJECT_TYPE]
Ein in TADDM dargestellter Modellobjekttyp, zum Beispiel ComputerSystem oder ITSystem. Weitere Beispiele finden Sie im TADDM-Datenverzeichnis unter http://*TADDMServerhost*:9430/cdm/datadictionary/model-object/index.html.

enabled
Dieses Attribut legt fest, ob das Senden von Ereignissen aktiviert ist. Zur Aktivierung des Listener muss dieses Attribut auf `true` gesetzt werden.

sendCauses
Dieses optionale Attribut legt fest, ob der Listener Ereignisse zu Änderungen sendet, die dem Modellobjekt übergeben wurden. Bewirkt eine Änderung an einem Windows-Betriebssystem beispielsweise eine Änderung an einem ComputerSystem-Objekt und das Attribut `sendCauses` ist für einen ComputerSystem-Listener auf `true` gesetzt, so sendet dieser Listener ein Ereignis für die Änderung sowohl am ComputerSystem-Objekt als auch am Windows-Betriebssystem. Der Standardwert des Attributs `sendCauses` ist `false`.

sendOriginGuid
Dieses optionale Attribut wird gemeinsam mit dem Attribut `sendCauses` verwendet. Wenn `sendOriginGuid` auf `true` gesetzt ist, wird ein Objekt, das mit dem Listener übereinstimmt, als logischer Ursprung der Änderungen betrachtet, die dem Objekt übergeben werden. Ereignisse, die zu übergebenen Änderungen gesendet werden, enthalten die eindeutige Kennung des Ursprungsobjekts. Bewirkt eine Änderung an einem ConfigFile-Objekt beispielsweise eine Änderung an einem ComputerSystem-Objekt und die Attribute `sendCauses` und `sendOriginGuid` sind beide für einen ComputerSystem-Listener auf `true` gesetzt, so enthält das Ereignis zu der ConfigFile-Änderung zusätzlich zur eindeutigen Kennung des ConfigFile-Objekts auch die eindeutige Kennung des ComputerSystem-Objekts. Diese Funktion steht nur für die Netcool/OMNIbus-Ereignisempfänger zur Verfügung. Der Standardwert des Attributs `sendOriginGuid` ist 'false'.

[RECIPIENT_SYSTEM_NAME]
Ein Alertempfänger. Siehe „Ereignisempfänger” auf Seite 208.

[ATTRIBUTE_NAME]
Der Name eines Attributs für `[OBJECT_TYPE]`, das abgefragt wird.

[OPERATOR]
Die Operatorbezeichnung einer TADDM MQL-Abfrage. Die folgenden Werte sind gültig.

Tabelle 49. Operatorbezeichnungen für TADDM MQL-Abfragen

Operator	**TADDM MQL-Entsprechung**
contains-with	contains
ends-with	ends-with
equals	equals
greater-or-equal	>=
greater-than	>
less-or-equal	<=
less-than	<
not-equals	not-equals
starts-with	starts-with

[ATTRIBUTE_VALUE]
Der Wert, mit dem das Attribut abgeglichen wird.

<causeFilter>
Mit diesem Attribut können die Objekttypen der Kausalereignisse gefiltert werden, die übertragen werden, wenn das Attribut `sendCauses` aktiviert ist. Wenn Sie dieses Attribut angeben, werden nur Kausalereignisse des angegebenen Objekttyps gesendet. Übergebene Ereignisse, wie diejenigen des im Listener angegebenen Objekttyps, werden jedoch nach wie vor gesendet. Ohne Angabe des Attributs `causeFilter` werden alle vom Listener vorgefundenen Kausalereignisse an den Empfänger gesendet.

Eine Änderung an einem WindowsService-Objekt verursacht beispielsweise eine Änderung am Windows-Betriebssystem und damit auch am ComputerSystem-Objekt. Ist `causeFilter` in diesem Fall auf WindowsService gesetzt, werden nur die Änderungen an ComputerSystem und WindowsService angezeigt, die Änderung am Windows-Betriebssystem hingegen nicht.

Wenn Sie das Attribut `causeFilter` angeben, können Sie optional auch einen Wert für das Attribut `sendOriginGuid` festlegen. Standardmäßig übernimmt das Attribut `causeFilter` die Einstellung des Attributs `sendOriginGuid` des dem Attribut `causeFilter` übergeordneten Listener. Wird das Attribut `sendOriginGuid` in einem `causeFilter`-Attribut gesetzt, so wird die Listener-Einstellung nur für dieses `causeFilter`-Attribut überschrieben.

Wenn Sie Objekte wie WindowsService oder ConfigFile aktualisieren möchten, deren Änderungen an höhere Objekte wie ComputerSystem übergeben werden, sollten Sie diese Objekte nicht durch einen eigenen Listener, sondern mit einer Kombination der Attribute `sendCauses` und `causeFilter` erfassen.

[CAUSEFILTER_OBJECT_TYPE]
Der Klassenname des Objekts wie im CDM definiert. Sie können den vollständigen Namen (z. B. `com.collation.platform.model.topology.sys.windows.WindowsService`) oder den Kurznamen (z. B. `WindowsService`) verwenden.

Beispiele für Ereignislistener

Im folgenden Beispiel wird jede erkannte Änderung an einem ComputerSystem-Objekt, dessen vollständig qualifizierter Domänenname die Zeichenfolge "`mycompany`" enthält, an den Empfänger "`enterprise-eventhost-itm`" gesendet.

```
<listener object="ComputerSystem" enabled="true">
  <alert recipient="enterprise-eventhost-itm"/>
  <attribute name="fqdn" operator="contains-with">
    <value>
     mycompany
    </value>
  </attribute>
</listener>
```

Im folgenden Beispiel werden Änderungen an allen Objekten des angegebenen Typs erkannt.

```
  <attribute name="guid" operator="not-equals">
    <value>
     0
    </value>
  </attribute>
```

Im folgenden Beispiel wird jede erkannte Änderung an einem ComputerSystem-Objekt an den Empfänger "`enterprise-eventhost-omnibus`" gesendet.

```
<listener object="ComputerSystem" enabled="true">
  <alert recipient="enterprise-eventhost-omnibus"/>
  <attribute name="guid" operator="not-equals">
    <value>
     0
    </value>
  </attribute>
</listener>
```

Im folgenden Beispiel werden nur Änderungen gesendet, die durch eine Änderung an einem ConfigFile-Objekt auf einem Linux-Computersystem verursacht wurden.

```
<listener object="ITSystem" enabled="true" sendCauses="true">
  <alert recipient="enterprise-eventhost-itm"/>
  <attribute name="name" operator="ends-with">
  <value>
    ShoppingCart
  </value>
  </attribute>
  <causeFilter object="ConfigFile" />
  <causeFilter object="LinuxUnitaryComputerSystem" />
</listener>
```

Ereignisempfänger

Ein Ereignisempfänger ist eine IBM Tivoli Monitoring- oder OMNIBus-Instanz, die Ereignisse des Änderungsereignismoduls empfangen kann. Sobald die Änderungslistener Änderungen erkennen, werden den betreffenden Empfängern Benachrichtigungen gesendet. Sie können auch mehrere Empfänger des gleichen oder unterschiedlicher Typen definieren. Damit eine Ereignisweiterleitung stattfinden kann, muss sowohl ein Listener- als auch ein zugehöriger Empfängerblock definiert sein.

Gehen Sie zur Definition eines Empfängers nach folgender Syntax vor.

```
<recipient name="[RECIPIENT_NAME]" type="[RECIPIENT_TYPE]">
  <address>[RECIPIENT_FQDN]</address>
  <port>[EVENT_ROUTING_PORT]</port>
  <config>[PATH_TO_EIF_CONFIGURATION]</config>
</recipient>
```

Dabei gilt:

[RECIPIENT_NAME]
Der Name des vom Listener referenzierten Systems.

[RECIPIENT_TYPE]
Der Typ der zum Ereignisempfang verwendeten Software. Folgende Typen werden unterstützt:

- `itm` - IBM Tivoli Monitoring 6 mit dem POST-Data-Provider Universal Agent.
- `omnibus` - Netcool/OMNIbus mit EIF-Adapter.

[RECIPIENT_FQDN]
(Nur IBM Tivoli Monitoring) Der vollständig qualifizierte Domänenname des Hosts, auf dem sich der Universal Agent befindet.

[EVENT_ROUTING_PORT]
(Nur IBM Tivoli Monitoring) Der Port, den der in KUMENV angegebene POST-Data-Provider Universal Agent verwendet (z. B. KUMP_POST_DP_PORT).

[PATH_TO_EIF_CONFIGURATION]
(Nur OMNIBUS) Der in der Eigenschaftendatei angegebene Pfad der EIF-Konfiguration. Geben Sie den vollständigen Pfad der Konfigurationsdatei an.

Beispiele für Ereignisempfänger

Das folgende Beispiel definiert einen Ereignisempfänger für Netcool/OMNIbus.

```
<recipient name="enterprise-eventhost-omnibus" type="omnibus">
  <config>/opt/IBM/taddm/dist/etc/omnibus.eif.properties</config>
</recipient>
```

Das folgende Beispiel definiert einen Ereignisempfänger für IBM Tivoli Monitoring.

```
<recipient name="enterprise-eventhost-itm" type="itm">
  <address>itm-ua.mycompany.com</address>
  <port>7575</port>
</recipient>
```

IBMTivoliNetcool/OMNIbus konfigurieren

Sie können IBMTivoliNetcool/OMNIbus Version 7.3 oder höher dafür konfigurieren, Änderungsereignisse zu empfangen, die von TADDM gesendet wurden. Sie können die in früheren Versionen von Tivoli Netcool/OMNIbus angezeigten Ereignisdaten zusammenfassen und anpassen sowie eine Logik zur Behandlung von Ereignissen definieren.

Vorbereitende Schritte

Informationen zur Konfiguration von IBM Tivoli Netcool/OMNIbus Version 7.3 oder höher für den Empfang von Änderungsereignissen, die TADDM sendet, finden Sie im Abschnitt *Enabling support for TADDM events* (Unterstützung für TADDM-Ereignisse aktivieren) in der IBM Tivoli Netcool/OMNIbus-Dokumentation unter http://www-01.ibm.com/support/knowledgecenter/SSSHTQ/landingpage/NetcoolOMNI-

bus.html?lang=en. Die Tivoli Netcool/OMNIbus-Dokumentation enthält auch Informationen zur Datei `tivoli_eif_taddm.rules`. Diese Datei beinhaltet die Logik zum Verarbeiten von Details von Konfigurationsänderungen, die während einer TADDM-Erkennung ermittelt wurden.

In Umgebungen mit hoher Verfügbarkeit oder Funktionsübernahme (Failover) kann TADDM für die Unterstützung der automatischen Funktionsübernahme konfiguriert werden. Diese Unterstützung tritt ein, wenn TADDM-Ereignisse an Tivoli Netcool/OMNIbus gesendet werden. Sie können primäre und sekundäre EIF-Testadressen und deren zugeordnete Ports in der EIF-Eigenschaftendatei angeben. Das folgende Beispiel zeigt, wo diese Eigenschaften hinzugefügt werden müssen:

```
# Hostname where the NetCool/OMNIbus EIF probe resides.  Specify up to 8 locations.
# Each location should be separated by a comma.
# The event is sent to the first available probe in the list.
# Example:
#     ServerLocation=netcool.mycompany.com,netcool2.mycompany.com
ServerLocation=netcool.mycompany.com,netcool2.mycompany.com

# Port the NetCool/OMNIbus EIF probe is listening on.
# There must be a port entry for each probe specified under ServerLocation.
# Example:
#     ServerPort=9998,9998
ServerPort=9998,9998
```

Für jede eingegebene Testadresse muss der zugeordnete Port in der Eigenschaft *ServerPort* angegeben werden. Wenn nicht für jede Testadresse der Port angegeben wird, führt dies beim Senden des Ereignisses zu einem Fehler. Wenn ein Ereignis nicht an den Primärport gesendet werden kann, wird es stattdessen an den ersten verfügbaren Port in der Liste gesendet. In der Eigenschaft *ServerLocation* können bis zu acht Testadressen angegeben werden.

Informationen zu diesem Vorgang

Das Standardverhalten in Versionen von IBMTivoli Netcool/OMNIbus vor Version 7.3 ist die Zusammenfassung aller Ereignisse eines Ereignismoduls zu einem einzigen Ereignis, wobei das Attribut 'Zähler' so festgelegt ist, dass die Zahl der in dem kombinierten Ereignis enthaltenen Ereignisse angezeigt wird. In den folgenden Schritten wird beschrieben, wie das Standardverhalten geändert wird.

Vorgehensweise

1. Öffnen Sie auf dem TADDM-Server die folgende Datei zur Bearbeitung: `$COLLATION_HOME/etc/omnibus.eif.properties`
2. Legen Sie folgende Eigenschaftswerte der TADDMEvent_Slot-Eigenschaften fest:

```
TADDMEvent_Slot_object_name=$TADDM_OBJECT_NAME
TADDMEvent_Slot_change_type=$TADDM_CHANGE_TYPE
TADDMEvent_Slot_change_time=$TADDM_CHANGE_TIME
TADDMEvent_Slot_class_name=$TADDM_CLASS_NAME
TADDMEvent_Slot_attribute_name=$TADDM_ATTRIBUTE_NAME
TADDMEvent_Slot_old_value=$TADDM_OLD_VALUE
TADDMEvent_Slot_new_value=$TADDM_NEW_VALUE
TADDMEvent_Slot_host=$TADDM_HOST
TADDMEvent_Slot_port=$TADDM_PORT
TADDMEvent_Slot_guid=$TADDM_GUID
TADDMEvent_Slot_origin=$TADDM_ORIGIN
```

Nächste Schritte

Lesen Sie bei eventuellen Problemen mit der Konfiguration von IBM Tivoli Netcool/OMNIbus den Abschnitt *Integrating TADDM with other products problems* (Probleme bei Integration von TADDM mit anderen Produkten) im TADDM *Troubleshooting Guide*.

IBM Tivoli Monitoring-Datenanbieter konfigurieren

Sie können die Initialisierungsdatei des Universal Agent für die Definition eines neuen Datenanbieters konfigurieren.

Vorbereitende Schritte
Bei Verwendung der Version 6.2.2 von Tivoli Monitoring oder einer älteren Version müssen Sie sicherstellen, dass in der Konfigurationsdatei KUMPOST keine Tabulator- oder Leerzeichen enthalten sind.

Vorgehensweise

Ist der Universal Agent auf einem Windows-System installiert, führen Sie die folgenden Schritte aus:

a) Klicken Sie auf dem Windows-System, auf dem der Universal Agent installiert ist, auf **Start** > **IBM Tivoli Monitoring** > **Manage Tivoli Monitoring Services** (Tivoli Monitoring-Services verwalten).

b) Klicken Sie mit der rechten Maustaste auf den Universal Agent und klicken Sie dann auf **Reconfigure** (Neu konfigurieren).

c) Klicken Sie in beiden erweiterten Konfigurationsfenstern des Agenten auf **OK**.

d) Klicken Sie auf **Ja**, um die Initialisierungsdatei des Universal Agent zu aktualisieren.

Die Datei KUMENV wird im Texteditor des Systems geöffnet.

e) Setzen Sie den Wert KUMA_STARTUP_DP auf POST:

```
KUMA_STARTUP_DP=POST
```

Anmerkung: Ist der Universal Agent bereits für die Verwendung eines anderen Datenanbieters konfiguriert, geben Sie wie im folgenden Beispiel beide Werte, durch ein Komma getrennt, an:

```
KUMA_STARTUP_DP=ASFS,POST
```

f) Fügen Sie die für den Parameter POST erforderlichen Angaben in die Datei KUMENV ein:

```
*---------------------------------------*
* TADDM POST DP Parameters *
*---------------------------------------*
KUMP_POST_DP_PORT=7575
KUMP_POST_GROUP_NAME=TADDM
KUMP_POST_APPL_TTL=14400
```

g) Speichern Sie die Datei KUMENV und schließen Sie sie.

h) Klicken Sie auf **Ja**, um den Agenten zu konfigurieren.

i) Klicken Sie im Fenster 'Manage Tivoli Enterprise Monitoring Services' auf **Universal Agent** > **Start**.

j) Erstellen Sie im Texteditor des Systems eine Textdatei. Geben Sie folgende Informationen in die Datei ein:

```
//APPl CONFIGCHANGE
//NAME dpPost E 3600
//ATTRIBUTES ';'
Post_Time T 16 Caption{Time}
Post_Origin D 32 Caption{Origination}
Post_Ack_Stamp D 28 Caption{Event time stamp}
Comp_Type D 512 Caption{Component type}
Comp_Name D 512 Caption{Component name}
Comp_Guid D 512 Caption{Component GUID}
Change_Type D 512 Caption{Change type}
Chg_Det_Time D 512 Caption{Change detection time}
Chg_Attr D 512 Caption{Changed attribute}
Srv_Addr D 512 Caption{TADDM server}
Srv_Port D 16 Caption{TADDM port}
```

k) Speichern Sie die Datei als `%ITM_HOME%\TMAITM6\metafiles\KUMPOST`.

Anmerkung: Vergewissern Sie sich, dass der Dateiname KUMPOST wie hier gezeigt in Großbuchstaben geschrieben ist.

l) Öffnen Sie eine Windows-Eingabeaufforderung und navigieren Sie zu dem Ordner `%ITM_HOME%\TMAITM6`.

m) Führen Sie das Programm `KUMPCON.exe` aus, um die Metadatei KUMPOST zu prüfen und zu importieren.

n) Klicken Sie im Fenster **Manage Tivoli Monitoring Services** (Tivoli Monitoring-Services verwalten) mit der rechten Maustaste auf den Universal Agent und wählen Sie die Option **Recycle** (Erneut starten) aus.

Ist der Universal Agent auf einem Linux- oder UNIX-System installiert, führen Sie die folgenden Schritte aus:

a) Konfigurieren Sie den Universal Agent mithilfe des folgenden Befehls neu:

```
itmcmd config –A um
```

Wenn Sie zur Eingabe des Datenanbieters aufgefordert werden, geben Sie POST ein.

Anmerkung: Ist der Universal Agent bereits für die Verwendung eines anderen Datenanbieters konfiguriert, müssen Sie beide Werte, durch ein Komma getrennt, angeben (Beispiel: ASFS,POST).

b) Erstellen Sie im Verzeichnis $ITM_HOME/config eine Sicherungskopie der Datei um.ini und fügen Sie anschließend der Originaldatei die folgenden Einträge hinzu:

```
# TADDM POST DP Parameters
KUMP_POST_DP_PORT=7575
KUMP_POST_GROUP_NAME=TADDM
KUMP_POST_APPL_TTL=14400
```

c) Erstellen Sie im Verzeichnis $ITM_HOME/*interp*/um/metafiles eine Textdatei. Geben Sie folgende Informationen in die Datei ein:

```
//APPl CONFIGCHANGE
//NAME dpPost E 3600
//ATTRIBUTES ';'
Post_Time T 16 Caption{Time}
Post_Origin D 32 Caption{Origination}
Post_Ack_Stamp D 28 Caption{Event time stamp}
Comp_Type D 512 Caption{Component type}
Comp_Name D 512 Caption{Component name}
Comp_Guid D 512 Caption{Component GUID}
Change_Type D 512 Caption{Change type}
Chg_Det_Time D 512 Caption{Change detection time}
Chg_Attr D 512 Caption{Changed attribute}
Srv_Addr D 512 Caption{TADDM server}
Srv_Port D 16 Caption{TADDM port}
```

d) Speichern Sie die Datei als KUMPOST.

Anmerkung: Vergewissern Sie sich, dass der Dateiname KUMPOST wie hier gezeigt in Großbuchstaben geschrieben ist.

e) Starten Sie den Universal Agent mithilfe der folgenden Befehle erneut:

```
itmcmd agent stop um
```

```
itmcmd agent start um
```

f) Führen Sie folgende Schritte aus, um die Metadatei KUMPOST zu prüfen und zu aktualisieren:

1) Führen Sie den Befehl $ITM_HOME/bin/um_console mit folgenden Parametern aus:

```
um_console -h <ITM-Verzeichnis>
```

2) Geben Sie folgenden Text in die Befehlszeile ein:

```
validate KUMPOST
```

Es werden Nachrichten angezeigt, die beispielsweise wie folgt lauten könnten:

```
KUMPS001I Console input accepted.
KUMPV025I Processing input metafile /opt/IBM/ITM//lx8266/um/metafiles/KUMPOST
KUMPV026I Processing record 0001 -> //APPl CONFIGCHANGE
KUMPV148I Note: APPL names starting with letters A-M are designated for
Best Practices and Business Partner UA solutions.
KUMPV026I Processing record 0002 ->     //NAME dpPost E 3600
KUMPV026I Processing record 0003 ->     //ATTRIBUTES ';'
```

```
KUMPV026I Processing record 0004 -> Post_Time T 16 Caption{Time}
KUMPV026I Processing record 0005 -> Post_Origin D 32 Caption{Origination}
KUMPV026I Processing record 0006 -> Post_Ack_Stamp D 28 Caption{Event time stamp}
KUMPV026I Processing record 0007 -> Comp_Type D 512 Caption{Component type}
KUMPV026I Processing record 0008 -> Comp_Name D 512 Caption{Component name}
KUMPV026I Processing record 0009 -> Comp_Guid D 512 Caption{Component GUID}
KUMPV026I Processing record 0010 -> Change_Type D 512 Caption{Change type}
KUMPV026I Processing record 0011 -> Chg_Det_Time D 512 Caption{Change detection
time}
KUMPV026I Processing record 0012 -> Chg_Attr D 512 Caption{Changed attribute}
KUMPV026I Processing record 0013 -> Srv_Addr D 512 Caption{TADDM server}
KUMPV026I Processing record 0014 -> Srv_Port D 16 Caption{TADDM port}
KUMPV000I Validation completed successfully
KUMPV090I Application metafile validation report saved in file
/opt/IBM/ITM//lx8266/um/metafiles/KUMPOST.rpt.
```

3) Geben Sie folgenden Text ein, wenn Sie aufgefordert werden, anzugeben, welche Aktion für die Metadatei ausgeführt werden soll:

```
Refresh
```

4) Geben Sie `Yes` (Ja) ein, um den Vorgang zu bestätigen.

Nächste Schritte

Überprüfen Sie anhand des Änderungsereignisberichts in Tivoli Enterprise Portal, ob der Universal Agent erfolgreich konfiguriert wurde.

Führen Sie folgende Schritte aus, um den Änderungsereignisbericht mithilfe von IBM Tivoli Monitoring 6.2.1 oder höher zu öffnen:

1. Navigieren Sie zu dem Universal Agent, der für das Senden und Empfangen von Ereignisbenachrichtigungen von TADDM konfiguriert wurde.
2. Erweitern Sie den Knoten CONFIGCHANGE.
3. Klicken Sie auf den Knoten **DPPOST**.

Konfigurationsänderungssituationen in IBM Tivoli Monitoring erstellen

Mit der Situationsfunktion in Tivoli Enterprise Portal können Sie Änderungsereignisse überwachen und Situationen auf der Grundlage der Informationen in einem Änderungsereignis auslösen.

Vorgehensweise

So erstellen Sie bei Verwendung von IBM Tivoli Monitoring 6.2.1 eine Konfigurationsänderungssituation:

a) Navigieren Sie im Navigatorfenster von IBM Tivoli Enterprise Portal zu dem Universal Agent, der für das Senden und Empfangen von Ereignisbenachrichtigungen von TADDM konfiguriert wurde.

b) Erweitern Sie den Knoten CONFIGCHANGE.

c) Klicken Sie mit der rechten Maustaste auf den Knoten DPPOST und klicken Sie auf **Situations** (Situationen).

d) Klicken Sie im Fenster "**Situations for** *Knotenname*" (Situationen für Knotenname) mit der rechten Maustaste auf **Universal Data Provider** (Universeller Datenanbieter). Klicken Sie auf **Create New** (Neue erstellen).

Das Fenster **Create Situation or Rule** (Situation oder Regel erstellen) wird angezeigt.

e) Geben Sie im Feld **Name** den Namen der Situation ein.
Beispielsweise `ConfigurationChanged`.

f) Geben Sie im Feld **Beschreibung** die Beschreibung der Situation ein.
Beispiel: `Durch TADDM wurde die Änderung eines überwachten Objekts festgestellt`.

g) Wählen Sie in der Liste **Monitored Application** (Überwachte Anwendung) die Option **Universal Data Provider** (Universeller Datenanbieter) aus.

h) Stellen Sie sicher, dass das Kontrollkästchen **Correlate Situations across Managed Systems** (Situationen verschiedener verwalteter Systeme korrelieren) nicht ausgewählt ist.

i) Klicken Sie auf **OK**.

Das Fenster "**Select condition**" (Bedingung auswählen) wird angezeigt.

j) Wählen Sie in der Liste **Attribute Group** (Attributgruppe) **DPPOST** aus.

k) Wählen Sie in der Liste **Attribute Item** (Attributelement) **Komponentenname** aus.

l) Klicken Sie auf **OK**.

Die Registerkarte **Formula** (Formel) für die Situation wird angezeigt.

m) Konfigurieren Sie die Situation so, dass sie ausgelöst wird, wenn der Komponentenname mit dem Namen der zu überwachenden Ressource in Ihrer Umgebung übereinstimmt.

n) Klicken Sie auf **OK**.

So erstellen Sie bei Verwendung von IBM Tivoli Monitoring 6.2.2 oder höher eine Konfigurationsänderungssituation:

a) Navigieren Sie im Navigatorfenster von IBM Tivoli Enterprise Portal zu dem Universal Agent, der für das Senden und Empfangen von Ereignisbenachrichtigungen von TADDM konfiguriert wurde.

b) Erweitern Sie den Knoten CONFIGCHANGE.

c) Klicken Sie mit der rechten Maustaste auf den Knoten DPPOST und klicken Sie auf **Situations** (Situationen).

d) Klicken Sie im Fenster "**Situations for** *Knotenname*" (Situationen für Knotenname) auf **Create new Situation** (Neue Situation erstellen).

Das Fenster **Create Situation** (Situation erstellen) wird angezeigt.

e) Geben Sie im Feld **Name** den Namen der Situation ein.
Beispielsweise `ConfigurationChanged`.

f) Geben Sie im Feld **Beschreibung** die Beschreibung der Situation ein.
Beispiel: `Durch TADDM wurde die Änderung eines überwachten Objekts festgestellt`.

g) Wählen Sie in der Liste **Monitored Application** (Überwachte Anwendung) die Option **Universal Data Provider** (Universeller Datenanbieter) aus.

h) Klicken Sie auf **OK**.

Das Fenster "**Select condition**" (Bedingung auswählen) wird angezeigt.

i) Wählen Sie in der Liste **Attribute Group** (Attributgruppe) **DPPOST** aus.

j) Wählen Sie in der Liste **Attribute Item** (Attributelement) **Komponentenname** aus.

k) Klicken Sie auf **OK**.

Die Registerkarte **Formula** (Formel) für die Situation wird angezeigt.

l) Konfigurieren Sie die Situation so, dass sie ausgelöst wird, wenn der Komponentenname mit dem Namen der zu überwachenden Ressource in Ihrer Umgebung übereinstimmt.

m) Klicken Sie auf **OK**.

n) Klicken Sie im Navigatorfenster von IBM Tivoli Enterprise Portal mit der rechten Maustaste auf den Knoten, der den Änderungsereignisbericht enthält. klicken Sie auf **Situations** (Situationen).

o) Klicken Sie im Fenster "**Situations for** *Knotenname*" (Situationen für Knotenname) mit der rechten Maustaste auf die von Ihnen erstellte **ConfigurationChanged**-Situation und klicken Sie auf **Start Situation** (Situation starten).

Ergebnisse

Beim Empfang von Konfigurationsänderungsereignissen wird deren Komponentenname überprüft. Wenn der Komponentenname mit dem der Komponente übereinstimmt, die Sie in der Situationsformel angegeben haben, wird die konfigurierte Situation ausgelöst.

Detaillinks in Konfigurationsänderungs-Ereignisberichten in IBM Tivoli Monitoring erstellen

Sie können Links in einer Berichtstabelle zu einem Arbeitsbereich, der ein Änderungsprotokoll sowie Details anzeigt, direkt über den TADDM-Server erstellen. Über diese Links sind detailliertere Informationen verfügbar als in einem Bericht.

Vorgehensweise

Gehen Sie wie folgt vor, um in einem Konfigurationsänderungs-Ereignisbericht einen Link zu detaillierteren Änderungsereignisinformationen zu erstellen:

1. Gehen Sie folgendermaßen vor, um einen Arbeitsbereich zur Anzeige der Informationen zu erstellen:
   a) Klicken Sie im Navigatorfenster mit der rechten Maustaste auf den Knoten, der den Arbeitsbereich enthalten soll. Klicken Sie auf **Datei** > **Save workspace as (Arbeitsbereich speichern als)**.

      Das Fenster **Save Workspace As** (Arbeitsbereich speichern als) wird angezeigt.
   b) Geben Sie im Feld **Name** den Namen des Arbeitsbereichs ein.
   Beispielsweise `ConfigChangeDetails`.
   c) Geben Sie im Feld **Beschreibung** eine Beschreibung des Arbeitsbereichs ein.
   Beispielsweise `Generic workspace for the change event table` (Allgemeiner Arbeitsbereich für die Änderungsereignistabelle).
   d) Wählen Sie das Kontrollkästchen **Only selectable as the target of a Workspace Link** (Nur als Ziel eines Arbeitsbereichslinks auswählbar) aus.
   e) Klicken Sie auf **OK**.
2. So konfigurieren Sie den Arbeitsbereich mithilfe von IBM Tivoli Monitoring 6.2.1 oder höher:
   a) Konfigurieren Sie den Arbeitsbereich so, dass er über ein Navigatorfenster und zwei Browserfenster verfügt.
   b) Klicken Sie auf **Bearbeiten** > **Eigenschaften**.
   c) Wählen Sie im **Browser**-Fenster die erste Instanz von **Getting Started** (Erste Schritte) aus.
   d) Wählen Sie im Fenster **Style** (Stil) die Option **Use Provided Location** (Bereitgestellte Position verwenden) aus.
   e) Klicken Sie auf **OK**.
   f) Geben Sie im Feld **Position** eines der Browserfenster die URL der Änderungsprotokollansicht in TADDM ein. Drücken Sie nach der Eingabe der URL in einer Zeile nicht die **Eingabetaste**.

      ```
      http://$taddm_server$:$taddm_port$/cdm/servlet/
      LICServlet?view=changehistory&hoursback=10000&console=web&guid=$taddm_guid$
      ```

      Der Parameter `hoursback` gibt in Stunden an, wie lange Änderungsereignisse angezeigt werden. Durch Setzen des Parameters `hoursback` auf 6 werden beispielsweise alle Änderungsereignisse der letzten sechs Stunden angezeigt.
   g) Wählen Sie im **Browser**-Fenster die zweite Instanz von **Getting Started** (Erste Schritte) aus.
   h) Wählen Sie im Fenster **Style** (Stil) die Option **Use Provided Location** (Bereitgestellte Position verwenden) aus.
   i) Klicken Sie auf **OK**.
   j) Geben Sie im Feld **Position** des zweiten Browserfensters die URL der Objektdetailansicht in TADDM ein. Drücken Sie nach der Eingabe der URL in einer Zeile nicht die **Eingabetaste**.

      ```
      http://$taddm_server$:$taddm_port$/cdm/servlet/LICServlet?console=web
      &guid=$taddm_guid$
      ```
   k) Klicken Sie auf **Datei** > **Speichern**, um den neuen Arbeitsbereich zu speichern.

   Drücken Sie direkt nach der Eingabe der URL im Feld **Position** nicht die **Eingabetaste**, sondern speichern Sie den Arbeitsbereich.
3. Öffnen Sie IBM Tivoli Enterprise Portal. Klicken Sie im Berichtsfenster mit der rechten Maustaste auf eine Zeile der **Berichts**-Tabelle.
4. Klicken Sie auf **Link To (Link zu)** > **Link Wizard (Linkassistent)**.

   Die Begrüßungsseite des Arbeitsbereichslinkassistenten wird angezeigt.
5. Klicken Sie auf **Create a new link** (Neuen Link erstellen). Klicken Sie auf **Weiter**.

   Die Linknamensseite des Arbeitsbereichslinkassistenten wird angezeigt.

6. Geben Sie im Feld **Name** den Namen des Links ein.
   Beispielsweise `Show Details` (Details anzeigen).
7. Geben Sie im Feld **Beschreibung** eine Beschreibung des Links ein.
   Beispielsweise `Link to details` (Link zu Details).
8. Klicken Sie auf **Weiter**.

   Die Linktypenseite des Arbeitsbereichslinkassistenten wird angezeigt.
9. Klicken Sie auf **Absolute** (Absolut). Klicken Sie auf **Weiter**.

   Die Zielarbeitsbereichsseite des Arbeitsbereichslinkassistenten wird angezeigt.
10. Wählen Sie im Navigatorfenster den Knoten aus, der den von Ihnen erstellten Arbeitsbereich enthält. Wählen Sie im Arbeitsbereichsfenster den von Ihnen erstellten Arbeitsbereich aus.
11. Klicken Sie auf **Next** (Weiter).

    Die Parameterseite des Arbeitsbereichs-Linkassistenten wird angezeigt.
12. Die folgenden drei Symbole müssen hinzugefügt werden: 'taddm_server', 'taddm_port' und 'taddm_guid'. Gehen Sie folgendermaßen vor, um ein Symbol hinzuzufügen:
    a) Klicken Sie auf **Add Symbol** (Symbol hinzufügen).

       Das Fenster **Add Symbol** (Symbol hinzufügen) wird angezeigt.
    b) Geben Sie im Feld **Symbol** den Namen des Symbols ein.
    c) Klicken Sie auf **OK**.
13. Sie müssen alle erstellten Symbole mit einem Attribut verknüpfen, das für die korrekte Spalte des Berichts steht.
    - Verknüpfen Sie das Symbol 'taddm_server' mit dem TADDM-Serverattribut.
    - Verknüpfen Sie das Symbol 'taddm_port' mit der Portnummer der TADDM-Webkonsole.
    - Verknüpfen Sie das Symbol 'taddm_guid' mit dem Komponenten-GUID-Attribut.

    Gehen Sie folgendermaßen vor, um ein Symbol mit einem Attribut zu verknüpfen:
    a) Wählen Sie auf der Parameterseite des Arbeitsbereichs-Linkassistenten das Symbol aus, das mit einer Berichtsspalte verknüpft werden soll.
    b) Klicken Sie auf **Modify Expression** (Ausdruck ändern).

       Das Fenster **Expression Editor** (Ausdrucksеditor) wird angezeigt.
    c) Klicken Sie auf **Symbol**.

       Das Fenster **Symbols** (Symbole) wird angezeigt.
    d) Navigieren Sie zu **Attributes** (Attribute) und wählen Sie das Attribut aus, das mit dem Symbol verknüpft werden soll. Klicken Sie auf **OK**.
    e) Klicken Sie im Fenster **Expression Editor** (Ausdrucksеditor) auf **OK**.

       Die Parameterseite des Arbeitsbereichslinkassistenten wird angezeigt.
14. Klicken Sie auf **Weiter**.

    Die Übersichtsseite des Arbeitsbereichslinkassistenten wird angezeigt.
15. Klicken Sie auf **Fertigstellen**.

Ergebnisse
Wenn sich aktive Ereignisse in Ihrem Änderungsereignisbericht befinden, wird neben jeder Tabellenzeile ein Linksymbol angezeigt. Um in den Zielarbeitsbereich zu wechseln, klicken Sie auf das Linksymbol und wählen Sie die Option **Details anzeigen** aus. In der Tabellenzeile werden die Werte durch Symbole ersetzt. Im Arbeitsbereich werden Änderungsprotokoll- und Objektdetailfenster gemeinsam gestartet.

Änderungsereignisse für ein Business-System konfigurieren
Mit der Änderungsereignisfunktion können Sie bei jeder Änderung eines Business-Systems ein Änderungsereignis senden.

Informationen zu diesem Vorgang
Standardmäßig zeigt TADDM keine Business-Systemänderungen an, wenn einer der Computer, von denen das System abhängig ist, geändert wurde.

Vorgehensweise

Führen Sie die folgenden Schritte aus, um das Senden von Änderungsereignissen für Business-Systeme zu aktivieren:

1. Öffnen Sie `$COLLATION_HOME/etc/propagationserver.xml` in einem entsprechenden Editor.
2. Setzen Sie im Abschnitt 'Computersystem' für die Elemente der Anwendungs- und Business-System-Beziehung den Wert des Attributs `enabled` (aktiviert) auf `true` (wahr).

   Beispiele dafür sind:

   ```
   <relationship enabled="true" source="sys.ComputerSystem" attribute="groups"
   target="app.Application" targetAttribute="true"
   collectionType="app.FunctionalGroup" radius="1"/>

   <relationship enabled="true" source="sys.ComputerSystem" attribute="components"
   target="sys.BusinessSystem" targetAttribute="true"/>
   ```
3. Starten Sie TADDM erneut.
4. Erstellen Sie für das Business-System in der Änderungsereigniskonfiguration `$COLLATION_HOME/etc/EventConfig.xml` einen Listener.

   Im folgenden Beispiel ist `mycompany-itm` der Ereignisempfänger und `MyBiz` der Name des Business-Systems.

   ```
   <listener object="ITSystem" enabled="true">
       <alert recipient="mycompany-itm"/>
       <attribute name="name" operator="equals">
           <value>MyBiz</value>
       </attribute>
   </listener>
   ```

Jobs mit IBM Tivoli Workload Scheduler planen

Zur Planung von Jobs in TADDM verwenden Sie IBM Tivoli Workload Scheduler. IBM Tivoli Workload Scheduler ist ein Softwareautomatisierungstool, das die Zentralverbindung für das automatisierte Auslastungsmanagement und die automatisierte Auslastungsüberwachung darstellt.

Verwenden Sie IBM Tivoli Workload Scheduler 8.5.1 oder höher. Sie müssen den Master Domain Manager und den Fault Tolerant Agent auf dem TADDM-Server installieren. Informationen zur Installation und Konfiguration von Tivoli Workload Scheduler finden Sie unter http://www-01.ibm.com/support/knowledgecenter/SSGSPN_8.5.1.1/com.ibm.tivoli.itws.doc_8.5.1.1/ic-homepage.html?lang=en. Zeitplanobjekte werden mit dem Composer-Befehlszeilenprogramm verwaltet und in Tivoli Workload Scheduler gespeichert.

Bei Tivoli Workload Scheduler-Jobs wird das Script `invokejob.sh` für die Ausführung der erforderlichen Operation verwendet. Das Script `invokejob.sh` wird bei der TADDM-Installation bereitgestellt.

Die folgenden Parameter gelten für alle Nutzungen des Scripts:

Erforderlich: -u *Benutzer*
Dieser Wert gibt den Benutzer an, der den API-Befehl ausführt.

Erforderlich: -p *Kennwort*
Dieser Wert gibt das Kennwort an, das den Benutzer authentifiziert.

Erforderlich: -profile *Profil*
Dieser Wert definiert das Erkennungsprofil.

Optional: -H *Host*
Dieser Wert gibt den Hostnamen des TADDM-Servers an. Der Standardname lautet `localhost`. Wenn Sie den Parameter **-T** verwenden, müssen Sie auch den Parameter **-H** angeben.

Optional: -P *Port*
Dieser Wert gibt den Port des TADDM-Servers an. Der Standardwert ist 9433.

Optional: -v *Version*
Dieser Wert gibt den Versionsnamen oder die Versionsnummer an. Der Standardwert ist `0`.

Optional: -t *Zeitlimit*
Dieser Wert gibt die Zeitspanne an, nach der der Job automatisch unterbrochen wird.

Optional: -T|--truststorefile *Truststore*
Dieser Wert gibt die Position der Truststore-Datei (`jssecacerts.cert`) mit einem Zertifikat für die Verbindung mit dem TADDM-Server an. Dieser Parameter ist für eine sichere Verbindung mit TADDM erforderlich. Wenn Sie diesen Parameter verwenden, müssen Sie auch den Parameter **-H** angeben.

Gehen Sie folgendermaßen vor, um einen Job zu planen:

1. Geben Sie über Tivoli Workload Scheduler die TADDM-Jobdefinitionsdatei in eine Bearbeitungsdatei ein. Das folgende Beispiel stellt eine Schablonenjobdefinition dar:

   ```
   WORKSTATION_ID#TADDM_JOB
    SCRIPTNAME "./opt/IBM/taddm/dist/bin/invokejob.sh -u
   ^TADDM-BENUTZERNAME^ -p ^TADDM-KENNWORT^ Befehl [Parameter]"
    STREAMLOGON taddmuser
    TASKTYPE UNIX
    RECOVERY STOP
   ```

   *^TADDM-BENUTZERNAME^* und *^TADDM-KENNWORT^* sind Variablen, die in Tivoli Workload Scheduler definiert werden müssen. Diese Variablen werden Werten zugeordnet, die in der Datenbank gespeichert sind. Verwenden Sie, vor allem bei der Codierung von Kennwörtern, aus Sicherheitsgründen Variablen, um sicherzustellen, dass die Werte nicht als Klartext sichtbar sind.
2. Fügen Sie die Bearbeitungsdatei mithilfe des Composers zur Datenbank hinzu.
3. Fügen Sie den Job zu einem Jobstrom hinzu und planen Sie die Ausführung des Jobstroms. Der IBM Tivoli Workload Scheduler-Agent startet und überwacht die Aktion des Scripts `invokejob.sh`.

Erkennungsjob planen

Im folgenden Beispiel wird eine Erkennung in Bereich 127.0.0.1 ausgeführt:

```
dist/bin/invokejob.sh -u USER -p PASSWORD --timeout 60000 discover start
--profile  “Level 3 Discovery” 127.0.0.1
```

Im folgenden Beispiel wird eine Erkennung in der Bereichsgruppe 'MyScopeSEt' ausgeführt, die bereits in der Bereichsliste vorhanden sein muss:

```
dist/bin/invokejob.sh -u USER -p PASSWORD --timeout 60000 discover start
--profile “Level 3 Discovery” MyScopeSet
```

In den vorhergehenden Beispielen gibt der letzte Parameter das Bereichselement oder die Bereichsgruppe an, das bzw. die in den Erkennungslauf miteinbezogen werden soll. Der Parameter **profile** ist erforderlich. Der Parameter **name**, der den Namen des Erkennungslaufs angibt, ist optional.

Der folgende Befehl ist ein Beispiel für das Stoppen einer Erkennung, die gerade ausgeführt wird:

```
dist/bin/invokejob.sh -u USER -p PASSWORD --timeout 60000 discover stop
```

Für den Befehl **discover stop** sind keine zusätzlichen Argumente erforderlich.

Domänensynchronisationsjob planen

Im folgenden Beispiel sind die Befehlszeilensyntax und die Befehlszeilenoptionen dargestellt, die für die Ausführung einer Domänensynchronisation in einer Synchronisationsserverimplementierung mit dem TADDM-Script `invokejob.sh` erforderlich sind:

```
dist/bin/invokejob.sh -u USER -p PASSWORD --timeout 60000 sync start TestDomain
```

Sowohl für den Befehl **sync start** als auch für den Befehl **sync stop** ist ein Argument erforderlich, nämlich der Name der Domäne, für die der Synchronisationsjob gestartet oder gestoppt werden soll.

Kombinierter Einsatz von TADDM mit IBM Tivoli Business Service Manager

Je nach den Tasks, die Sie in Ihrer IT-Umgebung ausführen möchten, können Sie die Integrationsfunktionen verwenden, die für TADDM und IBM Tivoli Business Service Manager verfügbar sind. Um diese Funktionen verwenden zu können, müssen Sie über den vorläufigen Fix 3 für IBM Tivoli Business Service Manager 4.2.1 verfügen, es ist jedoch keine zusätzliche Konfiguration von TADDM erforderlich.

Lebenszyklusstatus für Geschäftsanwendungen aktualisieren

Mithilfe des Lebenszyklusstatus können Sie Objekte für die Synchronisation in IBM Tivoli Business Service Manager von TADDM aus filtern. Mit dem Programm **BusinessServiceLifecycle** können Sie Informationen zu einem Geschäftsservice auflisten oder den Lebenszyklusstatus einer Geschäftsanwendung festlegen.

Die Anwendung 'ITsystems' von IBM Tivoli Business Service Manager enthält ausschließlich Geschäftsanwendungen. Daher unterstützt das Programm **BusinessServiceLifecycle** nur Geschäftsanwendungen.

Das Programm **BusinessServiceLifecycle** befindet sich an der folgenden Speicherposition:

- Bei Linux- und UNIX-Betriebssystemen befindet sich das Script `BusinessServiceLifecycle` im Verzeichnis `$COLLATION_HOME/bin`.
- Bei Windows befindet sich die Batchdatei `BusinessServiceLifecycle.bat` im Ordner `%COLLATION_HOME%\bin`.

Verwenden Sie das Programm **BusinessServiceLifecycle** mit den folgenden Befehlszeilenoptionen:

```
BusinessServiceLifecycle -u TADDM-Benutzername -p TADDM-Kennwort -l | -s GUID Status
```

Verwenden Sie die Option `-l` zum Auflisten von Informationen zum Lebenszyklus der Geschäftsanwendung oder verwenden Sie die Option `-s` zusammen mit einem GUID-Parameter und einem Statuscodeparameter zum Festlegen eines Lebenszyklusstatus. Sie können die Option `-l` und die Option `-s` nicht gleichzeitig verwenden.

In der folgenden Tabelle werden die gültigen Statuscodes aufgeführt:

Tabelle 50. Statuscodes

Code	Status
0	Unbekannt
1	Sonstige
2	Bestellt
3	Empfangen
4	Wird getestet
5	Getestet
6	Installiert
7	Aktiviert
8	Inaktiviert
9	Wird gewartet
10	Gesperrt
11	Archiviert
12	Akzeptiert

Tabelle 50. Statuscodes (Forts.)	
Code	**Status**
13	Build
14	Entwicklung
15	Entwurf
16	Bestand
17	Offline
18	Nachbereitung
19	Produktion
20	Produktion bereit
21	Sunset
22	Prüfen

Integration von TADDM in Jazz for Service Management

TADDM unterstützt die Integration in Plattformen mit Open Services for Lifecycle Collaboration (OSLC). Wenn OSLC zusammen mit TADDM verwendet wird, können Sie Erkennungsdaten abrufen, die im Format standardmäßiger Ressourcendefinitionen dargestellt werden. Bei der Jazz for Service Management-Plattform handelt es sich um ein IBM Integrationstool, das auf den Spezifikationen der offenen OSLC-Community basiert.

Jazz for Service Management stellt eine einheitliche Konfiguration und Verwaltung für alle Tivoli-Produkte sowie für die Produkte anderer Anbieter bereit. Jazz for Service Management zeigt eine umfassende Ansicht von IT-Ressourcen, Anwendungen und Geschäftsbeziehungen.

TADDM OSLC REST-Kommunikation

Der REST-Service (Representational State Transfer) von TADDM stellt eine OSLC-Integration in einer Vielzahl von OSLC REST-Feeds bereit. Der Service gibt die Medientypen an, die bei der Ausführung zurückgegeben werden, und beschreibt die Sicherheitsaspekte, die mit dem Service verbunden sind.

Bei 'Common Resource Type Vocabulary' (CRTV) handelt es sich um ein von IBM und der OSLC-Community definiertes Datenmodell, das von TADDM zusammen mit dem Tivoli Common Data Model (CDM) unterstützt wird. Die TADDM-Unterstützung für OSLC macht CDM-Erkennungsdaten im Format von CRTV definierten Ressourcen verfügbar.

OSLC REST-Schnittstelle

In TADDM for Open Services Lifecycle Collaboration (OSLC) ist eine REST-Schnittstelle verfügbar. Über die OSLC REST-Schnittstelle können Sie Informationen zu registrierten Konfigurationselementen, den zugehörigen Attributen und zum Ändern von Protokollen abrufen.

Sie können Informationen zu den Attributen der Konfigurationselemente nur abrufen, wenn diese Attribute von Common Resource Type Vocabulary (CRTV) oder vom TADDM-Vokabular unterstützt werden.

Jede gültige Anforderung muss eine GUID enthalten, mit der das jeweilige Konfigurationselement ermittelt wird.

Es sind zwei Servicetypen verfügbar:

Konfigurationsservice
Dieser Service stellt eine Schnittstelle bereit, über die erweiterte Attribute für eine CRTV-Ressourcen abgerufen werden können.

Änderungsprotokollservice
Dieser Service stellt eine Schnittstelle bereit, über die ein Änderungsprotokoll für einen angegebenen Zeitraum für eine CRTV-Ressource abgerufen werden kann.

Für jeden Service können Sie die folgenden drei Inhaltstypen anzeigen:

- RDF-Darstellung
- Kompakte OSLC-Anzeige
- HTML-Vorschau

Bei der folgenden URL handelt es sich um die Basisadresse:

```
http[s]://TADDM-Host:Port/cdm/oslc/Providername/CI_GUID
```

Dabei gilt:

- *Port* steht für den Port, an dem der Tomcat-Server (TADDM 7.3.0) oder WAS Liberty Profile-Server (TADDM 7.3.0.1 und höher) empfangsbereit ist. Der Standardwert ist 9430.
- *Providername* ist einer der beiden folgenden Werte, je nach dem, welchen Service Sie verwenden möchten:
  - `Konfiguration`
  - `Änderungsprotokoll`
- *CI_GUID* ist die ID des Konfigurationselements in TADDM

Rufen Sie die HTML-Vorschau für ein Konfigurationselement mit folgender URL auf:

- `http[s]://TADDM-Host:Port/cdm/oslc/Providername/CI_GIUD/preview`

Die OSLC REST-Schnittstelle akzeptiert nur HTTP-GET-Anforderungen. Mit dem HTTP-Accept-Header können Sie den Typ des zurückgegebenen Inhalts angeben.

Geben Sie folgenden Accept-Header an, um die kompakte OSLC-Anzeige für das gegebene Konfigurationselement anzuzeigen:

```
application/x-oslc-compact+xml
```

Geben Sie folgenden Accept-Header an, um die RDF-Darstellung für das gegebene Konfigurationselement anzuzeigen:

```
application/rdf+xml
```

Hierbei handelt es sich um das Standardverhalten, falls für den Accept-Header kein Wert bereitgestellt wurde.

Kompakte OSLC-Anzeige

Bei der kompakten OSLC-Anzeige handelt es sich um eine XML-Darstellung einer Zielressource.

Die kompakte OSLC-Anzeige ist eine Vorschau, die von der OSLC REST-Schnittstelle bereitgestellt wird. Um die Vorschau einer Zielressource anzuzeigen, muss der Provider eine Darstellung der Ressourcen gemäß der Definition in der OSLC-Spezifikation bereitstellen.

Sie können diese Darstellung der Ressource mithilfe einer HTTP GET-Anforderung an die URI der Zielressource zusammen mit dem Zugriffsheader `application/x-oslc-compact+xml` abrufen.

Wenn der Provider die Vorschau unterstützt, antwortet er mit einer kompakten Darstellung, in der Informationen enthalten sind, mit denen der Nutzer Links sowie eine Vorschau der Zielressource anzeigen kann.

HTML-Vorschau von Jazz for Service Management

Jazz for Service Management Registry Services stellt eine HTML-Benutzerschnittstelle bereit, um Informationen zu registrierten Elementen von verbundenen externen Systemen zu übermitteln.

Alle Elemente mit Daten, die von TADDM bereitgestellt werden, verfügen über eine HTML-Vorschau, die eine schnelle Übersicht über ausgewählte Elementdaten direkt vom TADDM-Server bereitstellt.

TADDM stellt Jazz for Service Management mit einem Feed-Service unter folgender Adresse bereit:

```
http[s]://Hostname:Port/cdm/oslc/configuration/GUID/preview
```

Dabei stehen *Hostname* und *Port* für den Hostnamen und die Portnummer des TADDM-Servers und *GUID* für die eindeutige Elementkennung.

Die URL zeigt eine Seite mit Informationen zur Übersicht des ausgewählten Elements an. Die Seite wird automatisch in der Benutzerschnittstelle von Jazz for Service Management angezeigt.

Der Seiteninhalt entspricht weitgehend der Registerkarte **General** (Allgemein) in der Ansicht 'Inventory Summary Details' (Einzelheiten zur Zusammenfassung des Bestands) im TADDM-Datenmanagementportal.

Sicherheit

Sie können TADDM so konfigurieren, dass auf den von der OSLC REST-Schnittstelle bereitgestellten Zugriff auf die Feeds eine Authentifizierung erforderlich ist.

Für den Zugriff auf die REST-Schnittstelle müssen Sie sich mit einer der folgenden Methoden authentifizieren:

HTTP-Basisauthentifizierung
: Die Berechtigungsnachweise müssen im Anforderungsheader für die Autorisierung festgelegt werden. Der Wert dieses Headers muss den Regeln für die HTTP-Basisauthentifizierung entsprechen.

Single Sign-on (einmalige Anmeldung)
: Bei der Verwendung von Single Sign-on müssen alle Anforderungen, die an die REST-Schnittstelle übergeben werden, ein LTPA-Token (Lightweight Third-Party Authentication) enthalten. Zur Überprüfung des Tokens muss TADDM für die Verwendung von WebSphere Virtual Member Manager (VMM) als Benutzerrepository konfiguriert sein.

  Weitere Informationen zur Konfiguration von VMM finden Sie im Abschnitt „TADDM-Server für die Verwendung eingebundener WebSphere-Repositorys konfigurieren” auf Seite 25.

Um angeforderte Feeds bereitzustellen, die ohne Authentifizierung präsentiert werden sollen, muss in der Datei `collation.properties` die folgende Eigenschaft mit einer gültigen Registry Services-URL konfiguriert werden:

```
com.ibm.cdb.topobuilder.integration.oslc.frsurl
```

Anschließend werden ein vorkonfigurierter Benutzername und ein vorkonfiguriertes Kennwort verwendet, falls in die Anforderung keine gültigen Berechtigungsnachweise integriert sind.

Der Benutzername und das Kennwort werden aus der Implementierungsdeskriptordatei `web.xml` der Common Data Model-Webanwendung übernommen. Sie können diese Anpassung mithilfe der folgenden OSLCFilter-Initialisierungsparameter konfigurieren:

OSLC_LOGIN_OFF
: Wenn dieser Parameter auf `true` gesetzt ist, wird die von den Parametern OSLC_USER und OSLC_PASSWORD angegebene Kombination aus Benutzername und Kennwort verwendet, falls eingehende Anforderungen keine eigenen gültigen Berechtigungsnachweise enthalten.

  Wenn dieser Parameter auf `false` gesetzt ist, muss die eingehende Anforderung gültige Berechtigungsnachweise enthalten.

  Der Standardwert ist `true`.

OSLC_USER
: Mit diesem Parameter wird der Benutzername festgelegt, der verwendet wird, falls in die Anforderung keine gültigen Berechtigungsnachweise integriert sind. Bei Bedarf kann der verwendete Benutzername geändert werden.

  Der Standardwert ist `administrator`.

OSLC_PASSWORD
Mit diesem Parameter wird das Kennwort festgelegt, das verwendet wird, falls in die Anforderung keine gültigen Berechtigungsnachweise integriert sind. Wenn Sie das Kennwort des Administrators mithilfe der TADDM-Benutzerschnittstelle ändern, müssen Sie den durch diesen Parameter festgelegten Kennwortwert aktualisieren.

Der Standardwert ist `collation`.

Daten mit OSLCAgent in Registry Services exportieren
Mit dem OSLCAgent-Topologieagenten können Sie Informationen zu Konfigurationselementen (KE) in Registry Services exportieren.

Bei OSLCAgent handelt es sich um eine automatisierte Lösung für den Export von Daten von TADDM in Registry Services. Der Agent führt regelmäßig die folgenden Tasks aus:

- Abfragen nach Objekten, die in Registry Services registriert werden können
- Übersetzung dieser Objekte in RDF-formatierte Nachrichten
- Übertragung dieser Nachrichten mithilfe von HTTP

Der OSLCAgent gehört zur Gruppe 'Integration'. Das Zeitintervall zwischen den Ausführungen wird in der Datei `collation.properties` in folgendem Eintrag angegeben:

```
com.ibm.cdb.topobuilder.groupinterval.integration
```

Der OSLCAgent kann als Konfigurations-Provider und als Provider zum Ändern von Protokollen fungieren. Diese beiden Rollen können separat aktiviert werden. Um die Rolle des Konfigurations-Providers zu aktivieren, setzen Sie die folgende Eigenschaft auf `true`:

```
com.ibm.cdb.topobuilder.integration.oslc.enable.configurationsp
```

Um die Rolle des Providers zum Ändern von Protokollen zu aktivieren, setzen Sie die folgende Eigenschaft auf `true`:

```
com.ibm.cdb.topobuilder.integration.oslc.enable.changehistorysp
```

Um OSLCAgent für die Verbindung zu Registry Services zu konfigurieren, müssen Sie die Registry Services-Adresse und Zugriffseingabedetails angeben.

Konfigurieren Sie die Registry Services-Adresse in der folgenden Eigenschaft:

```
com.ibm.cdb.topobuilder.integration.oslc.frsurl
```

Geben Sie die Registry Services-Adresse im folgenden Format an:

```
Protokoll://FQDN_oder_IP_oder_Hostname:Port
```

Beispiel: `http://192.0.2.24:9081`

Anmerkung: Der IP-Adresse vorzuziehen ist der vollständig qualifizierte Domänenname (FQDN) bzw. der vollständig qualifizierte Hostname. Dadurch vermeiden Sie Integrationsprobleme mit anderen Produkten. Verwenden jedoch alle anderen mit TADDM integrierten Produkte die IP-Adresse, so müssen Sie auch hier die IP-Adresse angeben. Werden mit TADDM noch keine anderen Produkte verwendet, so ist für den Fall, dass später weitere Produkte hinzugefügt werden, der vollständig qualifizierte Domänenname vorzuziehen.

Erstellen Sie einen Zugriffslisteneintrag vom Typ **Integration/Registry Service**. Geben Sie den Benutzernamen und das Kennwort für Registry Services an.

Mit folgenden Eigenschaften können Sie die Ausführung von OSLCAgent optimieren:

com.ibm.cdb.topobuilder.integration.oslc.maxtimeperrun
Diese Eigenschaft gibt die maximale Dauer (in Minuten) an, die der OSLCAgent ausgeführt werden darf. Diese Dauer kann für jeden Provider durch die Zeitlänge überschritten werden, die von den Jobs benötigt wird, die vor der Zeitlimitüberschreitung in den Pool übergeben werden. Wenn die Eigen-

schaft nicht konfiguriert oder auf den Wert `-1` gesetzt ist, hat der OSLCAgent für eine einzelne Ausführung unbegrenzt Zeit zur Verfügung.

com.ibm.cdb.topobuilder.integration.oslc.jobspoolsize
Diese Eigenschaft gibt die maximale Anzahl von Jobs an, die gleichzeitig ausgeführt werden können. Jeder Job registriert ein einzelnes Konfigurationselement. Wenn die Eigenschaft nicht konfiguriert ist, hat sie den Standardwert `10`.

com.ibm.cdb.topobuilder.integration.oslc.frshttptimeout
Diese Eigenschaft gibt in Millisekunden das Zeitlimit für HTTP-Verbindungen an. Der Standardwert ist `5000`.

com.ibm.cdb.topobuilder.integration.oslc.frsfailfastafter
Diese Eigenschaft gibt an, nach wie vielen aufeinanderfolgenden HTTP-Verbindungszeitlimits ein Agent Registry Services als nicht verfügbar erachtet und nicht mehr versucht, eine Verbindung herzustellen. Der Standardwert ist 5.

com.ibm.cdb.topobuilder.integration.oslc.unregisterableIPs
Diese Eigenschaft gibt IP-Adressen an, die nicht registriert werden sollten, da sie nicht öffentlich sind und zu einer übermäßigen Zusammenführung der Ressourcendatensätze führen können. Diese Eigenschaft kann reguläre Ausdrücke enthalten. Der Standardwert ist
`127.0.0.1,01111111000000000000000000000001,0:0:0:0:0:0:0:1,`
`0000000000000000000000000000000000000000000000000000000000`
`0000000000000000000000000000000000000000000000000000000000`
`000000000001`.

com.ibm.cdb.topobuilder.integration.oslc.unregisterableFQDNs
Diese Eigenschaft gibt vollständig qualifizierte Domänennamen an, die nicht registriert werden sollten, da sie nicht öffentlich sind und zu einer übermäßigen Zusammenführung der Ressourcendatensätze führen können. Diese Eigenschaft kann reguläre Ausdrücke enthalten. Der Standardwert ist `localhost.localdomain,localhost,loopback6(.?\u005c\w*)*`.

com.ibm.cdb.topobuilder.integration.oslc.enablecrtvtype.*CRTV-Typ*
Diese Eigenschaft gibt an, dass der OSLCAgent nur Konfigurationselemente eines bestimmten CRTV-Typs (Common Resource Type Vocabulary) verarbeitet. Wenn die Eigenschaft aktiviert ist, ist das Registrieren, Aktualisieren und Zurücknehmen der Registrierung nur für Konfigurationselemente dieses CRTV-Typs möglich.

Beispielsweise gibt die Einstellung

```
com.ibm.cdb.topobuilder.integration.oslc.enablecrtvtype.ComputerSystem=true
```

an, dass nur Computersysteme verarbeitet werden.

Folgende CRTV-Typen sind zulässig:

- ComputerSystem
- Database
- Fix Pack 2 SAPSystem
- ServiceInstance
- SoftwareModule
- SoftwareServer

com.ibm.cdb.topobuilder.integration.oslc.history.days_previous
Diese Eigenschaft gibt den Zeitraum in Tagen an, aus dem die Informationen des Änderungsprotokolls über die Launch-in-Context-URL (LIC-URL) zur Verfügung stehen. Der Standardwert ist 5.

Zur Steuerung der Menge an Speicherplatz und der Verarbeitungszeit, die zur Verwaltung der Informationen aus dem Änderungsprotokoll erforderlich sind, können Sie die Eigenschaft `days_previous` verwenden.

Die LIC-URL ist in der HTML-Vorschau des OSLC-Änderungsprotokolls angegeben.

Bei einem Wert größer als 0 wird der Parameter `days_previous` auf die LIC-URL angewendet, um die Menge der angezeigten Änderungsprotokolldaten einzugrenzen.

Bei 0 oder einem niedrigeren Wert bleibt der Parameter `days_previous` für die LIC-URL unberücksichtigt und das vollständige Änderungsprotokoll dieser CI wird angezeigt.

Befehlszeilenschnittstelle für OSLCAgent

Über die OSLCAgent-Befehlszeilenschnittstelle können Sie manuell Informationen zu Konfigurationselementen (KE) in Registry Services exportieren.

Für OSLCAgent können Sie eine Kombination aus Befehlen und Switches an das Script oder die Stapeldatei `runtopobuild` übergeben. Jeder Befehl und jeder Switch hat ein kurzes einstelliges Format und ein längeres beschreibendes Format. Sie können die Befehls- und Switchformate beliebig kombinieren.

Die folgenden Befehle stehen zur Verfügung:

- `-R | –refreshAll true|false`

  Dieser Befehl registriert alle zulässigen Konfigurationselemente, und zwar selbst dann, wenn sie bereits registriert wurden.
- `-r | -refreshGuid` *`GUID`*

  Dieser Befehl registriert das Konfigurationselement, das die angegebene GUID aufweist, und zwar selbst dann, wenn es bereits registriert wurde.
- `-l | -refreshIgnored true|false`

  Falls ein KE in einer Position erkannt wird, die nicht tief genug ist, sind die Benennungsregeln für das KE möglicherweise nicht korrekt gebildet. Standardmäßig werden solche KEs von OSLCAgent ignoriert. Mit diesem Befehl wird OSLCAgent gezwungen, diese KEs erneut zu verarbeiten.

Wenn Sie bestimmte Aktionen angeben möchten, können Sie mit jedem Befehl einen Switch übergeben. Es stehen zwei Arten von Switches zur Verfügung.

Mit den folgenden Switches können Sie die Verarbeitung bestimmter CRTV-Typen aktivieren oder inaktivieren:

- `-c | --enableComputerSystem true|false`
- `-d | --enableDatabase true|false`
- `-i | --enableServiceInstance true|false`
- `-m | --enableSoftwareModule true|false`
- `-s | --enableSoftwareServer true|false`

Wenn Sie Computersysteme beispielsweise nicht erneut registrieren möchten, verwenden Sie die Switches `-c false`.

Mit den folgenden Switches können Sie die Rollen für die Konfiguration und Änderungsprotokolle aktivieren oder inaktivieren:

- `-h | --enableChangeHistoryProvider true|false`
- `-p | --enableConfigurationProvider true|false`

Wenn Sie beispielsweise keine erneute Registrierung als Änderungsprotokollprovider vornehmen möchten, verwenden Sie die Switches `-h false`.

Falls Standardwerte verwendet werden sollen, wenn Sie bei der Ausführung des Scripts oder der Stapeldatei `runtopobuild` keinen Befehl oder keinen Switch übergeben, konfigurieren Sie die folgenden Eigenschaften in der Datei `collation.properties`:

- `com.ibm.cdb.topobuilder.integration.oslc.refreshAll=true|false`
- `com.ibm.cdb.topobuilder.integration.oslc.refreshGuid=`*`GUID`*
- `com.ibm.cdb.topobuilder.integration.oslc.enablecrtvtype.`*`CRTV-Typ`*

Wenn Sie eine vollständige Liste der verfügbaren Parameter und Switches wünschen, wechseln Sie in das Verzeichnis `$COLLATION_HOME/support/bin` und führen Sie das Script oder die Stapeldatei `runtopobuild` mit dem Switch `-H` aus. Beispiel:

```
./runtopobuild.sh -H
```

Konfigurationselemente mit Registry Services registrieren
In diesem Abschnitt sind die von TADDM erkannten Konfigurationselemente (KE) aufgeführt, die für die Registrierung mit Registry Services abgefragt werden. Außerdem sind die definierten Attribute sowie ausführliche Zuordnungsinformationen angegeben.

Wenn ein bestimmtes Konfigurationselement nicht registriert ist, erzeugt jeder Registrierungsthread Protokollinformationen zu den Ursachen, warum das Konfigurationselement nicht registriert wurde. Die Liste der nicht festgelegten Attribute der Namenskonventionen wird im Protokoll aufgeführt. Um die korrekte Protokollierungsstufe zu konfigurieren, legen Sie den folgenden Eigenschaftswert in der Datei `collation.properties` fest:

```
com.collation.log.level.vm.Topology=DEBUG
```

Die folgenden Attribute sind für jeden CRTV-Typ einheitlich:

GUID
Wird mit dem GUID-Wert für das Konfigurationselement festgelegt.

Name
Wird mit dem Wert für den Namen, die Bezeichnung oder das Attribut 'displayName' festgelegt.

Beschreibung
Wird mit dem Wert für das Attribut 'Beschreibung' festgelegt.

lastDiscoveredTime
Wird mit dem Wert für das Attribut 'lastModifiedTime' festgelegt.

SoftwareServer

Der CRTV-Typ 'SoftwareServer' enthält die folgenden TADDM-Klassen und -Attribute:

- WebSphereServer
  - host
  - node
  - node.cell
- Db2Instance
  - home
  - host
- MQQueueManager
  - displayName | label | name
- AppServer
  - displayName | label | name
  - host
- CommunityServer
  - displayName | label
- SametimeServer
  - displayName | label
- MeetingServer

  - displayName | label
- SpecialityServer
  - displayName | label | name
- AgentManager
  - displayName | label
- SharePointRole
  - displayName | label | name

TADDM-Attribute werden CRTV-Attributen folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
PrimarySAP	crtv:serverAccessPoint	Die Ressource 'serviceAccessPoint' wird zusammen mit der Ressource 'IpAddress' auf die sie verweist, mithilfe von 'crtv:ipAddress' registriert.
Version	crtv:version	
Anbietername	crtv:manufacturer	
Host	crtv:runsOn	crtv:runsOn verweist auf ComputerSystem
Ausgangsposition	crtv:instancePath	Nur für DatabaseServer und Db2Instance.
Datenpfad	crtv:instancePath	Nur für MQQueueManager.

rdf:type ist auf einen der folgenden Werte gesetzt:

- `J2EEServer`
- `WebSphereServer`
- `IBMHTTPServer`
- `WebServer`
- `Db2Instance`
- `OracleInstance`
- `MQQueueManager`
- `WebServer`
- `DatabaseInstance`
- `CICSRegion`

Computersystem

Der CRTV-Typ 'ComputerSystem' enthält die folgenden TADDM-Klassen und -Attribute:

- ComputerSystem

  Es ist eine der folgenden Kombinationen aus Attributen festgelegt:
  - System-ID &VMID
  - System-ID
  - Seriennummer & Modell & Hersteller &VMID
  - Seriennummer & Modell & Hersteller
  - Systemplatinen-UUID
  - IP-Schnittstellen

TADDM-Attribute werden CRTV-Attributen folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
Bezeichnung oder Anzeigename	crtv:name	
OSVersion oder OSRunning	crtv:version	
Hostsystem	crtv:dependsOn	
fqdn	crtv:fqdn	
name	crtv:shortHostname	Falls ein Name festgelegt ist und es ein gültiger Hostname ist. Nur für SunSPARCComputerSystem.
IP-Schnittstelle	crtv:ipAddress	Alle FQDNs für diese IP-Adressen werden in crtv:fqdn zusammengeführt.

crtv:type ist mit einem der folgenden Werte festgelegt

- `Generisch`
- `SunFire`
- `SunSPARC`
- `SystemP`
- `Unitär`
- `Virtuell`
- `WPAR`

Für eine LinuxUnitaryComputerSystem-Komponente werden zusätzliche Attribute folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
Hersteller	crtv:manufacturer	
Modell	crtv:model	
Seriennummer	crtv:serialNumber	
VMID	crtv:vmid	Wenn CPUType und Model festgelegt sind: • Für Intel wird VMID auf `null` gesetzt und es wird versucht, crtv:systemBoardUUID mit systemBoardUUID oder convertedUUID festzulegen. • Für Power wird CS ignoriert, falls VMID festgelegt ist.

Für eine SunSPARCUnitaryComputerSystem-Komponente werden zusätzliche Attribute folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
System-ID	crtv:hostid	
VMID	crtv:vmid	

Für andere Computersysteme werden zusätzliche Attribute folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
Hersteller	crtv:manufacturer	

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
Modell	crtv:model	
Seriennummer	crtv:serialNumber	
VMID	crtv:VMID	Wenn OSRunning auf `WindowsOperatingSystem` gesetzt ist, wird VMID auf `null` gesetzt. Wenn OSRunning auf HpUx gesetzt ist, werden VMID, Modell und Seriennummer auf `null` gesetzt.
systemBoardUUID oder convertedUUID	crtv:systemBoardUUID	
worldWideName	crtv:hostid	Nur für FCSwitch, TapeLibrary und TapeMediaChanger.

Datenbank

Der CRTV-Typ 'Database' enthält die folgenden TADDM-Klassen und -Attribute:

- Db2Database
  - name | displayName
- IDSDatabase
  - name | displayName
- IMSDatabase
  - name | displayName
- OracleDatabase
  - name | displayName
- SqlServerDatabase
  - name | displayName
- SybaseDatabase
  - name | displayName
- DominoDatabase
  - name | displayName

TADDM-Attribute werden CRTV-Attributen folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
name	crtv:name	
fileName	crtv:name	Nur für DominoDatabase.
parent	crtv:dbInstance	

ServiceInstance

Je nachdem, ob die Kompatibilität mit früheren Versionen aktiviert ist, enthält der CRTV-Typ ServiceInstance die folgenden TADDM-Klassen und Attribute:

- Kompatibilität mit früheren Versionen ist aktiviert:
  - BusinessSystem
    - name

  - Application
    - name
  - ServiceInstance
    - name
  - ServiceInfrastructure
    - name
  - SAPSystem
    - SAPSystemSID | systemHome
- Kompatibilität mit früheren Versionen ist inaktiviert:
  - CustomCollection (nur mit dem Typ „BusinessApplication”)
    - collectionId

TADDM-Attribute werden CRTV-Attributen folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
name	crtv:name	
SAPSystemSID:systemHome	crtv:name	Falls weder 'name' noch 'displayName' festgelegt ist. Nur für SAPSystem.
parentGUID oder NULL	crtv:parentServiceInstance	
collectionId	crtv:name	

SoftwareModule

Der CRTV-Typ 'SoftwareModule' enthält die folgenden TADDM-Klassen und -Attribute:

- SoftwareModule
  - fileName
  - name
  - parent.name
- MQQueue
  - name
  - queueManager

TADDM-Attribute werden CRTV-Attributen folgendermaßen zugeordnet:

TADDM-Attribut	CRTV-Attribut	Weitere Informationen
parent	deployedTo	
fileName	crtv:fileName	

rdf:type ist auf einen der folgenden Werte gestzt

- `J2EEApplication`
- `MQQueue`

Fehlerbehebung für OSLC

In diesem Abschnitt werden häufig auftretende Probleme, die mit OSLC auftreten können, sowie Lösungen für diese Probleme beschrieben.

Die konfigurierte TADDM-URL enthält keine Portnummer

Problem

Die in der Datei `collation.properties` konfigurierte Eigenschaft `taddmURL` für die TADDM-URL muss eine Portnummer enthalten.

Falls die Eigenschaft nicht mit einer Portnummer konfiguriert ist, müssen Sie die TADDM-URL entsprechend aktualisieren, sodass sie eine Portnummer umfasst, und die Informationen zu Registry Services oder bestimmten Providern sowie die TADDM-Zeitmarken löschen.

Lösung

Führen Sie die folgenden Schritte aus, um die TADDM-URL mit einer integrierten Portnummer zu aktualisieren:

1. Legen Sie in der Datei `collation.properties` die Eigenschaft `taddmURL` folgendermaßen fest:

   ```
   taddmURL=http://Server.Domäne:Port
   ```

2. Führen Sie auf dem Computer mit Registry Services die folgenden Schritte aus:
   a. Wechseln Sie zu `/opt/IBM/JazzSM/registry/etc`.
   b. Konfigurieren Sie in der Datei `CLI.properties` Berechtigungsnachweise für folgende Eigenschaften:
      - ds.jdbc.user
      - ds.jdbc.password
      - appserver.user
      - appserver.password
   c. Wechseln Sie zu `/opt/IBM/WebSphere/AppServer/bin`.
   d. Führen Sie das Script `stopServer.sh` aus, um den WebSphere Application Server zu stoppen.

      ```
      ./stopServer.sh Servername -user Benutzername -p Kennwort
      ```

      Beispiel:

      ```
      ./stopServer.sh server1 -user wasadmin -p passw0rd
      ```

   e. Wechseln Sie zu `/opt/IBM/JazzSM/registry/bin`.
   f. Führen Sie das Script `frs.sh` mit den entsprechenden Parametern aus:

      ```
      ./frs.sh uninstall -type db -properties ../etc/CLI.properties
      ```

   g. Überprüfen Sie, ob die Datenbank freigegeben wurde. Führen Sie andernfalls die folgenden Befehle aus:

      ```
      db2 drop db Datenbankname
      ```

      ```
      db2 create db Datenbankname
      ```

      Dabei ist *Datenbankname* der Name der Registry Services-Datenbank.
   h. Wechseln Sie zu `/opt/IBM/JazzSM/registry/bin`.
   i. Führen Sie das Script `frs.sh` mit den entsprechenden Parametern aus:

      ```
      ./frs.sh install -type db -properties ../etc/CLI.properties
      ```

   j. Wechseln Sie zu `/opt/IBM/WebSphere/AppServer/bin`.

k. Führen Sie das Script `startServer.sh` aus, um den WebSphere Application Server zu starten.

```
./startServer.sh Servername -user Benutzername -p Kennwort
```

Beispiel:

```
./startServer.sh server1 -user wasadmin -p passw0rd
```

l. Führen Sie das Script `frs.sh` mit den entsprechenden Parametern aus:

```
./frs.sh uninstall -type container -properties ../etc/CLI.properties
```

m. Führen Sie das Script `frs.sh` mit den entsprechenden Parametern aus:

```
./frs.sh install -type container -properties ../etc/CLI.properties
```

Mit dem folgenden Befehl können Sie möglicherweise ein Element aus Registry Services für einen bestimmten Provider entfernen:

```
./frs.sh deleteProvider -providerUrl URL - properties cli.properties
```

3. Führen Sie auf dem Computer mit der TADDM-Datenbank folgende Schritte aus:
   a. Wechseln Sie zu `$COLLATION_HOME/support/bin`.
   b. Führen Sie das Script oder die Stapeldatei `runtopobuild` mit den entsprechenden Parametern aus. Beispiel:

```
./runtopobuild.sh -a OSLCAgent -R
```

Tivoli Directory Integrator

Mit dem Erwerb von IBM Tivoli Application Dependency Discovery Manager (TADDM) erhalten Sie auch Tivoli Directory Integrator, mit dessen Hilfe Sie TADDM mit anderen Datenquellen integrieren können.

Tivoli Directory Integrator-Dokumentation im Knowledge Center
http://www-01.ibm.com/support/knowledgecenter/SSCQGF_7.1.0/KC_ditamaps/welcome.html?lang=en

TADDM-Integrationsszenarios in der Tivoli Application Dependency Discovery Manager-Wiki
https://github.com/TADDM/taddm-wiki/wiki/Integration-Scenarios

Kompatibilität von Geschäftsentitäten mit früheren Versionen

Für die Integration von TADDM mit Produkten, die die Daten aus TADDM über DataApi oder über SQL direkt aus der TADDM-Datenbank lesen, wurde eine neue Funktion eingeführt. Beispiele für diese Produkte sind IBM Tivoli Business Service Manager (TBSM), IBM SmartCloud Control Desk (SCCD) und Tivoli Directory Integrator (TDI). Das aktuelle Geschäftsanwendungsdatenmodell setzt auf der CustomCollection-Schnittstelle auf, die nichts mit den bisherigen Anwendungs- und ITSystem-Schnittstellen gemeinsam hat. Die neue Funktion ermöglicht die Integration mit anderen Produkten ohne Änderung an diesen Systemen.

In künftigen Versionen von TBSM und SCCD wird das Geschäftsanwendungsmodell mit neuen Funktionen eingeführt. Ziel ist die Generierung früherer Geschäftsentitäten (dies sind Kopien benutzerdefinierter Objektgruppeninstanzen).

Die neue Funktion, die die Kompatibilität mit früheren Versionen sicherstellt, umfasst folgende Neuerungen.

Zusätzlicher Schritt bei der Ausführung von `BizAppsAgent`
In diesem zusätzlichen Schritt werden für jede vom Agenten generierte benutzerdefinierte Objektgruppe Geschäftsentitäten (Services, Anwendungen, Objektgruppen) generiert.

Zur Aktivierung dieses Schritts wurde der Datei `collation.properties` die neue Eigenschaft `com.ibm.cdb.serviceinfrastructure.earlier.ver.compatibility` hinzugefügt. Der Standardwert dieser Eigenschaft ist *TRUE* bei einem Upgrade und *FALSE* bei einer Neuinstallation.

OSLC-Unterstützung
Der OSLC-Agent wurde geändert. Er kann nun entweder die alten Geschäftsentitäten oder die neuen benutzerdefinierten Objektgruppen registrieren. Wenn das Kompatibilitätsflag auf *TRUE* gesetzt ist, werden die alten Geschäftsentitäten registriert. Andernfalls werden benutzerdefinierte Objektgruppen als Inhalt für Jazz for Service Management (JazzSM) verwendet.

In Zukunft müssen die Geschäftsentitäten komplett neu geladen werden, wenn bei der Integration eines Produkts mit dem Laden von Daten begonnen wird, die neue Modellobjekte verwenden (benutzerdefinierte Objektgruppen und Knoten). Alte Geschäftsanwendungen (Anwendungen) und neue Geschäftsanwendungen (benutzerdefinierte Objektgruppen) dürfen nicht die gleichen GUIDs haben. Zur Vermeidung von Duplikaten müssen die Benutzer die alten Geschäftsanwendungen entfernen, bevor sie die neuen benutzerdefinierten Objektgruppen laden.

Funktionsgruppen erstellen

Die neuen Geschäftsanwendungen weisen im Gegensatz zu den alten Geschäftsanwendungen keine Funktionsgruppen auf. Einem ähnlichem Zweck dient jedoch die neue Schichtenfunktionalität. Zur Gewährleistung der Kompatibilität mit früheren Versionen wird für jede eindeutige Schicht eine Funktionsgruppe mit einem entsprechenden Namen erstellt.

Weitere Informationen finden Sie im *Benutzerhandbuch* zu TADDM im Abschnitt *Geschäftsanwendungsschichten*.

Integration von BigFix

Zweck
TADDM verwendet Anker und Gateways, um Maschinen/Anwendungen/Netze zu erkennen, die sich hinter der Firewall befinden. Die Verwendung von Ankern/Gateways kann mithilfe von IBM Netcool Monitoring Tools (ITM) derzeit verhindert werden. Alternativ dazu kann die TADDM-Integration mit der BigFix-Architektur verwendet werden, um die Verwendung von Ankern und Gateways zu vermeiden. Die BigFix-Architektur besteht aus einem BigFix-Server (BES-Server) und mehreren BigFix-Endpunkten (BES-Clients), wobei es sich bei den BES-Clients um die sicheren Systeme handelt, auf die über den BES-Server zugegriffen werden kann. Die BigFix-Infrastruktur kann für die Ausführung der TADDM-Scriptpakete auf den BES-Clients über BES-Server automatisch wiederverwendet/verwendet werden.

Die wichtigsten Vorteile dieser Integration für TADDM-Administratoren sind:

1. Zonen, die durch Firewalls geschützt sind, können ohne Anker erkannt werden.
2. Die BigFix-Architektur (z. B. offene sichere Ports) kann für den Zugriff auf Endpunkte wiederverwendet werden, wodurch die Dauer der Einrichtung für die Erkennung der gleichen Ziele mithilfe der TADDM-Standardmethode verringert wird.
3. Scriptbasierte TADDM-Sensoren können an der strategischen Ausrichtung ausgerichtet werden.
4. Es ist ein minimaler Eingriff vom TADDM-Administrator erforderlich.
5. Es wird eine alternative Methode zur Erkennung von Maschinen in Zonen, die durch Firewalls geschützt sind, ohne die Verwendung der TADDM-ITM-Integration bereitgestellt.

Referenz
Dokumentation für TADDM

Folgende Tabelle zeigt die unterstützten Versionen der Produkte, mit denen TADDM integriert werden kann.

Weitere Informationen zu Produkten, die Sie mit TADDM integrieren können, finden Sie in der zugehörigen Dokumentation:

- Knowledge Center zu TADDM 7.3 und Sensoren (offizielle Dokumentation) http://www-01.ibm.com/support/knowledgecenter/SSPLFC_7.3.0/com.ibm.taddm.doc_7.3/welcome_page/kc_welcome-444.html?lang=en
- Sensoren für TADDM 7.3 und unterstützte Zielsysteme https://github.com/TADDM/taddm-wiki/wiki/Sensors-and-Supported-Target-Systems
- TADDM-Konfiguration von asynchroner Script-Erkennung (ASD) https://www.ibm.com/support/knowledgecenter/SSPLFC_7.3.0/com.ibm.taddm.doc_7.3/SensorGuideRef/r_cmdb_async_script_sensors.html#sensorsthatcanbescripted
- IBM BigFix-Konfigurationshandbuch https://www.ibm.com/support/knowledgecenter/SSPLFC_7.3.0/com.ibm.taddm.doc_7.3/SensorGuideRef/r_cmdb_async_script_sensors.html#sensorsthatcanbescripted
- Support-Website zu TADDM http://www.ibm.com/software/sysmgmt/products/support/IBMTivoliApplicationDependencyDiscoveryManager.html
- TADDM-Wiki https://github.com/TADDM/taddm-wiki/wiki Hier finden Sie aktuelle Informationen und bewährte Verfahren für TADDM. Setzen Sie ein Lesezeichen für diese Seite und machen Sie sich mit den Funktionen vertraut.
- TADDM-Forum http://www.ibm.com/developerworks/forums/forum.jspa?forumID=1547&categoryID=15&ca=drs-fo
- Request for Enhancement Community http://www.ibm.com/developerworks/rfe/?BRAND_ID=90 In dieser Community können Erweiterungen zum Produkt direkt von IBM Entwicklern angefordert werden.

Lösungsarchitektur
Die TADDM-BigFix-Integration basiert auf der Verbesserung und Automatisierung des aktuellen Verhaltens der asynchronen Scripterkennung (Asynchronous Script Discovery, ASD), für das ein manueller Eingriff vom TADDM-Administrator erforderlich ist. In dieser Integration wird die Konnektivität genutzt, die durch die BigFix-Infrastruktur für die mit einer Firewall geschützten Maschinen bereitgestellt wird, damit die Erkennung über die TADDM-Scriptpakete ausgeführt werden kann.

In ASD muss der TADDM-Administrator die folgenden Schritte manuell ausführen:

1. Ausführung eines Scripts auf dem TADDM-Server, um ein Erkennungspaket mit allen Sensoren zu erstellen, die auf dem Ziel ausgeführt werden sollen.
2. Übertragung dieses Pakets an das Zielsystem.
3. Ausführung des Erkennungspakets auf dem Zielsystem.
4. Übertragung der Ergebnisdatei, die auf dem Zielsystem generiert wurde, zurück auf dem TADDM-Server.

Mit der aktuellen Lösung wurden die manuellen Schritte automatisiert und die Lösung wird daher auch als 'Automated Asynchronous Script Discovery' (AASD, automatische asynchrone Scripterkennung) bezeichnet. Der TADDM-Administrator muss nur ein Script ausführen, um die Erkennung auf dem TADDM-Server zu starten - die übrigen Schritte werden automatisch ausgeführt.

Schritte in der BigFix-Erkennung
Einzelheiten zu den Schritten der BigFix-Erkennung

Schritt 1: Script für die BigFix-Integration

Das Script “runBigFixDiscovery.sh” wurde entwickelt, um die automatische ASD-Erkennung (AASD) aus dem TADDM-Erkennungsserver zu starten. Das Script kann auf Anforderung ausgeführt werden. Dieses Script verwendet den Namen des Erkennungsbereichs und des Erkennungsprofils als Eingaben (neben BigFix-Zugriffsberechtigungsnachweisen) und unterstützt die folgenden Modi:

- Modus DISCOVER (Erkennung) – zur Einleitung der BigFix-Erkennung
- Modus POLL (Abfrage) – zur Abfrage der Ergebnisse der BigFix-Erkennung
- Modus CLEANUP (Bereinigung) – zur bedarfsgesteuerten Bereinigung der Pakete mit den Ergebnissen der Erkennung vom BES-Root-Server

- Modus REDISCOVER (erneutes Erkennen) – zur erneuten Ausführung der vorherigen Erkennung

a) Sensorpaket für automatisches ASD erstellen

- Mit dem angegebenen Erkennungsprofil wird die Liste der Sensoren abgerufen und nur eine gültige Untergruppe von scriptgesteuerten Sensoren wird für die Erstellung von AASD-Scriptpaketen berücksichtigt. Diese Funktion unterstützt nur eine Untergruppe der Scriptsensoren im ASD-Standardmodus.
- Andere, nicht scriptgesteuerte Sensoren im Erkennungsprofil werden ignoriert.
- Das AASD-Paket ist unabhängig vom Betriebssystem - daher schlagen einige Sensoren möglicherweise auf BigFix-Endpunkten fehl, wenn es nicht vorhanden ist.
- Das generierte AASD-Scriptpaket wird mit der REST-API '/api/upload' auf den BigFix-Root-Server hochgeladen.

b) Erstellen Sie eine BigFix-Task

- Mit dem angegebenen Erkennungsbereich wird die “Relevance”-XML erstellt, die von BigFix erkannt wird.
- Die BigFix-Task 'XML' wird mit “Relevance” und dem Dummy “ActionScript” generiert.
- Generieren Sie den Titel der Task auf Basis des aktuellen Datums und der aktuellen Uhrzeit.
- Erstellen Sie mithilfe der REST-API '/api/tasks/custom/TADDM' eine BigFix-Task auf der angepassten Site mit der Bezeichnung “TADDM” auf dem BigFix-Server.

c) Starten Sie die BigFix-Task

- Mit <SourcedFixletAction> starten Sie die Aktionsausführung auf die zuvor erstellte BigFix-Task
- Mit der BigFix-REST-API '/api/actions' wird die Ausführung von “ActionScript” auf dem Zielendpunkt gestartet.

Schritt 2: Script ausführen

- Im Rahmen der Ausführung von “ActionScript” wird das TADDM-AASD-Paket entpackt und darin enthaltene Sensorscripts (auf Basis des Erkennungsprofils) werden auf den BigFix-Endpunkten ausgeführt.

Schritt 3: ZIP-Datei erfassen

- Am Ende der Ausführung von “ActionScript” wird das Ereignispaket, das bei der Ausführung des TADDM-AASD-Pakets auf dem BES-Client generiert wurde, in den BES-Root-Server kopiert.

Schritt 4: Ergebnisse zurück in TADDM importieren

- TADDM fragt die Datenbank des BigFix-Servers kontinuierlich ab, um zu überprüfen, ob die Ergebnisdatei auf den BES-Server hochgeladen wurde.
- Wenn in der Datenbank angezeigt wird, dass neue Ergebnisdateien vorhanden sind, verwendet TADDM die HTTP-Anforderung für die Abfrage der verschlüsselten Ergebnisdateien sowie zum Entschlüssen und Speichern der Dateien.
- TADDM verarbeitet anschließend diese Ergebnisdateien auf Basis des konfigurierten Bereichs und Profils und speichert die erkannten Objekte in der Datenbank.

Intergration von TADDM Bigfix
Die BigFix-Lösung für die erweiterte TADDM-Erkennung konzentriert sich auf die durchgängige funktionale Ausführung der Erkennung für Windows, Linux, AIX und Solaris OS und der zugehörigen Sensoren (für mehrere Endpunkte) und auf die SSL-Kommunikation, die zwischen TADDM und BigFix über BigFix-REST-APIs unterstützt wird. Die Erkennung kann auf dem TADDM-Erkennungsserver eingeleitet werden und die Ergebnisse der Erkennung sollten automatisch abgerufen und in der grafischen Benutzerschnittstelle von TADDM angezeigt werden.

Voraussetzungen
Die folgenden Voraussetzungen werden beim Ausführen der Erkennung berücksichtigt:

1. Der BigFix-Server und die Clients haben die in Abschnitt 2.1 erwähnte Version.

2. BigFix-Clients verfügen über die entsprechenden Berechtigungen zur Ausführung der von dem BigFix-Server hochgeladenen Erkennungstask bzw. des Aktionsscripts.

3. Der in der Datei `collation.properties` konfigurierte Benutzer der BigFix-SQL-Datenbank muss über Lesezugriff für die Datenbank `BFEnterprise` verfügen.

4. Scriptpakete für Sensoren, die über BigFix-Agenten ausgeführt werden, benötigen Schreibzugriff auf das konfigurierte TEMP-Verzeichnis (z. B. “C:\Windows\Temp”). Das TEMP-Verzeichnis kann in der Datei `collation.properties` konfiguriert werden und es wird angenommen, dass der Verzeichnispfad keine Leerzeichen enthält.

5. Die Bereinigung des Anforderungspakets für das Script wird nicht auf dem TADDM-Erkennungsserver verarbeitet und es wird angenommen, dass es vom Administrator verwaltet wird.

6. Da die TADDM-BigFix-Integration auf dem aktuell vorhandenen ASD-Framework in TADDM basiert, basieren die Leistungsmerkmale dieser Integration auf den Benchmarks des ASD-Frameworks.

7. Nur "`taddmusr` kann für die Ausführung des BigFix-Erkennungsscripts auf dem TADDM-Erkennungsserver verwendet werden, der Rootbenutzer ist nicht zulässig.

8. Die Bereinigung des BigFix-Root-Servers wird bei jedem TADDM-Start und außerdem regelmäßig gemäß der konfigurierten Dauer (com.collation.bigfix.root.cleanup.interval = Standardwert 1 Tag) aufgerufen. Dadurch werden Ergebnisdateien gelöscht, die älter als die konfigurierte Zeit sind (`com.collation.bigfix.root.cleanup.days = Standard 5 Tage`).

9. Bereinigung aller Ergebnisdateien auf dem TADDM-Server, die während der Erkennung auf dem TADDM-Server erstellt/kopiert wurden und deren Name "`taddmasd`" enthält und mit "`DONE`" endet. (Für die Aktivierung der Bereinigung auf dem TADDM-Server sollte mindestens ein Schwellenwert konfiguriert sein, der in *Konfigurierbare Eigenschaften* > *Bereinigung* angegeben ist).

10. Die Bereinigung des Erkennungsendpunkts ist standardmäßig aktiviert und kann durch die Konfiguration der folgenden Eigenschafteneinstellungen gesteuert werden:

- Durch das Setzen von "`com.collation.bigfix.endpoint.cleanup`" auf “N” wird die Bereinigung auf dem Erkennungsendpunkt inaktiviert

11. Das Szenario für die Hochverfügbarkeit oder die Wiederherstellung nach einem Katastrophenfall wird nicht unterstützt. Ein Erkennungsserver stellt nur eine Verbindung mit dem einzelnen BigFix-Server her, der in der zugehörigen Eigenschaft `collation.properties` für den Start dieses erweiterten Erkennungsprozesses konfiguriert ist.

Voraussetzungen

Bevor Sie die Erkennung vom TADDM-Server aus starten, müssen folgende Voraussetzungen erfüllt sein:

1. Wählen Sie bei der Erstellung des Erkennungsprofils die Eigenschaften 'ASDSensor', 'ASDPingSensor', 'Generic Server Sensor' verbindlich aus und heben Sie die Auswahl der Eigenschaften 'PingSensor', 'PortSensor' und 'SessionSensor' verbindlich auf.

2. Alle im Abschnitt "*Konfigurierbare Eigenschaften, die in der Integration verwendet werden*" angegebenen Konfigurationsschritte wurden abgeschlossen.

3. Das BigFix-Aktionsscript hat den nativen PowerShell-Befehl auf dem Windows-Endpunkt zum Dekomprimieren des Anforderungspakets und den nativen TAR-Befehl in Linux verwendet.

Anmerkung: Auf Grundlage der spezifischen Anforderung kann die Anpassung des Aktionsscript ausgeführt werden. Diese Anpassung wird durch die Aktualisierung der vom Kunden änderbaren Datei 'ActionScript_Pre_Post.txt' unterstützt, die sich im Ordner `$COLLATION_HOME/etc/` befindet. Damit kann beispielsweise angepasste Software zum Entpacken heruntergeladen und angewendet werden (ausführbare Verteilung auf dem BigFix-Root-Server). Ein Beispielsnippet wird im Anschluss gezeigt:

```
%WIN_PRE_START%
if {not exists file "C:\Windows\System32\unzip.exe"}
prefetch unzip.exe sha1:e1652b058195db3f5f754b7ab430652ae04a50b8
```

```
size:167936 http://10.160.161.199:52311/Uploads/Unzip/unzip.exe

// Make sure that environment is set appropriately and “unzip”
utility is available in the windows PATH
copy "__Download\unzip.exe" "C:\Windows\System32\unzip.exe"

endif
%WIN_PRE_END%

%WIN_POST_START%
%WIN_POST_END%

%LIN_PRE_START%
%LIN_PRE_END%
…
```

4. Der Benutzer, der die Erkennung ausführt, benötigt Schreib-/Leseberechtigungen für den Ergebnisordner.

5. Es sollte ausreichender Plattenspeicherplatz, eine ausreichende Verarbeitungskapazität und genügend Hauptspeicher für die Anforderungs- und Ergebnispakete verfügbar sein, die auf dem TADDM-Server, dem BigFix-Root-Server und auf Erkennungszielen verarbeitet werden.

6. Der BigFix-Agent (Endpunktcomputer) sollte mit ausreichendem Wert für die Einstellung “_BESClient_ArchiveManager_MaxArchiveSize” konfiguriert sein, damit die Ergebnisse erfolgreich auf den BigFix-Root-Server hochgeladen werden können.

7. Es muss ein ordnungsgemäßer Bereich und ein korrektes Profil festgelegt sein (Informationen zum Festlegen von Bereich und Profil finden Sie im Abschnitt 4.1).

8. Der Sitename "TADDM" muss konfiguriert und auf dem BigFix-Server vorhanden sein.

9. Alle Voraussetzungen, die für standardmäßige ASD-Sensorscripts erforderlich sind, gelten auch für die BigFix-Erkennung.

- Die ausführbare Powershell-Datei sollte ordnungsgemäß für die Erkennung des Windows 2003-Endpunkts installiert und konfiguriert sein.

10. Für die erneute Erkennung oder die wiederholte Erkennung mit Intervall sollten der Bereich und das Profil, die für den Start der Big Fix-Erkennung verwendet werden, zwischenzeitlich nicht gelöscht worden sein. Im Falle eines Löschens muss ein Benutzer möglicherweise nicht verarbeitete Dateien löschen, die auf dem TADDM-Server vom Big Fix-Server empfangen werden.

Beschränkungen

Dem aktuellen Release sind die folgenden Einschränkungen zugeordnet:

1. Erkennungsziele, die während der Erkennung angegeben wurden und vom BigFix-Root-Server nicht erreichbar sind, werden im Erkennungsprotokoll nicht angezeigt.

2. 'PingSensor', 'PortSensor' und 'SessionSensor' werden automatisch aktiviert, wenn andere Sensoren ausgewählt und aktiviert werden, und diese müssen während der Erstellung des Erkennungsprofils manuell inaktiviert werden.

3. Die Bereinigung des Anforderungspakets auf dem TADDM-Erkennungsserver oder BigFix-Root-Server wird vom Entwurf nicht unterstützt. Bei der erneuten Erkennung (ausgelöst durch den Modus 'Rediscover') wird sie möglicherweise berücksichtigt.

4. Die erneute Erkennung wird nur von demselben TADDM-Erkennungsserver unterstützt, auf dem die ursprüngliche Erkennung eingeleitet wurde.

Konfiguration

Befolgen Sie die in diesem Abschnitt beschriebenen Schritte, um die gewünschte Konfiguration festzulegen.

Basiskonfigurationen für die BigFix-Erkennung

1. Legen Sie die folgenden verbindlichen Eigenschaften in der Datei `$COLLATION_HOME/etc/collation.properties` fest:

Einstellungen für die BigFix-Integrationsfunktion

```
com.collation.bigfix.enabled=true
```

BigFix-Servereinstellungen

- com.collation.bigfix.host=<IP oder vollständig qualifizierter Domänenname des BigFix-Servers>
- com.collation.bigfix.port=<Portnummer>
- com.collation.bigfix.uid=<Benutzer-ID für Zugriff auf BigFix-Serverkonsole>
- com.collation.bigfix.pwd=<Kennwort für Zugriff auf BigFix-Serverkonsole>

BigFix-Datenbankeinstellungen

- com.collation.bigfix.db.type=<MSSQL/DB2>
- com.collation.bigfix.db.host=<IP oder vollständig qualifizierter Domänenname der BigFix-Serverdatenbank>
- com.collation.bigfix.db.port=<Port, an dem TADDM eine Verbindung mit BigFix-Datenbank herstellt>
- com.collation.bigfix.db.dbname=<BigFix-Datenbankname>
- com.collation.bigfix.db.domain=<Benutzerdomäne> Optionaler Parameter - nur erforderlich, wenn Windows-basierte Authentifizierung für BigFix-Datenbank konfiguriert is>
- com.collation.bigfix.db.uid=<Benutzer-ID für den Zugriff auf die BigFix-Datenbank>
- com.collation.bigfix.db.pwd=<Kennwort für Zugriff auf BigFix-Datenbank>

Einstellungen für den Thread zur Verarbeitung von Ergebnissen

- com.ibm.cdb.discover.asd.ProcessUnreachableIPs=true
- com.ibm.cdb.discover.asd.autodiscovery.enabled=true

2. Legen Sie die folgenden Eigenschaften in der Datei $COLLATION_HOME/etc/collation.properties nur fest, wenn die SSL-Konfiguration auf dem BigFix-Server aktiviert ist:

Bigfix-Zertifikat

- com.collation.bigfix.certificate.type=<PKCS12/JKS>
- com.collation.bigfix.certificate.file=<Vollständiger Pfad zur Zertifikatsdatei>
- com.collation.bigfix.certificate.pwd=<Kennwort für die Verwendung des Zertifikats>

Anmerkung: ^Neben diesen verbindlichen Eigenschaften können noch weitere Eigenschaften konfiguriert werden. Eine vollständige Liste der Eigenschaften und der zugehörigen Einzelheiten finden Sie unter '*kKnfigurierbare Eigenschaften*'.

Anmerkung: Fix Pack 6 Ein mit einem leeren Kennwort generiertes Zertifikat wird unterstützt.

3. Führen Sie das Script “encryptprops.sh” aus, um die Eigenschaften zu verschlüsseln (unter '*Beispiele für die Scriptausführung*' finden Sie Informationen, um das Format zur Ausführung dieses Scripts zu überprüfen). Andernfalls schlagen Erkennungsscripts (runBigFixDiscovery.sh/.bat) mit einem Fehler aufgrund fehlender oder ungültiger Argumente fehl (siehe Einzelheiten zu Fehlercode '*Fehlercodes und Beschreibung*'), da nur verschlüsselte Kennwörter akzeptiert werden.

4. Erstellen Sie den Ordner $COLLATION_HOME/var/asdd, um Ergebnisdateien in TADDM zu speichern. Wenn der Ordner 'var' oder 'asdd' nicht verwendet werden soll, muss die Eigenschaft “com.ibm.cdb.discover.asd.AsyncDiscoveryResultsDirectory” mit dem bestimmten Ordner festgelegt werden, in den der Administrator die Ergebnisdateien herunterlädt.

5. Starten Sie TADDM erneut.

6. Erstellen Sie ein Erkennungsprofil mit den erforderlichen Sensoren, die in Abschnitt 2.1 aufgeführt sind. Das Profil sollte neben den erkannten Sensoren auch verbindliche Sensoren enthalten.

7. Erstellen Sie einen Erkennungsbereich mit BigFix-Zielendpunkt(en), auf dem/denen die Erkennung ausgeführt werden muss.

Sonstige Konfiguration
Auf dem BigFix-Server muss eine Site mit der Bezeichnung “TADDM” erstellt werden.

1. Öffnen Sie die “BigFix-Konsole” -> wechseln Sie zur Registerkarte “Tools” -> wählen Sie “Create Custom Site.”(Angepasste Site erstellen) aus -> geben Sie den Sitenamen “TADDM” an.

2. Klicken Sie auf “TADDM” -> wählen Sie die Registerkarte “Computer Subscription” (Computer-Subskription) aus -> subskribieren Sie die Computer auf Basis der Anforderung (es sollten alle Computer eingeschlossen sein, mit denen der BigFix-Server über TADDM verbunden sein soll).

Unterstützte Anpassungen
In der TADDM BigFix-Integration unterstützte Anpassungen.

Anpassung von Aktionsscripts

- Die Anpassung von Aktionsscripts wird für die Erkennung unterstützt
- Der Benutzer kann den Abschnitt 'Pre/Post' für jedes Betriebssystem in der Datei “`Action-Script_Pre_Post.txt`” im Ordner `$COLLATION_HOME/etc/` ändern, wenn einige bestimmte Aktionen vor bzw. nach der Erkennung ausgeführt werden sollen
  - Im folgenden Snippet wird beispielsweise die Implementierung von angepasster Dekomprimierungssoftware (ausführbare Linux-Distribution im BigFix-Root-Server) aktiviert, anstelle der Vorinstallation auf jedem Endpunkt

```
%WIN_PRE_START%

if{not exists file "C:\Windows\System32\unzip.exe"}
prefetchunzip.exe sha1:e1652b058195db3f5f754b7ab430652ae04a50b8
size:167936
http://10.160.161.199.52311/Uploads/Unzip/unzip.exe

//Make sue thet environment is set appropriatly and "unzip" utility is
available in the windows PATH
copy"_Download\unzip.exe""C:\Windows\System32\unzip.exe"

endif
%WIN_PRE_END%

%WIN_POST_START%
%WIN_POST_END%

%LIN_PRE_START%
%LIN_PRE_END%
...
```

Anpassung des Betriebssystems

Es werden standardmäßige Aktionsscripts für alle vier unterstützten Plattformen (**Linux**, **Windows**, **AIX** und **SunOS**) erstellt.

Wenn ein Benutzer die Logik von Aktionsscripts auf eine bestimmte Gruppe von Plattformen beschränken möchte, kann die folgende Sortiereigenschaft verwendet werden:

- com.collation.bigfix.action.enable.os

Beispiel: `com.collation.bigfix.action.enable.os=WIN,LIN,SUN,AIX`

Anpassung der Relevanz

- Der Bereich der TADDM-Erkennung basiert auf **IP-Adressen/Bereichen/Netzen/Masken**. Mit dieser Anpassung kann ein Benutzer die Erkennungsbereiche erweitern, die nicht notwendigerweise IP-basiert sind, sondern dynamisch auf einer bestimmten Relevanz/einem bestimmten Kriterium basieren können.
- Wenn ein Benutzer beispielsweise die Erkennung auf allen Windows-Computern ausführen und den angegebenen TADDM“**-Erkennungsbereich**” überspringen will, wird die folgende Sortiereigenschaft verwendet:
  - `com.collation.bigfix.relevance.appendscope=`**`false`**
  - `com.collation.bigfix.relevance=`**`Windows-Betriebssystem`**
- `com.collation.bigfix.relevance.appendscope`
  - Bei 'true' wird zusätzlich zum angegebenen Erkennungsbereich die **angepasste Relevanz** abgefragt
  - Bei 'false' wird anstelle des angegebenen Erkennungsbereichs nur die Abfrage der **angepassten Relevanz** verwendet
- `com.collation.bigfix.relevance`
  - Abfrage der **gültigen Relevanz** zur Ermittlung einer Gruppe von Endpunkten für eine bestimmte Erkennung

Anmerkung:

- Die angepasste Relevanz wird auf der Ebene des TADDM-Erkennungsservers unterstützt
- Die Validierung der angepassten Relevanz wird nicht unterstützt und wird transparent verwendet

Tipp für die Protokolldatei und zur Fehlerbehebung
Wenn während der Erkennung Fehler auftreten, können die folgenden Punkte überprüft werden. Bestätigen Sie, dass alle Voraussetzungen erfüllt sind:

TADDM-Erkennungsserver

- So überprüfen Sie Protokolle für die Ausführung des BigFix-Erkennungsscripts:
  - `$COLLATION_HOME/log/BigFixDiscovery.log`
- So überprüfen Sie, ob die Protokolle für die Ergebnisdatei auf dem TADDM-Server empfangen wurden:
  - `$COLLATION_HOME/log/services/ApiServer.log` (Suche nach den Schlüsselwörtern 'BigfixDiscoveryServerController' und 'AASDiscoveryServerController')

BigFix-Root-Server

So überprüfen Sie den Status der Erkennung und der Aktionsausführung mithilfe der `IBM BigFix-Konsole` auf dem **BigFix-Root-Server**

1. Öffnen Sie die **IBM BigFix-Konsole** .
2. Wählen Sie **Site** (Custom->TADDM) -> **Fixlets** und **Tasks** (Site > Angepasst > TADDM > Fixlets > Tasks) aus.
3. Wählen Sie **Task** aus (wird während der Scriptausführung angegeben).
4. Überprüfen Sie die Einträge in **Details** und **Action History**)Aktionsprotokoll).
5. Wählen Sie **Particular Action History** (Bestimmtes Aktionsprotokoll) -> **Reported Computers** (Gemeldete Computer) aus.
6. Überprüfen Sie den Status und **doppelklicken** Sie auf ihn für die zeilenweise Ausführung .
7. Klicken Sie auf **OK**, um zum Fenster 'Erkennungsprofile' zurückzukehren.

BigFix-Agent/Erkennungsziel

- Stellen Sie sicher, dass die Ergebnisdatei im Ordner %wintemp%"/taddm7.3.0.4/asd vorhanden ist (nur, wenn die Eigenschaft 'com.collation.bigfix.endpoint.cleanup' auf "N" gesetzt ist).
- Auf die Datei allErrors.txt (im Ordner %wintemp%"/taddm7.3.0.4/asd) kann verwiesen werden, wenn während der Scriptausführung für Sensoren Fehler auftreten.

BigFix-Erkennung ausführen

BigFix-Erkennung ausführen

Bereich erstellen
Öffnen Sie die grafische Benutzerschnittstelle des TADDM-Servers, um einen Erkennungsbereich zu erstellen. Der Bereich sollte alle BigFix-Zielendpunkte umfassen. Die Zielendpunkte können als einzelne Hosts oder als Domäne/Netzbereich angegeben werden.

Profil erstellen
Über die grafische Benutzerschnittstelle des TADDM-Servers sollte ein Erkennungsprofil erstellt werden. Das Profil sollte die Sensoren für die Anwendungen enthalten, die gemäß dem Administrator von TADDM erkannt werden sollen. In Abschnitt 2.1 finden Sie Einzelheiten zu Sensoren, die verbindlich in das erstellte Erkennungsprofil integriert bzw. verbindlich aus diesem ausgeschlossen werden müssen.

Script ausführen
Zur Ausführung der Erkennung muss das Script "runBigFixDiscovery.sh" aus dem Verzeichnis '$COLLATION_HOME/bin' ausgeführt werden. Das Script kann in vier Modi ausgeführt werden: 'Discovery' (Erkennung), 'Poll' (Abfrage), 'Cleanup' (Bereinigung) und 'Rediscovery' (Erneute Erkennung). Im Erkennungsmodus wird die Erkennung gestartet. Im Abfragemodus wird der aktuelle Erkennungsstatus abgerufen. Im Bereinigungsmodus wird die Bereinigung der Ergebnisdateien auf dem BigFix-Root-Server ausgelöst, und im Modus für die erneute Erkennung kann die zuvor ausgeführte Erkennung noch einmal ausgeführt werden.

1. Erkennungsmodus -

TADDM stellt Jazz for Service Management mit einem Feed-Service unter folgender Adresse bereit:

```
./runBigFixDiscovery.sh -d -o <Ausgabeverzeichnis> -s <Bereich> -p <Profil>
Dabei steht
-d – für den Erkennungsmodus
-o - für das Ausgabeverzeichnis, in dem Erkennungspakete erstellt werden
-s – für den Eingabebereich mit BigFix-Endpunktzielen, die erkannt werden sollen.
-p – für das Eingabeprofil mit Sensoren, die ausgeführt werden sollen
```

Sobald der Befehl im Erkennungsmodus ausgeführt wird, wird die Erkennung gestartet. Dadurch wird der Status des ausgeführten Schritts angezeigt und es wird die ID "Action" (Aktion) angegeben. Mit dieser Aktion kann im Abfragemodus der Status der Aktion auf jedem BigFix-Endpunkt überprüft werden.

Anmerkung:

- Die "ActionID" (Aktions-ID), die für die Erkennung erstellt wurde (wie in der Konsolenausgabe gezeigt, z. B. 2090 im folgenden Beispiel), kann für die Abfrage erneut verwendet werden.
- Behalten Sie den Namen der neu erstellten BigFix-Task (der in der Konsolenausgabe angezeigte "Task-Name", z. B. 20180130125432) bei, der dem angegebenen Bereich und Profil zugeordnet ist und für die erneute Erkennung wiederverwendet werden kann.

2. Abfragemodus -

```
./runBigFixDiscovery.sh -p -r <Wiederholung> -i <Aktions-ID>

Dabei steht
-p – für den Abfragemodus
-r – Anzahl der Abfragen auf dem BigFix-Server
-i - Aktions-ID aus dem Befehl im Erkennungsmodus.
```

3. Bereinigungsmodus -

```
./runBigFixDiscovery.sh -c -d <Anzahl der Tage>
Dabei steht
-c – für den Bereinigungsmodus
-d – Dateien, die älter als die angegebene Anzahl an Tagen sind und gelöscht werden sollen
```

4. Modus für die erneute Erkennung -

```
./runBigFixDiscovery.sh  -r/--rediscover -i <Taskname>
Dabei steht
-r – für den Modus für die erneute Erkennung
-i – TASKNAME, der der vorherigen Erkennung entspricht, die erneut ausgeführt werden muss
```

Anmerkung: Weitere Einzelheiten zu diesem Befehl mit allen möglichen Optionen finden Sie im Anhang B und ein Beispiel für die Ausführung des Befehls finden Sie im Anhang C.

Optionen für Scriptparameter in verschiedenen Modi
./runBigFixDiscovery.sh (or.bat) - TADDM-Tool für die Ausführung einer erweiterten BigFix-Erkennung oder für die Abfrage einer vorhandenen Erkennungsaktion.

Modus: DISCOVER

Syntax: bin/runBigFixDiscovery.sh -d/--discover [-c <Arg>] [-freq <Arg>] [-h] [-intr <Arg>] -o <Arg> -p <Arg> -s <Arg>

Dabei gilt:

Tabelle 51.

-ac,--actionConstraint <arg>	Datei wurde mit anderem Parameter für Aktionsvorgabe angegeben. Datei sollte folgenden Parameter mit Werten wie in diesem Beispiel enthalten:
	`com.collation.bigfix.action.constraint.starttime=T10M`
	`com.collation.bigfix.action.constraint.endtime=3DT5H20M`
	`com.collation.bigfix.action.constraint.timerange.starttime=01:15:00`
	`com.collation.bigfix.action.constraint.timerange.endtime=05:30:00`
	`com.collation.bigfix.action.constraint.days=sat,sun`
	Dies bedeutet, dass ein Script nur auf dem Ziel ausgeführt werden kann, wenn diese Bedingungen erfüllt sind, und zwar im Zeitraum von `10 Minuten` ab jetzt `(T10M) bis in 3 Tagen, 5 Stunden und 20 Minuten (3DT5H20M)`, und die Erkennung kann nur am Samstag und Sonntag zwischen `01:15:00 und 05:30:00(24-Stunden-Format)` ausgeführt werden. Diese Option ist sowohl bei der einmaligen Erkennung hilfreich, als auch bei der Erkennung, die mit der Intervalloption gestartet wurde.

-c,--compressMethod <Arg>	[Standardwert: ZIP] Gültige Werte: [ZIP, TAR].
-freq,--frequency <Arg>	[Standardwert: 1] Gibt an, wie oft die Erkennung ausgeführt werden muss.
-h,--help	Hilfe anzeigen.
-intr,--interval <Arg>	[Standardwert: P1D] Zeitintervall zwischen der erneuten Ausführung der Erkennung. Unterstützte Werte: [PT15M, PT30M, PT1H, PT2H, PT4H, PT6H, PT8H, PT12H, P1D, P2D, P3D, P5D, P7D, P15D, P30D].
-o,--output <Arg>	ERFORDERLICH: Ausgabeverzeichnis, in dem das BigFix-Erkennungspaket generiert wird.

Tabelle 51. (Forts.)	
-p,--profile <Arg>	ERFORDERLICH: Profilname, der für die Erstellung des Erkennungspakets verwendet wird, um Sensoren einzuschließen.
-s,--scope	ERFORDERLICH: Name(n) von Bereich/Bereichsgruppe (durch Kommas getrennt - Namen mit Leerzeichen müssen in Anführungszeichen gesetzt werden).

Modus: POLL

Syntax: bin/runBigFixDiscovery.sh -p/--poll [-h] -i <Arg> [-r <Arg>] [-t <Arg>]

Dabei gilt:

Tabelle 52.	
-d,--detail <Arg>	[Standardwert: true] Abfrageergebnis für jeden Endpunkt
-h,--help	Hilfe anzeigen.
-r,--repeat <Arg>	[Standardwert: 1] Gibt an, wie oft der Aktionsstatus abgefragt werden soll
-i,--id <Arg>	ERFORDERLICH: Aktions-ID für POLL
-t,--timeout <Arg>	[Standardwert: 1] Intervall zwischen aufeinanderfolgenden Abfragen in Sekunden.

Modus: CLEANUP

Syntax: bin/runBigFixDiscovery.sh -c/--cleanup [-d <Arg>] [-h]

Dabei gilt:

Tabelle 53.	
-h,--help	Hilfe anzeigen.
-d,--days <Arg>	[Standardwert: 5] Ergebnisdateien löschen, die älter als die angegebene Anzahl an Tagen sind

Modus: REDISCOVER

Syntax: bin/runBigFixDiscovery.sh -r/--rediscover [-freq <Arg>] [-h] [-intr <Arg>]

Dabei gilt:

Tabelle 54.	
-freq,--frequency <Arg>	[Standardwert: 1] Gibt an, wie oft die Erkennung ausgeführt werden muss
-h,--help	Hilfe anzeigen.
-intr,--interval <Arg>	[Standardwert: P1D] Zeitintervall zwischen der erneuten Ausführung der Erkennung. Unterstützte Werte: [PT15M, PT30M, PT1H, PT2H, PT4H, PT6H, PT8H, PT12H, P1D, P2D, P3D, P5D, P7D, P15D, P30D].

Verarbeitung von Ergebnissen der Erkennung
Der Abfragemodus des Befehls “runBigFixDiscovery.sh” gibt den Status der Aktion zurück, die auf jedem BigFix-Endpunkt ausgeführt wurde. Auf Grundlage des Status wird die Ergebnisdatei erstellt und verarbeitet.

1. Sobald die Aktion für den Endpunkt erfolgreich abgeschlossen wurde, wird eine Ergebnisdatei für diesen Endpunkt in den konfigurierten Ergebnisordner heruntergeladen (standardmäßig 'var/asdd'; Informationen zur Konfiguration des Ergebnisordners finden Sie unter '*Konfigurierbare Eigenschaften*').

2. Nachdem die Ergebnisdateien erfolgreich verarbeitet wurden, ist das Ergebnis in der grafischen Benutzerschnittstelle von TADDM auf der Registerkarte 'Protokoll' verfügbar>.

3. Die verarbeiteten Ergebnisdaten werden in der TADDM-Datenbank gespeichert und sind im Datenmanagementportal oder PSS von TADDM verfügbar.

Mögliches Fehlerszenario
Wenn im Erkennungs-, Abruf-, Bereinigungsmodus oder im Modus für das erneute Erkennen ein Fehler auftritt, können die folgenden Punkte überprüft werden:

1. Bestätigen Sie, dass alle in Abschnitt 2.2 angegebenen Voraussetzungen eingehalten werden:

2. Überprüfen Sie die TADDM-Protokolle im folgenden Pfad:

- `$COLLATION_HOME/log/BigFixDiscovery.log` – Protokolle zur Erkennung und Scriptausführung
- `$COLLATION_HOME/log/services/ApiServer.log` – Protokolle zum Abruf von Ergebnissen und zur Analyse

3. Überprüfen Sie die Protokolle des BigFix-Servers und der BigFix-Konsole auf einen Fehlerstatus.

4. Die Ausführungsprotokolle des **BigFix-Aktionsscripts** können überprüft werden, falls '**Action Polling**' (Abfrage von Aktion) einen Fehler vom '**Root-Server**' und '**Endpunkt**' empfängt.

5. Wenn die Erkennung über BigFix ausgeführt wird, können gelegentlich folgende Fehler oder ähnliche Fehler in den Protokollen angezeigt werden:

```
20191106144522 5#LinuxComputerSystemSensor-XX.XX.XXX.XXX DEBUG cdb.ScriptSensorUtils
- CTJTD0891E Error processing command: mem_size_demidecode due to error: sudo: sorry,
you must have a tty to run sudo
```

Dies kann vorkommen, wenn beide der folgenden Bedingungen auftreten:

1. sudo wird Befehlen hinzugefügt, die in 'collation.properties' konfiguriert so sind, dass sudo mit den auf fernen Systemen ausgeführte Befehlen verwendet werden soll. Beispiel: com.collation.discover.agent.command.dmidecode.Linux=sudo dmidecode
2. Außerdem ist “Defaults requiretty” auf dem Zielsystem in der Datei '/etc/sudoers' konfiguriert.

Bei der Erkennung über BigFix werden die Sensorscripts (mit den Befehlen mit sudo) von BigFix auf dem Ziel mit dem Rootbenutzer ausgeführt. Zur Problemlösung muss sudo vor der Ausführung der Befehle auf dem Ziel entfernt werden, wenn die Erkennung über BigFix ausgeführt wird. Dazu muss die Eigenschaft `com.ibm.cdb.aasd.RemoveSudoIfAny=true` in der Datei `collation.properties` konfiguriert werden.

Konfigurierbare Eigenschaften
Die folgenden konfigurierbaren Eigenschaften können in `collation.properties` konfiguriert werden.

1. BigFix-Funktion aktiviert

Tabelle 55.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation.bigfix.enabled	true/false	Bei 'true' wird die BigFix-Funktion aktiviert. Nachdem Sie diese Eigenschaft auf 'true' gesetzt haben, ist ein TADDM-Neustart erforderlich.	J

2. BigFix-Server

Tabelle 56.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation. bigfix.host	IP oder vollständig qualifizierter Domänenname	IP oder vollständig qualifizierter Domänenname des BigFix-Servers	J
com.collation. bigfix.port	<Portnummer> Standardwert=52311	Port, an dem TADDM die Anforderung an den BigFix-Server sendet.	N
com.collation. bigfix.uid	<Benutzer-ID>	Benutzer-ID für den Zugriff auf die BigFix-Serverkonsole.	J
com.collation. bigfix.pwd	<Kennwort>	Kennwort für den Zugriff auf die BigFix-Serverkonsole. Wird in verschlüsselter Form gespeichert.	J
com.collation. bigfix.connectTo	<Zeitraum> Standardwert=20 sec	TADDM wartet diesen Zeitraum in Sekunden, bevor das HTTP/RestAPI-Verbindungszeitlimit überschritten wird.	N
com.collation. bigfix.responseTo	<Zeitraum> Standardwert=20 sec	TADDM wartet diesen Zeitraum in Sekunden, bevor das HTTP-Antwortzeitlimit überschritten wird.	N
com.collation. bigfix.site.type	<Sitetyp> Standardwert=custom	Typ der Site auf dem BigFix-Server, zu der TADDM eine Verbindung herstellt.	N
Visibility.Control. Automation	<Sitename> Standardwert=TADDM	Name der Site auf dem BigFix-Server, zu der TADDM eine Verbindung herstellt.	N
com.collation. bigfix. aasdpkgmaxsize	<Größe des Anforderungspakets> Standardwert=1024	Maximal zulässige Größe des Anforderungspakets, das von Script für die BigFix-Erkennung generiert wird.	N

3. Big-Server-Zertifikat

Tabelle 57.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation. bigfix.certificate.type	<PKCS12/JKS> Standardwert=JKS	Typ des unterstützten Clientzertifikats.	N
com.collation. bigfix.certificate.file	<Pfad>	Position der Clientzertifikatsdatei.	N
com.collation. bigfix.certificate.pwd	<Kennwort>	Kennwort des Clientzertifikats	N

4. BigFix-Serverdatenbank

Tabelle 58.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation. bigfix.db.type	MSSQL oder DB2	Typ der vom BigFix-Server verwendeten Datenbank; MSSQL für den Windows-basierten BigFix-Server oder DB2 für Linux.	J
com.collation. bigfix.db.host	IP oder vollständig qualifizierter Domänenname	IP oder vollständig qualifizierter Domänenname der BigFix-Datenbank.	J
com.collation. bigfix.db.port	<Portnummer>	Port, an dem TADDM eine Verbindung mit BigFix-Datenbank herstellt>	J
com.collation. bigfix.db.dbname	<Datenbankname>	Name der BigFix-Datenbank.	J
com.collation. bigfix.db.domain	<Benutzerdomäne>	Domäne des Benutzers, die bei der Windows-basierten Authentifizierung verbindlich ist.	N
com.collation. bigfix.db.domain	<Benutzer-ID>	Benutzer-ID für den Zugriff auf die BigFix-Datenbank.	J
com.collation. bigfix.db.pwd	<Kennwort>	Kennwort für Zugriff auf die BigFix-Datenbank; wird in verschlüsselter Form gespeichert.	J

Anmerkung:

- Wenn die Verbindung zur TADDM-Datenbank unterbrochen wird, versucht TADDM durch die Einstellung “com.collation.bigfix.result.wait”, die Verbindung wiederherzustellen.
- Wenn in den oben genannten Einstellungen Änderungen vorgenommen werden, ist ein Neustart von TADDM erforderlich.

5. TADDM - Thread für Abruf oder Verarbeitung der Ergebnisse

Tabelle 59.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation. bigfix.result.wait	<Wert in Sek> Standardwert=60 Sek	Wenn “com.collation.bigfix.enabled” aktiviert ist, wird der Thread für den Abruf von Ergebnissen so generiert, dass die Dateien mit den Ergebnissen der Erkennung regelmäßig vom BigFix-Server abgerufen werden. Der “Results Fetching Thread” (Thread für den Abruf von Ergebnissen) ruft die Ergebnispakete vom BigFix-Server mit der konfigurierten Periodizität (definiert in Sekunden) ab.	N
com.ibm.cdb. discover.asd. autodiscovery.enabled	True/false	Bei 'true' wird der Thread für die Verarbeitung der gespeicherten ASD-Ergebnisdateien aktiviert.	J
com.ibm.cdb. discover.asd. ProcessUnreachableIPs	True/false	Der Thread verarbeitet das ASD-Ergebnis für Ziele, die nicht erreichbar sind.	J

Tabelle 59. (Forts.)

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.ibm.cdb.discover.asd.AsyncDiscoveryResultsDirectory	Standardpfad = var/asdd	Pfad, in dem Ergebnisdateien gespeichert werden sollen. Der Pfad ist konfigurierbar, ist aber standardmäßig auf 'var/asdd' gesetzt.	N
com.ibm.cdb.discover.asd.autodiscovery.asdScope	<Bereichsname> Standardwert = ASD	Der Thread wird das in diesem Bereich erwähnte Ziel auswählen, um die Ergebnisdatei zu verarbeiten. Wenn diese Eigenschaft nicht angegeben ist, wird der standardmäßige ASD-Bereich verarbeitet.	N
com.ibm.cdb.discover.asd.autodiscovery.asdProfile	<Profilname> Standardwert = ASD	Der Thread wird die in diesem Profil erwähnten Sensoren auswählen, um die Ergebnisdatei zu verarbeiten. Wenn diese Eigenschaft nicht angegeben ist, wird das ASD-Standardprofil verarbeitet.	N
com.ibm.cdb.discover.asd.autodiscovery.filesThreshold	<Dateischwellenwert> Standardwert=20	Mindestanzahl der für den Thread erforderlichen Dateien, um die Verarbeitung zu starten. Der Thread verarbeitet das Ergebnis, wenn entweder der Dateischwellenwert oder der Zeitschwellenwert erreicht wird.	N
com.ibm.cdb.discover.asd.autodiscovery.timeThreshold	<Zeitschwellenwert> Standardwert=60 Sek	Zeitschwellenwert, nach dem der Thread die Ergebnisdateien verarbeitet, auch wenn der Dateischwellenwert nicht erfüllt ist.	N

Anmerkung: 1. Die Ergebnisse der BigFix-Erkennung werden asynchron auf dem TADDM-Server empfangen, und wenn eine der Eigenschaften (com.ibm.cdb.discover.asd.autodiscovery.filesThreshold, com.ibm.cdb.discover.asd.autodiscovery.timeThreshold) erfüllt ist, werden alle verfügbaren Ergebnisdateien verarbeitet und es wird ein neuer Eintrag für “Discovery History” (Erkennungsverlauf) erstellt. Diese Eigenschaften werden gemäß den spezifischen Anforderungen für die Steuerung der Anzahl der Einträge in “Discovery History” endgültig bestimmt.

6. Bereinigung

Tabelle 60.

Seriennummer	Ressourcen	TADDM-Server		BES-Root-Server		BES-Endpunkt	
		Erstellt	Bereinigen	Erstellt	Bereinigen	Erstellt	Bereinigen
1.	Anforderungspaket	J	N[1]	J	N[2]	J	J
2.	Task	-	-	J	J [4]	-	-
3.	Aktion	-	-	J	J [3]	-	-
4.	Ergebnispaket	J	J	J	J	J	J
5.	Dateigruppe	-	-	-	-	J	J

Anmerkung:

- Die Bereinigung des Anforderungspakets wird auf dem TADDM-Server nicht unterstützt.
- Die Bereinigung des Anforderungspakets wird auf dem BES-Root-Server nicht unterstützt. (Das Anforderungspaket kann während der erneuten Erkennung wiederverwendet werden)
- Es werden nur Aktionen für die Bereinigung berücksichtigt, die von TADDM erstellt wurden und sich im Status 'Expired' (Abgelaufen) befinden (mit Ausnahme der Aktion, die mit dem Namen TADDMCLEANUP erstellt wurde).
- Tasks, die von TADDM erstellt werden, werden nur entfernt, wenn alle Aktionen, die dieser Task zugeordnet sind, bereits entfernt wurden.
- Tasks, die von TADDM erstellt werden, werden nur entfernt, wenn alle Aktionen, die dieser Task zugeordnet sind, bereits entfernt wurden.
  - Um eine bestimmte Task und die zugehörigen Aktionen von der Bereinigung auszuschließen (zur Unterstützung der erneuten Erkennung), können Sie 'retainBigFixTask.sh/.bat' wie im Anschluss gezeigt verwenden:

  Syntax: ./retainBigFixTask.sh <Taskname> <enable/disable>

Bereinigung auf TADDM-Server

Tabelle 61.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation. bigfix.taddm. cleanup.volume	<Größe mit Suffix begrenzen>	<Begrenzte Größe, nach der die alten verarbeiteten Dateien entfernt werden, z. B. 50 MB, 2 GB usw.>	N
com.collation. bigfix.taddm. cleanup.time	<Zeit mit Suffix begrenzen>	<Zum Überprüfen der verarbeiteten Dateien, die älter als konfigurierte Einheiten sind, z. B. 1D, 5H, 30M usw.>	N
com.collation. bigfix.taddm. cleanup.runtime	Anzahl der Minuten	Der Thread zur TADDM-Bereinigung wartet nach der Ausführung die konfigurierte Anzahl an Minuten.	N

Anmerkung:

- Die Bereinigung von Ergebnisdateien auf dem TADDM-Server wird nur ausgeführt, wenn mindestens eine der Eigenschaften (com.collation.bigfix.taddm.cleanup.volume, com.collation.bigfix.taddm.cleanup.time) konfiguriert ist.

Bereinigung auf Endpunkt

Tabelle 62.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation. bigfix.endpoint. cleanup	<J oder N> < Standardwert=J>	Wenn die Eigenschaft auf 'J' gesetzt ist, wird die ZIP-Datei des Anforderungspakets, das Verzeichnis mit dem extrahierten Anforderungspaket und eine neu erstellte ZIP-Datei des Ergebnispakets vom Endpunkt entfernt.	N

Bereinigung auf BigFix-Root-Server

Tabelle 63.			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
com.collation. bigfix.root. cleanup.interval	<Anzahl der Tage> <Standardwert=1>	Die Periodizität für die Ausführung der Bereinigungstask zum Entfernen von Ergebnispaketen, Tasks und abgelaufenen Aktionen vom BES-Root-Server.	N
com.collation. bigfix.root. cleanup.days	<Anzahl der Tage> <Standardwert=5>	Ergebnisdateien, die älter als eine angegebene Anzahl an Tagen sind, werden für die Entfernung berücksichtigt.	N

7. Angepasste Relevanz

Tabelle 64.			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
com.collation. bigfix.relevance. appendscope	true/false <Standard- wert=true>	Wenn 'true' festgelegt ist, wird zusätzlich zum angegebenen Bereich auch die angepasste Relevanz abgefragt. Bei 'false' wird anstelle des angegebenen Bereichs nur die Abfrage der angepassten Relevanz verwendet.	N
com.collation. bigfix. relevance	True/false	Relevanzabfrage zur Ermittlung einer Gruppe von Endpunkten für eine bestimmt Erkennung. Wenn Sie beispielsweise Endpunkte mit Computernamen erkennen möchten, die mit “nc04” beginnen, fügen Sie diese Eigenschaft in `collation.properties` als `com.collation.bigfix.relevance=(it starts with "nc04")` of (Computername in Kleinschreibung) hinzu. Dadurch wird die Aufnahme von Endpunkten, deren Computernamen mit “nc04” beginnen, zusammen mit dem Bereich aktiviert (abhängig vom Wert der Eigenschaft `com.collation.bigfix.relevance.appendscope`).	N

8. Temporäre Pfadeinstellungen für das BigFix-Paket

Tabelle 65.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
com.collation. bigfix.action. enable.os	<Aktionsscript für das konfigurierte Betriebssystem><Standardwert= Windows, AIX, Linux, SunOS>	Das Aktionsscript für das konfigurierte Betriebssystem wird in das BigFix-Aktionsscript integriert.	N
com.collation. bigfix.temp. Windows	Pfad zum Anforderungspaket<Standardwert= C:\Windows \Temp>	Der Pfad zum ASD-Anforderungspaket. *Hinweis: “\” muss im Windows-Pfad als “\\” angegeben werden.	N
com.collation. asd.temp.Windows	Pfad zum Ergebnispaket <Standardwert= C:\Windows \Temp>	Der Pfad zu den ASD-Ergebnispaketen.	N
com.collation. asd.temp.Unix	Pfad zum Ergebnispaket <Standardwert= /tmp>	Der Pfad zum ASD-Ergebnispaket.	N
com.collation. bigfix.temp.Linux	Pfad zum Ergebnispaket <Standardwert=/tmp>	Der Pfad zum ASD-Anforderungspaket.	N
com.collation. bigfix.temp. SunOS	Pfad zum Ergebnispaket <Standardwert= /tmp>	Der Pfad zum ASD-Anforderungspaket.	N
com.collation. bigfix.temp. AIX	Pfad zum Ergebnispaket <Standardwert= /tmp>	Der Pfad zum ASD-Anforderungspaket.	N
com.collation. asd.temp. Linux	Pfad zum Ergebnispaket <Standardwert ist der Wert der Eigenschaft 'com.collation.asd.temp.Unix' >	Der Pfad zum ASD-Ergebnispaket.	N
com.collation. asd.temp. SunOS	Pfad zum Ergebnispaket <Standardwert ist der Wert der Eigenschaft 'com.collation.asd.temp.Unix' >	Der Pfad zum ASD-Ergebnispaket.	N

Tabelle 65. (Forts.)			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
com.collation. asd.temp. AIX	Pfad zum Ergebnispaket <Standardwert ist der Wert der Eigenschaft 'com.collation.asd.temp.Unix' >	Der Pfad zum ASD-Ergebnispaket.	N

Anmerkung: Auf der Basis des oben konfigurierten temporären Pfads wird der Ordner auf Zielendpunkten erstellt, sofern er nicht vorhanden ist. Beispiel: Bei Windows 2003 wird der standardmäßige temporäre Pfad " C:\Windows\Temp \" verwendet, und dieser Ordner wird während der Erkennung erstellt.

Beispiele für die Scriptausführung
In den folgenden Beispielen wird die Ausführung von Scripts während dieser Integration gezeigt:

1. Script ausführen - encryptprops.sh

```
/opt/IBM/taddm/dist/bin/encryptprops.sh $COLLATION_HOME
```

2. Script ausführen – runBigFixDisocvery.sh

```
TADDM Server  – 9.167.42.227 (Linux)
BigFix server        – 10.160.161.195 (windows)
BigFix endpoints        – 10.160.161.196 (windows)
                         10.160.161.212 (windows)
Scope             – ASD (having both BigFix endpoints)
Profile              - ASD (having sensors mentioned in section 2.2)
Configuration        - done as per section 3.
```

a. Erkennung starten

```
[taddmusr@nc042227 bin]$ ./runBigFixDiscovery.sh -d -o /tmp -p ASD -s ASD
BigFix Action will be applied total [1] times with [P1D] interval

Task created on BES server with Name [20170828083852] and Action created with
ID [633]

DISCOVER: LAUNCH OK
The Bigfix Discovery script exited successfully.
```

b. Taskname – 20170828083852, Aktions-ID – 633

c. Abfrage starten:

```
[taddmusr@nc042227 bin]$ ./runBigFixDiscovery.sh -p -i 633
Repeatedly poll the BigFix Action [1] number of times for every [1] seconds
Total [2] Computers returned for Action with ID [633] has status: Open
```

```
Total [1] computers with status : The action executed successfully.
[Hostname] [Apply Count] [Line Number] [Start Time] [End Time]
[PNC161196] [1] [98] [Mon, 28 Aug 2017 14:43:56 +0000] [Mon, 28 Aug 2017
14:44:11 +0000]
```

```
Total [1] computers with status : The action failed.
[Hostname] [Apply Count] [Line Number] [Start Time] [End Time]
[PRODUCTIONWASB] [1] [37] [Mon, 28 Aug 2017 07:40:26 +0000] [Mon, 28 Aug 2017
07:40:26 +0000]

POLL FINISHED
The Bigfix Discovery script exited successfully
```

Bereinigung starten:

```
[taddmusr@nc042227 bin]$ ./runBigFixDiscovery.sh -c
CLEANUP TASK FOUND: TADDMCLEANUP with ID: 2067

Cleanup Action created with ID: [2068]

CLEANUP: LAUNCH OK
The Bigfix Discovery script exited successfully
```

Erneute Erkennung starten:

```
[taddmusr@nc042227 bin]$ ./runBigFixDiscovery.sh –r –i 20171117085907

TASK FOUND : 20171117085907 with ID : 2075

Action created with ID: [2086]

REDISCOVERY: LAUNCH OK
The Bigfix Discovery script exited successfully.
```

Fehlercodes und Beschreibung
In der folgenden Tabelle werden die Fehler- oder Nachrichten-IDs sowie die zugehörigen Beschreibungen aufgeführt, die auftreten können, wenn der Benutzer das BigFix-Script (`runBigFixDisoc-very.sh(oder .bat)`) aufruft:

Tabelle 66.

Nachrichten-ID	**N: Nachricht, U:Ursache, A:Auswirkung**
CTJTD1260E	M: BigFix-Erkennung ist nicht aktiviert. Konfigurieren Sie `com.collation.bigfix.enabled in collation.properties` A: Das Script für die BigFix-Erkennung wird nicht ausgeführt und der Thread für den Abruf des Ergebnisses wird nicht aufgerufen.
CTJTD1261E	N: Fehlende oder falsche Argumente U: Es wird versucht, Script mit einem anderen Modus als 'Discover', 'Poll', 'Cleanup' oder 'Rediscover' auszuführen. A: Script wird nicht ausgeführt
CTJTD1262E	N: Falsches Zahlenformat angegeben U: Eigenschaften oder Argumente sind im Zeichenfolgeformat angegeben, es ist aber ein Zahlenformat erforderlich A: Das Script für die BigFix-Erkennung wird nicht ausgeführt und der Thread für den Abruf des Ergebnisses funktioniert nicht korrekt.
CTJTD1263E	M: Eigenschaften der Befehlszeile konnten nicht analysiert werden: <Eigenschaftsname> U: Argumente, die während der Ausführung von Scripts übergeben wurden, werden nicht unterstützt. A: Scriptmodus wird nicht aufrufen.
CTJTD1264E	M: <Eigenschaftsname> fehlt in `collation.properties` C: Bei der Ausführung des Scripts fehlen die erforderlichen Eigenschaften oder sind ungültig (siehe '*Konfigurierbare Eigenschaften*') A: Das Script für die BigFix-Erkennung wird nicht ausgeführt und der Thread für den Abruf des Ergebnisses funktioniert nicht korrekt.

Tabelle 66. (Forts.)	
Nachrichten-ID	**N: Nachricht, U:Ursache, A:Auswirkung**
CTJTD1265I	N: Anstelle des angegebenen Bereichs wird nur angepasste Relevanz verwendet
CTJTD1266I	N: Zusätzlich zum angegebenen Bereich wird auch angepasste Relevanz verwendet
CTJTD1267E	N: Leerer Bereich angegeben, keine Elemente gefunden U: Der angegebene Bereich/die angegebene Bereichsgruppe enthält kein Element zum Definieren des Endpunkts. A: Die Erkennung wird nicht aufgerufen.
CTJTD1268E	N: Keine Sensoren im angegebenen Profil vorhanden oder aktiviert U: Das angegebene Profil enthält keine Sensoren. A: Die Erkennung wird nicht aufgerufen.
CTJTD1269E	N: Paket für AASD-Anforderung nicht vorhanden U: Problem beim Erstellen des Anforderungspakets oder es ist keine Berechtigung vorhanden oder es sich keine Elemente zum Hochladen vorhanden A: Die Erkennung wird nicht aufgerufen.
CTJTD1270E	M: Größe des AASD-Pakets übersteigt den konfigurierten Schwellenwert `com.collation.bigfix.aasdpkgmaxsize` U: Die Größe des erstellten Anforderungspakets übersteigt die konfigurierte Größe A: Die Erkennung wird nicht aufgerufen.
CTJTD1271E	M: Verbindung zu BigFix kann nicht hergestellt werden; Ursache: <Ursache> C: Verbindungsproblem mit dem Bigfix-Web-Service aufgrund ungültiger Parameter bzw. eines ungültigen Zertifikats. A: Das Paket wird nicht hochgeladen und die Erkennung nicht aufgerufen.
CTJTD1272E	M: Fehler bei der Einrichtung ermittelt: <Ursache> U: Nicht erwartetes Szenario für Code, der nicht verarbeitet wird. A: Die Scriptausführung funktioniert nicht ordnungsgemäß.
CTJTD1273I	N: Hauptscript zur (erneuten) Ausführung einer BigFix-Erkennung oder zur Abfrage einer angegebenen Erkennungsaktion oder zur Ausführung einer manuellen Bereinigung

Fix Pack 6 Integration von TADDM in ServiceNow

TADDM 7.3 Fixpack 6 enthält zusätzlich die Unterstützung für ein Integrations-Plug-in oder ein Tool zur Integration von Daten zur TADDM-Erkennung in der ServiceNow-CMDB.

Zweck

TADDM bietet eine umfassende Lösung für die Erkennung, in der das gesamte Spektrum der vom Kunden implementierten Infrastruktur abgedeckt wird. ServiceNow stellt eine flexible Software as a Service-Lösung bereit. Durch die Integration von TADDM in die ServiceNow-CMDB kann die TADDM-Erkennung, die

über die ServiceNow-Benutzerschnittstelle verfügbar ist, zusammen mit anderen Funktionen zur Verarbeitung oder Analyse aktiviert werden.

Wesentliche Vorteile der ServiceNow-Integration

- Mit dieser Integration können die Daten aus der TADDM-Erkennung in der ServiceNow-Benutzerschnittstelle für alle folgenden Formate zur Verarbeitung oder Darstellung von Daten einfach angezeigt werden>
- Reduzierung des Entwicklungsaufwands bei der Integration neuer CMDB-Tabellen, da dieser Vorgang durch die Aktualisierung der Dateien zur Datenkonvertierung vorgenommen werden kann.
- Mit dem Integrations-Plug-in können TADDM-Daten in vordefinierten, angepassten Berichten dargestellt und rollenbasierte Dashboards sofort in der ServiceNow-Benutzerschnittstelle erstellt werden.

1. Mit dieser Integration können die Daten aus der TADDM-Erkennung in der ServiceNow-Benutzerschnittstelle für alle folgenden Formate zur Verarbeitung oder Darstellung von Daten einfach angezeigt werden.
2. Reduzierung des Entwicklungsaufwands bei der Integration neuer CMDB-Tabellen, da dieser Vorgang durch die Aktualisierung der Zuordnungsdateien vorgenommen werden kann. Mit dem Integrations-Plug-in können TADDM-Daten in vordefinierten, angepassten Berichten dargestellt und rollenbasierte Dashboards sofort in der ServiceNow-Benutzerschnittstelle angezeigt werden.

Referenz

Referenzlinks für TADDM und ServiceNow.

TADDM-Referenzlinks

Weitere Informationen zu TADDM finden Sie in den folgenden Dokumentationen:

- Knowledge Center zu TADDM 7.3 und Sensoren (offizielle Dokumentation) http://www-01.ibm.com/support/knowledgecenter/SSPLFC_7.3.0/com.ibm.taddm.doc_7.3/welcome_page/kc_welcome-444.html?lang=en
- Support-Website zu TADDM http://www.ibm.com/software/sysmgmt/products/support/IBMTivoliApplicationDependencyDiscoveryManager.html
- TADDM-Wiki https://github.com/TADDM/taddm-wiki/wiki. Hier finden Sie aktuelle Informationen und bewährte Verfahren für TADDM. Setzen Sie ein Lesezeichen für diese Seite und machen Sie sich mit den Funktionen vertraut.
- Request for Enhancement Community http://www.ibm.com/developerworks/rfe/?BRAND_ID=90 In dieser Community können Erweiterungen zum Produkt direkt von IBM Entwicklern angefordert werden.

ServiceNow-Referenzlinks

Weitere Informationen zu ServiceNow finden Sie in den folgenden Dokumentationen:

- ServiceNow-CMDB https://docs.servicenow.com/bundle/kingston-servicenow-platform/page/product/configuration-management/concept/c_ITILConfigurationManagement.html
- OAuth-Konfiguration https://docs.servicenow.com/bundle/london-platform-administration/page/administer/security/concept/c_OAuthApplications.html
- CMDB-Instanz-API https://docs.servicenow.com/bundle/kingston-application-development/page/integrate/inbound-rest/concept/cmdb-instance-api.html
- ServiceNow-Community https://community.servicenow.com/community
- Identifikationsregel https://docs.servicenow.com/bundle/london-servicenow-platform/page/product/configuration-management/task/t_CreateCIIdentificationRule.html

Lösungsarchitektur

Die Integration von ServiceNow wird durch das Erstellen eines generischen Tools oder eines Plug-in-Frameworks erreicht, das über Software-Connectors für Quellen- und Zieldatenflüsse verwendet werden kann. Das Plug-in sollte Daten aus Quellendaten-Connectors empfangen, die konfigurierten Datenkonver-

tierungen ausführen und die konvertierten Daten an den Zieldaten-Connector übergeben können. Es ist die Aufgabe der Connectors, die Verbindung mit den Endpunkten zur Extraktion und Integration der Daten über die Schnittstelle herzustellen. Diese Integration der TADDM-Erkennungsdaten mit der ServiceNow-CMDB unterstützt die ursprüngliche Massenmigration und alle nachfolgenden dynamischen Aktualisierungen von Konfigurationselementen (Configuration Item, CI) mithilfe eines Verfahrens zur Übermittlung von Daten mit einer Push-Operation.

Migration
Ursprüngliche Datenmigration von TADDM-Datenbank an ServiceNow-CMDB.

1. Konfigurieren Sie die verschiedenen `<Name>.property`-Dateien für eine ordnungsgemäße Initialisierung und die Bereitstellung von Standardwerten.
2. Konfigurieren Sie die Konfigurationsdateien `mapping.xml und <CI>.xml` für die unterstützten Konfigurationselemente und die entsprechenden Transformationszuordnungen.
3. Das Tool oder das Integrations-Plug-in ruft Daten vom TADDM-Speicherserver mithilfe der standardmäßigen APIs in Form von Modellobjekten ab.
4. Die Modellobjektinformationen werden mit den jeweiligen definierten Zuordnungsregeln umgewandelt.
5. Die umgewandelten Daten werden mit einer Push-Operation über REST-APIs an die ServiceNow-Instanz gesendet und in die ServiceNow-CMDB eingefügt.

Dynamische Aktualisierungen
Durch dynamische Daten wird die Integration aus neuen TADDM-Erkennungen in der ServiceNow-CMDB aktualisiert.

1. Konfigurieren Sie die verschiedenen `<Name>.property`-Dateien für eine ordnungsgemäße Initialisierung und die Bereitstellung von Standardwerten.
2. Konfigurieren Sie die Konfigurationsdateien `mapping.xml und <CI>.xml` für die unterstützten Konfigurationselemente und die entsprechenden Transformationszuordnungen.
3. Konfigurieren Sie die Datei `<CI>.xml` für die Auflistung der Konfigurationselemente, für die Änderungsereignisse vom Framework für TADDS-Änderungsereignisse generiert werden.
4. Das Tool oder das Integrations-Plug-in empfangen Ereignisse vom Framework für TADDM-Änderungsereignisse mit dem aktualisierten Konfigurationselement.
5. Die relevanten Informationen werden mithilfe standardmäßiger APIs in Form von Modellobjekten vom TADDM-Speicherserver abgerufen.
6. Die Modellobjektinformationen werden mit den jeweiligen definierten Zuordnungsregeln umgewandelt.
7. Die umgewandelten Daten werden mit einer Push-Operation über REST-APIs an die ServiceNow-Instanz gesendet und in die ServiceNow-CMDB eingefügt.

Integrations-Plug-in ausführen

Gehen Sie für die Ausführung des Integrations-Plug-ins folgendermaßen vor:

Vorgehensweise

1. **ZIP-Datei erfassen**. Die TADDM-ISO-Datei und das Fixpack-Paket enthalten die ZIP-Datei 'IntegrationPlugin'. Nach dem Installieren oder Anhängen befindet sich das IntegrationPlugin-Paket im Verzeichnis `/opt/IBM/taddm/dist/tools`.
2. **ZIP-Datei extrahieren**.

```
Befehl: unzip IntegrationPlugin.zip
```

```
 Das Integration-Paket wird dekomprimiert und sollte die folgende Struktur aufwei
sen:
        /lib        lib directory contains all the required jars and Integration
Plugin.jar
    /plugin.sh    IntegrationPlugin script
```

```
    /resources    resources directory contains properties files and configuration
files
    /security    security directory contains Java security policy file
       /external       external directory contains IBM Java JDK
```

3. Der Benutzer muss die erforderlichen Konfigurationsschritte ausführen.
   - Konfigurieren Sie die unterstützten Konfigurationselemente (Configuration Items, CIs) und die zugehörigen Typen in der Datei `mapping.xml` für die Umsetzung von der TADDM-Datenbank in die ServiceNow-CMDB
   - Aktualisieren Sie die zugehörige Datei `<CI>.xml` für die Attributzuordnung
   - Geben Sie die erforderlichen Eigenschaften für das Ziel, die Quelle und das Plug-in in Eigenschaftendateien an.
   - Konfigurieren Sie die Zugriffsliste mit den erforderlichen Parametern auf der TADDM-Seite
   - Kopieren Sie `jdk-Linux-x86_64.zip` aus der TADDM-Hostmaschine (`Pfad:/opt/IBM/taddm/dist/external/jdk`) in das Verzeichnis `<IntegrationPlugin-Pfad>/external` auf der Maschine, auf der sich die Tools bzw. das Integrations-Plug-in befinden. Legen Sie den Pfad IBM_JAVA in 'plugin.sh' fest.
     – Entpacken Sie `jdk-Linux-x86_64.zip` in einem externen Ordner
     – Befehl: `unzip jdk-Linux-x86_64.zip`

     `javaHOME = /opt/IBM/taddm/dist/tools/IntegrationPlugin/IntegrationPlugin/external/jdk-Linux-x86_64`
   - Kopieren Sie die Datei `TADDMSec.properties` aus der TADDM-Hostmaschine (`Pfad:/opt/IBM/TADDM/dist/etc`) in das Verzeichnis `<IntegrationPlugin-Pfad>/security/etc` auf der Maschine, auf der sich das Tool oder das Integrations-Plug-in befinden.
4. Aktivieren Sie den Migrations- und Änderungsereignisprozess in der Datei `plugin.properties`.
5. Führen Sie das Script aus.

```
Integration Plugin package contains plugin.sh script in /script.
Starting the Integration Plugin:
--------------------------------
  Option 1:
  [Command] ./plugin.sh start
  Taddm User Id and Password – will read from internal properties file
  [Command] ./plugin.sh start &
  Taddm User Id and Password – will read from internal properties file
  & - for running the plugin in background

  Note: User is recommended to exit gracefully from the terminal from which
Integration plugin is invoked and put in background. This
can be done using commands “exit” or “disown”.

  Option 2:
  [Command] ./plugin.sh -u <username> start
  -u, --user <Taddm User Id>
 Password – to be entered on subsequent command line prompt (for safety rea
sons)


  Option 3:
  [Command] ./plugin.sh -u <username> -p <password> start
  -u, --user <Taddm User Id>
  -p, --password <Taddm User password>

 Stopping the Integration Plugin:
 --------------------------------
  Option 1:
  [Command] ./plugin.sh stop

  Option 2:
  [Command] ./plugin.sh stop force

 Status of Integration Plugin:
```

```
----------------------------
 Option 1:
 [Command] ./plugin.sh status
```

Integration von TADDM in die ServiceNow-CMDB

In TADDM 7.3 Fixpack 6 konzentriert sich diese Integration auf die Migration oder das Übergeben von Erkennungsdaten mit einer Push-Operation von TADDM zur ServiceNow-Instanz über REST-APIs.

Voraussetzungen

1. Die ServiceNow-Instanz sollte ausgeführt werden und vom Tool oder Integrations-Plug-in aus zugänglich sein, damit die umgewandelten Daten gesendet werden können.
2. Der TADDM-Speicherserver muss ausgeführt werden und vom Tool oder Integrations-Plug-in aus zugänglich sein, damit die Erkennungsdaten abgerufen werden können.
3. Es muss eine Netzkonnektivität zwischen dem TADDM-Server und der Maschine mit dem Tool oder dem Integrations-Plug-in bestehen, damit die dynamischen Änderungsereignisse empfangen werden können.
4. Kopieren Sie die Datei `jdk-Linux-x86_64.zip` von der TADDM-Hostmaschine (Pfad: `/opt/IBM/taddm/dist/external/jdk`) in das Verzeichnis `<IntegrationPlugin-Pfad>/external` auf der Maschine, auf der sich die Tools bzw. das Integrations-Plug-in befinden. Legen Sie den Pfad von IBM JAVA in 'plugin.sh' fest.
   - `Entpacken Sie die Datei 'jdk-Linux-x86_64.zip'` in einen externen Ordner.
   - Befehl: `unzip jdk-Linux-x86_64.zip`

   javaHome=<IntegrationPlugin-Pfad>/external/jdk-Linux-x86_64
5. Wenn das Integrations-Plug-in auf einer TADDM-Maschine ausgeführt wird, geben Sie den Ausgangspfad `taddmHome` an: `IntegrationPlugin/resources/config/taddm.properties`. `taddmHome=/opt/IBM/taddm/dist`.
6. Wenn das Integrations-Plug-in nicht auf einem TADDM-Erkennungsserver ausgeführt wird, kopieren Sie die Datei `TADDMSec.properties` von der TADDM-Maschine (`/opt/IBM/taddm/dist/etc`) in das Verzeichnis `<IntegrationPlugin-Pfad>/security/etc` auf der Maschine, auf der das Plug-in installiert ist. `taddmHome=<IntegrationPlugin-Pfad>/security/etc.`

   Anmerkung: Wenn die Maschine über keine Netzverbindung verfügt, aktivieren Sie zuerst die Netzkonnektivität und geben die folgende Konfiguration unter `/etc/hosts.XX.XX.XX.XX servicenow service-now.dev345.com` ein.
7. Stellen Sie sicher, dass alle erforderlichen Konfigurationsschritte ausgeführt wurden.

Beschränkungen

Dem aktuellen Release sind die folgenden Einschränkungen zugeordnet:

1. Unterstützung für die Integration von TADDM und die ServiceNow-CMDB startet mit dem Release von TADDM 7.3 Fixpack 6.
2. Das Integrations-Plug-in oder das Tool für die ServiceNow-CMDB wird aktuell nur für die Plattformen RHEL 7.3, 7.4 und 7.5 unterstützt.
3. Nicht alle TADDM-Konfigurationselemente werden derzeit mithilfe dieser Integration in diesem FP unterstützt.

Konfiguration

Befolgen Sie die in diesem Abschnitt beschriebenen Schritte, um die gewünschte Konfiguration festzulegen.

Zugriffsberechtigungsnachweise konfigurieren

Sie können eine Zugriffsliste konfigurieren.

Informationen zu diesem Vorgang

Für die Ausführung der Integration sind die TADDM-Berechtigungsnachweise und die ServiceNow-Berechtigungsnachweise erforderlich. Das Integrations-Plug-in kann die TADDM-Berechtigungsnachweise wiederverwenden, aber die ServiceNow-Berechtigungsnachweise müssen über die Zugriffsliste konfiguriert werden. Auf Basis der Zugriffslistenkonfiguration stellt TADDM eine Verbindung mit der ServiceNow-Instanz her.

Vorgehensweise

1. Klicken Sie im Fenster **Discovery Management Console** auf **Erkennung** > **Zugriffsliste**.
2. Klicken Sie im Fenster 'Zugriffsliste' auf **Hinzufügen**. Das Fenster mit den Einzelheiten zum Zugriff wird angezeigt.
3. Wählen Sie in der Liste **Komponententyp** den Eintrag **Integration** aus.
4. Wählen Sie in der Liste **Lieferant** den Eintrag **ServiceNow** aus.
5. Geben Sie im folgenden Feld die erforderlichen Einzelheiten an.

   Name
   Geben Sie einen eindeutigen Namen für die Zugriffsliste ein.

   Benutzer
   Geben Sie den Namen des Benutzers der ServiceNow-Instanz ein.

   Client-ID
   (Optional) Client-ID, die automatisch vom ServiceNow-OAuth-Server generiert wird.

   Geheimer Clientschlüssel
   (Optional) Geheimer Clientschlüssel für die OAuth-Anwendung.

   Kennwort
   Geben Sie das Kennwort der ServiceNow-Instanz ein.

   Wichtig: 'Client-ID' und 'Geheimer Clientschlüssel' sind Pflichtfelder für die tokenbasierte Anmeldung. Weitere Informationen zur tokenbasierten Konfiguration finden Sie unter *OAuth-Konfiguration*.
6. Klicken Sie auf **OK**.

Nächste Schritte

Konfigurieren Sie die verschiedenen Eigenschaftendateien. Siehe *Eigenschaften konfigurieren*.

Plug-in-Eigenschaften konfigurieren

Damit das Integrations-Plug-in ausgeführt werden kann, müssen Sie verschiedene Eigenschaftendateien in TADDM und im Integrations-Plug-in konfigurieren.

Informationen zu diesem Vorgang

Im Integrations-Plug-in müssen Sie 'plugin.properties', 'taddm.properties' und 'serviceNow.properties' konfigurieren. In TADDM müssen Sie 'collation.properties' konfigurieren. Nach dem Dekomprimieren von 'intergrationPlugin.zip' in das Ressourcenverzeichnis konfigurieren Sie die folgenden Eigenschaften:

Vorgehensweise

1. Öffnen Sie die Datei <*IntegrationPlugin-Pfad*>/resources/config/plugin.properties.
2. Konfigurieren Sie die folgenden Eigenschaften:

   Für das Integrations-Plug-in:

   - enableChangeEventManagement = *<Verarbeitung für Aktualisierungen von Änderungsereignissen aktivieren und inaktivieren>*
   - enableMigration = *<Migration von Daten des Konfigurationselements aktivieren und inaktivieren>*
   - runMigrationAgain = *<Migrationsprozess erneut für alle Konfigurationselemente ausführen>*
   - migrationThreadPoolSize = *<Anzahl der Threads für die Verarbeitung von Migrationsdaten>*

- changeEventThreadPoolSize = *<Anzahl der Threads für die Verarbeitung der Aktualisierungen von Änderungsereignissen >*
- sourceConnector = *<Quellenendpunkt>*
- targetConnector = *<Zielendpunkt>*
- javaHome = *<IBM Java-Ausgangspfad>*
- useMagicMethod = *<Auf 'true' setzen, um 'magicMethods' auszuführen>*
- sessionRetryTime = *<Festlegen des Zeitpunkts von Sitzungswiederholung für Quellen- und Zielconnector>*

Für Protokolleinstellungen:

- fileSize = *<Protokolldateigröße in MB>*
- backupIndex = *<Anzahl der Sicherungsprotokolldateien, bevor sie gelöscht oder durch die aktuellsten Protokolle ersetzt werden>*
- logLevel = *<Protokollebene>*

Für ordnungsgemäße Beendigung:

- gracefulEventProcessingCount = *<Gesamtzahl der Ereignisse, die während dem Herunterfahren verarbeitet werden sollen>*
- gracefulEventProcessingTime = *<Dauer der Verarbeitung für ordnungsgemäße Beendigung (in Millisekunden)>*

Anmerkung: Nicht alle Eigenschaften sind verbindlich. Einzelheiten finden Sie im *Anhang A: In der Integration verwendete Eigenschaften*.

3. Öffnen Sie die Datei *<IntegrationPlugin-Pfad>*/resources/taddm_snow/taddm.properties.
4. Konfigurieren Sie die folgenden Eigenschaften:

   Für TADDM-Konfigurationen (Quelle):

   - taddmHost = *<IP-Adresse der TADDM-Maschine>*
   - taddmPort = *<TADDM-Server-Port>*
   - taddmUserName = *<Benutzername>*
   - taddmPassword = *<TADDM-Benutzerkennwort oder in verschlüsselter Form gespeichertes Kennwort>*
   - taddmUseSSl = *<Zum Herstellen einer sicheren Verbindung>*
   - taddmTrustStorePath = *<Pfad zum SSL-Zertifikat angeben>*
   - taddmHome = *<Pfad zum Collation-Ausgangsverzeichnis>*
   - tlsVersion = *<Entspricht der TLS-Version, die in der TADDM-Datei 'collation.properties' festgelegt ist>*

   So konfigurieren Sie ein Änderungsereignis:

   - listenIp = *<IP-Adresse des Systems, auf dem Plug-in ausgeführt wird und für Verarbeitung des Änderungsereignisses empfangsbereit ist>*
   - listenPort = *<Port, auf dem das Plug-in für die Verarbeitung des Änderungsereignisses empfangsbereit ist>*

   Anmerkung: Nicht alle Eigenschaften sind verbindlich. Einzelheiten finden Sie im *Anhang A: In der Integration verwendete Eigenschaften*.

5. Öffnen Sie die Datei *<IntegrationPlugin-Pfad>*/resources/taddm_snow/serviceNow.properties.
6. Konfigurieren Sie die folgenden Eigenschaften:

   Für ServiceNow-Konfiguration (Ziel):

   - serviceNowInstanceUrl = *<URL der ServiceNow-Instanz>*

- serviceNowOAuthUrl = *<oAuth-URL zum Generieren eines Zugriffstokens>*
- discoverySource = *<Erkennungsquellentyp>*
- relationTable = *<Tabelle mit Beziehungstyp>*
- refreshTokenLifespan = *<Lebenszyklus des Aktualisierungstoken in Sekunden>*
- serviceNow.attribute.softdelete.status = *<Status, der für die Angabe von vorläufigem Löschen von Eintrag verwendet wird>*
- serviceNow.attribute.softdelete.name = *<ServiceNow-Attribut für das Beibehalten des Status für das vorläufige Löschen>*
- serviceNow.attribute.guid.name = justification *<ServiceNow-Attribut zum Speichern der TADDM-GUID, der eindeutigen ID für jeden Eintrag>*
- securityProtocol= *<TLS-Protokoll eingeben>*

Für Proxy-Einstellungen:

- enableProxy = *<Proxy beim Verbinden mit ServiceNow aktivieren>*
- proxyHost = *<Proxy-IP>*
- proxtPort = *<Proxy-Host>*
- proxyUserName = *<Proxy-Benutzername>*
- proxyPassword = *<Proxy-Kennwort>*

Anmerkung: Nicht alle Eigenschaften sind verbindlich. Einzelheiten finden Sie im *Anhang A: In der Integration verwendete Eigenschaften*.

7. Starten Sie das Integrations-Plug-in erneut.
8. Öffnen Sie auf dem TADDM-Server die Datei `$COLLATION_HOME/etc/collation.properties`.
9. Konfigurieren Sie die folgenden Eigenschaften:

   So aktivieren Sie ChangeEvent:

   - com.ibm.cdb.omp.changeevent.enabled= *<ChangeEvent-Modul aktivieren, um Ereignisse aus TADDM zu senden>*
   - com.ibm.cdb.omp.changeevent.optimized.update = *<Eigenschaft für die Konsolidierung von Aktualisierungsereignissen für mehrere Attribute des gleichen Modellobjekts>*
   - com.ibm.cdb.omp.changeevent.classnames.nonfriendly = *<Diese Eigenschaft gibt anstelle des nicht benutzerfreundlichen Klassennamens den Klassennamen des Konfigurationselements zurück>*
   - com.ibm.cdb.omp.changeevent.deletefromportal = *<Eigenschaft für das Senden von Löschereignissen, wenn das Konfigurationselement aus dem Portal gelöscht wird>*
   - com.ibm.cdb.omp.changeevent.GenerationType = later *<Diese Eigenschaft sollte auf 'later' gesetzt sein, damit die Änderungsereignisse separat verarbeitet werden, da die Erkennung andernfalls blockieren kann>*

   So verschlüsseln Sie das Kennwort:

   - com.collation.integrationplugin.taddm.password= *<Verschlüsseltes TADDM-Kennwort>*
   - com.collation.integrationplugin.target.proxy.password= *<Verschlüsseltes Kennwort für das Zielproxy>*

   Anmerkung: Nicht alle Eigenschaften sind verbindlich. Einzelheiten finden Sie im *Anhang A: In der Integration verwendete Eigenschaften*.

10. Starten Sie TADDM erneut.

Erkennungsquelle hinzufügen

In der ServiceNow-Schnittstelle werden Daten, die aus der TADDM-CMDB migriert werden, durch den Typ der Erkennungsquelle ermittelt. Der Standardwert des Erkennungsquellentyps ist IBM TADDM. Damit können Daten unterschieden werden, die aus anderen Erkennungsquellen migriert wurden.

Informationen zu diesem Vorgang

In dieser Task wird die Vorgehensweise bei der Konfiguration der Erkennungsquelle in der ServiceNow-Instanz beschrieben. Wenn Sie die Konfiguration das erste Mal durchführen, können Sie die ServiceNow-Instanz mit dem Standardwert 'discoverySource' konfigurieren. In dieser Task wird auch beschrieben, wie der Wert der Erkennungsquelle angepasst werden kann.

Vorgehensweise

1. Melden Sie sich bei der ServiceNow-Instanz an.
2. Geben Sie im Suchfeld 'Filter Navigator' dem Wert **System Definition** ein.
3. Klicken Sie in der Suchliste unter 'System Definition' auf **Tables** (Tabellen).
4. Wählen Sie im Feld **Go To** (Wechseln zu) den Wert **Name** aus der Liste aus, geben Sie dann im Suchfeld den Wert *cmdb_ci* ein und drücken Sie die Eingabetaste, um die Suche auszuführen.
5. Klicken Sie auf den Link mit der Bezeichnung, die *cmdb_ci* entspricht.
6. Wählen Sie auf der Registerkarte **Columns** (Spalte) im Feld **Go To** (Wechseln zu) den Eintrag **Column label** (Spaltenbeschriftung) aus der Liste aus und geben Sie **Discovery source** (Erkennungsquelle) ein.
7. Klicken Sie auf **Discovery source** (Erkennungsquelle).
8. Klicken Sie unter den zugehörigen Links auf die Registerkarte **Choices** (Auswahlmöglichkeiten) und klicken Sie anschließend auf **New** (Neu).
9. Füllen Sie die folgenden Felder aus:

   Beschreibung
   Geben Sie einen geeigneten Wert ein.

   Wert
   Geben Sie einen geeigneten Wert ein (verwenden Sie diesen Wert für die Eigenschaft 'discoverySource') oder geben Sie *IBM TADDM* ein, wenn Sie das Plug-in das erste Mal konfigurieren und den Standardwert “IBM TADDM” verwenden möchten.
10. Klicke Sie auf **Übergeben**.

Wenn Sie den Wert für die Erkennungsquelle anpassen, führen Sie die folgenden Schritte aus. In der Schnittstelle für das Integrations-Plug-in:

11. Öffnen Sie die Datei 'ServiceNow.properties' und suchen Sie anschließend die Eigenschaft 'discoverySource'.
12. Legen Sie für die Eigenschaft 'discoverySource' folgenden Wert fest: 'discoverySource = <*Gleicher Wert wie im Feld 'Value'*>.
13. Starten Sie das Plug-in erneut, damit die Änderungen wirksam werden.

Kennwörter verschlüsseln

Die Kennwortverschlüsselung verhindert den unbefugten Zugriff auf Klartextkennwörter. Verschlüsselte Kennwörter sorgen für eine sichere Verbindung zwischen TADDM und dem Integrations-Plug-in.

Informationen zu diesem Vorgang

Gehen Sie zum Verschlüsseln von Kennwörtern folgendermaßen vor:

Vorgehensweise

1. Suchen Sie in der Datei 'collation.properties' die Eigenschaft 'com.collation.integrationplugin.taddm.password' und wenn ein Wert für diese Eigenschaft vorhanden ist, löschen Sie diesen. Geben Sie anschließend ein neues Kennwort ein.
2. Suchen Sie die Datei `$COLLATION_HOME/bin directory/encryptprops.sh` und führen Sie anschließend das Script zum Verschlüsseln des Kennworts aus.
3. Kopieren Sie das verschlüsselte Kennwort aus der Eigenschaft 'com.collation.integrationplugin.taddm.password'.

4. Suchen Sie in der Datei '*<IntegrationPlugin-Pfad>*/resources/config/taddm_snow/taddm.properties' die Eigenschaft 'taddmPassword' und geben Sie das verschlüsselte Kennwort ein.
5. Starten Sie das Plug-in erneut, damit die Änderungen wirksam werden.

TLS-Zertifikat konfigurieren

Mithilfe des TLS-Protokolls können Sie eine sicher Verbindung zwischen dem Integrations-Plug-in und TADDM herstellen.

Informationen zu diesem Vorgang

Führen Sie zum Aktivieren von TLS die folgenden Schritte aus:

Vorgehensweise

In der TADDM-Schnittstelle:

1. Öffnen Sie die Datei '$COLLATION_HOME/etc/collation.properties', suchen Sie die folgenden Eigenschaften und stellen Sie sicher, dass sie konfiguriert sind.
   - com.ibm.cdb.secure.server= true
   - com.ibm.cdb.rmi.ssl.protocol = *<TLS-Version>*
   - com.ibm.cdb.ssl.protocol = *<TLS-Version>*
2. Starten Sie TADDM erneut.
3. Öffnen Sie einen Web-Browser und geben Sie die URL und die Portnummer des Systems ein, auf dem TADDM installiert ist. Beispiel: http://system.company.com:9430.
4. Klicken Sie im Abschnitt 'Discovery Management Console' auf **SSL-Optionen anzeigen**.
5. Klicken Sie auf den Link **Truststore herunterladen**. Geben Sie das Verzeichnis an, in dem Sie die Datei speichern möchten, wenn Sie von Ihrem Browser dazu aufgefordert werden.
6. Geben Sie im Eingabefeld neben dem Link **Truststore herunterladen** den Pfad zum Verzeichnis ein, in dem die Zertifikatsdatei gespeichert ist.

Im Integrations-Plug-in:

7. Öffnen Sie die folgende Datei:

   <Integrationplugin-Pfad>`/resources/taddm_snow/taddm.properties`. Suchen Sie die folgenden Eigenschaften und stellen Sie sicher, dass diese konfiguriert sind:

   - taddmTrustStorePath = *<Pfad mit der Zertifizierung eingeben>*
   - /resources/taddm_snow/taddm.properties

   Wenn TADDM auf einer anderen Maschine installiert ist, kopieren Sie das Zertifikat und speichern es auf der Maschine, auf der das Integrations-Plug-in ausgeführt wird. Anschließend geben Sie diesen Pfad in der Eigenschaft 'taddmTruststorePath' ein.

   - taddmUseSSl *<Setzen Sie den Wert auf 'true'>*
   - taddmUseSSl *<Setzen Sie den Wert auf 'true'>*
8. Starten Sie das Integrations-Plug-in erneut.

OAuth in ServiceNow konfigurieren

Mit OAuth können Sie einmalig eine Benutzer-ID und ein Kennwort übergeben und anschließend für nachfolgende REST-Anforderungen ein Token verwenden, statt bei jeder Anforderung Berechtigungsnachweise zu übergeben.

Vorgehensweise

1. Melden Sie sich bei der ServiceNow-Instanz an.
2. Aktivieren Sie das OAuth 2.0-Plug-in.
3. Setzen Sie in der Tabelle 'sys_properties' die Systemeigenschaft `com.snc.platform.security.oauth.is.active` auf **true**.

4. Klicken Sie auf **system OAuth** > **Application Registry** (System-OAuth > Anwendungsregistry).
5. Klicken Sie auf **NEW** (Neu) und klicken Sie anschließend auf einen **OAuth-API**-Endpunkt für externe Clients.
6. Geben Sie die folgenden Parameter an:

   Name
   Geben Sie einen eindeutigen Namen ein.

   Client-ID
   Client-ID, die automatisch vom ServiceNow-OAuth-Server generiert wird.

   Geheimer Clientschlüssel
   Geheimer Clientschlüssel für die OAuth-Anwendung.

   Refresh Token Lifespan (Lebensdauer des Aktualisierungstokens)
   Zeit in Sekunden, in der das Aktualisierungstoken gültig ist.

   Access Token Lifespan (Lebensdauer des Zugriffstokens)
   Zeit in Sekunden, in der das Zugriffstoken gültig ist.
7. Erfassen Sie den Wert der Client-ID und des geheimen Clientschlüssels aus dem vorherigen Schritt.
8. Klicke Sie auf **Übergeben**.

   Anmerkung: Die OAuth-Konfiguration ist für die Herstellung der Verbindung zu ServiceNow nicht verbindlich. Die OAuth-Konfiguration ist nur für die tokenbasierte Anmeldung erforderlich, und andernfalls werden die ServiceNow-Berechtigungsnachweise zum Herstellen der Verbindung verwendet.

Regel für angepasste ID erstellen

Mit Identifikationsregel werden Konfigurationselemente in der ServiceNow-CMDB während des Identifikations- und Abstimmungsprozesses eindeutig angegeben.

Informationen zu diesem Vorgang

Jede Klasse einer ServiceNow-CMDB kann einer einzelnen Identifikationsregel zugeordnet werden.

Prozedur

- Informationen zum Erstellen einer Regel für angepasste ID finden Sie im Abschnitt 'ServiceNow' über den folgenden Link:
  https://docs.servicenow.com/bundle/london-servicenow-platform/page/product/configurationmanagement/task/t_CreateCIIdentificationRule.html

XML-Datei für das Konfigurationselement bearbeiten

Mit den Dateien des Typs <*Konfigurationselement*>.xml werden die Zuordnungsattribute für die Übertragung der Daten aus der TADDM-Erkennung an die ServiceNow-CMDB und zum Herstellen von Beziehungen zwischen TADDM und den ServiceNow-Konfigurationselementen konfiguriert. Diese XML-Dateien haben normalerweise den gleichen Namen wie die Konfigurationselemente, denen sie zugeordnet sind.

Informationen zu diesem Vorgang

Diese XML-Dateien sind vorkonfiguriert. Sie können die Elemente anpassen und Attribute entsprechend Ihren Anforderungen zuordnen.

Vorgehensweise

1. Öffnen Sie die Datei <*IntegrationPlugin-Pfad*>/resources/config/mappingFiles/<Konfigurationselement>.xml.
2. Bearbeiten Sie die Elemente und die zugehörigen Attribut basierend auf Ihren Anforderungen.

   Anmerkung: Die in diesem Abschnitt beschriebenen Elemente und Attribute sind vorkonfiguriert. Ändern Sie diese nur, wenn es erforderlich ist.

   - Sie können die folgenden Attribute des Elements <*Konfigurationselement*> bearbeiten:

  - [name] Name des Konfigurationselements oder Name des untergeordneten Konfigurationselements
  - [targetTable] Zieltabelle in der ServiceNow-Instanz
- Sie können die folgenden untergeordneten Element des Elements *<Konfigurationselement>* bearbeiten:
  - [relationship] Wenn 'true' festgelegt ist, wird die Beziehung zum übergeordneten Konfigurationselement gesucht
  - [outbound] oder [inbound] Stellt den Typ der Beziehung dar. Beispielsweise enthält 'AppServer.xml' eine ausgehende Beziehung, 'FileSystem.xml' enthält eine eingehende Beziehung
  - [relation] Zeigt den Beziehungstyp oder den Namen eines Konfigurationselements auf ServiceNow-Seite an
  - [srcRelationType] Definiert den Beziehungsnamen auf TADDM-Seite
  - [srcAttributeType] Definiert die Beziehung zu anderen Konfigurationselementen innerhalb der Konfigurationselemente. Definiert das andere Konfigurationselement (auf TADDM-Seite), zu dem die Beziehung für dieses Konfigurationselement besteht
  - [targetAttributes] Stellt das andere Konfigurationselement dar, mit dem eine Beziehung festgestellt wurde. Daraus ergibt sich die 'sys_ID' des anderen Konfigurationselements.
  - [targetRelationType] Stellt die 'sys_id' der Beziehung dar (definiert auf ServiceNow-Seite)
  - [targetClass] Stellt den Namen der ServiceNow-Tabelle dar, mit der die Beziehung zu diesem Konfigurationselement besteht
  - [srcAttribute] Attributzuordnung für das Konfigurationselement
- Untergeordnete Elemente des Elements <srcAttribute >:
  - [targetAttr] Stellt das Zielattribut in ServiceNow dar.
  - [defaultValue] Standardwert des Zielattributs.
  - [defaultValue] Standardwert des Zielattributs.

3. Starten Sie das Plug-in erneut, damit die Änderungen wirksam werden.

Beispiel

Im Folgenden sehen Sie ein Beispiel für den Beispielcode:

```
<ciMapping>
<configurationItem name="AppServer" targetTable="cmdb_ci_appl">
<!-- AppServer relation with Computer System -->
<relationship value="true">
<outbound value="true">
<relation name="outbound_relations">
<srcRelationType name="RunsOn" />
<srcAttributeType name="ComputerSystem" />
<targetAttribute name="target" value="Host.Guid" />
<targetRelationType name="type" value="Runs on" />
<targetClass name="sys_class_name" value="cmdb_ci_hardware" />
</relation>
</outbound>
</relationship>
<srcAttribute name="ConfigFile.URI" type="String">
<targetAttribute>config_directory</targetAttribute>
<defaultValue></defaultValue>
<magicMethod></magicMethod>
</srcAttribute>
</configurationItem>
<!-- Sql Server -->
<configurationItem name="SqlServer" targetTable="cmdb_ci_db_mssql_server">
<srcAttribute name="Name" type="String">
<targetAttribute>instance</targetAttribute>
<defaultValue></defaultValue>
<magicMethod></magicMethod>
</srcAttribute>
</configurationItem>
</ciMapping>
```

Datei 'mapping.xml' bearbeiten

In der Datei 'Mapping.xml' können Sie unterstützte Konfigurationselemente (Configuration Items, CIs) konfigurieren. Diese Datei ist normalerweise vorkonfiguriert, Sie können die Elemente und Attribute aber entsprechend Ihren Anforderungen anpassen.

Informationen zu diesem Vorgang

Wenn Sie beispielsweise den Wert des Tiefenattributs erhöhen, werden weitere hierarchische Daten für ein angegebenes Konfigurationselement abgerufen, und wenn Sie den Wert verringern, werden weniger Daten abgerufen und die Ausführungszeit wird reduziert.

Vorgehensweise

1. Öffnen Sie die Datei *<IntegrationPlugin-Pfad>*/resources/config/mappingFiles/Mapping.xml.
2. Bearbeiten Sie die Elemente und die zugehörigen Attribut basierend auf Ihren Anforderungen.

   Anmerkung: Die in diesem Abschnitt beschriebenen Elemente und Attribute sind vorkonfiguriert. Ändern Sie diese nur, wenn es erforderlich ist.

   - Sie können die folgenden Attribute des Elements *<Dateiname>* bearbeiten:
     - [name] Name des Konfigurationselements (CI)
     - [file] Umsetzungsdatei oder Datei '*<Konfigurationselement>*.xml', mit der das TADDM-Attribut den entsprechenden ServiceNow-Attributen zugeordnet wird
     - [id] Eindeutige ID für das unterstützte 'ModelObject' des Konfigurationselements ist das am häufigsten unterstützte Konfigurationselement. Der erste Eintrag ist immer für 'modelObject', und Sie dürfen kein ID-Attribut für das 'modelObject'-Konfigurationselement bereitstellen.

     Im Anschluss finden Sie ein Codebeispiel:

     ```
      <fileName>
     <configurationItem name="ModelObject" file="ModelObject.xml" />
     <configurationItem name="ComputerSystem" file="ComputerSystem.xml" id = "0"/>
     <configurationItem name="AppServer" file="AppServer.xml" id = "1"/>
     <configurationItem name="StorageVolume" file="StorageVolume.xml" id= "2" />
     </fileName>
     ```

   - Sie können die folgenden Attribute des Elements *<Hierarchie>* bearbeiten:
     - [name] Name des Konfigurationselements (CI)
     - [ref] Übergeordnetes Konfigurationselement (des im Namensattribut definierten Konfigurationselements)
     - [ignore] Standardwert ist 'false'. Wenn das Integrations-Plug-in das im Namensattribut definierte Konfigurationselement nicht verarbeiten soll, setzen Sie den Wert auf 'true'.
     - [Depth] Ruft rekursive untergeordnete Objekte des Konfigurationselements auf Basis der Tiefenebene ab.
     - [top-level] Zeigt an, ob es sich bei dem im Konfigurationselement definierten Modellobjekt um ein übergeordnetes Konfigurationselement handelt.
3. Starten Sie das Plug-in erneut.

Beispiel

Im folgenden Beispiel handelt es sich beim Modellobjekt 'LinuxUnitaryComputerSystem' um ein untergeordnetes Element von 'computerSystem'. Durch den Wert des Attributs 'Ignore' wird sichergestellt, dass das Konfigurationselement bei der Ausführung berücksichtigt wird. Da der Wert der Ausgangsebene auf 'false' gesetzt ist, wird der Code rekursiv ausgeführt, bis das Konfigurationselement der Ausgangsebene erreicht wird. Das Konfigurationselement der Ausgangsebene wird im Attribut [ref] definiert.

```
<hierarchy>
<configurationItem name="ComputerSystem" ref="ComputerSystem" ignore="true" top-level="true" />
<configurationItem name="LinuxUnitaryComputerSystem" ref="ComputerSystem" ignore="false" top-le
vel="false" />
<configurationItem name="DiskDrive" ref="DiskDrive" ignore="false" top-level="true" />
<configurationItem name="AppServer" ref="AppServer" ignore="true" depth="3" top-level="true" />
…
</hierarchy>
```

'magicMethod' konfigurieren

'magicMethod' wird im Integrations-Plug-in verwendet, um die Werte während der Laufzeit dynamisch zu aktualisieren. Dies ist eine hilfreiche Funktion, um den Wert von Zuordnungsattributen zu bearbeiten. Mit 'magicMethod' wird ein Wert vor der Übertragung auf die Zielseite in das gewünschte Format für die Größe konvertiert. Sie können 'magicMethod' auf Basis einiger Voraussetzungen anpassen.

Informationen zu diesem Vorgang

Gehen Sie zum Konfigurieren von 'magicMethod' folgendermaßen vor.

Vorgehensweise

1. Erstellen Sie eine Java-Funktion für Ihre Anforderungen und definieren Sie einen geeigneten Namen für die Methode. Das folgende Beispiel zeigt einen Basiscode:

   ```
   public static <returntype> methodName(String origValue){
   return <manipulatedValue>;
   }
   ```

2. Öffnen Sie die Datei `<Integrations-Plug-in>/resources/config/magicMethod/MagicMethod.txt`.
3. Kopieren Sie den Java-Code, den Sie erstellt haben, und fügen Sie ihn in die Datei 'MagicMethod.txt' ein.
4. Gehen Sie in der Datei '<Konfigurationselement>.xml' folgendermaßen vor:
   a) Ersetzen Sie den Wert des 'magicMetod'-Elements durch den Methodennamen des Java-Codes. Im folgenden Beispiel werden einige Zeilen aus der Datei '<Konfigurationselement>.xml' gezeigt:

      ```
      <srcAttribute name="MemorySize" type="Long">
      <targetAttribute>ram</targetAttribute>
      <defaultValue></defaultValue>
      <magicMethod>convertToMB</magicMethod>
      </srcAttribute>
      ```

   b) Stellen Sie sicher, dass der Datentyp in den Attributen [type] dem Datentyp entspricht, der im Java-Code definiert ist.
5. Öffnen Sie die Datei `<Integrations-Plug-in>/resources/config/plugin.properties` und konfigurieren Sie die folgenden Eigenschaften:

   `useMagicMethod = true`

   Der Wert ist standardmäßig 'false', aber der Code wird nur ausgeführt, wenn der Wert auf 'true' gesetzt ist.

   `Javahome= < Geben Sie den IBM Java-Pfad ein, in dem das Plug-in ausgeführt wird/jre >`

   Im folgenden Beispiel wird ein Pfad gezeigt: `/opt/IBM/taddm/dist/tools/Integrations-Plug-in/IntegrationPlugin/external/jdk-Linux-x86_64/jre`
6. Öffnen Sie den Ordner '*<Integrations-Plug-in>*/lib' und suchen Sie die Datei 'magic-method.jar'.
7. Erstellen Sie eine Sicherungskopie der Datei 'magic-method.jar' und löschen Sie anschließend die Datei 'magic-method.jar' aus dem lib-Verzeichnis.
8. Starten Sie das Plug-in erneut.

Im folgenden Beispiel wird ein Java-Code gezeigt, mit dem der Speicherwert von Megabyte in Gigabyte umgewandelt wird:

```
“public static float convertMBToGB(Long value){
return value/1024;
}”
```

In diesem Code gibt der Wertparameter den ursprünglichen Wert an, der aus der Quelle empfangen wurde. Im Back-End wird die Datei 'magic-method.jar' erstellt; diese JAR-Datei konvertiert den Wert beim Übertragen der Daten an das Ziel.

Nächste Schritte

Gehen Sie zum Testen des Plug-ins folgendermaßen vor:

- Prüfen Sie, ob 'console.log' erfolgreich kompiliert wurde.
- Prüfen Sie 'plugin.log', um sicherzustellen, dass 'magicMethod' gestartet wurde.
- Prüfen Sie 'transform.log', um zu bestätigen, dass die Werte gemäß 'magicMethod' geändert wurden.

Wichtig: Nach Abschluss der Ausführung setzen Sie die Eigenschaft 'magicMethod' auf 'false'. Wenn Sie weiterhin den Wert 'true' verwenden, wird bei jedem Start des Plug-ins die 'magicMethod'-Kompilierung automatisch im Back-End gestartet.

Threads konfigurieren

Im Integrations-Plug-in wird die Multithread-Architektur zur Verarbeitung von Massendaten verwendet. Sie können die Verarbeitung von Threads konfigurieren, um die Leistung zu verbessern und die Ausführung zu beschleunigen. Sie können die Anzahl der Threads für die Datenmigration und für die Verarbeitung von Änderungsereignissen angeben.

Vorgehensweise

1. Öffnen Sie die Datei *<IntegrationPlugin-Pfad>*`/resources/config/plugin.properties`.
2. Konfigurieren Sie die folgenden Eigenschaft gemäß Ihren Anforderungen:
   - migrationThreadPoolSize = *<Anzahl der Threads für die Verarbeitung der Datenmigration>*
   - changeEventThreadPoolSize = *<Anzahl der Threads für die Verarbeitung der Aktualisierungen von Änderungsereignissen >*
3. Starten Sie das Plug-in erneut.

Protokollkonfiguration und Tipp zur Fehlerbehebung

Integrationsprotokolle werden im Verzeichnis '/log' erstellt. Bei jedem Start des Integrations-Plug-ins wird ein neues Protokollverzeichnis mit der Zeitmarke als Name erstellt.

Protokollkonfiguration: `<IntegrationPlugin-Pfad>/resources/config/plugin.properties`

- fileSize = Größe der Protokolldatei

  Standardgröße ist 20 MB. Sie kann je nach Anforderung erhöht oder verringert werden.
- backupIndex = Zähler der Sicherungsdatei

  Standardanzahl ist 5.
- logLevel = Definition der Protokollebene für Integrations-Plug-in

  Standardwert ist "info". Unterstützte Protokollebenentypen sind "info" und “debug”.

Fehlerbehebung

Wenn während des Integrationsprozesses Fehler auftreten, können die folgenden Punkte überprüft werden.

1. Stellen Sie sicher, dass alle Voraussetzungen erfüllt sind.
2. Es kann auf die Protokolle für die Integrationsverarbeitung verwiesen werden.

- Wichtige Protokolle für das Plug-in

  `<IntegrationPlugin-Pfad>/log/20190128095240/plugin.log`
- Migrationsprotokolle

  `<IntegrationPlugin-Pfad>/log/20190128095240/migration.log`
- Protokolle für die Verwaltung von Änderungsereignissen

  `<IntegrationPlugin-Pfad>/log/20190128095240/changeEvent.log`
- Umsetzungsprotokolle

  `<IntegrationPlugin-Pfad>/log/20190128095240/tansform.log`
- Fehlerprotokolle. Es enthält alle Fehler im Integrations-Plug-in

  `<IntegrationPlugin-Pfad>//log/20190128095240/error.log`

Ausführung der Integration
Das Integrations-Plug-in kann bei der Übertragung der Anfangsdaten im Migrationsprozess zusammen mit dem Senden der nachfolgenden Aktualisierungen von Konfigurationselementen verwendet werden, die von TADDM an die ServiceNow-CMDB gesendet werden.

Migration
Im Migrationsprozess werden Daten aus der TADDM-Erkennung für die unterstützten Konfigurationselemente an die ServiceNow-CMDB migriert. Dies ist ein einmaliger Prozess.

Migrationsverwaltung

Zum Aktivieren des Migrationsprozesses setzen Sie die Eigenschaft "enableMigration" unter `/resources/config/plugin.properties` auf "true". Damit diese Änderung wirksam wird, muss das Tool/Integrations-Plug-in erneut gestartet werden.

Für die erneute Ausführung des Migrationsprozesses setzen Sie die Eigenschaft "runMigrationAgain" unter `/resources/config/plugin.properties` auf "true".

Der Migrationsstatus kann in der Datei "MigrationStatus.txt" angezeigt werden. Darin wird der Name des Konfigurationselements und der entsprechende Status angezeigt. Bei einer erneuten Ausführung der Migration wird diese Datei gelöscht.

Verwaltung von Änderungsereignissen
TADDM-Ereignisse werden generiert, wenn nach Ausführung der Erkennung in einer IT-Umgebung eine Konfigurationsänderung erkannt wird. Für die Generierung dieser dynamischen Ereignisse ist eine Vorkonfiguration erforderlich.

Konfiguration auf TADDM-Seite

Um Änderungsereignisse senden zu können, muss TADDM mit Angaben zu den Ereignisbehandlungssystemen konfiguriert werden, an die Änderungsereignisse gesendet werden sollen.

Vorgehensweise:

1. Um Änderungsereignisse zu aktivieren, muss in der Datei `$COLLATION_HOME/etc/collation.properties` die folgende Eigenschaft festgelegt sein: `com.ibm.cdb.omp.changeevent.enabled=true`.
2. Bearbeiten Sie die Datei `$COLLATION_HOME/etc/EventConfig.xml`, um zu konfigurieren, bei welchen Ressourcen Änderungen protokolliert und an welche Ereignisbehandlungssysteme die Ereignisse gesendet werden sollen.

   Informationen zu dem Format, in dem Angaben in der Datei 'EventConfig.xml' angegeben werden müssen, finden Sie unter EVENT LISTNERS.

   Geben Sie in der Datei 'EventConfig.xml' den Namen des unterstützten Konfigurationselements an und konfigurieren Sie einen Listener sowie einen zugehörigen Empfängerblock mit den erforderlichen Informationen.

```
Erstellen Sie einen Eintrag für das Konfigurationselement, in dem wie folgt angegeben
ist, welches Ereignis erforderlich ist:
                <!-- ComputerSystem -->
                <listener object="ComputerSystem" enabled="true">
                <alert recipient="taddm-snow-plugin-host" />
                <attribute name="guid" operator="not-equals">
                <value>
                0
               </value>
               </attribute>
               </listener>
               Template for TADDM-SNOW Event recipient with IP and Port
              <recipient name="taddm-snow-plugin-host" type="itm">
              <address>Plugin host IP</
address>
              <port>7575</
port>

              </recipient>
```

Referenz: https://www.ibm.com/support/knowledgecenter/en/SSPLFC_7.2.1/com.ibm.taddm.doc_721/AdminGuide/r_cmdb_changeevent_config.html

3. Für den Abruf der Aktualisierungsereignisse für das Konfigurationselement mit dem Attributnamen legen Sie in der Datei `$COLLATION_HOME/etc/collation.properties` die folgende Eigenschaft fest:`com.ibm.cdb.omp.changeevent.optimized.update=true`.
4. Zum Speichern und anschließenden Abrufen der Klassennamen des Konfigurationselements aus Ereignissen legen Sie in der Datei `$COLLATION_HOME/etc/collation.properties` die folgende Eigenschaft fest: `com.ibm.cdb.omp.changeevent.classnames.nonfriendly=true`.
5. Für den Abruf von Löschereignissen für die Konfigurationselemente, die manuell im TADDM-Datenmanagementportal gelöscht wurden, führen Sie die folgende Konfiguration aus:
   - Legen Sie in der Datei `$COLLATION_HOME/etc/collation.properties` die folgende Eigenschaft fest: `com.ibm.cdb.omp.changeevent.deletefromportal=true`
   - Geben Sie in der Datei `/opt/IBM/taddm/dist/etc/CiNamesForDeleteEvents.txt` die Namen der Konfigurationselemente für den Abruf von Löschereignissen aus dem Portal ein.
6. Fix Pack 8 Wenn Sie die Änderungsereignisse parallel zu der nachfolgenden oder aktuellen Erkennung senden möchten, legen Sie die folgende Eigenschaft in der Datei `$COLLATION_HOME/etc/collation.properties` fest: `com.ibm.cdb.omp.changeevent.GenerationType=later`.

   Anmerkung: Setzen Sie die Eigenschaft immer auf '**later**' (später), um das Änderungsereignis auszuführen. Der Standardwert dieser Eigenschaft ist '**inTransaction**'.

   Wenn der Standardwert '**inTransaction**' konfiguriert ist, sind nach einer umfangreichen Erkennung die nächsten oder zeitgleich ausgeführten Erkennungen blockiert, bis die Änderungsereignisse der gerade abgeschlossenen Erkennung beendet sind.

   Wenn Sie den Wert dieser Eigenschaft auf '**later**' setzen, wird die Erkennung bei der Beendigung als abgeschlossen markiert, ohne dass auf den Abschluss der Änderungsereignisse gewartet wird. Die Änderungsereignisse werden parallel zu aktuellen und nachfolgenden Erkennungen ausgeführt.
7. Starten Sie TADDM nach der Einstellung dieser Änderungen in den Dateien erneut.

Ereignisverarbeitung auf der Seite des Integrations-Plug-ins

Im Integrations-Plug-in werden alle vier Typen von Ereignissen verarbeitet, die von TADDM ausgelöst werden.

1. Erstellungsereignis: Entspricht den neuen Konfigurationselementen nach der TADDM-Erkennung.
2. Aktualisierungsereignis: Für eine Änderung in einem beliebigen Attribut desselben Konfigurationselements (nicht in gespeicherten Referenzen).
3. Löschereignis: Entspricht einem Konfigurationselement, das während der Erkennung oder manuell im TADDM-Datenmanagementportal gelöscht wird.

Für Einträge im ServiceNow-Datensatz wird ein vorläufiges Löschen vorgenommen. Das bedeutet, dass der Eintrag nicht ausdrücklich entfernt wird, sondern das vorhandenes, aber nicht verwendetes Attribut (Leasingvertrag) aus der Basistabelle des ServiceNow-Konfigurationselements verwendet wird, um den Status zu markieren.

4. Weitergegebenes Ereignis: Dies ist ein impliziertes Aktualisierungsereignis für Szenarios, in denen Referenzen zu Konfigurationselementen aktualisiert werden.

Script ausführen

Für die Ausführung der Erkennung muss das Script “plugin.sh” aus dem Verzeichnis 'IntegrationPlugin' ausgeführt werden. Das Script kann mit 3 verschiedenen Argumenten ausgeführt werden: 'start', 'status' und 'stop'. Hier finden Sie die Einzelheiten dazu:

Integrations-Plug-in starten

```
Option 1
  [Command] ./plugin.sh start
  start – parameter to start the Integration plugin
  TADDM User Id and Password – will be read from internal properties file

  [Command] ./plugin.sh start &
  start – parameter to start the Integration plugin
  TADDM User Id and Password – will be read from internal properties file
  & - for running the plugin in background

  Option 2:
  [Command] ./plugin.sh -u <Benutzername> start
  -u, --user <TADDM-Benutzer-ID>
  start – parameter to start the Integration plugin
 Password – to be entered on subsequent command line prompt (for safety
rea
sons)


  Option 3:
  [Command] ./plugin.sh -u <Benutzername> -p <Kennwort> start
  -u, --user <TADDM-Benutzer-ID>
  -p, --password <TADDM-Benutzerkennwort>
  start – parameter to start the Integration plugin
```

Integrations-Plug-in stoppen

```
Option 1:
  [Command] ./plugin.sh stop
  stop - parameter to gracefully stop the Integration plugin

  Option 2:
  [Command] ./plugin.sh -stop force
  stop - parameter to stop the Integration plugin
```

Status des Integrations-Plug-ins

```
Option 1:
  [Command] ./plugin.sh status
  status – parameter for the status of Integration plugin
```

Mögliche Fehlerszenarios

Wenn während der Ausführung des Integrations-Plug-ins ein Fehler auftritt, können die folgenden Punkte überprüft werden:

1. Überprüfen Sie, dass alle Voraussetzungen erfüllt sind.
2. Überprüfen Sie die Fehlerprotokolle für das Integrations-Plug-in unter `<IntegrationPlugin-Pfad>/log/<Zeitmarke>/error.log`.

Fehlerszenarios und zugehörige Lösungen

Dies sind die häufigsten Fehler, die während der Initialisierung des Integrations-Plug-ins auftreten können:

Verschlüsselungsausnahmebedingung

Ausnahmebedingung: Die API-Sitzung konnte nicht abgerufen werden. `com.collation.platform.util.FIPSEncryptionException: CTJOP0165E` "Beim Laden des TADDM-Verschlüsselungsschlüssels ist ein E/A-Fehler aufgetreten". security/TADDMSec.properties (Datei oder Verzeichnis nicht vorhanden).

Lösung:

1. Wenn das Integrations-Plug-in und TADDM auf der gleichen Maschine installiert sind, setzen Sie `<Integrationplugin-Pfad>/resources/taddm_snow/taddm.properties` als TADDM-Ausgangspfad fest.
2. Wenn das Integrations-Plug-in auf einer anderen Maschine installiert ist, überprüfen Sie, ob die Datei 'TADDMSec.properties' in der Position `<Integrationplugin-Pfad>/security`/etc vorhanden ist.
3. Ist dies nicht der Fall, kopieren Sie die Datei 'TADDMSec.properties' auf der TADDM-Maschine und suchen Sie sie an der Position `<Integrationplugin-Pfad>/security/etc` auf der Maschine, auf der das Integrations-Plug-in installiert ist.

Fehler wegen nicht gefundenen Berechtigungsnachweisen

Fehler: [TSI] Der Berechtigungsnachweis wurde nicht gefunden.

Lösung: Konfigurieren Sie die Zugriffsliste für ServiceNow-Berechtigungsnachweise.

Die Verbindung ist mit der Wiederholung des Zugriffs auf das Ziel fehlgeschlagen

Fehler: [TSI] Die Verbindung mit Wiederholungszähler 5 für das Ziel fehlgeschlagen.

Lösung: Melden Sie sich bei der TADDM Discovery Management Console an und überprüfen Sie, ob die korrekten ServiceNow-Berechtigungsnachweise unter Discovery > Access List (Erkennung > Zugriffsliste) konfiguriert sind und eine gültige Instanz-URL unter `/resources/taddm_snow/serviceNow.properties` konfiguriert ist.

Anhang A: In der Integration verwendete Eigenschaften

In der Integration verwendete Eigenschaften.

Plugin.properties

Tabelle 67.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
enableChange EventManagement.	true/false Standardwert=true	Bei 'true' wird die Verarbeitung von Änderungsereignissen aktiviert. Nach dem Ändern des Eigenschaftswerts ist kein Neustart des Plug-ins erforderlich.	J
enableMigration	true/false Standardwert=true	Bei 'true' wird die Verarbeitung der Migration von Konfigurationselementen aktiviert.	J
runMigrationAgain	true/false Standardwert = false	Bei 'true' wird die Migration für alle konfigurierten Konfigurationselemente erneut ausgeführt.	J

Tabelle 67. (Forts.)

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
migrationThread PoolSize	Empfohlener Wert >1 Standardwert=5	Geben Sie den ThreadPool-Zähler für die Verarbeitung der Migration von Konfigurationselementen an. Erhöhen Sie die Anzahl, falls dies für die Verarbeitung der Massendaten erforderlich ist.	J
changeEvent ThreadPoolSize	Empfohlener Wert >1 Standardwert=5	Geben Sie die ThreadPool-Anzahl für die Verarbeitung von Änderungsereignissen für Konfigurationselemente an.	J
sourceConnector	<Quelle> Unterstützt = TADDM	Aktuell unterstützter Quellentyp ist TADDM.	J
targetConnector	<Ziel> Unterstützt = SNOW	Aktuell unterstützter Zieltyp ist SNOW.	J
javaHome	<Java-Ausgangspfad> Standardwert = /opt/IBM/taddm/dist/ external/jdk-Linux-x86_64/jre	Festlegen des IBM-Ausgangspfads für die Kompilierung von Code, der in magischen Methoden geschrieben ist.	
useMagicMethod	true/false Standardwert = false	Festlegen von 'true' für die Ausführung von magischen Methoden, damit die umgewandelten Werte dynamisch aktualisiert werden.	
taddmSession RetryTime	<Zeit in Millisekunden> Standardwert: 2000	Legt die Dauer für die Sitzungswiederholung für Quellen- und Ziel-Connector fest. Die Zeit sollte Millisekunden angegeben werden.	J

Protokollkonfiguration

Tabelle 68.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
fileSize	<Dateigröße> Standardwert = 20 MB	Angabe der Dateigröße für Protokolldateien.	J
backupIndex	<Anzahl> Standardwert = 5	Gibt die Anzahl der Sicherungsdateien an.	J
logLevel	<Protokollebene> Standardwert = info	Gibt die Protokollebene an. Unterstützte Protokollebenen sind 'info' oder 'debug'.	J

Einstellungen für die ordnungsgemäße Beendigung

Tabelle 69.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
gracefulEvent ProcessingCount	<Warteschlangengröße> Standardwert = 5	Anzahl der anstehenden Ereignisse, die verarbeitet werden sollen, bevor die Threads ordnungsgemäß beendet werden	J

taddm.properties

Tabelle 70.

Eigenschaftsname	Zulässige Werte	Beschreibung	Verbindlich
taddmHost	IP	IP-Adresse des TADDM-Erkennungsservers.	J
taddmPort	Port Standardwert=-1	Port auf dem TADDM-Server, an den Abfragen gesendet werden müssen. Port 9433 wird für Abfragen verwendet. Wenn der Port im Code auf -1 gesetzt wird, wird er bedingt auf 9433 gesetzt.	J
taddmUserName	<Benutzer-ID>	Benutzername für den Zugriff auf TADDM.	J
taddmPassword	<Kennwort>	Kennwort für den Zugriff auf TADDM. Wird in verschlüsselter Form gespeichert.	J
taddmUseSSl	true/false Standardwert = false	Bei 'true' wird eine sichere Verbindung hergestellt.	N
taddmTrustStorePath	<Pfad>	Gibt den Pfad zum SSL-Zertifikat an.	N
taddmHome	<Pfad>	Geben Sie den Pfad '$COLLATION_HOME' auf der TADDM-Hostmaschine an. Geben Sie '<Integrations-Plug-in-Pfad>/security/etc' an , wenn das Plug-in unabhängig ausgeführt wird. .	J
tlsVersion	TLS-Version Standardwert: TLSv1.0	Geben Sie den gleichen Wert für die TLS-Version ein, der auch in TADDM festgelegt ist (z. B. TLSv1.0, TLSv1.1, TLSv1.2)	N

Einstellungen für Änderungsereignisse

Tabelle 71.			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
listenIp	IP	IP der Maschine, die für Änderungsereignisse empfangsbereit ist.	J
listenPort	<Portnummer>	Port der Maschine, die für Änderungsereignisse empfangsbereit ist.	J

ServiceNow.properties
ServiceNow-Konfigurationen

Tabelle 72.			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
serviceNow InstanceUrl	URL	URL der ServiceNow-Instanz.	J
serviceNow OAuthUrl	URL	URL von OAuth zum Generieren von Zugriffstoken.	J
discoverySource	<Erkennungsquelle> Standardwert: IBM TADDM	Festlegen des Typs der Erkennungsquelle für die Nutzdaten.	J
relationTable	<Tabelle> Standardwert: cmdb_rel_type	ServiceNow-Tabellenname mit den Beziehungstypen für alle ServiceNow-CMDB-Beziehungen.	J
refreshToken Lifespan	<Anzahl>	Lebenszyklus des Aktualisierungstoken in Sekunden.	J
serviceNow.attribute. softdelete.status	<Status> Standardwert: InActive	Markierung eines Datensatzes als inaktiv bei einem vorläufigen Löschen.	J
serviceNow.attribute. softdelete.name	<Attributname> Standardwert: lease_id	Konfiguration des ServiceNow-Attributs zum Speichern des Status für das vorläufige Löschen.	J
serviceNow.attribute. guid.name	<Attributname> Standardwert: justification	Konfiguration des ServiceNow-Attributs zum Speichern der ID für das TADDM-Konfigurationselement (GUID).	J
securityProtocol	<TLS-Version> Standardwert: TLSv1.2	Geben Sie den gleichen Wert für die TLS-Version an, der auch in TADDM festgelegt ist. Beispiel: TLSv1.0, TLSv1.1, TLSv1.2.	J

Proxy-Einstellungen

Tabelle 73.			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
enableProxy	True/false	Bei 'true' wird das Proxy beim Herstellen einer Verbindung zu ServiceNow aktiviert.	N
proxyHost	<IP oder Host>	Proxy-IP.	N
proxyPort	<Port>	Proxy-Port.	N
proxyUserName	<Benutzername>	Benutzername für den Zugriff über eine Proxy-Verbindung.	N
proxyPassword	<Kennwort>	Festlegen des Proxy-Kennworts. Wird in verschlüsselter Form gespeichert. Ein verschlüsseltes Kennwort wird in der Eigenschaft `com.collation.integrationplugin.target.proxy.password` in der Datei `collation.properties` generiert.	N

Collation.properties
Collation.properties für TADDM.

Konfiguration von Änderungsereignissen

Tabelle 74.			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
com.ibm.cdb.omp.changeevent.enabled	True/false Standardwert: false	Bei 'true' kann das Änderungsereignismodul Änderungsereignisse senden. Nach dem Ändern des Eigenschaftswerts ist ein Neustart des Plug-ins erforderlich.	J
com.ibm.cdb.omp.changeevent.optimized.update	True/false Standardwert: false	Eigenschaft für die Konsolidierung von Aktualisierungsereignissen für mehrere Attribute des gleichen Modellobjekts. Nach dem Ändern des Eigenschaftswerts ist ein Neustart des Plug-ins erforderlich.	J
com.ibm.cdb.omp.changeevent.classnames.nonfriendly	True/false Standardwert: false	Diese Eigenschaft gibt anstelle des nicht benutzerfreundlichen Klassennamens den Klassennamen des Konfigurationselements zurück. Nach dem Ändern des Eigenschaftswerts ist ein Neustart des Plug-ins erforderlich.	J

Tabelle 74. (Forts.)			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
com.ibm.cdb. omp.changeevent. deletefromportal	True/false Standardwert: false	Eigenschaft für das Senden von Löschereignissen, wenn das Konfigurationselement aus dem Portal entfernt wird. Nach dem Ändern des Eigenschaftswerts ist ein Neustart des Plug-ins erforderlich.	J

Kennworteigenschaft

Tabelle 75.			
Eigenschaftsname	**Zulässige Werte**	**Beschreibung**	**Verbindlich**
com.collation. integrationplugin. taddm.password	<Kennwort>	TADDM-Benutzerkennwort. Wird in verschlüsselter Form gespeichert.	J
com.collation. integrationplugin. target.proxy.password	<Kennwort>	Proxy-Kennwort für Zielverbindung. Wird in verschlüsselter Form gespeichert. Das verschlüsselte Kennwort wird generiert, indem `encryptprops.sh` ausgeführt und dieses Kennwort in `serviceNow.properties` auf `proxyPassword property` gesetzt wird.	J

Anhang B: Hilfe für den Parameter des Integrations-Plug-ins bei verschiedenen Modi

./plugin.sh – Das Script kann mit 3 verschiedenen Argumenten ausgeführt werden: 'start', 'status' und 'stop'.

Argument: START

```
Syntax: ./plugin.sh -u <Benutzername> -p <Kennwort> start [-h]
Dabei gilt:
-u <Benutzername>    Name des TADDM-Benutzers bereitstellen.
-p <Kennwort>    Kennwort des TADDM-Benutzers bereitstellen. Aus Gründen der Sicherheit nicht
empfohlen.
-h,--help        show help.

Example to check Integration plugin execution
Start:
./plugin.sh start
```

Argument: STATUS

```
Integration Plugin status displays CI name, their running status and processed
number of records and count of failure and success of records.

./plugin.sh status
Migration Enabled=true
Change Event Management=true
Migration CI Status
ComputerSystem=COMPLETED|TOTAL=7|SUCCESS=4|FAILED=3
AppServer=COMPLETED|TOTAL=4|SUCCESS=4|FAILED=0
StorageVolume=COMPLETED|TOTAL=14|SUCCESS=14|FAILED=0
StoragePool=COMPLETED|TOTAL=0|SUCCESS=0|FAILED=0
DiskDrive=COMPLETED|TOTAL=6|SUCCESS=6|FAILED=0
FileSystem=COMPLETED|TOTAL=9|SUCCESS=9|FAILED=0
```

```
AppServerCluster=COMPLETED|TOTAL=0|SUCCESS=0|FAILED=0
ComputerSystemCluster=COMPLETED|TOTAL=0|SUCCESS=0|FAILED=0

Plugin Status: Running
```

Argument: STOP

```
./plugin.sh stop
Integration Plugin stopped gracefully

./plugin.sh -stop force
Integration Plugin stopped forcefully
```

Anhang C: Fehlercodes und Beschreibung

REST-Nachrichten, die an die ServiceNow-Instanz gesendet werden, geben die unten angegebenen HTTP-Antwortcodes zurück.

Tabelle 76.

Statuscode	Nachricht	Details
200	Erfolg	Erfolg mit Antworthauptteil.
201	Erstellt	Erfolg mit Antworthauptteil.
204	Erfolg	Erfolg ohne Antworthauptteil.
400	Ungültige Anforderung	Die Anforderungs-URI stimmt nicht mit den APIs im System überein oder die Operation ist aus unbekannten Gründen fehlgeschlagen. Dieser Fehler kann auch durch ungültige Header verursacht worden sein.
401	Nicht berechtigt	Der Benutzer ist nicht für die Verwendung der API berechtigt.
403	Nicht zulässig	Die angeforderte Operation ist für den Benutzer nicht zulässig. Dieser Fehler kann auch durch ACL-Fehler oder Einschränkungen der Geschäftsregeln oder Datenrichtlinien verursacht werden.
404	Nicht gefunden	Die angeforderte Ressource wurde nicht gefunden. Möglicherweise wurde dieser Fehler durch eine ACL-Einschränkung verursacht oder die Ressource ist nicht vorhanden.
405	Methode nicht zulässig	Die HTTP-Aktion ist für die angeforderte REST-API nicht zulässig oder wird von keiner API unterstützt.
406	Nicht akzeptabel	Der Endpunkt unterstützt nicht das Antwortformat, das im Accept-Header der Anforderung angegeben ist.
415	Nicht unterstützter Datenträgertyp	Der Endpunkt unterstützt nicht das Format des Anforderungshauptteils.

Anhang D: Liste der unterstützten Konfigurationselemente

Name des Konfigurationselements und unterstützter Typ:

1. ComputerSystem

Typ:

- WindowsComputerSystem
- NetwareUnitaryComputerSystem
- SunSPARCUnitaryComputerSystem

- HpUxUnitaryComputerSystem
- AixUnitaryComputerSystem
- StorageSubSystem
- UnitaryComputerSystem
- VmwareUnitaryComputerSystem
- Fix Pack 8 ZSeriesComputerSystem

2.AppServer

Typ:

- SqlServer
- OracleInstance
- Db2Instance
- SybaseServer
- WebLogicAdminServer
- WebsphereServer
- OracleAppHTTPServer
- VirtualCenter
- MySql
- JavaServer
- KVM
- IBMTivoliMonitoringAgent
- Fix Pack 8 PostgreSQL
- Fix Pack 8 Tomcat
- Fix Pack 8 CICSRegion
- Fix Pack 8 ApacheServer
- Fix Pack 8 IIsWebService

3. AppServerCluster

Typ:

- ComputerSystemCluster

4. StorageVolume

5. StoragePool

6. DiskDrive

7. FileSystem

Typ:

- NFSFileSystem
- SMBFileSystem
- Fix Pack 8 UnixFileSystem
- Fix Pack 8 WindowsFileSystem

8. IpInterface

9. L2Interface

10. Funktion

Typ:

- IpV4Router
- IpV6Router
- IpDevice
- Bridge
- Fix Pack 8 Vlan
- Fix Pack 8 VMWareVirtualSwitch

Fix Pack 6 Datenzugriffsportal

Für den Zugriff auf das TADDM-Datenzugriffsportal benötigt der Benutzer eine Rolle mit Anzeigeberechtigung.

Fix Pack 6 Benutzer mit der Rolle mit Anzeigeberechtigung können nur auf das Portal zugreifen. Benutzer werden beendet.

Rolle für Anzeigeberechtigte erstellen

Zur Bereitstellung des Zugriffs durch Benutzer können Sie die neue Rolle für Anzeigeberechtigte mit der Berechtigung '*Lesezugriff*' im TADDM-Datenmanagementportal erstellen.

Vorgehensweise

Anmerkung: Die Rolle für Anzeigeberechtigte ist die Standardeinstellung in neuen TADDM-Installationen; bei der Fixpack-Migration müssen die neuen Rollen im TADDM-Datenmanagementportal erstellt werden.

Gehen Sie folgendermaßen vor, um die Rolle für Anzeigeberechtigte zu erstellen:

1. Öffnen Sie das **Datenmanagementportal**.
2. Klicken Sie auf **Administration** > **Rollen**. Eine Liste mit Rollen wird angezeigt.
3. Klicken Sie auf **Rolle erstellen**. Das Fenster 'Rolle erstellen' wird angezeigt.
4. Geben Sie den Namen '**viewer**' (Anzeigeberechtigter) für die neue Rolle ein und wählen Sie anschließend die Berechtigung **Lesezugriff** aus.
5. Klicken Sie auf **Rolle erstellen**. Die Liste mit den Rollen wird erneut angezeigt und enthält jetzt die neue Rolle.

Rolle für Anzeigeberechtigte zuordnen

Nachdem die neue Rolle für Anzeigeberechtigte erstellt wurde, ordnen Sie diese den TADDM-Benutzern zu, damit diese auf das Datenzugriffsportal zugreifen können.

Vorgehensweise

Gehen Sie zum Zuordnen der Rolle für Anzeigeberechtigte zu einem vorhandenen TADDM-Benutzer folgendermaßen vor:

1. Öffnen Sie das **Datenmanagementportal**.
2. Klicken Sie auf **Administration** > **Benutzer**. Eine Liste mit Benutzern wird angezeigt.
3. Klicken Sie auf den **Benutzernamen**, den Sie bearbeiten möchten, und klicken Sie anschließend auf **Bearbeiten**. Die Angaben zu diesem Benutzer werden angezeigt.
4. Ändern Sie die Rolle in '**viewer**' (Anzeigeberechtigter).
5. Um die **Eigenschaften zur Rollenzuordnungsänderung** anzuwenden, klicken Sie auf '**Rolle ändern**'.

Anmerkung: Dieselben Schritte müssen für neue Benutzer ausgeführt werden.

Informationen zur Erstellung eines neuen Benutzers finden Sie unter https://www.ibm.com/support/knowledgecenter/SSPLFC_7.3.0/com.ibm.taddm.doc_7.3/UserGuide/t_cmdb_user_create.html

Datenbank konfigurieren

Das Datenzugriffsportal wird standardmäßig mit der TADDM-Datenbank konfiguriert. Sie können die konfigurierte Datenbank gemäß den Voraussetzungen ändern.

Vorgehensweise

Gehen Sie zum Ändern der Datenbank folgendermaßen vor:

1. Geben Sie die folgenden Werte in die Datei 'utility.properties' im Verzeichnis */opt/IBM/taddm/dist/apps/dap/etc* an.
   - **Database User Name:** (Datenbankbenutzername) *com.utility.db.user*
   - **Database User Password:** (Kennwort des Datenbankbenutzers) *com.utility.db.password*
   - **JDBC URL of Database to use:** (JDBC-URL der zu verwendenden Datenbank) *com.utility.db.url*
   - **JDBC Driver Name:** (Name des JDBC-Treibers): Die JAR-Datei mit dem JDBC-Treiber sollte im LIBS-Pfad der Anwendung *om.utility.db.driver* vorhanden sein.
   - **Database Type:** (Datenbanktyp) *com.utility.db.type*
2. Speichern Sie die Änderungen.
3. Starten Sie TADDM erneut.

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 andere 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 Défense
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 ohne weitere Mitteilung jederzeit 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
2Z4A/101
11400 Burnet Road
Austin, TX 78758
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 im Dokument aufgeführten 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 er-

© Copyright IBM Corp. 2006, 2020

zielten 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.

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

Wird dieses Dokument als Softcopy (Book) angezeigt, sind Fotografien oder Farbabbildungen möglicherweise nicht sichtbar.

Marken

IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der International Business Machines Corporation. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Herstellern sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite "Copyright and trademark information" unter http://www.ibm.com/legal/copytrade.shtml.

Java sowie alle auf Java basierenden Marken und Logos sind in den USA und/oder anderen Ländern Marken der der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

Linux ist in den USA und/oder anderen Ländern eine eingetragene Marke von Linus Torvalds.

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

UNIX ist in den USA und anderen Ländern eine eingetragene Marke von The Open Group.

Weitere Unternehmens-, Produkt- oder Servicenamen können Marken anderer Hersteller sein.