

HP Business Availability Center

für Windows- und Solaris-Betriebssysteme

Softwareversion: 8.00

TransactionVision-Bereitstellung

Dokument-Releasedatum: Januar 2009

Software-Releasedatum: Januar 2009



Rechtliche Hinweise

Garantie

Die Garantiebedingungen für Produkte und Services von HP sind in der Garantieerklärung festgelegt, die diesen Produkten und Services beiliegt. Keine der folgenden Aussagen kann als zusätzliche Garantie interpretiert werden. HP haftet nicht für technische oder redaktionelle Fehler oder Auslassungen.

Die hierin enthaltenen Informationen können ohne vorherige Ankündigung geändert werden.

Eingeschränkte Rechte

Vertrauliche Computersoftware. Gültige Lizenz von HP für den Besitz, Gebrauch oder die Anfertigung von Kopien erforderlich. Entspricht FAR 12.211 und 12.212; kommerzielle Computersoftware, Computersoftwareokumentation und technische Daten für kommerzielle Komponenten werden an die US-Regierung per Standardlizenz lizenziert.

Fremdanbieter-Websites

Zum Bereitstellen zusätzlicher Informationen verwendet HP Links zu externen Websites von Fremdanbietern. Die Inhalte und die Verfügbarkeit dieser Seiten können ohne vorherige Ankündigung geändert werden. Für die Inhalte und die Verfügbarkeit dieser Seiten übernimmt HP keinerlei Haftung über Garantien.

Urheberrechtshinweise

© Copyright 2000 - 2009 Hewlett-Packard Development Company, L.P.

Marken

TransactionVision® ist eine eingetragene Marke der Hewlett-Packard Company.

Java™ ist eine US-Marke der Sun Microsystems, Inc.

Microsoft® und Windows® sind US-Marken der Microsoft Corporation.

Oracle® ist eine eingetragene Marke der Oracle Corporation und/oder der zugehörigen Tochtergesellschaften.

UNIX® ist eine eingetragene Marke von The OpenGroup.

Dokumentationsaktualisierungen

Die Titelseite dieses Handbuchs enthält die folgenden Informationen:

- Software-Versionsnummer zur Angabe der Software-Version.
- Dokument-Releasedatum, das sich mit jeder Aktualisierung des Dokuments ändert.
- Software-Releasedatum zur Angabe des Releasedatums der Software-Version.

Um nach Aktualisierungen des Dokuments zu suchen, oder um zu überprüfen, dass Sie die aktuellste Version des Dokuments verwenden, wechseln Sie zu:

<http://h20230.www2.hp.com/selfsolve/manuals>

Für die Anmeldung an dieser Website benötigen Sie einen HP Passport. Um sich für eine HP Passport-ID zu registrieren, wechseln Sie zu:

<http://h20229.www2.hp.com/passport-registration.html>

Alternativ können Sie auf den Link **New user registration** (Neue Benutzer registrieren) auf der HP Passport-Anmeldeseite klicken.

Wenn Sie sich beim Support-Service eines bestimmten Produkts registrieren, erhalten Sie ebenfalls aktualisierte Softwareversionen und überarbeitete Ausgaben der zugehörigen Dokumente. Weitere Informationen erhalten Sie bei Ihrem HP-Kundenbetreuer.

Support

Die HP-Website zur Software-Unterstützung finden Sie unter:

<http://www.hp.com/go/hpsoftwaresupport>

Auf dieser Website finden Sie Kontaktinformationen und Details zu Produkten, Services und Supportleistungen von HP Software.

Der Online-Support von HP Software bietet Kunden mithilfe interaktiver technischer Support-Werkzeuge die Möglichkeiten, ihre Probleme intern zu lösen. Als Kunde mit Supportvertrag stehen Ihnen auf der HP-Website zur Software-Unterstützung folgende Optionen zur Verfügung:

- Suchen nach interessanten Wissensdokumenten
- Absenden und Verfolgen von Support-Fällen und Erweiterungsanforderungen
- Herunterladen von Software-Patches
- Verwalten von Supportverträgen
- Nachschlagen von HP-Supportkontakten
- Einsehen von Informationen über verfügbare Services
- Führen von Diskussionen mit anderen Softwarekunden
- Suchen und Registrieren für Softwareschulungen

Bei den meisten Support-Bereichen ist die Registrierung und Anmeldung als HP-Passport-Benutzer erforderlich. Einige Angebote setzen den Abschluss eines Supportvertrags voraus.

Weitere Informationen über die für den Zugriff erforderlichen Voraussetzungen erhalten Sie unter:

http://h20230.www2.hp.com/new_access_levels.jsp

Um sich für eine HP Passport-ID zu registrieren, wechseln Sie zu:

<http://h20229.www2.hp.com/passport-registration.html>

Inhalt

Willkommen bei diesem Handbuch	13
Wie dieses Handbuch aufgebaut ist	13
Wer dieses Handbuch lesen sollte	14
TransactionVision-Dokumentation	14
Weitere Onlineressourcen	16
Dokumentationsaktualisierungen.....	17

TEIL I: EINFÜHRUNG IN TRANSACTIONVISION

Kapitel 1: Einführung in TransactionVision	21
Informationen zu TransactionVision.....	21
Architekturdiagramm	23
TransactionVision in der Bereitstellungsumgebung	25
Installationspakete	27
Abwärtskompatibilität.....	28
Aktualisieren von früheren Versionen	29

Kapitel 2: Überprüfen der Systemanforderungen	31
Unterstützte Analyzer-Plattformen	33
Unterstützte TransactionVision UI/Job Server-Plattformen	33
Unterstützte Datenbankmanagementsysteme.....	34
Unterstützte Provider für Message Oriented Middleware.....	35
Unterstützte Plattformen für WebSphere MQ-Sensoren	35
Unterstützte WebSphere Message Broker-Konfigurationen.....	37
Unterstützte Plattformen für Servlet-Sensoren	38
Unterstützte Plattformen für EJB-Sensoren.....	39
Unterstützte Plattformen für JMS-Sensoren.....	40
Unterstützte Plattformen für JDBC-Sensoren	41
Unterstützte Plattformen für CICS-Sensoren.....	41
Unterstützte Plattformen für BEA Tuxedo-Sensoren	42
Unterstützte Plattformen für NonStop TMF-Sensoren	42
Unterstützte Plattformen für .NET-Agents.....	42
Unterstützte Browserkonfigurationen.....	42
LDAP-Unterstützung	43
Java-Unterstützung.....	43
Flash Player-Unterstützung	43
Lokalisierungs- und Internationalisierungsunterstützung	43

TEIL II: INSTALLATION UND KONFIGURATION VON ANALYZER

Kapitel 3: Vorbereiten der Installation von TransactionVision Analyzer	47
Informationen zu TransactionVision Analyzer.....	47
TransactionVision Analyzer in der Bereitstellungsumgebung.....	48
Übersicht über die Analyzer-Installation und -Konfiguration.....	48
Kapitel 4: Installieren von Analyzer unter Windows	51
Starten des Analyzer-Installationsprogramms unter Windows	51
Erstinstallation	52
Upgrade-Installation.....	53
Deinstallieren von Analyzer.....	54
Kapitel 5: Installieren von Analyzer auf UNIX-Plattformen	57
Installationsdateien	57
Starten des Analyzer-Installationsprogramms unter UNIX	58
Erstinstallation	59
Upgrade-Installation.....	60
Deinstallieren von Analyzer.....	61

Kapitel 6: Konfigurieren von Datenbanken	63
Informationen zum Konfigurieren von Datenbanken	63
Unterstützte Datenbanken	64
Konfigurieren des Datenbankzugriffs.....	64
Festlegen von DB2-Variablen	66
Festlegen von Oracle-Variablen.....	68
DBMS-Leistungsoptimierung	68
DBMS-Speicherplatzanforderungen	72
Konfigurieren von Datenbanken für Unicode-Daten	73
Kapitel 7: Konfigurieren von Analyzer	75
Informationen zum Konfigurieren von Analyzer	75
Dateien, die durch "TVisionSetupInfo" geändert werden	76
Informationen, die von "TVisionSetupInfo" angefordert werden	76
Ausführen von "TVisionSetupInfo"	83
Verwalten von Analyzer	89
Weitere Analyzer-Konfigurationsschritte.....	89
Verkleinern der Ereignisdatenbank.....	102
Kapitel 8: Konfigurieren der Analyzer-Protokollierung	105
Protokolldateien	105
Umlaufprotokollierung	106
Verwenden von Windows- und UNIX-Systemprotokollen	108
Aktivieren der SMTP-Protokollierung	109
Aktivieren der SNMP-Protokollierung.....	111
Aktivieren der JMS-Protokollierung	112

TEIL III: INSTALLATION UND KONFIGURATION VON UI/JOB SERVER

Kapitel 9: Vorbereiten der Installation von TransactionVision UI/Job Server	117
Informationen zu TransactionVision UI/Job Server	117
TransactionVision UI/Job Server in der Bereitstellungsumgebung.....	117
Kapitel 10: Installieren von UI/Job Server auf UNIX-Plattformen	119
Installationsdateien	119
Starten des UI/Job Server-Installationsprogramms unter UNIX	120
Erstinstallation	121
Upgrade-Installation.....	121
Deinstallieren von UI/Job Server.....	123

Kapitel 11: Installieren von UI/Job Server unter Windows	125
Starten des UI/Job Server-Installationsprogramms	
unter Windows	125
Erstinstallation	126
Deinstallieren von UI/Job Server.....	127
Kapitel 12: Konfigurieren von UI/Job Server	129
Informationen zum Konfigurieren von UI/Job Server.....	129
Dateien, die durch "TVisionSetupInfo" geändert werden	130
Informationen, die von "TVisionSetupInfo" angefordert	
werden.....	130
Ausführen von "TVisionSetupInfo"	131
Verwalten von UI/Job Server.....	137
Kapitel 13: Konfigurieren der UI/Job Server-Protokollierung	139
Protokolldateien	139
Umlaufprotokollierung	139
Ablaufprotokollierung.....	141
Verwenden von Windows- und UNIX-Systemprotokollen	142

TEIL IV: INSTALLATION UND KONFIGURATION VON SENSOREN UND AGENTS

Kapitel 14: Vorbereiten der Installation von	
TransactionVision-Sensoren	147
Applikationen, die überwacht werden können	148
Verfügbare Sensor- und Agenttypen	149
Kapitel 15: Installieren und Konfigurieren des Java-Agents.....	155
Informationen zum Installieren und Konfigurieren	
des Java-Agents	156
Installieren und Konfigurieren des Java-Agents unter	
Windows	157
Installieren und Konfigurieren des Java-Agents unter UNIX	172
Unbeaufsichtigte Installation des Java-Agents.....	183
Ausführen des JRE-Instrumentierers	184
Konfigurieren der Applikationsserver	195
Konfigurieren von Nachrichtensystemprovidern	195
Konfigurieren von benutzerdefinierten Benutzerereignissen	197

Kapitel 16: Installieren von WebSphere MQ- und User Event-Sensoren unter Windows	199
Starten des Installationsprogramms unter Windows.....	199
Erstinstallation	200
Upgrade-Installation.....	201
Bearbeiten der Installation	203
Deinstallieren von Sensoren	205
Kapitel 17: Installieren von WebSphere MQ- und User Event-Sensoren auf UNIX-Plattformen	207
Installieren von Sensoren.....	207
Deinstallieren von Sensoren	210
Kapitel 18: Installieren von Sensoren unter i5/OS	213
Starten des Installationsprogramms unter i5/OS.....	213
Kapitel 19: Installieren und Konfigurieren von Sensoren unter z/OS	215
Informationen zu Sensoren in der z/OS-Umgebung.....	215
Übersicht über die Basiskomponenteninstallation	216
Basiskomponenteninstallation – Verfahrensbeschreibung.....	217
Konfigurieren der SLD-Sensorkomponenten unter z/OS: CICS, WebSphere MQ Batch und WebSphere MQ IMS	224
Konfigurieren der SLM-Sensorkomponenten unter z/OS: WebSphere MQ CICS Bridge und WebSphere MQ IMS Bridge	226
Hintergrundinformationen: WebSphere MS-Sensor für CICS.....	229
Konfigurieren von SLMC für CICS.....	230
Hintergrundinformationen: WebSphere MQ IMS Bridge-Sensor	231
Kapitel 20: Konfigurieren des CICS-, WMQ Batch- und WMQ IMS Bridge-Sensors	233
Übersicht	233
Allgemeine Sensorkomponenten	234
Befehle für den CICS-Sensor	236
Sensorbetrieb	243
Überlegungen zur Pufferwarteschlange	245
Startprozedur für TransactionVision Manager.....	247
Startprozedur für den Sensortreiber	248

Kapitel 21: Installieren und Konfigurieren von Sensoren in BEA Tuxedo	251
Vorbereiten der Installation	251
Ausführen der Installation	252
Erneutes Binden des Tuxedo-Sensors	253
Deinstallieren von Sensoren	253
Kapitel 22: Installieren und Konfigurieren des .NET-Agents	259
Informationen zum Installationsprogramm des .NET-Agents.....	260
Installieren des .NET-Agents	260
Konfigurieren des .NET-Agent.....	267
Erneutes Starten von IIS	275
Bestimmen der Version des .NET-Agent	276
Deinstallieren des .NET-Agents	276
SSL-Konfiguration für .NET-Agents.....	277
Kapitel 23: Installieren und Konfigurieren von Sensoren in NonStop TMF	279
Informationen zum NonStop TMF-Sensor.....	279
Vorbereiten der Installation	280
Installieren des NonStop TMF-Sensors.....	280
Starten/Beenden	282
Konfigurieren des NonStop TMF-Sensors	283
Kapitel 24: Konfigurieren von WebSphere MQ-Sensoren	285
Konfigurieren der WebSphere MQ-Sensorbibliothek	286
Konfigurieren des Exit-Sensors für WebSphere MQ-APIs	298
WebSphere MQ-Sensoren und FASTPATH_BINDING	306
Verwenden von Sensoren mit WebSphere MQ-Beispielen.....	306
Überwachen von WebSphere MQ-Clientapplikationen	307
Verwenden des WebSphere MQ IMS Bridge-Sensors	313
Verwenden des WebSphere Business Integration-Sensors.....	320
Kapitel 25: Konfigurieren des Proxy-Sensors	323
Informationen zum Proxy-Sensor	323
Applikationsanforderungen	324
Aktivieren des Proxy-Sensors	324
Konfigurieren der Proxydefinitionsdatei	325
Konfigurieren der Benutzeroberfläche	328

Kapitel 26: Konfigurieren der Agent- und Sensorprotokollierung	329
Protokolldateien	329
Umlaufprotokollierung	330
Ablaufprotokollierung	332
Konfigurieren getrennter Protokolldateien für mehrere Sensorinstanzen	333
Verwenden von Windows- und UNIX-Systemprotokollen	334

TEIL V: SICHERHEIT

Kapitel 27: Sicherheit	339
Informationen zur Sicherheit in TransactionVision	339
SSL-Konfiguration für TransactionVision	340
Verwaltung von TransactionVision-Benutzerrechten	342
Standardrollen	346
TransactionVision-Authentifizierung und Lightweight Single Sign-On	347
Schützen der TransactionVision-Konfigurationsdateien	348
Schützen von TransactionVision Analyzer	351
Schützen der TransactionVision-Datenbank	353

TEIL VI: ANHÄNGE

Anhang A: Referenz der Dienstprogramme	359
CreateSqlScript	360
DB2RunStats	363
DB2Test	365
MigrateDB	366
nanny	367
OracleRunStats	370
OracleTest	372
PassGen	374
rebind_sensor	375
rebind_tux_sensor	376
runSupportSnapshot	378
ServicesManager	381
SetupModule	385
SQLServerTest	386
TVisionSetupInfo	387

Anhang B: Konfigurationsdateien	391
Analyzer.properties.....	392
CacheSize.properties.....	398
Database.properties	399
JobManager.properties	403
Sensor.properties	403
SensorConfiguration.xml	403
Setup.properties.....	405
StatisticsCache.properties.....	406
UI.properties	407
Anhang C: Datenbankmigration.....	409
Dauer und Speicherplatzanforderungen	409
Deaktivieren nicht verwendeter Integrationsspalten	410
Migration benutzerdefinierter Datenbankschemas	410
Datenbankmigration – Technische Details.....	410
Optimieren von TransactionVision in Umgebungen ohne Integration	413
Anhang D: Zusätzliche z/OS-Einstellungen.....	415
RACF-Autorisierungen.....	415
Firewalleinstellungen	418
MIPS-Anforderungen.....	418
Anhang E: Agents für die uCMDB-Discovery.....	419
Installieren und Konfigurieren des Agents für die uCMDB-Discovery	420
Komponenten und Betrieb des Agents für die uCMDB-Discovery	435
Sicherheitsanforderungen des Agents für die uCMDB-Discovery	437
Zusammenfassung der uCMDB-Befehle.....	441
Konsolenmeldungen des Agents für uCMDB-Mainframe-Dienste	443
Fehlermeldungen bei der uCMDB-Discovery unter z/OS	451
Index.....	465

Willkommen bei diesem Handbuch

Willkommen beim Handbuch zur Bereitstellung von HP TransactionVision. In diesem Handbuch wird Ihnen TransactionVision vorgestellt, Sie erhalten Informationen zu den ersten Schritten, eine Beschreibung der Konfiguration und der Installation von Servern und Komponenten sowie die Details zum Upgrade.

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Wie dieses Handbuch aufgebaut ist auf Seite 13
- ▶ Wer dieses Handbuch lesen sollte auf Seite 14
- ▶ TransactionVision-Dokumentation auf Seite 14
- ▶ Weitere Onlineressourcen auf Seite 16
- ▶ Dokumentationsaktualisierungen auf Seite 17

Wie dieses Handbuch aufgebaut ist

Dieses Handbuch umfasst die folgenden Kapitel/Teile:

- | | |
|---------|---|
| Teil I | Einführung in TransactionVision
Gibt eine Einführung in TransactionVision und bietet eine Übersicht über die TransactionVision-Plattform und -Komponenten. |
| Teil II | Installation und Konfiguration von Analyzer
Beschreibt die Installation und Konfiguration von TransactionVision Analyzer. |

Willkommen bei diesem Handbuch

- Teil III Installation und Konfiguration von UI/Job Server
Beschreibt die Installation und Konfiguration von UI/Job Server.
- Teil IV Installation und Konfiguration von Sensoren und Agents
Beschreibt die Installation und Konfiguration von TransactionVision-Sensoren und -Agents.
- Teil V Sicherheit
Beschreibt das Sichern der TransactionVision-Komponenten.
- Teil VI Anhänge
Dienstprogramme, Konfigurationsdateien und andere Informationen, die mit TransactionVision im Zusammenhang stehen.

Wer dieses Handbuch lesen sollte

Dieses Handbuch richtet sich an die folgenden TransactionVision-Benutzer:

- Applikationsentwickler oder -konfiguratoren
- System- oder Instanzadministratoren
- Datenbankadministratoren

Die Leser dieses Handbuchs sollten gewisse Kenntnisse über die Applikationsentwicklung in Unternehmen haben und über sehr gute Fähigkeiten hinsichtlich der Unternehmenssystem- und Datenbankverwaltung verfügen.

TransactionVision-Dokumentation

Die TransactionVision-Dokumentation enthält Informationen zum Verwenden der TransactionVision-Applikation von Business Availability Center und zum Entwickeln und Verwalten der TransactionVision-spezifischen Komponenten in der Business Availability Center-Bereitstellungsumgebung.

Zur TransactionVision-Dokumentation gehört Folgendes:

- ▶ Im *Handbuch zur Bereitstellung von HP TransactionVision* wird die Installation und Konfiguration der TransactionVision-spezifischen Komponenten in der Business Availability Center-Bereitstellungsumgebung beschrieben. Dieses Handbuch ist als PDF-Datei in der Dokumentationsbibliothek von Business Availability Center verfügbar.
- ▶ Im Handbuch *Using TransactionVision Guide* wird beschrieben, wie TransactionVision eingerichtet und konfiguriert wird, um Transaktionen zu verfolgen. Darüber hinaus wird erläutert, wie Reports und Topologien von Geschäftstransaktionen angezeigt und angepasst werden. Dieses Handbuch ist als TransactionVision-Portal oder als PDF-Datei in der Online-Dokumentationsbibliothek von Business Availability Center verfügbar.
- ▶ Das Handbuch *TransactionVision Planning Guide* enthält wichtige Informationen zur Dimensionierung und Planung neuer Installationen. Dieses Handbuch kann von der Website mit den Produkthandbüchern für HP Software heruntergeladen werden.

Zusätzliche TransactionVision-Dokumentationen finden Sie in den folgenden Bereichen von Business Availability Center:

Infodatei. Enthält eine Liste der Versionsbeschränkungen und die letzten Aktualisierungen. Doppelklicken Sie im Stammverzeichnis der HP Business Availability Center-DVD auf **readme800.html**. Auf die aktuelle Version dieser Datei können Sie auf der Website für HP Software-Unterstützung zugreifen.

Neuerungen. Enthält eine Liste der neuen Funktionen und der Besonderheiten der Version. Wählen Sie in HP Business Availability Center **Help > What's New**.

Onlinedokumentationsbibliothek. Bei der Dokumentationsbibliothek handelt es sich um ein Onlinehilfesystem, in dem die Verwendung von HP Business Availability Center und der TransactionVision-Applikation beschrieben wird. Sie können auf die Dokumentationsbibliothek mit einem Webbrowser zugreifen. Eine Liste der Überlegungen zur Anzeige finden Sie im Abschnitt, der sich mit der Anzeige der HP Business Availability Center-Website beschäftigt, in Kapitel 6 im *HP Handbuch zur Bereitstellung von HP Business Availability Center* (PDF).

Um auf die Dokumentationsbibliothek zuzugreifen, wählen Sie in HP Business Availability Center **Help > Documentation Library**. Eine kontextbezogene Hilfe ist auf verschiedenen HP Business Availability Center-Seiten verfügbar, indem Sie auf der jeweiligen Seite auf **Help > Help on this page** klicken, und in verschiedenen Fenstern, indem Sie auf die Schaltfläche **Help** klicken. Weitere Details zur Verwendung der Dokumentationsbibliothek finden Sie im Abschnitt "Working with the HP Business Availability Center Documentation Library" im Handbuch *Platform Administration*.

Weitere Onlinere Ressourcen

Mit dem Befehl **Troubleshooting & Knowledge Base** öffnen Sie die Seite "Troubleshooting" auf der HP-Website zur Software-Unterstützung, auf der Sie die Wissensdatenbank nach Lösungen zu Ihrem Problem durchsuchen können. Wählen Sie **Help > Troubleshooting & Knowledge Base**. Der URL für diese Website lautet <http://h20230.www2.hp.com/troubleshooting.jsp>.

Der Befehl **HP Software Support** öffnet die HP-Website zur Software-Unterstützung. Auf dieser Website finden Sie die Wissensdatenbank, die Sie nach Lösungen zu Ihrem Problem durchsuchen können. Sie können zudem eigene Beiträge in das Forum einstellen und die Beiträge des Forums durchsuchen, Support-Anfragen stellen sowie Patches, aktuelle Dokumentation usw. herunterladen. Wählen Sie **Help > HP Software Support**. Die URL für diese Website lautet www.hp.com/go/hpsoftwaresupport.

Bei den meisten Support-Bereichen ist die Registrierung und Anmeldung als HP-Passport-Benutzer erforderlich. Einige Angebote setzen den Abschluss eines Supportvertrags voraus.

Weitere Informationen über die für den Zugriff erforderlichen Voraussetzungen erhalten Sie unter:

http://h20230.www2.hp.com/new_access_levels.jsp

Um sich für eine HP Passport-Benutzer-ID zu registrieren, wechseln Sie zu: <http://h20229.www2.hp.com/passport-registration.html>

Der Befehl **HP Software Web site** öffnet die HP-Software-Website. Auf dieser Website finden Sie die aktuellsten Informationen über HP-Softwareprodukte. Hierzu zählen neue Software-Releases, Seminare und Verkaufsvorführungen, Kundenunterstützung usw. Wählen Sie **Help > HP Software Web site**. Die URL für diese Website lautet www.hp.com/go/software.

Dokumentationsaktualisierungen

HP Software aktualisiert die Produktdokumentation stetig mit neuen Informationen.

Um nach Aktualisierungen zu suchen, oder um zu überprüfen, ob Sie die aktuellste Version eines Dokuments verwenden, wechseln Sie zur HP Software-Website für Produkthandbücher (<http://h20230.www2.hp.com/selfsolve/manuals>).

Willkommen bei diesem Handbuch

Teil I

Einführung in TransactionVision

1

Einführung in TransactionVision

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zu TransactionVision auf Seite 21
- Architekturdiagramm auf Seite 23
- TransactionVision in der Bereitstellungsumgebung auf Seite 25
- Installationspakete auf Seite 27
- Abwärtskompatibilität auf Seite 28
- Aktualisieren von früheren Versionen auf Seite 29

Informationen zu TransactionVision

HP TransactionVision ist eine Lösung zur Transaktionsverfolgung, mit der die Interaktion zwischen allen Komponenten des Systems grafisch dargestellt wird.

HP TransactionVision zeichnet ohne Eingriffe einzelne elektronische Ereignisse auf, die von einer Transaktion erzeugt werden, während sie in einem Computernetzwerk abläuft. Am wichtigsten ist jedoch, dass der patentierte Algorithmus "Transaction Constructor" von TransactionVision diese Ereignisse zu einer schlüssigen Geschäftstransaktion zusammensetzt.

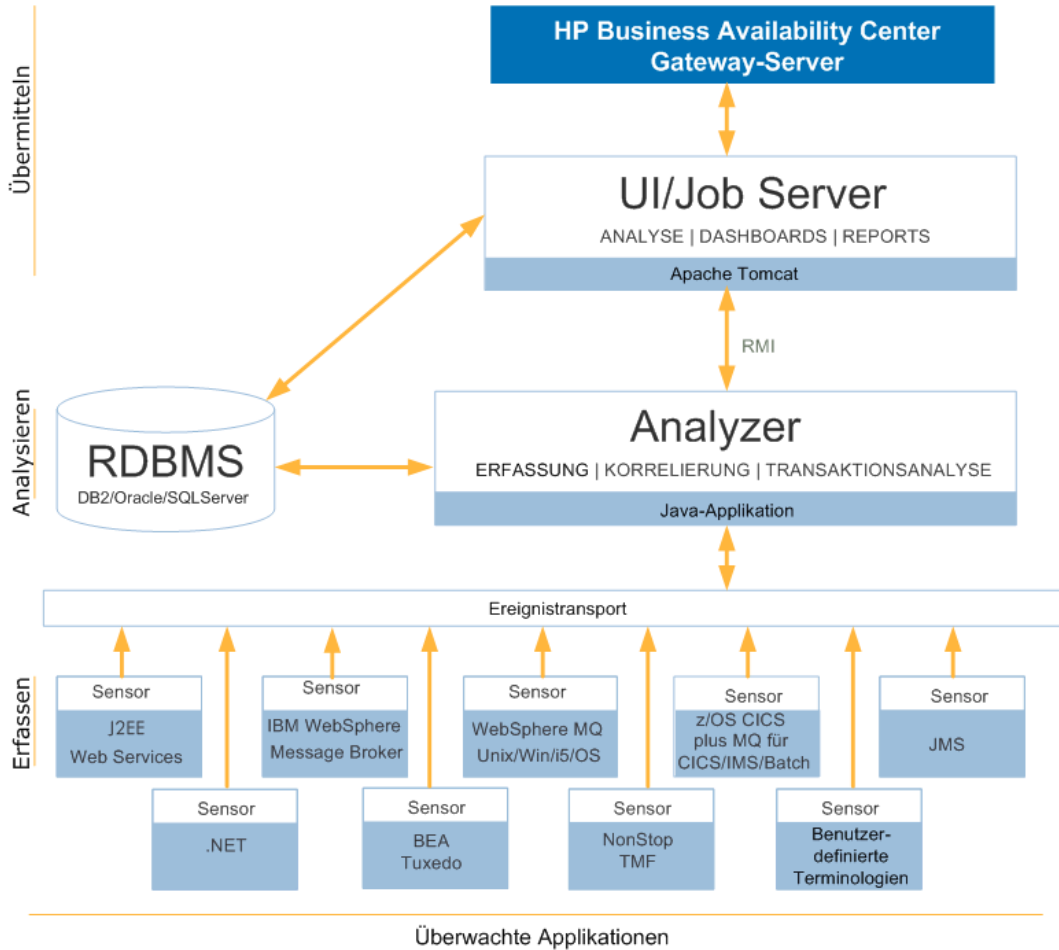
Wichtige Funktionen von HP TransactionVision:

- Vollständige Transparenz einzelner Geschäftstransaktionen.
- Verfolgung einzelner Geschäftstransaktionen in jedem Verarbeitungsschritt ohne Eingriff.

- Automatische Korrelation von Applikationsereignissen in Geschäftstransaktionen.
- Umfassende Transparenz, um die durchschnittliche Dauer der Problemissolierung und -lösung für Geschäftstransaktionen zu verkürzen.
- Erfassung von technischen und Geschäftsdaten für jede Transaktion, sodass der geschäftliche Kontext der Transaktionen bestimmt wird (Kunden-ID, finanzieller Wert usw.).
- Umfassende Integration in HP Business Availability Center und wichtige Applikationen, darunter End User Monitoring, Diagnostics und Business Process Insight.

Architekturdiagramm

Im folgenden Diagramm sind die wichtigsten Komponenten von TransactionVision dargestellt.



UI/Job Server

TransactionVision UI/Job Server ist eine Webapplikation, die mit einer Instanz des Apache Tomcat-Servlet/JSP-Containers ausgeführt wird. Der Server kommuniziert mit Analyzer, um die Konfigurationsinformationen für die Datenerfassung bereitzustellen, dazu gehören

Kommunikationsverbindungen und Datenerfassungsfilter. Er stellt auch die Verbindungen mit den Datenbankschemas des Projekts her, um Projektanalysen anzuzeigen und Reports zu Ergebnissen zu erstellen.

In Teil III finden Sie Informationen zur Installation und Konfiguration von UI/Job Server.

Analyzer

TransactionVision Analyzer ist ein Dienst unter Windows (oder ein Daemon unter UNIX), der über Message Oriented Middleware mit TransactionVision-Sensoren kommuniziert. Diese Komponente erzeugt und übermittlelt Konfigurationsnachrichten an Sensoren, indem sie einer bestimmten Warteschlange hinzugefügt werden. In den Konfigurationsnachrichten werden Konfigurationsinformationen für Sensoren, wie der Name der Ereigniswarteschlange, in der der Sensor die Ereignisnachrichten platzieren soll, sowie Filterdefinitionen zur Datenerfassung für das Projekt angegeben.

Standardmäßig verwendet TransactionVision SonicMQ als Provider für Message Oriented Middleware. TIBCO EMS und WebSphere MQ werden ebenfalls unterstützt.

Zudem ruft Analyzer Ereignisse ab, die von Sensoren einer Ereigniswarteschlange hinzugefügt wurden, und verarbeitet sie zur Analyse und Anzeige in der Webbenutzeroberfläche. Unmarshalling-, Korrelations-, Analyse- und Datenverwaltungsfunktionen werden damit ausgeführt.

In Teil II finden Sie Informationen zur Installation und Konfiguration von Analyzer.

Sensoren

TransactionVision-Sensoren erfassen Transaktionsereignisse aus verschiedenen Applikationen, die an den verteilten Transaktionen beteiligt sind. Sensoren sind einfache Bibliotheken oder Exitprogramme, die auf jedem Computer in der Umgebung installiert werden.

Jeder Sensor überwacht Aufrufe, die von den unterstützenden Technologien des Systems ausgehen, und vergleicht sie mit den Filterbedingungen. Falls der Aufruf die Filterbedingungen erfüllt, erfasst der Sensor über den Aufruf Informationen zum Eintritt und übergibt den Aufruf dann zur Verarbeitung an die entsprechende Bibliothek. Wenn der Aufruf zurückgegeben wird, erfasst der Sensor über den Aufruf Informationen zum Verlassen. Diese Informationen werden dann zu einem TransactionVision-Ereignis kombiniert, das an Analyzer weitergeleitet wird, indem es in eine bestimmte Warteschlange aufgenommen wird.

Der .NET-Sensor und der Java-Sensor werden als .NET-Agent bzw. Java-Agent bezeichnet. Mit Agents werden die Funktionen von HP Diagnostics Probe und des TransactionVision-Sensors in einer Komponente kombiniert.

In Teil IV finden Sie Informationen zur Installation und Konfiguration von Sensoren und Agents.

RDBMS (Datenbank)

TransactionVision nutzt eine RDBMS-Datenbank von Drittanbietern zum Speichern von Daten. Analyzer ruft die von Sensoren erfassten Ereignisse ab, verarbeitet sie und platziert sie in ereignisbezogenen Tabellen. Mithilfe von Schemas zur Partitionierung von Ereignisdaten nach Projekt können Sie den Zugriff auf Ereignisdaten steuern, die von den einzelnen Projekten erfasst werden.

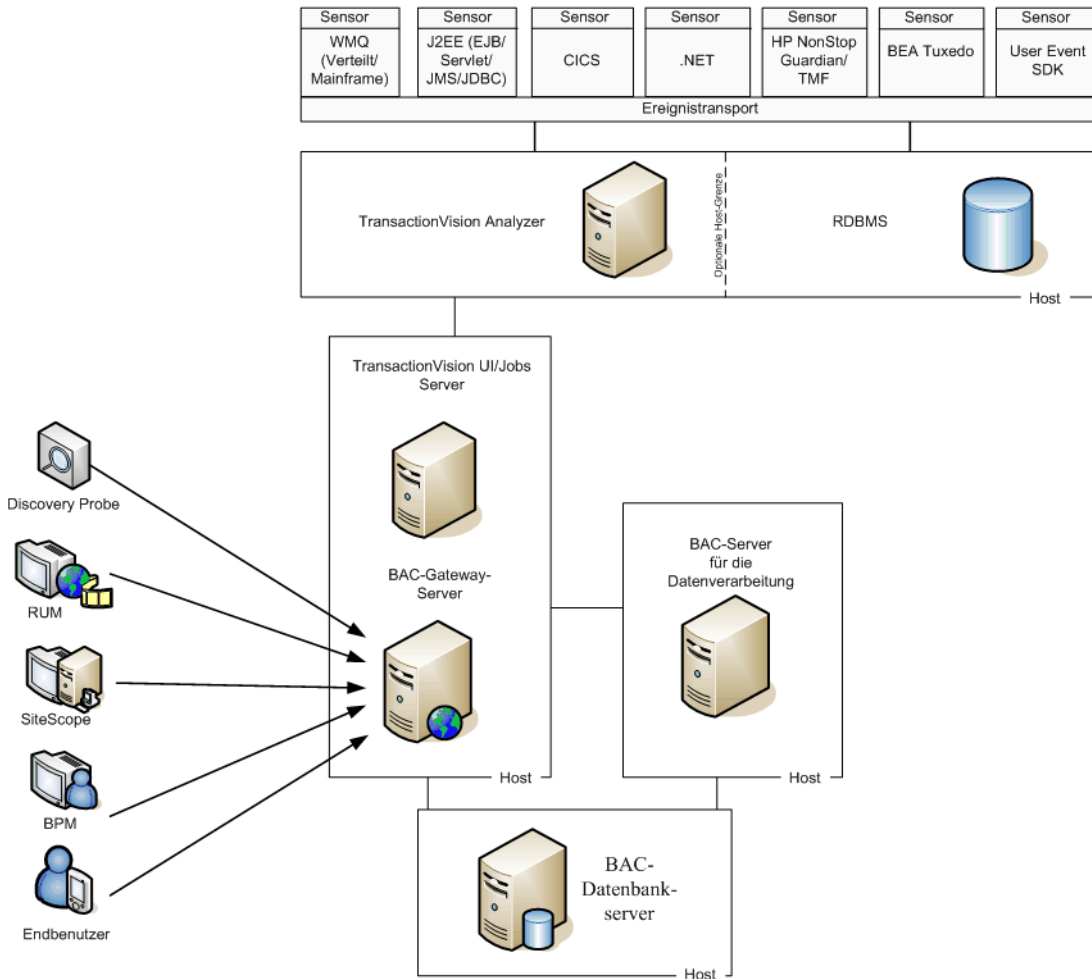
Weitere Informationen finden Sie unter "Konfigurieren von Datenbanken" auf Seite 63.

TransactionVision in der Bereitstellungsumgebung

TransactionVision arbeitet in der Business Availability Center-Bereitstellungsumgebung wie folgt:

- ▶ TransactionVision Analyzer ist auf einem anderen Host installiert als der Business Availability Center-Gateway-Server.
- ▶ TransactionVision UI/Job Server ist auf dem Host des Business Availability Center-Gateway-Servers installiert.

- ▶ TransactionVision verwendet eine andere Datenbank als Business Availability Center und die Datenbank für TransactionVision ist auf einem Host installiert, auf den TransactionVision Analyzer optimal zugreifen kann.
- Im folgenden Diagramm sind die TransactionVision-Komponenten in einer typischen Business Availability Center-Bereitstellungsumgebung dargestellt.



In Bereitstellungsumgebungen mit mehreren BAC-Gateway-Servern, wie in einem Szenario mit hoher Verfügbarkeit, kann zu einem bestimmten Zeitpunkt nur eine Instanz von TransactionVision UI/Job Server ausgeführt werden. Sie können eine gespiegelte UI/Job Server-Instanz auf den anderen Hosts der BAC-Gateway-Server einrichten, sie kann aber erst gestartet werden, wenn die primäre TransactionVision UI/Job Server-Instanz angehalten wurde.

Weitere Informationen zu den Business Availability Center-Bereitstellungsumgebungen finden Sie im *HP Handbuch zur Bereitstellung von HP Business Availability Center* (PDF).

Installationspakete

TransactionVision Analyzer, UI/Job Server und die Sensoren werden unabhängig von den Business Availability Center-Komponenten in der Bereitstellungsumgebung installiert.

Die TransactionVision-Komponenten sind Bestandteil von Paketen, die für eine bestimmte Plattform gelten. Für jeden Satz von Installationsanweisungen in diesem Handbuch ist angegeben, welches Installationspaket verwendet werden soll und wo es zu finden ist.

Bevor Sie ein Paket installieren, stellen Sie sicher, dass die Systeme, auf denen Sie die Installation durchführen, die Systemanforderungen für TransactionVision erfüllen.

Abwärtskompatibilität

In der folgenden Tabelle wird die Abwärtskompatibilität von TransactionVision-Komponenten dargestellt.

	Analyzer 8.00	UI/Job Server 8.00	Sensoren/ Agents 8.00	.NET-Agents 8.00
7.50-Version von Analyzer				✓
7.50-Version der Webkomponente				✓
7.50-Version von Sensoren/Agents	✓	✓		
5.0.0-Version von Analyzer				
5.0.0-Version der Webkomponente				
5.0.0-Version von Sensoren/Agents	✓	✓		
5.0.0 SPC-Version von Analyzer				
5.0.0 SPC-Version der Webkomponente				
5.0.0 SPC-Version von Sensoren/Agents	✓	✓		
5.0.0 SPE-Version von Analyzer				
5.0.0 SPE-Version der Webkomponente				
5.0.0 SPE-Version von Sensoren/Agents	✓	✓		
5.0.0 SPH-Version von Analyzer				

	Analyzer 8.00	UI/Job Server 8.00	Sensoren/ Agents 8.00	.NET-Agents 8.00
5.0.0 SPH-Version der Webkomponente				
5.0.0 SPH-Version von Sensoren/Agents	✓	✓		

Die Komponenten Analyzer und UI/Job Server von TransactionVision 8.00 sind mit Analyzer und Webapplikationskomponenten vorheriger Versionen nicht kompatibel.

Außer dem .NET-Agent können Sensoren von TransactionVision 8.00 nicht mit früheren Versionen von TransactionVision Analyzer verwendet werden. Die 8.00-Version des .NET-Agents wird mit der 7.50-Version von Analyzer unterstützt.

Andere Sensoren und Agents der Versionen 5.0.0, 5.0.0 SPC, 5.0.0 SPE, 5.0.0 SPH und 7.50 können mit Analyzer und UI/Job Server von TransactionVision 8.00 verwendet werden. Es empfiehlt sich aber, Sensoren der Version 8.00 zu installieren, um die letzten Aktualisierungen und Verbesserungen nutzen zu können.

Informationen zu den neuen Funktionen von TransactionVision in dieser Version finden Sie in der Business Availability Center *Infodatei*.

Aktualisieren von früheren Versionen

TransactionVision-Webkomponenten und Analyzer aus Version 7.50 können problemlos zu Version 8.0 migriert werden. Sensoren enthalten keine Konfigurationen, die migriert werden können.

Im Allgemeinen verläuft das Upgrade wie folgt:

- 1 Führen Sie die Installationsprogramme für Analyzer und UI/Job Server aus. Sie finden frühere Installationen dieser Komponenten auf dem Host und bieten eine Option zur Sicherung alter Konfigurationsdateien und zur Migration dieser Dateien zu Version 8.00.

- 2 Führen Sie auf Hosts, auf denen eine frühere Version der TransactionVision-Webbenutzeroberfläche ausgeführt wird, **TVisionSetupInfo -cleanweb** aus, um die alte Applikation der TransactionVision-Webbenutzeroberfläche zu entfernen.

Hinweis: Die TransactionVision-Webbenutzeroberfläche früherer Versionen ist keine Produktkomponente mehr. Diese Funktionalität wird jetzt von Business Availability Center und TransactionVision UI/Job Server bereitgestellt.

In Version 8.00 wird die Komponente TransactionVision UI/Job Server mit einem eigenständigen Tomcat-Applikationsserver installiert, WebSphere oder WebLogic sind in der Bereitstellung also nicht mehr erforderlich.

- 3 Migrieren Sie Projektdatenbanken mit dem Skript **MigrateDB**, nachdem TransactionVision installiert und konfiguriert wurde. Sie sollten die Datenbanken sichern, bevor Sie das Skript **MigrateDB** ausführen.
- 4 Führen Sie nach der Migration der Komponenten das Skript **TVisionSetupInfo** aus.
- 5 Führen Sie auf Windows-Plattformen **SupervisorStart** und auf UNIX-Plattformen **run_topaz** aus.

In den Installationskapiteln dieses Handbuchs finden Sie die ausführlichen Schritte für die Upgrades.

Im Abschnitt zu den bekannten TransactionVision-Problemen der *Infodatei* finden Sie Informationen zur Aktualisierung oder Neuinstallation von TransactionVision.

2

Überprüfen der Systemanforderungen

In diesem Kapitel werden die Systemanforderungen beschrieben, die erfüllt sein müssen, um die TransactionVision-Komponenten der HP Business Availability Center-Plattform auszuführen.

Hinweis: In der Infodatei zu HP Business Availability Center, die im HP Business Availability Center-Paket enthalten ist, finden Sie weitere Informationen zu Systemanforderungen für die aktuelle und frühere Versionen von HP Business Availability Center.

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Unterstützte Analyzer-Plattformen auf Seite 33
- ▶ Unterstützte TransactionVision UI/Job Server-Plattformen auf Seite 33
- ▶ Unterstützte Datenbankmanagementsysteme auf Seite 34
- ▶ Unterstützte Provider für Message Oriented Middleware auf Seite 35
- ▶ Unterstützte Plattformen für WebSphere MQ-Sensoren auf Seite 35
- ▶ Unterstützte WebSphere Message Broker-Konfigurationen auf Seite 37
- ▶ Unterstützte Plattformen für Servlet-Sensoren auf Seite 38
- ▶ Unterstützte Plattformen für EJB-Sensoren auf Seite 39
- ▶ Unterstützte Plattformen für JMS-Sensoren auf Seite 40
- ▶ Unterstützte Plattformen für JDBC-Sensoren auf Seite 41
- ▶ Unterstützte Plattformen für CICS-Sensoren auf Seite 41
- ▶ Unterstützte Plattformen für BEA Tuxedo-Sensoren auf Seite 42

- Unterstützte Plattformen für NonStop TMF-Sensoren auf Seite 42
- Unterstützte Plattformen für .NET-Agents auf Seite 42
- Unterstützte Browserkonfigurationen auf Seite 42
- LDAP-Unterstützung auf Seite 43
- Java-Unterstützung auf Seite 43
- Flash Player-Unterstützung auf Seite 43
- Lokalisierungs- und Internationalisierungsunterstützung auf Seite 43

Hinweis: Die Systemanforderungen für den Java-Agent sind unter den einzelnen Java-Technologien aufgeführt: Servlet, EJB, JMS und JDBC.

Unterstützte Analyzer-Plattformen

Betriebssystemumgebung	WebSphere MQ	TIBCO EMS	SonicMQ
Windows 2003 Server x86 64-Bit	6.0, 7.0	4.2.0, 4.4.2	7.5.2
AIX 5L 5.3, 6.1 POWER	6.0, 7.0	4.2.0, 4.4.2	7.5.2
Solaris 9, 10 SPARC	6.0, 7.0	4.2.0, 4.4.2	7.5.2
RedHat Enterprise Linux 5.0 WS/ES/AS x86 64-Bit	6.0, 7.0	4.2.0, 4.4.2	7.5.2

Unterstützte TransactionVision UI/Job Server-Plattformen

Plattformen	Betriebssystemumgebungen
Microsoft Windows	Windows 2003 Server x86 64-Bit
Sun Solaris	Solaris 9, 10 SPARC

Die TransactionVision UI/Job Server-Komponente war Teil der TransactionVision-Webapplikation früherer Versionen. Beginnend mit dieser Version wird die TransactionVision-Benutzeroberfläche über Business Availability Center bereitgestellt. TransactionVision UI/Job Server muss zusätzlich zu Business Availability Center installiert werden, um Zugriff auf die TransactionVision-Benutzeroberfläche zu erhalten. Informationen hierzu finden Sie unter "TransactionVision in der Bereitstellungsumgebung" auf Seite 25.

Unterstützte Datenbankmanagementsysteme

Die folgenden Datenbanken sowie die zugehörigen Plattformen werden von TransactionVision unterstützt. Der Zugriff auf diese Datenbankserver-Konfigurationen kann remote über den DB2-Client oder die Oracle OCI- oder Thin Client-Schnittstellen erfolgen. TransactionVision unterstützt nur JDBC 2.0.

DBMS-Client/-Server
DB2 9.1
Oracle 9.2
Oracle 10g
Oracle RAC 10g
Microsoft SQL Server 2005

Unterstützte Provider für Message Oriented Middleware

Agent/Sensor	WebSphere MQ	Transaction-Vision SonicMQ	SonicMQ	TIBCO EMS	WebLogic JMS	HTTP
Java-Agent	✓	✓	✓	✓	✓	
.NET-Agent	✓	✓	✓			
WebSphere MQ-Agent	✓					
BEA Tuxedo-Sensor						✓
NonStop TMF-Sensor						✓

Unterstützte Plattformen für WebSphere MQ-Sensoren

Plattform	Betriebssystem-umgebung	WebSphere MQ	Unterstützt den Exit-Sensor für WMQ-APIs
Microsoft Windows	Windows 2003 Server x86 32- und 64-Bit	6.0, 7.0 (32-Bit) ^{1, 6}	Ja
Sun Solaris ⁴	Solaris 9, 10 SPARC	6.0, 7.0 ^{1, 6} (32-Bit) 6.0, 7.0 ^{1, 6} (64-Bit)	Ja
Hewlett-Packard HP-UX ⁴	HP-UX 11i v3 PA-RISC und Itanium	6.0, 7.0 ^{1, 6} (32-Bit) 6.0, 7.0 ¹ (64-Bit nur für Bibliotheks-Agent)	Ja
IBM AIX ⁴	AIX 5L 5.3, 6.1 POWER	6.0, 7.0 ^{1, 6} (32-Bit) 6.0, 7.0 ^{1, 6} (64-Bit)	Ja

Plattform	Betriebssystem-umgebung	WebSphere MQ	Unterstützt den Exit-Sensor für WMQ-APIs
RedHat Linux ⁴	Enterprise Linux 5.0 WS/ES/AS x86 32- und 64-Bit	6.0, 7.0 ^{1, 6} (32-Bit), 6.0, 7.0 ^{1,6} (64-Bit)	Ja
IBM i5/OS ²	i5/OS V5R4 iSeries	6.0, 7.0 ^{1, 6}	Ja
IBM z/OS ^{3, 5}	z/OS 1.7, 1.8, 1.9 zSeries z/OS 1.7, 1.8, 1.9 Batch z/OS 1.7, 1.8, 1.9 CICS TS 2.x, 3.x z/OS 1.7, 1.8, 1.9 RRS z/OS 1.7, 1.8, 1.9 IMS 7.x, 8.x, 9.x	6.0, 7.0 ¹	n/v

¹ TransactionVision 8.00 unterstützt die Ausführung von WMQ-Applikationen mit WMQ 7.0 nur, wenn die Applikationen keine versionspezifischen neuen API-Features der Version 7.0 verwenden.

² Der C-Bibliotheks-Agent wird für die Überwachung von Java-Applikationen auf i5/OS-Systemen nicht unterstützt. Verwenden Sie stattdessen den API-Exit-Agent.

³ BTTRACE und BTMQEXIT werden unter z/OS nicht unterstützt.

⁴ Wichtig! Bei der Verwendung von Multithread-Applikationen mit WebSphere MQ auf UNIX-Systemen müssen Sie sicherstellen, dass die Applikationen über genug Stapelspeicher für die Threads verfügen. Von IBM wird eine Stapelgröße von mindestens 256 KB empfohlen, wenn Multithread-Applikationen MQI-Aufrufe absetzen. Wird der TransactionVision WebSphere MQ-Agent verwendet, kann sogar noch mehr Stapelspeicher erforderlich sein; die empfohlene Stapelgröße liegt hier bei mindestens 512 KB. Weitere Informationen finden Sie in Kapitel 7, "Connecting to and disconnecting from a queue manager", im Handbuch *WebSphere MQ Application Programming Guide*.

⁵ Wenn Sie den WebSphere MQ CICS-Agent für z/OS verwenden, setzen Sie sich bitte mit dem technischen Support für Hewlett-Packard TransactionVision in Verbindung, um sicherzustellen, dass Sie über die aktuellste und effizienteste Version dieser z/OS-Komponente verfügen.

⁶ Für WebSphere MQ 7.0 ist Fixpack 7.0.0.1 erforderlich, um den Exit-Sensor für WebSphere MQ-APIs verwenden zu können.

Unterstützte WebSphere Message Broker-Konfigurationen

TransactionVision verfügt über die Funktionalität zur Überwachung von WebSphere MQ 6-API-Aufrufen zu und von WebSphere Message Broker. In der folgenden Tabelle sind die von TransactionVision unterstützten WebSphere Message Broker-Konfigurationen aufgeführt.

Betriebssystemumgebung	WebSphere Message Broker-Version
AIX 5L 5.3, 6.1 POWER	6.1
Windows 2003 Server x86 32- und 64-Bit	6.1

Unterstützte Plattformen für Servlet-Sensoren

Plattform	Betriebssystemumgebung	Applikationsserver
Microsoft Windows	Windows 2003 Server x86 32- und 64-Bit	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10
Sun Solaris	Solaris 9, 10 SPARC	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10
IBM AIX	AIX 5L 5.3, 6.1 POWER	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10
RedHat Linux	RedHat Enterprise Linux 5.0 WS/ES/AS x86 32- und 64-Bit	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10

¹ TransactionVision 8.00 bietet keine Unterstützung für IBM WebSphere Application Server Community Edition.

² Wenn WebSphere Application Server 6.1 verwendet wird, ist Fixpack 9 oder höher für TransactionVision 8.00 erforderlich, um Leistungsprobleme zu vermeiden, die bei früheren Versionen festgestellt wurden.

Unterstützte Plattformen für EJB-Sensoren

Plattform	Betriebssystemumgebung	Applikationsserver
Microsoft Windows	Windows 2003 Server x86 32- und 64-Bit	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10
Sun Solaris	Solaris 9, 10 SPARC	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10
IBM AIX	AIX 5L 5.3, 6.1 POWER	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10
RedHat Linux	RedHat Enterprise Linux 5.0 WS/ES/AS x86 32- und 64-Bit	IBM WebSphere Application Server V5.1, 6.0, 6.1 FP9+ ^{1,2} BEA WebLogic Application Server 8.1.6, 9.2.3, 10

¹ TransactionVision 8.00 bietet keine Unterstützung für IBM WebSphere Application Server Community Edition.

² Wenn WebSphere Application Server 6.1 verwendet wird, ist Fixpack 9 oder höher für TransactionVision 8.00 erforderlich, um Leistungsprobleme zu vermeiden, die bei früheren Versionen festgestellt wurden.

Unterstützte Plattformen für JMS-Sensoren

Plattform	Betriebssystemumgebung	JMS-Dienstprovider
Microsoft Windows	Windows 2003 Server x86 32- und 64-Bit	WebSphere MQ 6.0 ¹ TIBCO EMS 4.2.0, 4.4.2 SonicMQ 7.5.2 WebLogic JMS 8.1.6, 9.2.3, 10
Sun Solaris	Solaris 9, 10 SPARC	WebSphere MQ 6.0 (32-Bit und 64-Bit) ¹ TIBCO EMS 4.2.0, 4.4.2 SonicMQ 7.5.2 WebLogic JMS 8.1.6, 9.2.3, 10
IBM AIX	AIX 5L 5.3, 6.1 POWER	WebSphere MQ 6.0 (32-Bit und 64-Bit) ¹ TIBCO EMS 4.2.0, 4.4.2 SonicMQ 7.5.2 WebLogic JMS 8.1.6, 9.2.3, 10
RedHat Linux	RedHat Enterprise Linux 5.0 WS/ES/AS x86 32- und 64-Bit	WebSphere MQ 6.0 (32-Bit und 64-Bit) ¹ TIBCO EMS 4.2.0, 4.4.2 SonicMQ 7.5.2 WebLogic JMS 8.1.6, 9.2.3, 10

¹ WebSphere JMS, die in WebSphere Application Server eingebettete JMS, wird nicht unterstützt.

Unterstützte Plattformen für JDBC-Sensoren

Plattform	Datenbankversion	Betriebssystemumgebung
Oracle	9.2, 10g, RAC 10g	Windows 2003 Server 32- und 64-Bit Solaris 9, 10 AIX 5L 5.3, 6.1 RedHat Enterprise Linux x86 5.0 32- und 64-Bit
DB2	9.1	Windows 2003 Server 32- und 64-Bit Solaris 9, 10 AIX 5L 5.3, 6.1 RedHat Enterprise Linux 5.0 x86 32- und 64-Bit

Unterstützte Plattformen für CICS-Sensoren

Plattform	Betriebssystemumgebung	WebSphere MQ
z/OS	z/OS 1.7, 1.8, 1.9 CICS TS 2.x, 3.x zSeries ¹	6.0, 7.0

¹ Um TransactionVision mit CICS TS 3.2 unter z/OS ausführen zu können, muss PTF UK37779 angewendet werden (dieser PTF schließt APAR PK66562 ein, mit dem das Problem eigentlich behoben wird). PTF UK37616 ist ebenfalls erforderlich und eine Voraussetzung für die Installation von UK37779.

Unterstützte Plattformen für BEA Tuxedo-Sensoren

Plattform	Betriebssystemumgebung	BEA Tuxedo-Version
Sun Solaris	Solaris 9, 10 SPARC	8.1 (32- und 64-Bit)
IBM AIX	AIX 5L 5.3 POWER	8.1 (32- und 64-Bit)
HP HP-UX	HP-UX 11i v2, v3 PA-RISC	8.1, 9.1 (32- und 64-Bit)

Unterstützte Plattformen für NonStop TMF-Sensoren

Plattform	Betriebssystem- umgebung	TMF
NonStop	Guardian G06.29.02	T8652G08^10JUN2006^TMFCOM^AGL
NonStop	Guardian G06.30	T8608G08^08JAN2007^TMP^AGS

Unterstützte Plattformen für .NET-Agents

Plattform	Betriebssystemumgebung	.NET
Microsoft Windows	Windows 2003 Server x86 32- und 64-Bit	1.1, 2.0, 3.0 ¹

¹ TransactionVision 8.00 unterstützt die Ausführung von .NET-Applikationen mit .NET 3.0 nur, wenn die Applikationen keine versionspezifischen neuen API-Features der Version 3.0 verwenden.

Unterstützte Browserkonfigurationen

Die von TransactionVision unterstützten Browserkonfigurationen sind mit den von Business Availability Center unterstützten Konfigurationen identisch. Informationen hierzu finden Sie im Handbuch *HP Business Availability Center Hardening Guide* (PDF).

LDAP-Unterstützung

Die LDAP-Unterstützung für TransactionVision wird von Business Availability Center verwaltet. Informationen hierzu finden Sie im Handbuch *HP Business Availability Center Hardening Guide* (PDF).

Java-Unterstützung

TransactionVision umfasst die Version 1.5 der Java Runtime-Umgebung, um die Analyser- und UI/Job Server-Komponenten auszuführen.

Die unterstützten JVMs für die JMS-, JDBC-, Servlet- und EJB-Agents müssen denjenigen Versionen entsprechen, die mit WebSphere Application Server, Version 5.1, 6.0 oder 6.1, und WebLogic Application Server, Version 8.1.6, 9.2.2 oder 10, bereitgestellt werden.

Die Applets, die die Ansichten **Component Topology Analysis**, **Aggregated Topology** und **Instance Topology** anzeigen, verwenden JRE 1.6.0_x (aktuellste Version wird empfohlen).

Flash Player-Unterstützung

Für einige TransactionVision-Reports und -Topologien ist Adobe Flash Player erforderlich. Informationen zu den Versionen finden Sie im Handbuch *HP Business Availability Center Hardening Guide* (PDF).

Lokalisierungs- und Internationalisierungsunterstützung

Die Benutzeroberfläche von TransactionVision 8.00 wurde lokalisiert und ist in den Sprachen Englisch, Französisch, Japanisch, Koreanisch und vereinfachtes Chinesisch verfügbar. TransactionVision 8.00 ist internationalisierungskompatibel und unterstützt die Anzeige von Ereignisfeldern, Benutzerdaten und Reports in nichtenglischen Gebietschemas sowie nichtenglische Projekt- und Abfragenamen.

Die folgenden TransactionVision-Inhalte sind nicht lokalisiert:

Kapitel 2 • Überprüfen der Systemanforderungen

- Produktnamen, Applikationsnamen
- Benutzerdefinierbare Objekte (z. B. Jobs, Filter und Kommunikationsverbindungen)
- Das Dienstprogramm **TVisionSetupInfo** und Konfigurationsdateien
- Die meisten Datumsformate

Teil II

Installation und Konfiguration von Analyzer

3

Vorbereiten der Installation von TransactionVision Analyzer

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zu TransactionVision Analyzer auf Seite 47
- TransactionVision Analyzer in der Bereitstellungsumgebung auf Seite 48
- Übersicht über die Analyzer-Installation und -Konfiguration auf Seite 48

Informationen zu TransactionVision Analyzer

TransactionVision Analyzer kommuniziert mit TransactionVision-Sensoren und verarbeitet die von den Sensoren erfassten Ereignisdaten, um aussagekräftige Analysen zu erstellen. Analyzer wird unter Windows als Windows-Dienst und unter UNIX als Daemon ausgeführt.

In TransactionVision wird RMI (Remote Method Invocation, Remote-Methodenaufruf) verwendet, um über einen angegebenen Port (standardmäßig 21100) mit Analyzer zu kommunizieren. TransactionVision UI/Job Server sowie die Befehlszeilenprogramme verwenden RMI für die Kommunikation mit Analyzer, um das Verhalten des Analyzer-Dienstes zu steuern, die Beendigung des Dienstes zu initiieren, das Starten und Beenden der Datenerfassung zuzulassen und Statusinformationen abzurufen.

Der Host, auf dem TransactionVision Analyzer installiert ist, muss über eine statische IP-Adresse verfügen. Wenn Analyzer für ein TransactionVision-Projekt registriert wird, wird der zugehörige Hostname in die zugrunde liegende IP-Adresse aufgelöst, die dann als eindeutiger Bezeichner für Analyzer verwendet wird. Weitere Informationen finden Sie im Handbuch *Using TransactionVision*.

TransactionVision Analyzer in der Bereitstellungsumgebung

In den meisten Bereitstellungsumgebungen ist TransactionVision Analyzer auf einem anderen Host als der Business Availability Center-Gateway-Server und TransactionVision UI/Job Server installiert.

Informationen zu den Systemanforderungen für den Analyzer-Host finden Sie unter "Unterstützte Analyzer-Plattformen" auf Seite 33.

Übersicht über die Analyzer-Installation und -Konfiguration

Im Folgenden ist das allgemeine Verfahren zum Installieren und Konfigurieren von Analyzer aufgeführt:

- 1** Ermitteln Sie den Host, auf dem der Business Availability Center-Gateway-Server installiert ist. Auf diesem Host wird auch Analyzer installiert.
- 2** Lesen Sie die Systemanforderungen für Analyzer. Informationen hierzu finden Sie unter "Unterstützte Analyzer-Plattformen" auf Seite 33.
- 3** Installieren Sie Analyzer und migrieren Sie, falls gewünscht, die Konfigurationseinstellungen einer früheren Version.

Informationen zur Installation auf einem Windows-basierten Host finden Sie unter "Installieren von Analyzer unter Windows" auf Seite 51.

Informationen zur Installation auf einem UNIX-basierten Host finden Sie unter "Installieren von Analyzer auf UNIX-Plattformen" auf Seite 57.

- 4** (Optional) Installieren Sie ein unterstütztes Produkt für Message Oriented Middleware, falls Sie nicht SonicMQ verwenden möchten, das im Lieferumfang von TransactionVision enthalten ist.
- 5** Richten Sie die Datenbank ein.
Informationen hierzu finden Sie im Kapitel "Konfigurieren von Datenbanken".
- 6** Führen Sie das Dienstprogramm **TVisionSetupInfo** aus.

Informationen hierzu finden Sie unter "Konfigurieren von Analyzer" auf Seite 75.

7 Nehmen Sie nach Bedarf optionale Konfigurationsänderungen vor.

Informationen hierzu finden Sie unter "Weitere Analyzer-Konfigurationsschritte" auf Seite 89.

8 Starten Sie Analyzer.

Informationen hierzu finden Sie unter "Verwalten von Analyzer" auf Seite 89.

4

Installieren von Analyzer unter Windows

Dieses Kapitel umfasst die folgenden Themen:

- Starten des Analyzer-Installationsprogramms unter Windows auf Seite 51
- Erstinstallation auf Seite 52
- Upgrade-Installation auf Seite 53
- Deinstallieren von Analyzer auf Seite 54

Starten des Analyzer-Installationsprogramms unter Windows

Führen Sie die folgenden Schritte aus, um Analyzer zu installieren:

- 1** Stellen Sie sicher, dass Sie entweder als Administrator oder als Benutzer mit Administratorberechtigungen am Zielsystem angemeldet sind.
- 2** Wenn Sie für Analyzer ein Upgrade von einer früheren Version durchführen, müssen Sie sicherstellen, dass die Umgebungsvariable `JAVA_HOME` festgelegt oder Java im Pfad angegeben ist.
- 3** Schließen Sie alle Windows-Programme, die momentan auf dem Computer ausgeführt werden. Dies umfasst auch Programme für die automatische Sicherung. Antiviren-, Antispyware- und andere Schutzprogramme müssen nicht heruntergefahren werden.
- 4** Doppelklicken Sie in Windows-Explorer auf `tvalzr_800_win.exe`. Die InstallShield-Willkommenseite wird angezeigt.
- 5** Klicken Sie auf **Next>** und warten Sie dann, bis die Willkommenseite des TransactionVision-Setupprogramms angezeigt wird.

- 6 Klicken Sie, wenn die InstallShield-Seite **Save Files** angezeigt wird, auf **Next>**, um den Standardordner zum Extrahieren der Installationsdateien zu verwenden (z. B. **C:\TEMP\Hewlett-Packard\TransactionVision**), oder klicken Sie auf **Change**, um den gewünschten Ordner auszuwählen. Klicken Sie danach auf **Next>**, um fortzufahren.

Wenn TransactionVision Analyzer erstmalig auf diesem Host installiert wird, fahren Sie gemäß der Beschreibung unter "Erstinstallation" auf Seite 52 fort. Falls eine frühere Version von Analyzer auf dem Computer installiert ist, fahren Sie gemäß der Beschreibung unter "Upgrade-Installation" auf Seite 53 fort.

Erstinstallation

Bei einer Erstinstallation wird die Willkommenseite des Setupprogramms angezeigt.

- 1 Klicken Sie auf der Willkommenseite des Setupprogramms auf **Next>**, um die Lizenzvereinbarung für TransactionVision anzuzeigen.
- 2 Klicken Sie auf **Yes**, um der Lizenzvereinbarung zuzustimmen. Die Seite **User Information** wird angezeigt.
- 3 Geben Sie Ihren Namen und den Namen Ihres Unternehmens ein und klicken Sie dann auf **Next>**. Die Seite **Destination Location** wird angezeigt.
- 4 Wenn Sie den Standardinstallationsordner (**C:\Programme\HP\TransactionVision**) verwenden möchten, klicken Sie auf **Next>**. Wenn Sie einen anderen Installationsordner verwenden möchten, klicken Sie auf **Browse....** Wählen Sie den gewünschten Installationsordner aus und klicken Sie dann auf **Next>**.

Die ausgewählten Pakete werden am angegebenen Speicherort installiert. Die Seite **Setup Complete** wird angezeigt.

- 5 Klicken Sie auf **Finish**, um die Installation abzuschließen.

Upgrade-Installation

Bei einer Upgrade-Installation zeigt der Installations-Assistent das Wartungsmenü des Setupprogramms an.

- 1** Wenn Sie für diese Installation von TransactionVision andere Einstellungen als in der vorherigen Installation verwenden möchten, wählen Sie **Remove** aus. Klicken Sie auf **Next>**, um die vorherige Installation zu deinstallieren, und beginnen Sie dann erneut mit dem Installationsverfahren. Wenn Sie ein Upgrade für eine frühere Version durchführen, wählen Sie **Reinstall** aus. Klicken Sie dann auf **Next>**, um TransactionVision mit den Einstellungen der vorherigen Installation zu installieren. Das Dialogfeld **Configuration File Migration** wird angezeigt:
- 2** Klicken Sie auf **Ja**, um die Konfigurationsinformationen aus der vorherigen Installation zu übernehmen. Der Installations-Assistent erstellt eine Sicherungskopie der vorhandenen Konfigurationsdateien, installiert die neue Version von TransactionVision und öffnet ein MS-DOS-Fenster, um vorhandene Konfigurationsdateien zur neuen Version zu migrieren. Wenn dieser Vorgang abgeschlossen ist, wird die Seite **Setup Complete** angezeigt.

Klicken Sie auf **No**, um vorhandene Konfigurationsdateien zu überschreiben. In einem Meldungsfeld des Installations-Assistenten werden Sie aufgefordert, anzugeben, ob Sie eine Sicherungskopie vorhandener Konfigurationsdateien erstellen möchten, bevor Sie mit der Installation fortfahren.

Klicken Sie auf **Yes**, um eine Sicherungskopie zu erstellen, oder auf **No**, um mit der Installation fortzufahren, ohne die Konfigurationsdateien zu sichern. Der Installations-Assistent installiert nun die neue Version von TransactionVision (wobei vorhandene Konfigurationsdateien überschrieben werden) und zeigt danach die Seite **Setup Complete** an.

- 3** Klicken Sie auf **Finish**, um die Installation abzuschließen.
- 4** Nachdem Sie die Upgrade-Installation durchgeführt haben, müssen Sie die vorhandenen Datenbankschemas mit dem Skript **MigrateDB.bat** migrieren. Informationen zu diesem Dienstprogramm finden Sie in Anhang A, "Referenz der Dienstprogramme".

Wenn Sie größere Datenmengen migrieren müssen, finden Sie in Anhang C, "Datenbankmigration.", ausführlichere Informationen zum Migrationsverfahren.

Deinstallieren von Analyzer

Führen Sie die folgenden Schritte aus, um Analyzer zu deinstallieren:

- 1** Wählen Sie im Startmenü den Befehl **Systemsteuerung** aus.
- 2** Doppelklicken Sie auf **Software**.
- 3** Wählen Sie das HP TransactionVision-Paket aus, das Sie deinstallieren möchten, und klicken Sie auf **Ändern/Entfernen**. Die Seite mit dem Wartungsmenü wird angezeigt.
- 4** Wählen Sie **Remove** aus und klicken Sie auf **Next>**, um TransactionVision-Komponenten zu entfernen.
- 5** Klicken Sie auf **OK**, um die Deinstallation des angegebenen Pakets zu bestätigen. Das angegebene Paket wird deinstalliert. Die folgenden Dateien werden nicht gelöscht:
 - ▶ Alle Dateien, die nach der Installation hinzugefügt wurden.
 - ▶ Alle gemeinsam genutzten Dateien, die Paketen zugeordnet sind, die noch installiert sind.

Wenn es den Anschein hat, dass gemeinsam genutzte Dateien keinem installierten Paket zugeordnet sind (z. B. wenn alle anderen TransactionVision-Pakete deinstalliert wurden), wird die Seite **Shared File Detected** angezeigt.

- ▶ Wenn die gemeinsam genutzten Dateien installiert bleiben sollen, aktivieren Sie **Don't display this message again** und klicken Sie dann auf **No**.
- ▶ Wenn Sie die aktuelle Datei beibehalten möchten, diese Meldung jedoch für jede weitere gemeinsam genutzte Datei angezeigt werden soll, klicken Sie auf **No**.
- ▶ Zum Löschen der gemeinsam genutzten Datei klicken Sie auf **Yes**.

Wenn Sie den Servlet-Sensor deinstallieren, wird er zuerst in der WebSphere Application Server-Instanz deaktiviert, die der Sensor überwacht. Die instrumentierten Klassen unter **\$WAS_HOME/classes** werden zusammen mit ihren übergeordneten Verzeichnissen entfernt, falls sie leer sind.

- 6** Die Seite **Uninstallation Complete** wird angezeigt. Klicken Sie auf **Finish**, um die Deinstallation abzuschließen.

Hinweis: Nach der Deinstallation der TransactionVision-Webbenutzeroberfläche müssen Sie den temporären Cache und das Distributionsverzeichnis leeren. Dieser Vorgang wird nicht automatisch von WebSphere durchgeführt; ggf. vorhandene alte Dateien können dazu führen, dass eine neue Installation nicht ordnungsgemäß funktioniert.

5

Installieren von Analyzer auf UNIX-Plattformen

Dieses Kapitel umfasst die folgenden Themen:

- Installationsdateien auf Seite 57
- Starten des Analyzer-Installationsprogramms unter UNIX auf Seite 58
- Erstinstallation auf Seite 59
- Upgrade-Installation auf Seite 60
- Deinstallieren von Analyzer auf Seite 61

Installationsdateien

In der folgenden Tabelle sind die Namen der Installationsdateien der TransactionVision Analyzer-Pakete für die einzelnen verteilten Plattformen aufgeführt.

Plattform	Dateiname
AIX	tvalzr_800_aix_power.tar.gz
Linux	tvalzr_800_linux_x86.tar.gz
Solaris	tvalzr_800_sol_sparc.tar.gz

Starten des Analyzer-Installationsprogramms unter UNIX

Unter Solaris und Linux wird Analyzer im folgenden Verzeichnis installiert: **/opt/HP/TransactionVision**. Unter AIX wird Analyzer im Verzeichnis **/usr/lpp/HP/TransactionVision** installiert.

Führen Sie die folgenden Schritte aus, um TransactionVision Analyzer auf UNIX-Plattformen zu installieren.

- 1** Wechseln Sie zum Verzeichnis, in dem sich die TransactionVision-Installationsdateien befinden (entweder ein DVD-Laufwerk oder ein Downloadverzeichnis). **HINWEIS:** Unter Solaris und HP-UX müssen Sie die Installationsdateien stattdessen vom DVD-Laufwerk in ein temporäres Verzeichnis auf der Festplatte des Hostsystems kopieren.
- 2** Entpacken Sie die Pakete für Ihre Plattform; weitere Informationen finden Sie unter "Installationsdateien" auf Seite 57. Beispiel:

```
gunzip tvalzr_800_sol_sparc.tar.gz  
tar xvf tvalzr_800_sol_sparc.tar
```

- 3** Melden Sie sich als Superuser an:

```
su
```

- 4** Geben Sie den folgenden Befehl ein, um das Installationsverfahren zu starten:

```
./tvininstall_800_unix.sh
```

Nachdem das Paket entzippt wurde, wird ein Menü angezeigt, in dem die verfügbaren Komponenten aufgeführt sind. Beispiel:

```
The following TransactionVision packages are available for installation:
```

1. TransactionVision Analyzer
2. TransactionVision UI/Job Server
3. TransactionVision WebSphere MQ Agent
4. TransactionVision User Event Agent

```
99. All of above
```

```
q. Quit install
```

```
Please specify your choices (separated by,) by number/letter:
```

Wenn TransactionVision-Komponenten erstmalig auf diesem Host installiert werden, fahren Sie gemäß der Beschreibung unter "Erstinstallation" fort. Falls eine frühere Version von TransactionVision auf dem Computer installiert ist, fahren Sie gemäß der Beschreibung unter "Upgrade-Installation" auf Seite 60 fort.

Erstinstallation

- 1** Geben Sie **1** ein und drücken Sie die EINGABETASTE.

Das Paket wird vom Installationskript installiert und die folgende Meldung wird angezeigt:

```
Installation of <TVANLZR> was successful.
```

Das TransactionVision-Komponentenmenü wird angezeigt.

- 2** Geben Sie **q** ein und drücken Sie die EINGABETASTE, um den Installationsvorgang zu beenden.

Informationen zum Installieren weiterer Komponenten auf diesem Host finden Sie in den Installationsanweisungen für die betreffenden Komponenten.

Upgrade-Installation

- 1 Geben Sie **1** ein und drücken Sie die EINGABETASTE.

Wenn durch das Installationsskript festgestellt wird, dass eine frühere Analyzer-Version installiert ist, wird die folgende Meldung angezeigt:

```
There is an earlier version of TransactionVision installed on the system.  
The earlier version has to be uninstalled before installing the current package(s).  
Continue with the uninstallation? (y/n) [n]:
```

- 2 Geben Sie **Y** ein und drücken Sie die EINGABETASTE, um die vorherige Version zu deinstallieren. Die folgende Meldung wird angezeigt:

```
Installation has detected a previous installation of TransactionVision.  
Do you want to migrate existing TransactionVision configuration files? (y/n) [n]:
```

- 3 Geben Sie **Y** ein und drücken Sie die EINGABETASTE, um die Konfigurationsdateien zur neuen Version zu migrieren. Durch das Installationsskript wird automatisch eine Sicherungskopie vorhandener Analyzer-Konfigurationsdateien für die Migration erstellt. Anschließend wird die vorherige Analyzer-Version deinstalliert und das TransactionVision-Komponentenmenü wird angezeigt. Fahren Sie mit Schritt 4 fort.

Wenn Sie **N** eingeben und die EINGABETASTE drücken, erhalten Sie die Möglichkeit, zu Referenzzwecken eine Sicherungskopie der Konfigurationsdateien zu erstellen:

```
Although migration will not be performed at this time, you may optionally back up  
configuration files from your previous installation for reference purposes. Note that  
answering N will overwrite these files, causing any existing setup information to be lost.  
Do you wish to back up configuration files from the previous installation? (y/n) [n]:
```

Geben Sie **Y** ein und drücken Sie die EINGABETASTE, um eine Sicherungskopie der Konfigurationsdateien zu erstellen. Sie werden vom Installationsskript aufgefordert, einen Speicherort für die Sicherung anzugeben:

```
Enter the directory to which existing TransactionVision configuration files should be  
backed up [/opt/TVision/migrate_tv800_yyyymmdd_HHMMSS]:
```

Drücken Sie die EINGABETASTE, um den Standardspeicherort zu verwenden, oder geben Sie den Pfad für das gewünschte Sicherungsverzeichnis ein. Vom Installationsdienstprogramm werden die folgenden Aufgaben ausgeführt:

- ▶ Kopieren der aktuellen Konfigurationsdateien in das angegebene Verzeichnis.
 - ▶ Deinstallieren der vorherigen Installation von TransactionVision.
 - ▶ Erneutes Anzeigen des TransactionVision-Komponentenmenüs.
- 4 Geben Sie **q** ein und drücken Sie die EINGABETASTE, um das Installationsverfahren zu beenden.
 - 5 Migrieren Sie die Datenbankschemas mit dem Skript **MigrateDB.sh**. Informationen zu diesem Dienstprogramm finden Sie in Anhang A, "Referenz der Dienstprogramme".

Wenn Sie größere Datenmengen migrieren müssen, finden Sie in Anhang C, "Datenbankmigration.", ausführlichere Informationen zum Migrationsverfahren.

Deinstallieren von Analyzer

Führen Sie die folgenden Schritte aus, um TransactionVision-Komponenten zu deinstallieren:

- 1 Melden Sie sich als Superuser an:

```
su
```

- 2 Geben Sie den folgenden Befehl ein:

```
./tvinstall_800_unix.sh -u
```

Das folgende Menü wird angezeigt (beachten Sie, dass die tatsächlich angezeigten Optionen von den installierten TransactionVision-Komponenten abhängen).

The following TransactionVision packages are available for installation:

1. TransactionVision Analyzer
2. TransactionVision UI/Job Server
3. TransactionVision WebSphere MQ Agent
4. TransactionVision User Event Agent

99. All of above

q. Quit install

Please specify your choices (separated by,) by number/letter:

Hinweis: Beachten Sie, dass die tatsächlich angezeigten Optionen und Nummern von den auf dem Computer verfügbaren Installationsdateien abhängen.

3 Geben Sie **1** ein und drücken Sie die EINGABETASTE.

Zum Deinstallieren aller TransactionVision-Komponenten geben Sie **99** ein und drücken dann die EINGABETASTE.

Die angegebenen Pakete werden durch das Installationskript deinstalliert. Anschließend wird erneut das Menü angezeigt.

4 Geben Sie **q** ein und drücken Sie die EINGABETASTE, um die Installation zu beenden.

6

Konfigurieren von Datenbanken

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zum Konfigurieren von Datenbanken auf Seite 63
- Unterstützte Datenbanken auf Seite 64
- Konfigurieren des Datenbankzugriffs auf Seite 64
- Festlegen von DB2-Variablen auf Seite 66
- Festlegen von Oracle-Variablen auf Seite 68
- DBMS-Leistungsoptimierung auf Seite 68
- DBMS-Speicherplatzanforderungen auf Seite 72
- Konfigurieren von Datenbanken für Unicode-Daten auf Seite 73

Informationen zum Konfigurieren von Datenbanken

Bevor Sie TransactionVision Analyzer konfigurieren, sollten Sie sicherstellen, dass die DBMS-Umgebung ordnungsgemäß für die Arbeit mit TransactionVision konfiguriert ist. Diese Konfiguration variiert in Abhängigkeit davon, ob DB2-, Oracle- und SQL Server-Datenbanken verwendet werden.

Mit dem Skript **TVisionSetupInfo** wird keine Datenbank erstellt; es ist auf eine Verbindung mit einer vorhandenen Datenbank angewiesen. Außerdem müssen Datenbankvariablen oder Optimierungsparameter angepasst werden, um sicherzustellen, dass die Datenbank im Hinblick auf die jeweilige Applikation ein angemessenes Leistungsverhalten zeigt. Mit TransactionVision werden verschiedene Werkzeuge bereitgestellt, die Ihnen beim Identifizieren und Beseitigen von Datenbank-Leistungsgpässen helfen.

Unterstützte Datenbanken

Die folgenden Datenbanken sowie die zugehörigen Plattformen werden von TransactionVision unterstützt.

- DB2 9.1
- Oracle 9.2
- Oracle 10g
- Oracle RAC 10g
- Microsoft SQL Server 2005

Konfigurieren des Datenbankzugriffs

In TransactionVision wird die Verbindung mit der Datenbank über JDBC hergestellt. Wenn Sie das Dienstprogramm **TVisionSetupInfo** zum Konfigurieren von TransactionVision Analyzer und UI/Job Server verwenden, werden Sie aufgefordert, Datenbanktyp (Oracle, DB2, SQL Server), Hostnamen, Datenbanknamen, Datenbankport, Benutzernamen und Kennwort anzugeben. Durch diese Eingabe werden die Einstellungen in `<TVISION_HOME>/config/datamgr/Database.properties` geändert, die zum Herstellen der JDBC-Verbindung mit der Datenbank verwendet werden. Das Benutzerkennwort wird in verschlüsselter Form in dieser Datei gespeichert.

In TransactionVision werden JDBC-Drittanbietertreiber von DataDirect für den Zugriff auf die Datenbank verwendet. Wenn es notwendig sein sollte, stattdessen JDBC-Treiber des Originalanbieters zu verwenden, können Sie den von TransactionVision verwendeten JDBC-Treiber manuell konfigurieren.

So konfigurieren Sie den von TransactionVision verwendeten JDBC-Treiber manuell:

- 1 Bearbeiten Sie die folgenden Eigenschaften in `<TVISION_HOME>/config/datamgr/Database.properties`:
 - Legen Sie die Eigenschaft **jdbc_driver** auf die JDBC-Treiberklasse fest. Die Eigenschaftendatei enthält verschiedene Beispieleinstellungen für die gängigsten Treiber.
 - Legen Sie die Eigenschaft **jdbc_url** auf die vollständige URL der JDBC-Verbindung fest. Die Eigenschaftendatei enthält verschiedene URL-Beispiele für die gängigsten Treiber.

Wenn Sie TransactionVision beispielsweise so konfigurieren möchten, dass der DB2 Universal JDBC-Treiber verwendet wird, müssen Sie die folgenden Eigenschaften festlegen:

```
jdbc_driver=com.ibm.db2.jcc.DB2Driver
jdbc_url=jdbc:db2://host:port/
dbname:retrieveMessagesFromServerOnGetMessage=true;
```

- 2 Fügen Sie die JAR-Dateien des JDBC-Treibers zur CLASSPATH-Einstellung für TransactionVision hinzu. Dies kann erfolgen, indem Sie die JAR-Dateien zur benutzerdefinierten CLASSPATH-Variablen hinzufügen, wenn eine entsprechende Aufforderung durch **TVisionSetupInfo** angezeigt wird:

```
The Analyzer can optionally be customized by plugin beans in JAR files.

The location of these JAR files needs to be added to the Analyzer CLASSPATH.

Please specify a semicolon delimited list of any additional JAR files you wish
to be added to the CLASSPATH. (for example,
'C:\TVision\myext.jar;C:\TVision\myutil.jar'): C:\oracle\10g\jdbc\lib\ojdbc14.jar
```

Hierdurch werden die JAR-Dateien des JDBC-Treibers zur CLASSPATH-Einstellung in **SetupEnv.bat/.sh** hinzugefügt.

Festlegen von DB2-Variablen

Vor der Verwendung von TransactionVision müssen Sie für die folgenden DB2-Variablen die in der nachfolgenden Tabelle aufgeführten Werte festlegen:

Variable	Wert	Beschreibung
APP CTL HEAP SZ	1024	Maximale Heap-Größe für die Applikationssteuerung. Dieser Wert gibt die Anzahl der 4-KB-Blöcke an.
MAXAPPLS	150	Maximale Anzahl aktiver Applikationen.
APPLHEAPSZ	1024	Standardmäßige Heap-Größe für die Applikation. Dieser Wert gibt die Anzahl der 4-KB-Blöcke an.
LOCKLIST	Schätzung in Abhängigkeit vom System	Umfang des Speichers (in 4-KB-Einheiten), der der Sperrliste zugeordnet ist.

Sie sollten darüber hinaus die Größe des Pufferpools heraufsetzen.

APP CTL HEAP SZ, MAXAPPLS und APPLHEAPSZ

Verwenden Sie die folgenden Befehle, um diese Werte festzulegen. Beachten Sie, dass durch die drei letzten Befehle alle aktiven Datenbankverbindungen getrennt und anschließend der DB2-Server beendet und neu gestartet wird. Stellen Sie sicher, dass diese Schritte zu einem geeigneten Zeitpunkt ausgeführt und keine anderen Datenbankbenutzer beeinträchtigt werden.

```
db2 connect to tvision
db2 get db cfg for tvision
db2 update db cfg for tvision using APP_CTL_HEAP_SZ 1024
db2 update db cfg for tvision using MAXAPPLS 150
db2 update db cfg for tvision using APPLHEAPSZ 1024
db2 force application all
db2stop
db2start
```

LOCKLIST

Führen Sie die folgenden Schritte aus, um die Anzahl der für die Sperrliste erforderlichen Seiten zu bestimmen:

- 1 Berechnen Sie die untere Grenze für die Größe der Sperrliste:

$$(512 * 36 * \text{maxappls}) / 4096$$

Hierbei ist 512 eine Schätzung der durchschnittlichen Anzahl der Sperren je Applikation und 36 die Anzahl der Bytes, die für jede Sperre eines Objekts erforderlich sind, das über eine vorhandene Sperre verfügt.

- 2 Berechnen Sie die obere Grenze für die Größe der Sperrliste:

$$(512 * 72 * \text{maxappls}) / 4096$$

Hierbei ist 72 die Anzahl der Bytes, die für die erste Sperre für ein Objekt erforderlich sind.

- 3 Schätzen Sie den Umfang der Parallelität für den Zugriff auf die Daten ab. Wählen Sie ausgehend von Ihren Annahmen einen Anfangswert für LOCKLIST aus, der zwischen der berechneten oberen und unteren Grenze liegt. Verwenden Sie den folgenden Befehl, um diesen Wert festzulegen:

```
db2 update db cfg for Datenbankname using LOCKLIST n
```

Hierbei ist *n* der Wert für LOCKLIST.

- 4 Verwenden Sie den Datenbanksystemmonitor für die Feinabstimmung dieses Parameterwerts. Weitere Informationen finden Sie in Kapitel 32, "Configuring DB2 Capacity Management", im Handbuch *DB2 V7 Administration Guide*.

Pufferpool

Der DB2-Pufferpool umfasst standardmäßig nur 250 Seiten. Erhöhen Sie den Wert auf 1024 (oder mehr, wenn ausreichend Arbeitsspeicher für DB2 verfügbar ist), um die Leistung zu verbessern. Durch den folgenden Befehl wird die Größe des Standardpufferpools von 250 Seiten auf 1024 Seiten heraufgesetzt:

```
db2 ALTER BUFFERPOOL IBMDEFAULTBP SIZE 1024
```

Festlegen von Oracle-Variablen

Wenn zu erwarten ist, dass TransactionVision in einer vergleichsweise einfachen Umgebung verwendet wird, in der eine kleine Anzahl an Datenbankverbindungen erforderlich ist, ist für das Oracle-DBMS keine besondere Konfiguration erforderlich. In Umgebungen, in denen eine große Anzahl von Benutzern gleichzeitig auf die Webbenutzerschnittstelle zugreifen, in denen mehrere Analyzer-Installationen im Einsatz sind oder in denen Analyzer eine größere als die Standardanzahl an Threads einbindet, ist es jedoch notwendig, den Wert des Datenbankparameters für geöffnete Cursor zu erhöhen. Wenn dieser Grenzwert überschritten wird, können die TransactionVision UI- oder Analyzer-Protokolle entsprechende Fehlermeldungen enthalten. Führen Sie den folgenden Befehl aus, um die Anzahl der geöffneten Cursor zu erhöhen:

```
alter system set open_cursors = 600
```

Diese Änderung kann dynamisch während der Ausführung des Oracle-Servers erfolgen. Ein Neustart des RDBMS ist nicht erforderlich.

DBMS-Leistungsoptimierung

Da TransactionVision im Rahmen der Datenerfassung und -analyse das DBMS ausgiebig nutzt, ist die Leistung des DBMS im Hinblick auf die Gesamtleistung von TransactionVision von zentraler Bedeutung. Das Einfügen und Aktualisieren von Datensätzen macht die Mehrzahl der mit TransactionVision verbundenen Datenbankoperationen aus. Aus diesem Grund haben die Geschwindigkeit der physischen Datenträger und der E/A-Schnittstelle beträchtliche Auswirkungen auf die Leistung.

Optimieren des E/A-Durchsatzes

Die Überwindung des E/A-Durchsatzlimits ist eine zentrale Aufgabe im Hinblick auf die Optimierung der Leistung eines DBMS. Normalerweise wird dieses Limit durch den physischen Datenträger und die E/A-Schnittstelle bestimmt.

Vor der Bereitstellung muss unbedingt sichergestellt werden, dass für das DBMS ein geeignetes E/A- und Datenträger-Subsystem verfügbar ist und dass das Subsystem für Schreibvorgänge optimiert wurde. In diesem Zusammenhang müssen Sie überprüfen, ob der Datenträger Teil einer auf Leistungsverbesserung ausgerichteten RAID-Konfiguration ist, ob der Schreibcache für die Datenträger aktiviert ist und ob es sich um eine schnelle E/A-Schnittstelle (vorzugsweise eine Glasfaserschnittstelle) handelt.

Um einen hohen E/A-Durchsatz zu erzielen, sollten Formen der Parallelverarbeitung verwendet werden:

- ▶ Verwenden Sie separate DBMS-Instanzen für separate Projekte. Wenn die Daten partitioniert werden können, können separate DBMS-Instanzen auf verschiedenen Hosts genutzt werden, um die Parallelität zu unterstützen. Für diese Konfiguration müssen mehrere Instanzen von TransactionVision eingerichtet werden.
- ▶ Verwenden Sie RAID-Datenträger für Tablespace-Container. RAID-Datenträger ermöglichen parallele E/A auf Hardwareebene.
- ▶ Trennen Sie Tablespace-Container und Protokolldateiverzeichnisse. DB2-Protokolldateien, Oracle-Rollback-Segmente und SQL Server-Transaktionsprotokolle enthalten noch nicht festgeschriebene Datenbankoperationen und werden im Rahmen von Einfüge- und Aktualisierungsoperationen üblicherweise stark beansprucht. Aus diesem Grund sollten sie über ihre eigenen Container auf physisch getrennten Datenträgern, vorzugsweise auf RAID-Datenträgern, verfügen.
- ▶ Verteilen Sie Tablespaces. Wenn parallele E/A auf Hardwareebene nicht verfügbar ist, kann das DBMS so konfiguriert werden, dass sich ein Tablespace über mehrere Datenträger erstreckt. Auf diese Weise wird die Parallelität auf Ebene der DBMS-Bereichsverwaltung unterstützt.

Es gibt zahlreiche weitere Datenbankparameter, die sich auf die Leistung von TransactionVision auswirken können. Insbesondere für DB2 sollten die zuvor unter "Festlegen von DB2-Variablen" auf Seite 66 erwähnten Variablen nacheinander untersucht werden, um sicherzustellen, dass sie optimale Werte aufweisen.

Auch die direkte Verwaltung der von TransactionVision verwendeten Tablespaces durch die Datenbank bietet einige Vorzüge (DMS- oder DMS RAW-Tablespaces).

Testen von DBMS und Diagnostizieren von Leistungsengpässen

HP stellt unabhängige Werkzeuge (**DB2Test**, **OracleTest** und **SQLServerTest**) bereit, mit deren Hilfe die DBMS-Leistung, die für TransactionVision relevant ist, getestet werden kann (insbesondere die Rate beim Einfügen von Datensätzen). Die Werkzeuge sind in Java geschrieben und sollten auf dem System ausgeführt werden, auf dem TransactionVision Analyzer installiert werden soll. Weitere Informationen zu diesen Werkzeugen finden Sie in Anhang A, "Referenz der Dienstprogramme".

Diese Werkzeuge simulieren die Datenbank-Aktualisierungsoperation, die von TransactionVision erzeugt wird. Führen Sie den Test mehrmals aus, um ein vollständiges Bild der DBMS-Leistung zu erhalten. Beachten Sie, dass das Ergebnis des Tests nicht direkt mit der TransactionVision-Verarbeitungsgeschwindigkeit korreliert; es handelt sich vielmehr um einen Indikator, der anzeigt, wie die DBMS-Leistung vor dem Hintergrund der jeweiligen Konfiguration zu bewerten ist.

Stellen Sie sicher, dass jeder Test mindestens mehrere Minuten dauert, um den relativen Aufwand für den Initialisierungsprozess möglichst gering zu halten. Der Test sollte für dieselbe Datenbank und dieselben Tabellengruppen ausgeführt werden, die mit TransactionVision Analyzer verwendet werden.

- ▶ Führen Sie den Einfügetest mit einem Thread und einer Datensatzgröße von 1 KB aus. Hierdurch wird die Einfügleistung für das Rohereignis gemessen.
- ▶ Führen Sie den Einfügetest mit mehreren Threads und einer Datensatzgröße von 1 KB aus. Hierdurch wird getestet, ob die Verwendung mehrerer Threads die Einfügeoperation begünstigt. Normalerweise wird die Thread-Anzahl auf das Doppelte der CPU-Anzahl festgelegt.
- ▶ Führen Sie den Einfügetest mit einem Thread und einer Datensatzgröße von (7 KB + durchschnittliche Nachrichtengröße) aus. Hierdurch wird die Einfügleistung für das analysierte Ereignis gemessen.
- ▶ Führen Sie den Einfügetest mit mehreren Threads und einer Datensatzgröße von (7 KB + durchschnittliche Nachrichtengröße) aus. Hierdurch wird getestet, ob die Verwendung mehrerer Threads die Einfügeoperation begünstigt. Normalerweise wird die Thread-Anzahl auf das Doppelte oder Vierfache der CPU-Anzahl festgelegt.

- ▶ Wenn sich Analyzer-Host und DBMS-Host unterscheiden, sollten die oben genannten Tests auf dem Analyzer-Host ausgeführt werden. Mindestens ein Test sollte jedoch auf dem DBMS-Host ausgeführt werden, um festzustellen, ob Probleme auftreten, die mit der Konfiguration der Kommunikationsverbindung oder des Datenbankclients in Zusammenhang stehen.

Die Einfügerate sollte sich nicht wesentlich von dem Ergebnis unterscheiden, das auf vergleichbaren Systemen erreicht wird, die von HP getestet wurden. Während des Tests sollten die folgenden Parameter des DBMS-Systems überwacht werden:

- ▶ Datenträger-E/A-Nutzung für alle beteiligten Datenträger (Tablespaces und Protokolldateien), insbesondere der Prozentsatz der E/A-Auslastung.
- ▶ CPU-Nutzung, einschließlich des Prozentsatzes der Wartezeit.

Wenn ein Datenträger eine Auslastung von 80 bis 90 % erreicht oder die E/A-Wartezeit außergewöhnlich hoch ist, weist dies auf einen Engpass bei der Datenträger-E/A hin. Wenn kein offensichtlicher Engpass erkennbar ist, könnte ein Problem mit der DBMS- oder der Betriebssystemkonfiguration vorliegen. Überprüfen Sie mithilfe von HP die Datenbank-, DBMS- und Kernel-Parameter im Hinblick auf mögliche Konfigurationsprobleme.

Ein weiteres hilfreiches Werkzeug für die Analyse der DBMS-Leistung ist der DB2-Monitor für Leistungsmomentaufnahmen.

Aktualisieren von Datenbankstatistiken

Jedes Datenbankprodukt bietet Werkzeuge zum Aktualisieren der Statistiken zu den physischen Merkmalen von Tabellen und zugeordneten Indizes. Zu diesen Merkmalen gehören die Anzahl der Datensätze, die Anzahl der Seiten und die durchschnittliche Datensatzlänge. Diese Statistiken werden vom Datenbank-Abfrageoptimierer verwendet, um die Zugriffspfade für die Daten zu bestimmen.

Die Datenbankstatistiken sollte immer aktualisiert werden, wenn an Tabellen viele Aktualisierungen vorgenommen wurden, z. B. wenn Daten fortlaufend durch TransactionVision Analyzer erfasst und in die Datenbank geschrieben werden.

Dies kann zu beträchtlichen Leistungsverbesserungen bei Abfragen, die von TransactionVision-Ansichten und -Reports ausgeführt werden, sowie bei Abfragen führen, die intern durch TransactionVision Analyzer zum Korrelieren von Ereignissen ausgeführt werden. Es empfiehlt sich, die vom Datenbankprodukt bereitgestellte Funktionalität (z. B. die automatische Verwaltung in DB2 oder der integrierte Job-Scheduler von Oracle) zu verwenden, um die Statistiken regelmäßig neu zu erzeugen. Alternativ dazu können Sie mithilfe des TransactionVision-Dienstprogramms **CreateSqlScript** auch SQL-Skripts für die Erzeugung von Statistiken erstellen und manuelle Zeitpläne als Tasks (im Taskplaner von Windows) oder als cron-Jobs (unter UNIX) einrichten. Informationen hierzu finden Sie unter "CreateSqlScript" auf Seite 360.

DBMS-Speicherplatzanforderungen

Ein weiterer Faktor, der endgültig festgelegt werden muss, ist der Umfang des Speicherplatzes auf dem Datenträger, der für TransactionVision-Ereignisse erforderlich ist. Dieser Wert wird durch die durchschnittliche Größe der Nachrichten, der Ereignisrate für die von TransactionVision erfassten Ereignisse sowie den Zeitraum bestimmt, für den TransactionVision-Ereignisdaten in der Datenbank aufbewahrt werden müssen.

Die Formel zum Berechnen der Speicherplatznutzung lautet wie folgt:

$(\text{Durchschnittliche Nachrichtengröße} + 7 \text{ KB}) \times \text{Ereignisrate} \times \text{Aufbewahrungsdauer für Ereignisse}$

Angenommen, die durchschnittliche Nachrichtengröße ist 2 KB und die Transaktionsrate beträgt 5 Transaktionen je Sekunde (für 8 Stunden pro Tag an jedem Wochentag sind dies 720.000 Transaktionen pro Woche). Wenn TransactionVision-Daten für einen Zeitraum von 4 Wochen gespeichert werden müssen, bevor sie archiviert oder gelöscht werden, beläuft sich der insgesamt erforderliche Speicherplatz auf ungefähr 52 GB $((2 + 7) \text{ KB} \times 720.000 \times 2 \times 4 = 52 \text{ GB})$.

Konfigurieren von Datenbanken für Unicode-Daten

In TransactionVision können Unicode-Daten in Ansichten und Reports angezeigt werden. Sie müssen jedoch die TransactionVision-Datenbank mit dem erforderlichen Code-/Zeichensatz erstellen und die entsprechenden Datenbankigenschaften in TransactionVision festlegen.

Datenbank-Codesatz/Zeichensatz

Wenn Sie die TransactionVision-Datenbank erstellen, müssen Sie die in der folgenden Tabelle aufgeführten Eigenschaften angeben:

Datenbankprovider	Erforderliche Einstellungen
DB2	Die TransactionVision-Datenbank muss mit dem Codesatz UTF-8 erstellt werden.
Oracle	<ul style="list-style-type: none"> ▶ Die TransactionVision-Datenbank muss mit dem Zeichensatz AL32UTF8 oder UTF8 erstellt werden. ▶ Für den Initialisierungsparameter NLS_LENGTH_SEMANTICS muss der Wert BYTE festgelegt werden.
SQL Server	Bei der Datenbankerstellung sind keine besonderen Einstellungen erforderlich.

Eigenschaften der TransactionVision-Datenbank

Zusätzlich zum Festlegen des richtigen Datenbank-Zeichensatzes zur Erstellungszeit muss die Eigenschaft **unicode_db** in **Database.properties** festgelegt werden. Alle zeichenbasierten XDM-Spalten, für die **unicode=true** festgelegt ist, werden mit der doppelten Bytengröße erzeugt, um das Speichern der angegebene Anzahl an Zeichen in der Datenbank zu ermöglichen. Legen Sie für Zeichensätze, die mehr als zwei Byte pro Zeichen erfordern, **unicode_bytes_per_character** in der Datei **Database.properties** auf die erforderliche Anzahl an Zeichen fest. Weitere Informationen zum Ändern der Datei **Database.properties** finden Sie in Anhang B, "Konfigurationsdateien."

7

Konfigurieren von Analyzer

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zum Konfigurieren von Analyzer auf Seite 75
- Dateien, die durch "TVisionSetupInfo" geändert werden auf Seite 76
- Informationen, die von "TVisionSetupInfo" angefordert werden auf Seite 76
- Ausführen von "TVisionSetupInfo" auf Seite 83
- Verwalten von Analyzer auf Seite 89
- Weitere Analyzer-Konfigurationsschritte auf Seite 89
- Verkleinern der Ereignisdatenbank auf Seite 102

Informationen zum Konfigurieren von Analyzer

Mit TransactionVision wird das Dienstprogramm **TVisionSetupInfo** bereitgestellt, das Sie durch die Konfiguration von Analyzer führt. Dieses Dienstprogramm fordert Sie zur Angabe von Informationen auf, die für die Konfiguration benötigt werden, und legt erforderliche Umgebungsvariablen fest.

Nach dem Ausführen des Dienstprogramms können Sie einige optionale Konfigurationen vornehmen, wie weiter unten in diesem Kapitel beschrieben wird.

Nach Abschluss der Konfiguration müssen Sie Analyzer starten, wie ebenfalls weiter unten beschrieben wird.

Dateien, die durch "TVisionSetupInfo" geändert werden

Beim Konfigurieren von Analyzer werden die folgenden Dateien durch das Dienstprogramm **TVisionSetupInfo** geändert:

- `TVISION_HOME/config/datamgr/Database.properties`
- `TVISION_HOME/config/setup/Setup.properties`

Darüber hinaus werden von **TVisionSetupInfo** folgende Operationen ausgeführt:

- Speichern des Installationspfads für Softwarewerkzeuge in `TVISION_HOME/config/setup/DefaultInstallPath.xml`.
- Erzeugen der Datei `TVISION_HOME/bin/SetupEnv.[sh|bat]`, die von **TVisionSetup** ausgeführt wird, um `JAVA_HOME`, `CLASSPATH` und die Umgebungsvariablen für den Systembibliothekspfad festzulegen, die von `TransactionVision` benötigt werden.

Informationen, die von "TVisionSetupInfo" angefordert werden

Vom Dienstprogramm **TVisionSetupInfo** werden Sie zur Eingabe der folgenden Informationen aufgefordert, die benötigt werden, um die Konfiguration abschließen zu können. Sie können diese Informationsanforderungen vor dem Ausführen von **TVisionSetupInfo** durchgehen, sodass Sie vorbereitet sind und die geeigneten Antworten eingeben können.

Lizenzierungsinformationen

Sie werden aufgefordert, Ihre `TransactionVision`-Lizenz einzugeben.

Wenn Sie die Lizenz nach dem Ausführen von **TVisionSetupInfo** anwenden möchten, müssen Sie `<TVISION_HOME>/config/license/License.properties` ändern.

Speicherort der Protokolldateien

Sie müssen den Pfad zu dem Verzeichnis angeben, in dem TransactionVision die Analyzer-Protokolldateien speichern soll. Der Standardwert ist **TVISION_HOME/logs**. Informationen hierzu finden Sie in Kapitel 8, "Konfigurieren der Analyzer-Protokollierung".

Datenbankinformationen

Sie müssen den Typ der Datenbank angeben, die Sie mit TransactionVision verwenden möchten (IBM DB2, Oracle oder SQL Server).

Hinweis: Durch das Dienstprogramm **TVisionSetupInfo** wird automatisch der DB2 Universal JDBC-Treiber (Typ 4) oder der Thin Client-Treiber für Oracle eingerichtet. Anweisungen zur Verwendung anderer Treiber, z. B. des DB2 Universal Driver (Typ 2) oder des Oracle OCI-Treibers, finden Sie unter "Ausführen von Analyzer mit einem Typ-2-Treiber für DB2 oder einem Oracle OCI-JDBC-Treiber" auf Seite 92.

In der folgenden Tabelle sind die Datenbankparameter aufgeführt, die für jeden unterstützten Datenbanktyp erfasst werden. Weitere Informationen zu jedem Parameter finden Sie im Anschluss an die Tabelle.

Parameter	Datenbank		
	DB2	Oracle	SQL Server
Speicherort der Protokolldateien			
Name der Datenbankinstanz	✓		✓
Name der Datenbankverbindung	✓	✓	
Datenbankname	✓	✓	✓
Datenbankhost	✓	✓	✓
Datenbank-Benutzername	✓	✓	✓
Kennwort des Datenbankbenutzers	✓	✓	✓

Parameter	Datenbank		
	DB2	Oracle	SQL Server
Datenbank-Listenerport	✓	✓	✓
Installationspfad der Datenbank	✓	✓	
Installationspfad des JDBC-Treibers	✓	✓	✓

Spezifische Informationen für DB2-Datenbanken

Für eine DB2-Datenbank müssen Sie die folgenden Informationen bereitstellen:

- ▶ Den Wert der Umgebungsvariablen DB2INSTANCE. Der Standardwert ist **DB2**. Bevor Sie den WebSphere 5-Server starten, muss die Umgebungsvariable DB2INSTANCE festgelegt werden. Wird diese Variable nicht festgelegt, führt dies zu einem Fehler beim Starten von TransactionVision. Außerdem kann der WebSphere-Server nicht gestartet und initialisiert werden.
- ▶ Den Namen der Datenbankverbindung, die von TransactionVision verwendet werden soll. Weitere Informationen zum Angeben des Verbindungs- und des Datenbanknamens finden Sie in Kapitel 6, "Konfigurieren von Datenbanken".
- ▶ Den Namen der Datenbank, mit der TransactionVision eine Verbindung herstellt. Dieser Name kann von dem Namen für **database_connection_name** abweichen, wenn eine Client-Datenbankverbindung verwendet wird. Weitere Informationen zum Angeben des Verbindungs- und des Datenbanknamens finden Sie in Kapitel 6, "Konfigurieren von Datenbanken".
- ▶ Den Namen des Hosts, auf dem die Datenbank ausgeführt wird. Der Standardwert ist **local_host**.
- ▶ Den Benutzernamen zum Herstellen der Datenbankverbindung. Wenn dieses Feld leer ist, wird die Datenbankverbindung für den momentan angemeldeten Benutzer hergestellt. Stellen Sie sicher, dass der angegebene oder momentan angemeldete Benutzer berechtigt ist, auf die Datenbank zuzugreifen.

- Das Kennwort für den Datenbank-Benutzernamen. Wenn dieses Feld leer ist, wird das Kennwort des momentan angemeldeten Benutzers verwendet, um die Datenbankverbindung herzustellen. Wenn von **TVisionSetupInfo** ein installierter JCE-Provider gefunden werden kann, der den DES-Verschlüsselungsalgorithmus unterstützt, wird der Wert für **jce_provider** in der Datei **Database.properties** auf den Klassennamen des JCE-Providers festgelegt, und das Kennwort wird verschlüsselt. Andernfalls wird die Kennwortverschlüsselung deaktiviert und das Kennwort wird als nicht verschlüsselter Text gespeichert.
- Die Nummer des Datenbankports. Es handelt sich hierbei um den TCP/IP-Port, der vom Datenbankserver abgehört wird. Ein gängiger Wert ist 50000. Zum Identifizieren des richtigen Werts wählen Sie die folgenden Menüelemente in der DB2-Steuerzentrale aus, während der DB2-Verwaltungsserver ausgeführt wird:

All Systems > *Systemname* > Instances > *Instanzname*

Klicken Sie mit der rechten Maustaste und wählen Sie im Kontextmenü **Setup Communications** aus. Klicken Sie auf **Properties**, um die Portnummer anzuzeigen, die von der Datenbankinstanz verwendet wird.

Spezifische Informationen für Oracle-Datenbanken

Für eine Oracle-Datenbank müssen Sie die folgenden Informationen bereitstellen:

- Den Namen der Datenbankverbindung, die von TransactionVision verwendet werden soll. Der Standardwert ist **TVISION**. Weitere Informationen zum Angeben des Verbindungs- und des Datenbanknamens finden Sie in Kapitel 6, "Konfigurieren von Datenbanken".
- Den Namen der Datenbank, mit der TransactionVision eine Verbindung herstellt. Der Standardwert ist **TVISION**. Dieser Name kann von dem Namen für **database_connection_name** abweichen, wenn eine Client-Datenbankverbindung verwendet wird. Weitere Informationen zum Angeben des Verbindungs- und des Datenbanknamens finden Sie in Kapitel 6, "Konfigurieren von Datenbanken".
- Den Namen des Hosts, auf dem die Datenbank ausgeführt wird. Der Standardwert ist **local_host**.

- ▶ Den Benutzernamen zum Herstellen der Datenbankverbindung. Wenn dieses Feld leer ist, wird die Datenbankverbindung für den momentan angemeldeten Benutzer hergestellt. Stellen Sie sicher, dass der angegebene oder momentan angemeldete Benutzer berechtigt ist, auf die Datenbank zuzugreifen.
- ▶ Das Kennwort für den Datenbank-Benutzernamen. Wenn dieses Feld leer ist, wird das Kennwort des momentan angemeldeten Benutzers verwendet, um die Datenbankverbindung herzustellen. Wenn von **TVisionSetupInfo** ein installierter JCE-Provider gefunden werden kann, der den DES-Verschlüsselungsalgorithmus unterstützt, wird der Wert für **jce_provider** in der Datei **Database.properties** auf den Klassennamen des JCE-Providers festgelegt, und das Kennwort wird verschlüsselt. Andernfalls wird die Kennwortverschlüsselung deaktiviert und das Kennwort wird als nicht verschlüsselter Text gespeichert.
- ▶ Die Nummer des Datenbankports. Es handelt sich hierbei um den TCP/IP-Port, der vom Datenbankserver abgehört wird. Ein gängiger Wert ist 1521. Zum Identifizieren des richtigen Werts führen Sie auf der Serverseite den Befehl **lsnrctl** aus. Verwenden Sie den Befehl **status**, um die Portnummer zu ermitteln.

Spezifische Informationen für SQL Server-Datenbanken

Für eine SQL Server-Datenbank müssen Sie die folgenden Informationen bereitstellen:

- ▶ Den Namen des Hosts, auf dem die Datenbank ausgeführt wird. Der Standardwert ist **local_host**.
- ▶ Den Namen der SQL Server-Instanz. Dieser Wert sollte normalerweise nur dann angegeben werden, wenn Sie mehrere Serverinstanzen ausführen.
- ▶ Die Portnummer zum Herstellen der Verbindung mit SQL Server.
- ▶ Den Namen der SQL Server-Datenbank, mit der TransactionVision eine Verbindung herstellt. Der Standardwert ist **TVISION**.
- ▶ Den Benutzernamen zum Herstellen der Datenbankverbindung. Wenn dieses Feld leer ist, wird die Datenbankverbindung für den momentan angemeldeten Benutzer hergestellt. Stellen Sie sicher, dass der angegebene oder momentan angemeldete Benutzer berechtigt ist, auf die Datenbank zuzugreifen.

- Das Kennwort für den Datenbank-Benutzernamen. Wenn dieses Feld leer ist, wird das Kennwort des momentan angemeldeten Benutzers verwendet, um die Datenbankverbindung herzustellen. Wenn von **TVisionSetupInfo** ein installierter JCE-Provider gefunden werden kann, der den DES-Verschlüsselungsalgorithmus unterstützt, wird der Wert für **jce_provider** in der Datei **Database.properties** auf den Klassennamen des JCE-Providers festgelegt, und das Kennwort wird verschlüsselt. Andernfalls wird die Kennwortverschlüsselung deaktiviert und das Kennwort wird als nicht verschlüsselter Text gespeichert.

LW-SSO-Einstellungen (Lightweight Single Sign-on)

Geben Sie die LW-SSO-Einstellungen an; diese Einstellungen sind für die Autorisierung des Zugriffs über Business Availability Center erforderlich.

Informationen zum Provider für Message Oriented Middleware

Von TransactionVision wird SonicMQ für die Kommunikation zwischen Sensoren und Analyzer bereitgestellt; die Software Progress SonicMQ wird zusammen mit Analyzer installiert. Wenn Sie diese in TransactionVision enthaltene Version von SonicMQ verwenden, müssen Sie keine Informationen zum Nachrichtensystemprovider angeben.

Hinweis: Für BEA Tuxedo- oder NonStop TMF-Sensoren oder für die Anbindung an RUM muss das integrierte Produkt SonicMQ als Nachrichtensystemprovider verwendet werden.

Separate Installationen von TIBCO EMS, WebSphere MQ und SonicMQ werden ebenfalls unterstützt; eines dieser Produkte kann als Provider für Message Oriented Middleware angegeben werden. Wenn Sie einen dieser Nachrichtensystemprovider verwenden, werden Sie, wie weiter unten beschrieben, zur Angabe des Installationspfads und weiterer Informationen aufgefordert.

WebSphere MQ-Installationsüberprüfung

Wenn Sie WebSphere MQ als Provider für Message Oriented Middleware verwenden, müssen Sie überprüfen, ob Analyzer mit dem angegebenen Warteschlangen-Manager kommunizieren kann. Sie können eine Server- oder eine Clientverbindung testen.

Sie benötigen hierfür die folgenden Informationen:

- Den Hostnamen für den WebSphere MQ-Warteschlangen-Manager.
- Den Kanalnamen für den WebSphere MQ-Warteschlangen-Manager.
- Die Portnummer für den WebSphere MQ-Warteschlangen-Manager.
- Die CCSID für den WebSphere MQ-Warteschlangen-Manager (bleibt leer, falls nicht bekannt)

TIBCO-Installationsüberprüfung

Wenn Sie TIBCO als Provider für Message Oriented Middleware verwenden, müssen Sie überprüfen, ob Analyzer mit dem angegebenen Host kommunizieren kann. Sie benötigen hierfür die folgenden Informationen:

- Den Hostnamen des TIBCO EMS-Servers
- Die Portnummer des TIBCO EMS-Servers
- Den Benutzernamen
- Das Kennwort

Die Agent- und Sensortypen

User Event-, .NET-, NonStop TMF- und Tuxedo-Ereignisse werden immer erfasst; die Ereigniserfassung für andere Technologien kann aktiviert oder deaktiviert werden.

Geben Sie die Sensortypen an, die Sie in der Umgebung verwenden und für die Ereignisse aktiviert sind. Folgende Optionen sind verfügbar:

- CICS
- EJB
- Servlet

- TIBCO EMS, SonicMQ JMS oder WebLogic JMS
- WebSphere MQ
- WebSphere MQ IMS Bridge
- WebSphere MQ JMS
- JDBC
- Alle der zuvor genannten Typen

Plugin-Beans zum Anpassen von Analyzer

Sie müssen den Speicherort aller JAR-Dateien zum CLASSPATH-Wert für Analyzer hinzufügen. Diese JAR-Dateien enthalten Plugin-Beans zum Anpassen von Analyzer.

Registrierung bei Business Availability Center

Sie müssen den Standort des Business Availability Center-Gateway-Servers angeben, der von den TransactionVision-Komponenten verwendet wird.

Ausführen von "TVisionSetupInfo"

- 1** Stellen Sie sicher, dass Sie entweder als Root-Benutzer, als Administrator oder als Benutzer mit Administratorberechtigungen am Zielsystem angemeldet sind.
- 2** Geben Sie **TVisionSetupInfo.[sh|bat]** ein, um das Skript **TVisionSetupInfo** auszuführen.

Betriebssystem	Skriptbefehl
AIX, Linux, Solaris	<TVISION_HOME>/bin/TVisionSetupInfo.sh
Windows	<TVISION_HOME>\bin\TVisionSetupInfo.bat

Hinweis: Bei der Installation wird die Umgebungsvariable `TVISION_HOME` auf den absoluten Pfad des Installationsverzeichnis festgelegt. Unter Solaris würde `TVISION_HOME` beispielsweise `/opt/HP/TransactionVision` entsprechen; unter AIX wäre der Wert `/usr/lpp/HP/TransactionVision`.

TVisionSetupInfo fordert Sie zur Eingabe von Informationen auf, die zum Konfigurieren der folgenden Elemente erforderlich sind:

- Protokolldateien
- Datenbankeigenschaften
- Informationen zum Drittanbieterprovider für Message Oriented Middleware
- Agents und Sensoren in der Zielumgebung
- Registrierung bei Business Availability Center

Wenn Eingabeaufforderungen von **TVisionSetupInfo** angezeigt werden, drücken Sie die `EINGABETASTE`, um die in Klammern angezeigten Standardwerte zu akzeptieren. Geben Sie andernfalls den richtigen Wert ein und drücken Sie dann die `EINGABETASTE`. Zum Angeben eines leeren Werts drücken Sie die `LEERTASTE` und drücken dann die `EINGABETASTE`. In den folgenden Abschnitten werden Beispieleingaben kursiv dargestellt.

Sobald **TVisionSetupInfo** einen Konfigurationsabschnitt abgeschlossen hat, werden Sie in einer Meldung darüber informiert, welche Dateien aktualisiert wurden.

Beispiel

Im Folgenden sehen Sie eine Beispielsitzung für die Ausführung von **TVisionSetupInfo** unter Windows.

This program collects configuration information in order to set up the TransactionVision environment. This includes:

- Location of software that TransactionVision depends upon such as the messaging middleware and the relational database system
- Parameters required to connect to the messaging middleware
- Parameters to connect to the database system
- Setup parameters for installed TransactionVision components

You will be prompted to input required configuration parameters.

If a default value is provided in [], pressing <Enter> will set the parameter to this default value. Pressing <Space><Enter> will set the parameter to an empty value.

Please specify name of the directory where you want to store your log files

[C:/PROGRA~1/HP/TRANSA~1\logs]:

TransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\config\setup\Setup.properties" has been successfully updated

Modifying *.Logging.xml files to use log file directory C:/PROGRA~1/HP/

TRANSA~1\logsTransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\config\logging\bpitveventimporter_loggingconfig.properties" has been successfully updated

Please provide your TransactionVision license key: PreSalesTraining@hp.com-020EF5-815082021D0482C

TransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\config\license\License.properties" has been successfully updated

Database Settings

Retrieving database configuration parameters...

Type of Database? (DB2/Oracle/SQLServer) [Oracle]:

Name of the host the database is running on [your_database_host]: ros89891duh.rose.hp.com

Name of database TransactionVision connects to.

This is the Oracle database SID [your_database_name]: orcl

Enter the port number that the database listener is on [1521]:

Database user name []: system

User password []: rpctest

TransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\config\datamgr\Database.properties" has been successfully updated

LW-SSO (Lightweight Single Sign-On) Settings

Enter the LW-SSO init string, used for the initialization of the enc algorithm. This value needs to be the same in all applications integ with LW-SSO. Unless you have modified the default value in other app such as BAC, you can just accept the default. [Xy6stqZ]:

Enter the LW-SSO application domain, used for LW-SSO cookie creation. This is the full or partial domain part of the fully-qualified hostn of the machine running the TransactionVision UI. (for example, 'mydomain.com') [hpqcorp.net]:

Enter a comma delimited list of LW-SSO protected domains. Applicable if other applications integrated with LW-SSO are running in different domains. It is not necessary to include the TransactionVision UI domain (for example, if BAC is in domain bac.com and Diagnostics is in domadiag.com, enter

'bac.com,diag.com'): TransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\cproperties" has been successfully updated

Generating script file for environment setup...

TransactionVision Info(NewFileCreated): File "C:\PROGRA~1\HP\TRANSA~nv.bat" has been successfully created

Initialize and verify database setup for TransactionVision

Do you wish to initialize the TransactionVision Database and add communication links for SonicMQ on this Analyzer ? [y]:

Default TVISION commlink has been added

Default HTTP commlink has been added

Verifying database initialization for TransactionVision...

TransactionVision Info(DBInitialized): Database has been properly initialized

Register TransactionVision with Business Availability Center

Do you wish to register TransactionVision with Business Availability Center at this time? [n]: y

Note: If your BAC server has Lightweight Single Sign On enabled, the hostnames of the TransactionVision UI and BAC servers must be a fully qualified domain name.

Please enter the hostname of the BAC server [] : ovresx1-vm11.rose.hp.com

Please enter the port of the BAC server [80] :

Do you wish to configure TransactionVision to access BAC using HTTPS? [n] :

TransactionVision has been configured to use the BAC server at: http://
ovresx1-vm11.rose.hp.com:80

Do you wish to register the TransactionVision UI with BAC? [y] :

Note: If your BAC server has Lightweight Single Sign On enabled, the hostnames of the TransactionVision UI and BAC servers must be a fully qualified domain name.

Enter TransactionVision UI host [ROS89891DUH.americas.hpqcorp.net] :

Do you wish to configure BAC to access TransactionVision using HTTPS? [n] :

Enter TransactionVision UI port [21000] :

Do you wish to configure BAC to access TransactionVision using HTTPS? [n] :

Enter TransactionVision UI port [21000] :

Do you wish to register a TransactionVision Analyzer for RUM Data Publishing with BAC? [y] :

Enter the TransactionVision Analyzer host for RUM Data Publishing
[ROS89891DUH.americas.hpqcorp.net] :

Enter RUM data publishing port [21113] :

User name for Basic Authentication to TransactionVision Analyzer
used by Real User Monitor :

Password for Basic Authentication to TransactionVision Analyzer used by Real User
Monitor :

The BAC server at <http://ovresx1-vm11.rose.hp.com:80> will be configured with:

TransactionVision Web Server: [http://
ROS89891DUH.americas.hpqcorp.net:21000](http://ROS89891DUH.americas.hpqcorp.net:21000)
TransactionVision URL for RUM: [http://
ROS89891DUH.americas.hpqcorp.net:21113/tv_rum](http://ROS89891DUH.americas.hpqcorp.net:21113/tv_rum)
TransactionVision User for RUM:
TransactionVision Password for RUM:

Please review the above settings and verify that they are correct.

Proceed with registration to BAC? [y] :

Registration with BAC was successful.

TransactionVision Info(TVisionSetupInfoSuccess): TVisionSetupInfo has completed
successfully.

All program output and user input has been logged to
"C:/PROGRA~1/HP/TRANSA~1/logs/setup.log".

Please start HP Business Availability Center by running
%TVISION_HOME%\bin\SupervisorStart.bat

Informationen bei Beendigung der Ausführung

Bei Beendigung der Ausführung von **TVisionSetupInfo** wird die folgende
Meldung angezeigt:

TransactionVision Info(TVisionSetupInfoSuccess): TVisionSetupInfo has completed
successfully.

Neustart von Business Availability Center

Nach der Ausführung des Skripts **TVisionSetupInfo** kann ein Neustart von Business Availability Center notwendig sein. Wenn dies der Fall ist, wird der entsprechende Befehl von **TVisionSetupInfo** folgendermaßen angezeigt:

Unter Windows:

```
Please start HP Business Availability Center by running  
%TVISION_HOME%\bin\SupervisorStart.bat
```

Unter UNIX:

```
Please start HP Business Availability Center by running $TVISION_HOME/bin/  
run_topaz start
```

Verwalten von Analyzer

Der Analyzer-Dienst wird vom HP Business Availability Center-Dienst verwaltet. Informationen hierzu finden Sie im *Handbuch zur Bereitstellung von HP Business Availability Center* (PDF).

Verwenden Sie den folgenden Befehl, um Analyzer zu beenden, ohne eine Nachricht zum Beenden der Datenerfassung an den Sensor zu senden (der Sensor fährt mit dem Erfassen von Ereignissen fort):

```
ServicesManager -exit -keepcollect
```

Beachten Sie, dass der Sensor nur so lange mit dem Erfassen von Ereignissen fortfährt, bis die letzte Konfigurationsnachricht abgelaufen ist, da von Analyzer keine neuen Konfigurationsnachrichten gesendet werden. Sie können den Ablauf von Konfigurationsnachrichten auf der Seite für Analyzer-Einstellungen der TransactionVision-Benutzeroberfläche ändern.

Weitere Analyzer-Konfigurationsschritte

Die folgenden Konfigurationen können nach Bedarf vorgenommen werden.

- Ändern der Heapeinstellungen für SonicMQ Broker

- Konfigurieren des Nachrichtenablaufs
- Festlegen der Thread-Anzahl für Analyzer
- Fehlerprotokollierung
- Ausführen von Analyzer mit einem Typ-2-Treiber für DB2 oder einem Oracle OCI-JDBC-Treiber
- Multithread-Servlet-/JMS-Ereignisse
- Deaktivieren nicht benötigter Analyzer-Beans
- Lokaler Transaktionsabgleich
- Analyzer-Fehlermodus
- Datenbankauflösung durch den JDBC-Sensor
- Optimierung für .NET-Agents

Ändern der Heapeinstellungen für SonicMQ Broker

- 1** Starten Sie die SonicMQ-Verwaltungskonsolle mithilfe eines der folgenden Skripts:
`<TVISION_HOME>\Sonic\MQ7.5\bin\startmc.bat` (unter Windows)
`<TVISION_HOME>/Sonic/MQ7.5/bin/startmc.sh` (unter UNIX)
- 2** Klicken Sie auf die Registerkarte **Configure**, die oben im Fenster der SonicMQ-Verwaltungskonsolle angezeigt wird.
- 3** Klicken Sie auf die zweite Schaltfläche mit einem Pluszeichen (+) (diese Schaltfläche trägt die Bezeichnung **Containers**), um die Liste der in SonicMQ konfigurierten Container zu erweitern.
- 4** Wählen Sie den Container aus, der denselben Namen wie der Host aufweist, und klicken Sie mit der rechten Maustaste auf das Dropdownmenü. Wählen Sie hier **Properties** aus.
- 5** Klicken Sie im Fenster **Edit Container Properties** auf die Registerkarte **Environment**.
- 6** Erhöhen Sie im Feld **Java VM Options** die Heap-Größe, indem Sie den Standardwert `-Xms32m -Xmx256m` ändern.
- 7** Verlassen Sie die Verwaltungskonsolle.

- 8** Fahren Sie den Dienst mithilfe des folgenden Befehls herunter:

<TVISION_HOME>\bin\SupervisorStop.bat (unter Windows)

<TVISION_HOME>/bin/run_topaz stop (unter UNIX)

- 9** Starten Sie den Dienst mithilfe des folgenden Befehls neu:

<TVISION_HOME>\bin\SupervisorStart.bat (unter Windows)

<TVISION_HOME>/bin/run_topaz start (unter UNIX)

Weitere Informationen zum Festlegen der Heap-Größe für SonicMQ Broker finden Sie im Handbuch *Progress SonicMQ Performance Tuning Guide*.

Konfigurieren des Nachrichtenablaufs

Für den Ablauf von Konfigurationsnachrichten sollte ein Wert festgelegt werden, der klein genug ist, sodass die Menge der von einer "verwaisten" Konfigurationsnachricht erzeugten Ereignisse in die Ereigniswarteschlangen aufgenommen werden können, ohne größere Produktionsprobleme zu verursachen. Weitere Informationen finden Sie im Handbuch *Verwenden von TransactionVision*.

Festlegen der Thread-Anzahl für Analyzer

Die Thread-Anzahl für Analyzer sollte so festgelegt werden, dass sie den Testergebnissen aus dem DBMS-Einfügetest entspricht. Weitere Informationen finden Sie im Handbuch *Verwenden von TransactionVision*.

Fehlerprotokollierung

Sobald Analyzer ausgeführt wird, werden Fehler unter **TVISION_HOME/logs/analyzer.log** protokolliert. Falls ein Fehler während der Initialisierung des Dienstabschnitts von Analyzer auftritt, werden diese jedoch in der Datei **analyzer_startup.log** aufgezeichnet.

Ausführen von Analyzer mit einem Typ-2-Treiber für DB2 oder einem Oracle OCI-JDBC-Treiber

Durch das Dienstprogramm **TVisionSetupInfo** wird der DB2 Universal JDBC-Treiber (Typ 4) oder der Thin Client-Treiber für Oracle eingerichtet. Führen Sie die folgenden Schritte aus, wenn Sie andere Treiber, z. B. den DB2 Universal Driver (Typ 2) oder den Oracle OCI-Treiber verwenden möchten:

- 1** Führen Sie **TVisionSetupInfo** mindestens einmal aus, um erforderliche Parameter, z. B. den Namen des Datenbankhosts, den Datenbanknamen usw., einzugeben.
- 2** Fügen Sie die Eigenschaft **jdbc_url** im folgenden Format zur Eigenschaftendatei **<TVISION_HOME>/config/datamgr/Database.properties** hinzu, um den DB2 Universal Driver (Typ 2) zu verwenden:

```
jdbc:db2:<Name_der_Datenbankverbindung>,
```

Hierbei entspricht **<Name_der_Datenbankverbindung>** dem von Analyzer verwendeten Aliasnamen für die Datenbank. Informationen zum Bestimmen des Namens der Datenbankverbindung finden Sie unter "Konfigurieren des Datenbankzugriffs" auf Seite 64.

- 3** Fügen Sie die Eigenschaft **jdbc_url** im folgenden Format zur Eigenschaftendatei **<TVISION_HOME>/config/datamgr/Database.properties** hinzu, um den Oracle OCI-Treiber zu verwenden:

```
jdbc:oracle:oci:<Benutzer>/  
<Kennwort>@<Name_der_Datenbankverbindung>
```

Hierbei ist **<Name_der_Datenbankverbindung>** der von Analyzer verwendete Name des Oracle Net-Dienstes. **<Benutzer>** und **<Kennwort>** entsprechen den Angaben zum Benutzernamen und zum Kennwort, die für die Oracle-Verbindung erforderlich sind. Informationen zum Bestimmen des Namens der Datenbankverbindung finden Sie unter "Konfigurieren des Datenbankzugriffs" auf Seite 64.

- 4** Nachdem Sie die Eigenschaft **jdbc_url** geändert haben, wird dieser Eigenschaftswert von Analyzer und dem Applikationsserver verwendet, um die Verbindung mit der Datenbank herzustellen. Von Applikationsservern wie WebSphere oder WebLogic werden 32-Bit-JVMs verwendet. Für diese Server sind keine Änderungen an der Umgebungsvariablen für den Bibliothekspfad erforderlich.

Multithread-Servlet-/JMS-Ereignisse

Wenn ein Servlet einen Thread ausgliedert, um JMS-Aufrufe abzusetzen, übergibt das Servlet standardmäßig Verfolgungsinformationen an den untergeordneten Thread. Dies führt dazu, dass sowohl Servlet- als auch JMS-Ereignisse zu derselben Geschäftstransaktion gehören. Mitunter kann es jedoch gewünscht sein, dass Sie diese Ereignisse in unterschiedlichen Transaktionen trennen möchten. So könnte beispielsweise aus einem Servlet ein lang ausgeführter Thread ausgegliedert werden, der jedoch nicht zur selben Transaktion wie das Servlet gehören soll.

Führen Sie die folgenden Schritte aus, um das Standardverhalten zu ändern, sodass der untergeordnete Thread durch TransactionVision nicht als Teil derselben Geschäftstransaktion wie das Servlet eingestuft wird, von dem er ausgegliedert wurde:

- 1** Öffnen Sie die Datei `<TVISION_HOME>/config/services/Analyzer.properties`.
- 2** Fügen Sie den Namen des Programms, das Threads ausgliedert, wie im folgenden Beispiel gezeigt zur Eigenschaft `separate_child_thread_txns` hinzu:

```
separate_child_thread_txns=programm1, programm2
```

Trennen Sie mehrere Programmnamen durch ein Komma.

- 3** Schließen und speichern Sie `Analyzer.properties`.
- 4** Starten Sie Analyzer neu.

Deaktivieren nicht benötigter Analyzer-Beans

Wenn Sie Analyzer während der Ausführung von `TVisionSetupInfo` so konfiguriert haben, dass JMS-Ereignisse erfasst werden, wird `JMSPubSubRelationBean` standardmäßig aktiviert. Diese Bean wird benötigt, um JMS-Publish- und Subscribe-Ereignisse während der Ereignisanalyse zu korrelieren.

Angenommen, Folgendes trifft zu:

- ▶ Sie erfassen keine JMS-Publish/Subscribe-Ereignisse von den Sensoren.
– UND –

- Sie benötigen maximale Leistung von Analyzer. In diesem Fall können Sie diese Bean deaktivieren, um den Aufwand zu vermeiden, der mit der Berücksichtigung des Publish/Subscribe-Mechanismus verbunden ist.

So deaktivieren Sie "JMSPubSubRelationBean":

Bearbeiten Sie den folgenden Abschnitt von `<TVISION_HOME>/config/Beans.xml`:

```
<!-- this bean is used to correlate Publish/Subscribe events -->
<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.JMSPubSubRel
ationshipBean"/>
```

Neu:

```
<!-- this bean is used to correlate Publish/Subscribe events -->
<!--Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.JMSPubSubRel
ationshipBean"/-->
```

Lokaler Transaktionsabgleich

Standardmäßig wird in Analyzer der strikte lokale Transaktionsabgleich verwendet, um WebSphere MQ-Ereignisse zu lokalen Transaktionen zu gruppieren. Mitunter kann es jedoch gewünscht sein, den strikten lokalen Transaktionsabgleich zu deaktivieren und stattdessen die MQ-Standardbean für lokale Transaktionen zu verwenden, durch die Ereignisse gemäß Arbeitseinheit zu lokalen Transaktionen gruppiert werden.

Zum Deaktivieren des strikten lokalen Transaktionsabgleichs für bestimmte WebSphere MQ-Ereignisse müssen Sie Kriterien in der Datei **TransactionVision MQStrictLocalTxnExclude.xml** definieren. Wenn ein Ereignis einem der in dieser Datei angegebenen Kriterien entspricht, wird es von Analyzer nicht einer separaten lokalen Transaktion zugeordnet; stattdessen wird die MQ-Standardbean für lokale Transaktionen verwendet.

Die Kriterien bestehen aus einer Kombination aus Programmname, Warteschlangen-Manager und Objektname. Im folgenden Beispiel wird der strikte lokale Transaktionsabgleich für Ereignisse deaktiviert, die eines der folgenden Kriterien erfüllen:

- ▶ Programmname = amqcrsta, Warteschlangen-Manager = QM1 und Objekt = TEST.Q1
- ▶ Programmname = amqcrsta, Warteschlangen-Manager = ALT_QM und Objekt = TEST.Q

```

<Criteria>
  <Match id="0">
    <value xpath="/Event/StdHeader/ProgramName">amqcrsta</value>
    <value xpath="/Event/Technology/MQSeries/MQObject/@queueManager">
QM1</value>
    <value xpath="/Event/Technology/MQSeries/MQObject/@objectName">
TEST.Q1</value>
  </Match>

  <Match id="1">
    <value xpath="/Event/StdHeader/ProgramName">amqcrsta</value>
    <value xpath="/Event/Technology/MQSeries/MQObject/@queueManager">
ALT_QM</value>
    <value xpath="/Event/Technology/MQSeries/MQObject/@objectName">
TEST.Q1</value>
  </Match>
</Criteria>

```

Analyzer-Fehlermodus

Es gibt zwei Betriebsmodi für Analyzer: den Standardmodus und den Fehlermodus. Im Fehlermodus werden nur Ereignisdaten für fehlgeschlagene Geschäftstransaktionen (oder Transaktionen die einen SLA verletzen) gespeichert. Für erfolgreiche Geschäftstransaktionen werden nur die entsprechenden Geschäftstransaktionszeilen gespeichert. Durch die Verwendung des Fehlermodus können die Anforderungen an den Datenspeicher für ein Projekt reduziert werden. Weitere Informationen zum Aktivieren des Fehlermodus finden Sie im Handbuch *Verwenden von TransactionVision*.

Datenbankauflösung durch den JDBC-Sensor

In allen JDBC-Ereignissen wird die aktuelle Datenbankverbindungs-URL gemeldet.

Für DB2 kann die URL-Syntax eines der folgenden Formate aufweisen:

- ▶ **jdbc:db2:<db2_Datenbankalias>** für Typ-2-Treiber.
- ▶ **jdbc:db2://<Host>:<Port>/<db2_Datenbankalias>** für Typ-4-Treiber.

Für Oracle kann die URL-Syntax eines der folgenden Formate aufweisen:

- ▶ **jdbc:oracle:thin:@//<Host>:<Port>/<Dienstname>** für Typ-4-Treiber.
- ▶ **jdbc:oracle:thin:<Host>:<Port>:<SID>** für Typ-4-Treiber.
- ▶ **jdbc:oracle:oci:@<TNS_Name>** für Typ-2-Treiber.
- ▶ **jdbc:oracle:oci:@** für die lokale Standardverbindung.

Der JDBC-Sensor verwendet die Verbindungs-URL, um die Datenbank zu identifizieren, der die entsprechenden Ereignisse zugeordnet sind. In vielen Fällen enthält die URL alle Informationen, die zum Identifizieren der Datenbank notwendig sind. Dies gilt in der Regel für eine URL im Typ-4-Format, die den Host, den Port und den Datenbanknamen enthält. In diesen Fällen sind Änderungen an dieser Datei nicht erforderlich. Es ist jedoch auch möglich, dass die URL diese Informationen nicht enthält. In diesen Fällen werden von TransactionVision Analyzer eine Reihe von Regeln, die in **JDBCSystemModelDefinition.xml** definiert sind, befolgt, um den Datenbanknamen zu bestimmen. Ob Sie diese Datei konfigurieren müssen, hängt davon ab, wie die jeweilige Applikation die Verbindung mit der Datenbank herstellt und ob es gewünscht ist, dass eventuell verwendete Datenbank-Aliasnamen in den tatsächlichen Namen der zugrunde liegenden Datenbank aufgelöst werden sollen.

TVISION_HOME/config/services/ JDBCSystemModelDefinition.xml

Diese Datei enthält Daten, um JDBC-Verbindungs-URLs zu Objekten auf Datenbankebene zuzuordnen. Sie enthält die folgenden vier Abschnitte:

- ▶ Den Namen der Standard-Datenbankinstanz für verschiedene Anbieter (optional).

- Eine Liste der Datenbankinstanz-Objekte in der überwachten Umgebung, auf die im Abschnitt für die Zuordnung der lokalen Standarddatenbank verwiesen werden kann.
- Eine Liste der lokalen Standard-Datenbankverbindungen Dies gilt nur für Oracle.
- Eine Zuordnung zwischen JDBC-URL-Attributen und der entsprechenden Datenbank.

Standard-Datenbankinstanz

In diesem optionalen Abschnitt wird die Zeichenfolge für die Standard-Datenbankinstanz definiert, die für einen Datenbankanbieter und eine Datenbankplattform verwendet wird:

- `<defdb2instwin>db2</defdb2instwin>` – Hierdurch wird der Name der Standard-Datenbankinstanz für DB2 auf einer Microsoft Windows-Plattform definiert.
- `<defdb2inst>db2inst1</defdb2inst>` – Hierdurch wird der Standardname für DB2 auf allen anderen Plattformen definiert.
- `<deforacleinst>1521</deforacleinst>` – Hierdurch wird die Standardportnummer für die Oracle-Instanz definiert.

Falls diese Einträge nicht vorhanden sind, geht TransactionVision von den oben aufgeführten Standardwerten aus.

Datenbankinstanz-Objekte

Im zweiten Abschnitt werden die verschiedenen Datenbankinstanzen in der Umgebung definiert:

```
<dbinstance vendor="oracle" host=" myhost " name="1544">
  <protocol name="tcp">
    <param name="host" value=" myhost "/>
    <param name="port" value="1544"/>
  </protocol>
</dbinstance>
```

Für jedes Datenbankinstanz-Element sollten die folgenden Attribute definiert sein: **vendor** (**db2** oder **oracle**), **host** und **name**. Für die Instanz können eine oder mehrere Protokollabschnitte angegeben sein.

Für DB2 wird der Instanzname auf die DB2-Instanz festgelegt. Für Oracle wird der Name auf die Portnummer des Listenerprozesses festgelegt.

Lokale Standard-Datenbankverbindungen

Im dritten Abschnitt wird die Standarddatenbank für den jeweiligen Host und die jeweilige Datenbankinstanz definiert. Beachten Sie, dass eine Datenbankinstanz-Definition (siehe den Abschnitt weiter oben) für die Instanz vorhanden sein muss, auf die in der Definition der lokalen Standarddatenbank verwiesen wird.

```
<localdefdb vendor="oracle" host="myhost" inst="1544"  
  name="mydb"/>
```

Anwendung findet die Einstellung für die lokale Standarddatenbank beispielsweise, wenn ein JDBC-Programm eine Verbindung mit einer URL wie **jdbc:oracle:oci:@** herstellt. In diesem Fall kann der verwendete Datenbankname von TransactionVision nicht identifiziert werden, sodass die Datei **JDBCSystemModelDefinition.xml** herangezogen wird, um festzustellen, welche Standarddatenbank für den Host definiert ist, auf dem das Ereignis aufgetreten ist. Angenommen, es handelt sich um **myhost** aus dem vorherigen Beispiel; in diesem Fall würde dieses Ereignis der Oracle-Datenbank **mydb** zugeordnet.

JDBC-URL-Zuordnung

Im vierten Abschnitt wird die Zuordnung zwischen dem Datenbankalias in der JDBC-URL und dem tatsächlichen Datenbanknamen definiert.

Das Zuordnungselement hat folgendes Format:

```
<mapping vendor="vendor">  
  <src host="host name" dbalias="db alias"/>  
  <dst host="host name" inst="instance name" dbname="db name"/>  
</mapping>
```

- Der Wert des Attributs **vendor** kann auf **db2** oder **oracle** festgelegt werden.

- Das Attribut **dbalias** im Element <src> wird auf den TNS-Namen (Oracle) oder den Datenbankalias (DB2) festgelegt.
- Das Attribut **host** im Element <src> entspricht dem Host, auf dem die überwachte Applikation ausgeführt wird.
- Das Instanzattribut wird nur in DB2-Zuordnungseinträgen benötigt.
- Das Element <dst> enthält die Definition, der eine Übereinstimmung zugeordnet wird; die entsprechenden Host- und Instanzattribute müssen dem zuvor definierten <dbinstance>-Eintrag entsprechen.

Angenommen, es wurde der folgende Zuordnungseintrag definiert:

```
<mapping vendor="oracle">
  <src host="apphost" dbalias="DBNAME"/>
  <dst host="myhost" inst="1544" dbname="mydb"/>
</mapping>
```

Wenn in diesem Fall eine überwachte Applikation, die auf **apphost** ausgeführt wird, eine JDBC-URL-Verbindung wie **jdbc:oracle:oci:@DBNAME** verwenden würde, könnte diese Datenbank von Analyzer als Oracle-Datenbank **mydb** auf **myhost** aufgelöst werden.

Auflösung eines Datenbanknamens

Wenn nach dem Definieren von Regeln in **JDBCSystemModelDefinition.xml** Ereignisse in Analyzer verarbeitet werden, folgt diese Verarbeitung einer Reihe von Regeln, um den zu meldenden Datenbanknamen zu bestimmen. Die Auflösung erfolgt in folgenden Schritten:

Datenbankauflösung für Typ-2-Treiber für DB2

- 1** Zuerst wird die Datenbank-Zuordnungsliste durchsucht, um festzustellen, ob ein Eintrag vorhanden ist, dessen Quellattribute dem Ereignishost und dem Datenbank-Aliasnamen entsprechen.
- 2** Wird ein übereinstimmender Eintrag gefunden, wird mithilfe der Informationen aus dem Abschnitt für die Zielzuordnung ein Datenbankobjekt erstellt.

- 3 Wird kein passender Eintrag gefunden, wird ein Datenbankobjekt erstellt, wobei der Ereignishost, der Name der lokalen Datenbankinstanz (oder der Standardwert, falls diese Daten im Ereignis nicht vorhanden sind) und der Datenbank-Aliasname verwendet werden.

Datenbankauflösung für Typ-4-Treiber für DB2

- 1 Zuerst werden die Schritte 1 und 2 der Auflösungsanweisungen für einen Typ-2-Treiber für DB2 ausgeführt.
- 2 Wenn Schritt 1 nicht erfolgreich ist, werden die folgenden Schritte ausgeführt.
- 3 Die URL wird analysiert, um den Servernamen und die Nummer des Listenerports abzurufen.
- 4 In der Datenbankinstanz-Liste wird ein übereinstimmender Host und eine entsprechende Portnummer gesucht.
- 5 Wird in Schritt 2 ein übereinstimmender Eintrag gefunden, wird das Ereignis der ermittelten Datenbankinstanz zugeordnet. Der Datenbankname wird auf den Namen festgelegt, der in der Ereignis-URL gemeldet wurde.
- 6 Wenn in Schritt 2 kein übereinstimmender Eintrag gefunden wird und der Ereignishost mit dem URL-Host identisch ist, werden die Daten zur lokalen Datenbankinstanz aus dem Ereignis als Datenbankinstanz übernommen. Sind keine entsprechenden Daten im Ereignis vorhanden, wird der Standardwert verwendet. Anhand der zuvor ermittelten Angaben zum Host, zur Datenbankinstanz und zum URL-Datenbanknamen wird ein Datenbankobjekt erstellt.
- 7 Wenn in Schritt 2 kein übereinstimmender Eintrag gefunden wird und der Ereignishost nicht mit dem URL-Host identisch ist, wird mit dem URL-Host, dem Namen der Standard-Datenbankinstanz und dem URL-Datenbanknamen ein Datenbankobjekt erstellt.

Datenbankauflösung für Typ-2-Treiber für Oracle

- 1 Wenn die URL einen Datenbanknamen (TNS-Namen) enthält, wird geprüft, ob die Datenbank-Zuordnungsliste einen Eintrag enthält, für den der Quellhost und der Datenbank-Aliasname dem Ereignishost und dem Datenbanknamen entsprechen.

- 2** Wird ein übereinstimmender Eintrag gefunden, werden die Daten aus dem Abschnitt für die Hostzuordnung (Host, Instanz, Datenbankname) verwendet, um ein Datenbankobjekt für das Ereignis zu erstellen.
- 3** Wird kein übereinstimmender Eintrag gefunden, wird ein Datenbankobjekt erstellt, wobei (a) der JDBC-Ereignishost, (b) die standardmäßige Oracle-Datenbankinstanz (Port) und (c) der Name der Ereignisdatenbank verwendet werden.
- 4** Wenn die URL keinen Datenbanknamen enthält, wird überprüft, ob der Abschnitt für die lokale Standarddatenbank einen Eintrag mit einem übereinstimmenden Hostnamen enthält. Wird mehr als ein Eintrag gefunden, wird die erste zurückgegebene Übereinstimmung verwendet. Anschließend wird ein Datenbankobjekt verwendet, wobei die Daten für den Host, die Datenbankinstanz und den Datenbanknamen aus dem übereinstimmenden Eintrag verwendet werden.
- 5** Wird kein übereinstimmender Eintrag gefunden, wird ein Datenbankobjekt erstellt, wobei (a) der JDBC-Ereignishost, (b) der Standardwert für die Oracle-Datenbankinstanz und (c) die Zeichenfolge **default** als Datenbankname verwendet werden.

Datenbankauflösung für Typ-4-Treiber für Oracle

- 1** In diesem Fall ist keine Auflösung erforderlich. Der Datenbankhost, die Nummer des Listenerports und die SID/der Dienstname sollten durch die URL bereitgestellt werden.
- 2** Es wird überprüft, ob die Konfigurationsdaten Detailinformationen zur Oracle-Datenbankinstanz bereitstellen, die durch den Host und die Portnummer identifiziert wurde. Ist ein solcher Datensatz vorhanden, werden die entsprechenden Informationen verwendet.
- 3** In der Systemmodelltabelle wird anhand der Informationen zur Datenbankinstanz und zur SID/zum Dienstnamen ein Datenankobjekt erstellt.

Optimierung für .NET-Agents

Wenn Analyzer in einer Bereitstellungsumgebung mit .NET-Agents verwendet wird, können die Ereignisübermittlungs- und -verarbeitungszeiten verkürzt werden, indem nicht die in TransactionVision enthaltene Message Oriented Middleware SonicMQ, sondern WebSphere MQ als Provider für Message Oriented Middleware verwendet wird.

Wenn die Zugriffe auf die .NET-Applikation sporadisch erfolgen, sollten Sie darüber hinaus eine Vergrößerung der Ereigniswarteschlange in Erwägung ziehen. Informationen zum Vergrößern der Ereigniswarteschlange finden Sie in der Dokumentation zum Provider für Message Oriented Middleware.

Verkleinern der Ereignisdatenbank

In TransactionVision wird eine Bean zur XML-Ereigniskomprimierung bereitgestellt. Mit dieser Bean können Sie die Datenbankgröße für jedes Ereignis reduzieren.

Hinweis: Wenn die Bean für die XML-Ereigniskomprimierung aktiviert ist, ist es nicht möglich, Benutzerdaten aus dem Puffer abzufragen.

Öffnen Sie nach der Ausführung von **TVisionSetupInfo** die Datei `<TVISION_HOME>/config/services/Beans.xml`, navigieren Sie zum folgenden Abschnitt und ändern Sie den Inhalt folgendermaßen:

Originalsegment:

```
<Module name="DBWriteEventCtx" type="Context">

  <!-- This context contains beans that write the XML event (or part of it) to the
  database. -->
  <!-- Each registered bean in the chain is called. -->

  <Module
class="com.bristol.tvision.services.analysis.dbwrite.DBWriteEventDefaultBea
n" type="Bean"/>
```

```

<!-- Replace the default bean with this one if you want ZIP compression for
the XML event -->
<!--Module type="Bean"
class="com.bristol.tvision.services.analysis.dbwrite.DBWriteEventCompress
edBean" /-->
</Module>

```

Geändertes Segment:

```

<Module name="DBWriteEventCtx" type="Context">

  <!-- This context contains beans that write the XML event (or part of it) to the
database. -->
  <!-- Each registered bean in the chain is called. -->

  <!--<Module
class="com.bristol.tvision.services.analysis.dbwrite.DBWriteEventDefaultBea
n" type="Bean"/-->

  <!-- Replace the default bean with this one if you want ZIP compression for
the XML event -->
  Module type="Bean"
class="com.bristol.tvision.services.analysis.dbwrite.DBWriteEventCompress
edBean" />
</Module>

```

Hinweis: Stellen Sie sicher, dass nicht mehrere Beans gleichzeitig aktiviert sind. Wenn sowohl **DBWriteEventDefaultBean** als auch **DBWriteEventCompressedBean** aktiviert sind, wird ein Ausnahmefehler ausgelöst.

Wenn Sie die Komprimierungsbean verwenden, muss die Datei **DatabaseDef.xml** aktualisiert werden, damit die Ereignisdaten im BLOB-Format gespeichert werden. Die Definitionen der Tabellen **EVENT** und **EVENT_OVERFLOW** müssen aktualisiert werden und der Typ der Spalte **event_data** muss von CLOB zu BLOB geändert werden.

Nachdem Sie diese Änderungen vorgenommen haben, müssen Sie die Analyzer- und UI/Job Server-Komponenten neu starten und ein neues Projekt erstellen, damit die Änderungen wirksam werden.

Das folgende Beispiel zeigt die Datei **DatabaseDef.xml**, bevor sie so geändert wurde, dass Ereignisdaten im BLOB-Format gespeichert werden:

```
<Table name="EVENT" volatile="true">
[...]
  <Column name="event_data" type="CLOB" size="1M"/>
[...]
</Table>
<Table name="EVENT_OVERFLOW" volatile="true">
[...]
  <Column name="event_data" type="CLOB" size="1M" notNull="true"/>
[...]
</Table>
```


8

Konfigurieren der Analyzer-Protokollierung

Dieses Kapitel umfasst die folgenden Themen:

- Protokolldateien auf Seite 105
- Umlaufprotokollierung auf Seite 106
- Verwenden von Windows- und UNIX-Systemprotokollen auf Seite 108
- Aktivieren der SMTP-Protokollierung auf Seite 109
- Aktivieren der SNMP-Protokollierung auf Seite 111
- Aktivieren der JMS-Protokollierung auf Seite 112

Protokolldateien

Standardmäßig werden alle Fehler- und Warnmeldungen von TransactionVision-Komponenten in den entsprechenden Protokolldateien protokolliert. Der Speicherort der Protokolldateien wird beim Ausführen von **TVisionSetupInfo** oder **SensorSetup** angegeben und in der Datei **Setup.properties** gespeichert.

Analyzer protokolliert Fehlermeldungen in der Datei **analyzer.log**. Unter Windows werden von Analyzer drei zusätzliche Protokolldateien verwendet:

- **analyzer_startup.log** enthält Informationen zur Ausführung des Windows-Dienstabschnitts von Analyzer. Die Datei enthält üblicherweise Informationen zu den Optionen, mit denen Analyzer gestartet wird. Wenn beim Initialisieren des Dienstabschnitts von Analyzer Fehler festgestellt werden, sind sie in dieser Datei zu finden.

- **analyzer_stderr.log** und **analyzer_stdout.log** sind die Standardausgabe- und Fehlerprotokolldateien des Analyzer-Prozesses. Wenn Sie benutzerdefinierte Analysebeans nutzen, die die Konsolenausgabe oder die Standardfehlerausgabe verwenden, finden Sie die Ausgabe dieser Beans in diesen Dateien. Sie sollten diese Dateien auch heranziehen, um nach weiteren Informationen zu suchen, falls Probleme beim Starten oder Ausführen von Analyzer auftreten und die Standardprotokolldatei **analyzer.log** keine Hinweise auf einen Fehler enthält.

Umlaufprotokollierung

In Analyzer wird standardmäßig eine Form der Umlaufprotokollierung verwendet. Wenn die Protokolldatei die konfigurierte Maximalgröße erreicht, wird sie als Sicherungsdatei umbenannt, und es wird eine neue, leere Protokolldatei erstellt. Die maximale Protokollgröße beträgt standardmäßig 10 MB und es gibt eine Sicherungsprotokolldatei.

Wenn die Standardeinstellungen verwendet werden und eine Protokolldatei (z. B. die Analyzer-Protokolldatei **analyzer.log**) die Größe von 10 MB erreicht, wird sie in **analyzer.log.1** umbenannt, und es wird eine neue Datei mit dem Namen **analyzer.log** erstellt. Wenn Sie die Konfiguration ändern, sodass zwei Sicherungsdateien verwendet werden, werden die folgenden Ereignisse ausgelöst, sobald **analyzer.log** eine Größe von 10 MB erreicht:

- Falls vorhanden wird **analyzer.log.2** entfernt.
- **analyzer.log.1** wird in **analyzer.log.2** umbenannt.
- **analyzer.log** wird in **analyzer.log.1** umbenannt.
- Eine neue Datei mit dem Namen **analyzer.log** wird erstellt.

Wenn Sie die Umlaufprotokollierung *nicht* verwenden möchten, können Sie die Konfiguration so ändern, dass die lineare Protokollierung verwendet wird. Hierbei wird nur eine einzige Protokolldatei erzeugt.

In den Dateien `<TVISION_HOME>/config/logging/*.Logging.xml` werden die Art der verwendeten Protokollierung, die maximale Größe der Protokolldatei und die Anzahl der Sicherungsprotokolldateien für jede Komponente angegeben. So gibt z. B. **Sensor.Logging.xml** die Konfiguration

für die Servlet- und JMS-Sensoren an. Diese Datei enthält Einträge ähnlich den folgenden:

```
<appender class="tvision.org.apache.log4j.RollingFileAppender"
name="SENSOR_LOGFILE">
  <param name="File" value="c:/Program Files/Hewlett-Packard/TransactionVision/
logs/sensor.log"/>
  <param name="Append" value="true"/>
  <param name="MaxBackupindex" value="2"/>
  <param name="MaxFileSize" value="10MB"/>
  <layout class="tvision.org.apache.log4j. PatternLayout">
  <param name="ConversionPattern" value="%d [%t] %-5p %c %x - %m%n"/>
  </layout>
</appender>
```

Maximale Größe der Protokolldatei

Wenn Sie die maximale Größe der Protokolldatei ändern möchten, ändern Sie den Wert des Parameters **MaxFileSize**, sodass er der gewünschten Dateigröße entspricht. Die angegebenen Werte sollten mit "MB" oder "KB" enden, um zwischen Mega- und Kilobytes zu unterscheiden.

Maximale Anzahl der Sicherungsprotokolldateien

Wenn Sie die Anzahl der Sicherungsdateien ändern möchten, ändern Sie den Wert des Parameters **MaxBackupIndex**, sodass er der gewünschten Anzahl an Sicherungsdateien entspricht.

Wechsel von der Umlauf- zur linearen Protokollierung

Führen Sie folgende Schritte aus, um die lineare Protokollierung anstelle der Umlaufprotokollierung zu verwenden:

- 1 Ändern Sie im **appender class**-Wert **RollingFileAppender** in **FileAppender**. Ändern Sie z. B. die erste Zeile des vorher aufgeführten Beispiels, sodass sie folgender Zeile entspricht:

```
<appender class="tvision.org.apache.log4j.FileAppender"
name="SENSOR_LOGFILE">
```

- 2 Entfernen Sie die Einträge für die Parameter **MaxBackupIndex** und **MaxFileSize**.

Verwenden von Windows- und UNIX-Systemprotokollen

Auf UNIX- und Windows-Plattformen können Sie TransactionVision so konfigurieren, dass die Ausgabe im Protokollierungsmechanismus für Systemereignisse – das Ereignisprotokoll unter Windows oder **syslog** unter UNIX – aufgezeichnet wird. Beispiele für die Protokollierungskonfigurationsdateien, die hierfür erforderlich sind, finden Sie in **TVISION_HOME/config/logging/system/*/Sensor.Logging.xml**.

Sowohl für Windows als auch für UNIX müssen Sie einen speziellen Ereignisappender definieren.

Windows-Ereignisappender

Im folgenden Beispiel wird gezeigt, wie der Windows-Ereignisappender so konfiguriert wird, dass das Ereignisprotokoll verwendet wird:

```
<appender name="NT_EVENT_LOG" class="tvision.org.apache.log4j.nt
.NTEventLogAppender">
  <layout class="tvision.org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%d [%t] - %m%n"/>
  </layout>
</appender>
```

Anschließend können Sie in einer Kategoriedefinition Ihrer Wahl auf NT_EVENT_LOG verweisen. Beispiel:

```
<category additivity="false" class="com.bristol.tvision.util.log.XCategory"
name="sensorLog">
  <priority class="com.bristol.tvision.util.log.XPriority" value="info"/>
  <appender-ref ref="NT_EVENT_LOG"/>
</category>
```

Unter Windows müssen Sie außerdem eine spezielle DLL zum Pfad hinzufügen. Diese DLL, **NTEventLogAppender.dll**, befindet sich im Verzeichnis **config\logging\system\bin**. Beispiel:

```
set path=%TVISION_HOME%\config\logging\system\bin;%PATH%
```

UNIX-Ereignisappender

Das folgende Beispiel zeigt, wie ein UNIX-Ereignisappender so konfiguriert wird, dass **syslog** verwendet wird:

```
<appender name="SYSLOG" class="tvision.org.apache.log4j.net. SyslogAppender">
  <param name="SyslogHost" value="localhost"/>
  <param name="Facility" value="local0"/>
  <layout class="tvision.org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="[%t] %-5p %c %x
- %m%n"/>
  </layout>
</appender>
```

Geben Sie für die Parameter **SyslogHost** und **Facility** die für Ihre Umgebung geeigneten Werte an.

Aktivieren der SMTP-Protokollierung

Der SMTP-Appender sendet eine E-Mail an den SMTP-Server, wenn ein Fehlerprotokollereignis mit dem angegebenen Schwellenwert den Appender erreicht. Fügen Sie Folgendes zur Datei **Analyzer.logging.xml** hinzu, um den SMTP-Appender zu aktivieren:

```
<appender name="EMAIL"
class="tvision.org.apache.log4j.net.SMTPAuthenticateAppender">
  <param name="SMTPHost" value="smtp.myserver.net"/>
  <param name="To" value="analyzer_log4j@myserver.net"/>
  <param name="From" value="administrator@myserver.net"/>
  <param name="UserName" value="smtp_user"/>
  <param name="Password" value=""/>
  <param name="Authenticate" value="true"/>
  <param name="BufferSize" value="1"/>
  <param name="Threshold" value="info"/>
  <layout class="tvision.org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%d [%t] %-5p %c %x - %m%n"/>
  </layout>
</appender>
```

Mit dem Schwellenwertparameter wird die Protokollierungsebene angegeben, bei der ein Eintrag im SMTP-Appender zulässig ist.

Fügen Sie den folgenden **EvaluatorClass**-Parameter hinzu, um eine angepasste Auswertung für das auslösende Ereignis zu definieren:

```
<param name="EvaluatorClass" class="tvision.org.apache.  
log4j.spi.TriggeringEventEvaluator"/>
```

Diese Schnittstelle stellt die folgende Funktion zur Verfügung, um zu bestimmen, wann eine E-Mail gesendet werden soll:

```
public boolean isTriggeringEvent(LoggingEvent event) {  
    long l = 0;  
    synchronized(lock) {  
        l = (msgCount ++);  
    }  
    return (((l + 1)%msgPkgSize) == 0); // fire email on every msgPkgSize events.  
}
```

Aktivieren der SNMP-Protokollierung

Der Appender **JoeSNMPTrapSender** sendet eine E-Mail, wenn ein Fehler einer angegebenen Fehlerebene auftritt. Fügen Sie die folgende Definition zur Datei **Analyzer.logging.xml** hinzu, um **JoeSNMPTrapSender** zu aktivieren:

```
<!-- SNMP TRAP appender !-->
<appender name="TRAP_LOG"
class="tvision.org.apache.log4j.ext.SNMPTrapAppender">
  <param name="ImplementationClassName"
value="tvision.org.apache.log4j.ext.JoeSNMPTrapSender"/>
  <param name="ManagementHost" value="127.0.0.1"/>
  <param name="ManagementHostTrapListenPort" value="162"/>
  <param name="EnterpriseOID" value="1.3.6.1.4.1.24.0"/>
  <param name="LocalIPAddress" value="127.0.0.1"/>
  <param name="LocalTrapSendPort" value="161"/>
  <param name="GenericTrapType" value="6"/>
  <param name="SpecificTrapType" value="12345678"/>
  <param name="CommunityString" value="public"/>
  <param name="ForwardStackTraceWithTrap" value="true"/>
  <param name="Threshold" value="DEBUG"/>
  <param name="ApplicationTrapOID" value="1.3.6.1.4.1.24.12.10.22.64"/>
  <layout class="tvision.org.apache.log4j.PatternLayout">
  <param name="ConversionPattern" value="%d,%p,[%t],[%c],%m%n"/>
  </layout>
</appender>
```

Hinweis: Sie müssen **joesntp.jar** zur CLASSPATH-Einstellung hinzufügen, da diese Datei für **JoeSNMPTrapSender** erforderlich ist. Diese JAR-Datei kann vom Projekt **JoeSNMP** unter <http://sourceforge.net/projects/joesntp> heruntergeladen werden.

Aktivieren der JMS-Protokollierung

TransactionVision bietet einen Mechanismus, um Protokollnachrichten über einen JMS-Nachrichtensystemprovider zu senden. Dies erfolgt über den Appender **com.bristol.tvision.appender.JMSAppender**, der in **Analyzer.logging.xml** konfiguriert wird.

Für die Konfiguration des JMS-Appenders stehen zwei Verfahren zur Verfügung. Normalerweise verwenden Sie JNDI-Einstellungen, um die JMS-Konnektivität zu konfigurieren; die direkte WMQ JMS-Konfiguration ist jedoch ebenfalls zulässig.

Wählen Sie aus, wie die Konfiguration erfolgen soll:

- ▶ über JNDI
- oder –
- ▶ direkt über WMQ JMS.

Hinweis: Wenn beide Methoden angegeben werden, haben die WMQ JMS-Einstellungen Vorrang; die JNDI-Einstellungen werden ignoriert.

Zum manuellen Konfigurieren der BPI JMS-Konnektivität oder zum Konfigurieren eines separaten Protokollierungsmechanismus, um Protokolle über JMS-Warteschlangen bereitzustellen, verwenden Sie die folgenden **JMSAppender**-Optionen:

- ▶ **ConnectionRetryDelay** gibt den Zeitraum an, der verstreicht, bevor im Falle eines Verbindungsfehlers ein Wiederholungsversuch unternommen wird.
- ▶ **ConnectionRetryTimeout** gibt an, wie lange gewartet wird, bevor bei einer nicht reagierenden Verbindung eine Zeitüberschreitung eintritt.
- ▶ **QueueName** ist der Name des JNDI-Objekts (falls JNDI verwendet wird) oder der tatsächliche Name der Warteschlange (bei WMQ JMS).
- ▶ **Username** und **Password** sind optionale Einstellungen, falls die Authentifizierung beim JMS-Provider erforderlich ist.

Für JNDI-Einstellungen

- **InitialContextFactoryName** ist der Klassenname der JNDI-Kontextfactory. Dieser Wert hängt vom verwendeten JMS-Anbieter ab (Details finden Sie in der entsprechenden Dokumentation). Beispiele hierfür sind **com.sun.jndi.fscontext.ReffSContextFactory** und **com.tibco.tibjms.naming.TibjmsInitialContextFactory**.
- **ProviderURL** ist die URL, um die Verbindung zum JNDI-Repository herzustellen. Der Wert hängt vom verwendeten JMS-Anbieter ab. Die URL einer **ReffSContextFactory**-Kontextfactory ähnelt der Folgenden: **file:/C:/jndi**. Für TIBCO könnte die URL z. B. folgendermaßen lauten: **tibjmsnaming://host:7222**.
- **QueueConnectionFactoryName** ist der Name des JNDI-Objekts für die Warteschlangen-Verbindungsfactory.

WMQ JMS-Einstellungen

Die WMQ JMS-spezifischen Einstellungen geben den Namen des Warteschlangen-Managers, den Host, den Port und den Kanal an, den Sie für die Verbindung mit WMQ JMS verwenden. Mit **TargetMQClient** aktivieren/deaktivieren Sie, ob MQ RFH2-Header in der JMS-Nachricht verwendet.

```
<appender class="com.bristol.tvision.appender.JMSAppender"
name="JMS_APPENDER">
  <!--connection retry interval in milliseconds -->
  <param name="ConnectionRetryDelay" value="0"/>
  <param name="ConnectionRetryTimeout" value="0"/>
  <param name="QueueName" value=""/>
  <param name="UserName" value=""/>
  <param name="Password" value=""/>
  <!-- enable the following to provide JNDI context parameters for
    JMS connection -->
  <!--<param name="InitialContextFactoryName" value="" />
  <param name="ProviderURL" value="" />
    <param name="QueueConnectionFactoryName" value=""/>
  -->

  <!-- enable the following to provide WebSphere MQ parameters for
    JMS connection -->
  <!--
  <param name="MQQueueManagerName" value="" />
  <param name="MQClientConnectionHost" value="" />
  <param name="MQClientConnectionPort" value="" />
  <param name="MQClientConnectionChannel" value="" />
  <param name="TargetMQClient" value="false"/>
  -->

</appender>
```

Teil III

Installation und Konfiguration von UI/Job Server

9

Vorbereiten der Installation von TransactionVision UI/Job Server

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zu TransactionVision UI/Job Server auf Seite 117
- TransactionVision UI/Job Server in der Bereitstellungsumgebung auf Seite 117

Informationen zu TransactionVision UI/Job Server

TransactionVision UI/Job Server ist eine Webapplikation, die im Paket mit dem Apache Tomcat-Servlet/JSP-Container verfügbar ist. TransactionVision UI/Job Server unterstützt die Kommunikation zwischen TransactionVision Analyzer und dem Business Availability Center-Gateway-Server.

TransactionVision UI/Job Server in der Bereitstellungsumgebung

In den meisten Bereitstellungsumgebungen sollte TransactionVision UI/Job Server auf demselben Host installiert werden, auf dem der Business Availability Center-Gateway-Server installiert ist.

Informationen zu den Systemanforderungen von TransactionVision UI/Job Server finden Sie unter "Unterstützte TransactionVision UI/Job Server-Plattformen" auf Seite 33.

10

Installieren von UI/Job Server auf UNIX-Plattformen

Dieses Kapitel umfasst die folgenden Themen:

- Installationsdateien auf Seite 119
- Starten des UI/Job Server-Installationsprogramms unter UNIX auf Seite 120
- Erstinstallation auf Seite 121
- Upgrade-Installation auf Seite 121
- Deinstallieren von UI/Job Server auf Seite 123

Installationsdateien

In der folgenden Tabelle sind die Namen der Installationsdateien der TransactionVision UI/Job Server-Pakete für die unterstützten Plattformen aufgeführt.

Plattform	Dateien
Solaris	tvuijs_800_sol_sparc.tar.gz

Starten des UI/Job Server-Installationsprogramms unter UNIX

Unter Solaris und Linux wird UI/Job Server im folgenden Verzeichnis installiert: **/opt/HP/TransactionVision**. Unter AIX wird UI/Job Server im Verzeichnis **/usr/lpp/HP/TransactionVision** installiert.

Führen Sie die folgenden Schritte aus, um TransactionVision UI/Job Server auf UNIX-Plattformen zu installieren.

- 1** Wechseln Sie zum Verzeichnis, in dem sich die TransactionVision-Installationsdateien befinden (entweder ein DVD-Laufwerk oder ein Downloadverzeichnis). **HINWEIS:** Unter Solaris und HP-UX müssen Sie die Installationsdateien stattdessen vom DVD-Laufwerk in ein temporäres Verzeichnis auf der Festplatte des Hostsystems kopieren.
- 2** Entpacken Sie die Pakete für Ihre Plattform; weitere Informationen finden Sie unter "Installationsdateien" auf Seite 119. Beispiel:

```
gunzip tvuijs_800_sol_sparc.tar.gz t
```

- 3** Melden Sie sich als Superuser an:

```
su
```

- 4** Geben Sie den folgenden Befehl ein, um das Installationsverfahren zu starten:

```
./tvininstall_800_unix.sh
```

Nachdem das Paket entzippt wurde, wird ein Menü angezeigt, in dem die verfügbaren Komponenten aufgeführt sind. Beispiel:

```
The following TransactionVision packages are available for installation:
```

1. TransactionVision Analyzer
2. TransactionVision UI/Job Server
3. TransactionVision WebSphere MQ Agent
4. TransactionVision User Event Agent

```
99. All of above
```

```
q. Quit install
```

```
Please specify your choices (separated by,) by number/letter:
```


Wenn TransactionVision-Komponenten erstmalig auf diesem Host installiert werden, fahren Sie gemäß der Beschreibung unter "Erstinstallation" fort. Falls eine frühere Version von TransactionVision auf dem Computer installiert ist, fahren Sie gemäß der Beschreibung unter "Upgrade-Installation" auf Seite 121 fort.

Erstinstallation

- 1 Geben Sie **2** ein und drücken Sie die EINGABETASTE.

Das Paket wird vom Installationskript installiert und die folgende Meldung wird angezeigt:

```
Installation of <TVUIJOB> was successful.
```

Das TransactionVision-Komponentenmenü wird angezeigt.

- 2 Geben Sie **q** ein und drücken Sie die EINGABETASTE, um den Installationsvorgang zu beenden.

Informationen zum Installieren weiterer Komponenten auf diesem Host finden Sie in den Installationsanweisungen für die betreffenden Komponenten.

Upgrade-Installation

- 1 Geben Sie **1** ein und drücken Sie die EINGABETASTE.

Wenn durch das Installationskript festgestellt wird, dass eine frühere UI/Job Server-Version installiert ist, wird die folgende Meldung angezeigt:

```
There is an earlier version of TransactionVision installed on the system.  
The earlier version has to be uninstalled before installing the current package(s).  
Continue with the uninstallation? (Y/N) [N]:
```

- 2** Geben Sie **Y** ein und drücken Sie die **EINGABETASTE**, um die vorherige Version zu deinstallieren. Die folgende Meldung wird angezeigt:

```
Before uninstalling the previous version, TransactionVision provides the option of
migrating configuration to the new installation:
Installation has detected a previous installation of TransactionVision.
Do you want to migrate existing TransactionVision configuration files? (y/n) [n]:
```

- 3** Geben Sie **Y** ein und drücken Sie die **EINGABETASTE**, um die Konfigurationsdateien zur neuen Version zu migrieren. Durch das Installationskript wird automatisch eine Sicherungskopie vorhandener UI/Job Server-Konfigurationsdateien für die Migration erstellt. Anschließend wird die vorherige UI/Job Server-Version deinstalliert und das TransactionVision-Komponentenmenü wird angezeigt. Fahren Sie mit Schritt 4 fort.

Wenn Sie **N** eingeben und die **EINGABETASTE** drücken, erhalten Sie die Möglichkeit, zu Referenzzwecken eine Sicherungskopie der Konfigurationsdateien zu erstellen:

```
Although migration will not be performed at this time, you may optionally back up
configuration files from your previous installation for reference purposes. Note that
answering N will overwrite these files, causing any existing setup information to be lost.
Do you wish to back up configuration files from the previous installation? (y/n) [n]:
```

Geben Sie **Y** ein und drücken Sie die **EINGABETASTE**, um eine Sicherungskopie der Konfigurationsdateien zu erstellen. Sie werden vom Installationskript aufgefordert, einen Speicherort für die Sicherung anzugeben:

```
Enter the directory to which existing TransactionVision configuration files should be
backed up [/opt/TVision/migrate_tv780_date_time]:
```

Drücken Sie die **EINGABETASTE**, um den Standardspeicherort zu verwenden, oder geben Sie den Pfad für das gewünschte Sicherungsverzeichnis ein. Vom Installationsdienstprogramm werden die folgenden Aufgaben ausgeführt:

- ▶ Kopieren der aktuellen Konfigurationsdateien in das angegebene Verzeichnis.
- ▶ Deinstallieren der vorherigen Installation von TransactionVision.

- Erneutes Anzeigen des TransactionVision-Komponentenmenüs.
- 4** Geben Sie **q** ein und drücken Sie die EINGABETASTE, um das Installationsverfahren zu beenden.

Deinstallieren von UI/Job Server

Führen Sie die folgenden Schritte aus, um eine beliebige TransactionVision-Komponente zu deinstallieren:

- 1** Melden Sie sich als Superuser an:

```
su
```

- 2** Geben Sie den folgenden Befehl ein:

```
./tvinstall_800_unix.sh -u
```

Das folgende Menü wird angezeigt (beachten Sie, dass die tatsächlich angezeigten Optionen von den installierten TransactionVision-Komponenten abhängen).

The following TransactionVision packages are available for installation:

1. TransactionVision UI/Job Server
3. TransactionVision WebSphere MQ Agent
4. TransactionVision User Event Agent

99. All of above
q. Quit install

Please specify your choices (separated by,) by number/letter:

Hinweis: Die tatsächlich angezeigten Optionen und Nummern hängen von den auf dem Computer verfügbaren Installationsdateien ab.

- 3** Geben Sie **1** ein und drücken Sie die EINGABETASTE.

Zum Deinstallieren aller TransactionVision-Komponenten geben Sie **99** ein und drücken dann die EINGABETASTE.

Die angegebenen Pakete werden durch das Installationsskript deinstalliert.
Anschließend wird erneut das Menü angezeigt.

- 4** Geben Sie **q** ein und drücken Sie die EINGABETASTE, um die Installation zu beenden.

11

Installieren von UI/Job Server unter Windows

Dieses Kapitel umfasst die folgenden Themen:

- Starten des UI/Job Server-Installationsprogramms unter Windows auf Seite 125
- Erstinstallation auf Seite 126
- Deinstallieren von UI/Job Server auf Seite 127

Starten des UI/Job Server-Installationsprogramms unter Windows

Führen Sie die folgenden Schritte aus, um UI/Job Server zu installieren:

- 1** Stellen Sie sicher, dass Sie entweder als Administrator oder als Benutzer mit Administratorberechtigungen am Zielsystem angemeldet sind.
- 2** Wenn Sie für UI/Job Server ein Upgrade von einer früheren Version durchführen, müssen Sie sicherstellen, dass die Umgebungsvariable JAVA_HOME festgelegt oder Java im Pfad angegeben ist.
- 3** Schließen Sie alle Windows-Programme, die momentan auf dem Computer ausgeführt werden. Dies umfasst auch Programme für die automatische Sicherung. Antiviren-, Antispyware- und andere Schutzprogramme müssen nicht heruntergefahren werden.
- 4** Doppelklicken Sie in Windows-Explorer auf **tvuijs_800_win.exe**. Die InstallShield-Willkommenseite wird angezeigt.
- 5** Klicken Sie auf **Next>** und warten Sie dann, bis die Willkommenseite des TransactionVision-Setupprogramms angezeigt wird.

- 6 Klicken Sie, wenn die InstallShield-Seite **Save Files** angezeigt wird, auf **Next>**, um den Standardordner zum Extrahieren der Installationsdateien zu verwenden (z. B. **C:\TEMP\Hewlett-Packard\TransactionVision**), oder klicken Sie auf **Change**, um den gewünschten Ordner auszuwählen. Klicken Sie danach auf **Next>**, um fortzufahren.

Wenn TransactionVision UI/Job Server erstmalig auf diesem Host installiert wird, fahren Sie gemäß der Beschreibung unter "Erstinstallation" auf Seite 126 fort. Falls eine frühere Version von UI/Job Server auf dem Computer installiert ist, fahren Sie gemäß der Beschreibung unter "Deinstallieren von UI/Job Server" auf Seite 127 fort.

Erstinstallation

Bei einer Erstinstallation wird die Willkommenseite des Setupprogramms angezeigt.

- 1 Klicken Sie auf der Willkommenseite des Setupprogramms auf **Next>**, um die Lizenzvereinbarung für TransactionVision anzuzeigen.
- 2 Klicken Sie auf **Yes**, um der Lizenzvereinbarung zuzustimmen. Die Seite **User Information** wird angezeigt.
- 3 Geben Sie Ihren Namen und den Namen Ihres Unternehmens ein und klicken Sie dann auf **Next>**. Die Seite **Destination Location** wird angezeigt.
- 4 Wenn Sie den Standardinstallationsordner (**C:\Programme\HP\TransactionVision**) verwenden möchten, klicken Sie auf **Next>**. Wenn Sie einen anderen Installationsordner verwenden möchten, klicken Sie auf **Browse...** Wählen Sie den gewünschten Installationsordner aus, und klicken Sie dann auf **Next>**.

Die ausgewählten Pakete werden am angegebenen Speicherort installiert. Die Seite **Setup Complete** wird angezeigt.

- 5 Klicken Sie auf **Finish**, um die Installation abzuschließen.

Deinstallieren von UI/Job Server

Führen Sie die folgenden Schritte aus, um UI/Job Server zu deinstallieren:

- 1** Wählen Sie im Startmenü den Befehl **Systemsteuerung** aus.
- 2** Doppelklicken Sie auf **Software**.
- 3** Wählen Sie das HP TransactionVision-Paket aus, das Sie deinstallieren möchten, und klicken Sie auf **Ändern/Entfernen**. Die Seite mit dem Wartungsmenü wird angezeigt.
- 4** Wählen Sie **Remove** aus, und klicken Sie auf **Next>**, um TransactionVision-Komponenten zu entfernen.
- 5** Klicken Sie auf **OK**, um die Deinstallation des angegebenen Pakets zu bestätigen. Das angegebene Paket wird deinstalliert. Die folgenden Dateien werden nicht gelöscht:
 - ▶ Alle Dateien, die nach der Installation hinzugefügt wurden.
 - ▶ Alle gemeinsam genutzten Dateien, die Paketen zugeordnet sind, die noch installiert sind.

Wenn es den Anschein hat, dass gemeinsam genutzte Dateien keinem installierten Paket zugeordnet sind (z. B. wenn alle anderen TransactionVision-Pakete deinstalliert wurden), wird die Seite **Shared File Detected** angezeigt.

- ▶ Wenn die gemeinsam genutzten Dateien installiert bleiben sollen, aktivieren Sie **Don't display this message again**, und klicken Sie dann auf **No**.
 - ▶ Wenn Sie die aktuelle Datei beibehalten möchten, diese Meldung jedoch für jede weitere gemeinsam genutzte Datei angezeigt werden soll, klicken Sie auf **No**.
 - ▶ Zum Löschen der gemeinsam genutzten Datei klicken Sie auf **Yes**.
- 6** Die Seite **Uninstallation Complete** wird angezeigt. Klicken Sie auf **Finish**, um die Deinstallation abzuschließen.

Hinweis: Nach der Deinstallation der TransactionVision-Webbenutzeroberfläche müssen Sie den temporären Cache und das Distributionsverzeichnis leeren. Dieser Vorgang wird nicht automatisch von WebSphere durchgeführt; ggf. vorhandene alte Dateien können dazu führen, dass eine neue Installation nicht ordnungsgemäß funktioniert.

12

Konfigurieren von UI/Job Server

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zum Konfigurieren von UI/Job Server auf Seite 129
- Dateien, die durch "TVisionSetupInfo" geändert werden auf Seite 130
- Informationen, die von "TVisionSetupInfo" angefordert werden auf Seite 130
- Ausführen von "TVisionSetupInfo" auf Seite 131
- Verwalten von UI/Job Server auf Seite 137

Informationen zum Konfigurieren von UI/Job Server

Mit TransactionVision wird das Dienstprogramm **TVisionSetupInfo** bereitgestellt, das Sie durch die Konfiguration von UI/Job Server führt. Dieses Dienstprogramm fordert Sie zur Angabe von Informationen auf, die für die Konfiguration benötigt werden, und legt erforderliche Umgebungsvariablen fest.

Wichtig: Vor der Konfiguration von UI/Job Server sollten Sie Analyzer konfigurieren. Informationen hierzu finden Sie in Kapitel 7, "Konfigurieren von Analyzer".

Nach dem Ausführen des Dienstprogramms können Sie einige optionale Konfigurationen vornehmen, wie weiter unten in diesem Kapitel beschrieben wird.

Nach Abschluss der Konfiguration müssen Sie UI/Job Server starten, wie ebenfalls weiter unten beschrieben wird.

Dateien, die durch "TVisionSetupInfo" geändert werden

Beim Konfigurieren von UI/Job Server werden die folgenden Dateien durch das Dienstprogramm **TVisionSetupInfo** geändert:

- ▶ `TVISION_HOME/config/datamgr/Database.properties`
- ▶ `TVISION_HOME/config/setup/Setup.properties`

Darüber hinaus werden von **TVisionSetupInfo** folgende Operationen ausgeführt:

- ▶ Speichern des Installationspfads für Softwarewerkzeuge in `TVISION_HOME/config/setup/DefaultInstallPath.xml`.
- ▶ Erzeugen der Datei `TVISION_HOME/bin/SetupEnv.[sh|bat]`, die von **TVisionSetup** ausgeführt wird, um `JAVA_HOME`, `CLASSPATH` und die Umgebungsvariablen für den Systembibliothekspfad festzulegen, die von `TransactionVision` benötigt werden.

Informationen, die von "TVisionSetupInfo" angefordert werden

Vom Dienstprogramm **TVisionSetupInfo** werden Sie zur Eingabe der folgenden Informationen aufgefordert, die benötigt werden, um die Konfiguration abschließen zu können. Sie können diese Informationsanforderungen vor dem Ausführen von **TVisionSetupInfo** durchgehen, sodass Sie vorbereitet sind und die geeigneten Antworten eingeben können.

Speicherort der Protokolldateien

Sie müssen den Pfad zu dem Verzeichnis angeben, in dem `TransactionVision` die UI/Job Server-Protokolldateien speichern soll. Der Standardwert ist `TVISION_HOME/logs`. Informationen hierzu finden Sie in Kapitel 8, "Konfigurieren der Analyser-Protokollierung".

Datenbankinformationen

Sie müssen den Typ der Datenbank, die Sie mit TransactionVision verwenden möchten (IBM DB2, Oracle oder SQL Server), sowie die datenbankspezifischen Eigenschaften angeben.

Die hier erfassten Informationen sind mit den Informationen identisch, die beim Einrichten von Analyzer erfasst werden. Informationen hierzu finden Sie unter "Datenbankinformationen" auf Seite 77.

Registrierung bei Business Availability Center

Sie müssen den Standort des Business Availability Center-Gateway-Servers angeben, der von den TransactionVision-Komponenten verwendet wird.

Ausführen von "TVisionSetupInfo"

- 1 Stellen Sie sicher, dass Sie entweder als Root-Benutzer, als Administrator oder als Benutzer mit Administratorberechtigungen am Zielsystem angemeldet sind.
- 2 Geben Sie `TVisionSetupInfo.[sh|bat]` ein, um das Skript `TVisionSetupInfo` auszuführen.

Betriebssystem	Skriptbefehl
AIX, Linux, Solaris	<TVISION_HOME>/bin/TVisionSetupInfo.sh
Windows	<TVISION_HOME>\bin\TVisionSetupInfo.bat

Hinweis: Bei der Installation wird die Umgebungsvariable `TVISION_HOME` auf den absoluten Pfad des Installationsverzeichnisses festgelegt. Unter Solaris würde `TVISION_HOME` beispielsweise `/opt/HP/TransactionVision` entsprechen; unter AIX wäre der Wert `/usr/lpp/HP/TransactionVision`.

TVisionSetupInfo fordert Sie zur Eingabe von Informationen auf, die zum Konfigurieren der folgenden Elemente erforderlich sind:

- Protokolldateien
- Datenbankeigenschaften
- Registrierung bei Business Availability Center

Wenn Eingabeaufforderungen von **TVisionSetupInfo** angezeigt werden, drücken Sie die EINGABETASTE, um die in Klammern angezeigten Standardwerte zu akzeptieren. Geben Sie andernfalls den richtigen Wert ein und drücken Sie dann die EINGABETASTE. Zum Angeben eines leeren Werts drücken Sie die LEERTASTE und drücken dann die EINGABETASTE. In den folgenden Abschnitten werden Beispieleingaben kursiv dargestellt.

Sobald **TVisionSetupInfo** einen Konfigurationsabschnitt abgeschlossen hat, werden Sie in einer Meldung darüber informiert, welche Dateien aktualisiert wurden.

Beispiel

Im Folgenden sehen Sie eine Beispielsitzung für die Ausführung von **TVisionSetupInfo** unter Windows.

This program collects configuration information in order to set up the TransactionVision environment. This includes:

- Location of software that TransactionVision depends upon such as the messaging middleware and the relational database system
- Parameters required to connect to the messaging middleware
- Parameters to connect to the database system
- Setup parameters for installed TransactionVision components

You will be prompted to input required configuration parameters.

If a default value is provided in [], pressing <Enter> will set the parameter to this default value. Pressing <Space><Enter> will set the parameter to an empty value.

Please specify name of the directory where you want to store your log files

[C:/PROGRA~1/HP/TRANSA~1\logs]:

TransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\config\setup\Setup.properties" has been successfully updated

Modifying *.Logging.xml files to use log file directory C:/PROGRA~1/HP/

TRANSA~1\logsTransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\config\logging\bpm\eventimporter_loggingconfig.properties" has been successfully updated

Please provide your TransactionVision license key: PreSalesTraining@hp.com-020EF5-815082021D0482C

TransactionVision Info(FileUpdated): File

"C:\PROGRA~1\HP\TRANSA~1\config\license\License.properties" has been successfully updated

Database Settings

Retrieving database configuration parameters...

Type of Database? (DB2/Oracle/SQLServer) [Oracle]:

Name of the host the database is running on [your_database_host]: ros89891duh.
rose.hp.com

Name of database TransactionVision connects to.

This is the Oracle database SID [your_database_name]: orcl

Enter the port number that the database listener is on [1521]:

Database user name []: system

User password []: rpctest

TransactionVision Info(FileUpdated): File
"C:\PROGRA~1\HP\TRANSA~1\config\datamgr\Database.properties" has been
successfully updated

LW-SSO (Lightweight Single Sign-On) Settings

Enter the LW-SSO init string, used for the initialization of the enc algorithm. This value
needs to be the same in all applications integ with LW-SSO. Unless you have modified
the default value in other app such as BAC, you can just accept the default. [Xy6stqZ]:

Enter the LW-SSO application domain, used for LW-SSO cookie creation. This is the full
or partial domain part of the fully-qualified hostn of the machine running the
TransactionVision UI. (for example, 'mydomain.com') [hpqcorp.net]:

Enter a comma delimited list of LW-SSO protected domains. Applicable if other
applications integrated with LW-SSO are running in different domains. It is not
necessary to include the TransactionVision UI domain (for example, if BAC is in domain
bac.com and Diagnostics is in domadiag.com, enter

'bac.com,diag.com'): TransactionVision Info(FileUpdated): File
"C:\PROGRA~1\HP\TRANSA~1\cproperties" has been successfully updated

Generating script file for environment setup...

TransactionVision Info(NewFileCreated): File "C:\PROGRA~1\HP\TRANSA~nv.bat"
has been successfully created

Initialize and verify database setup for TransactionVision

Do you wish to initialize the TransactionVision Database ? [y] :

Verifying database initialization for TransactionVision... TransactionVision
Info(DBInitialized): Database has been properly initialized

Register TransactionVision with Business Availability Center

Do you wish to register TransactionVision with Business Availability
Center at this time? [n] : y

Note: If your BAC server has Lightweight Single Sign On enabled, the hostnames of the TransactionVision UI and BAC servers must be a fully qualified domain name.

Please enter the hostname of the BAC server [] : ovresx1-vm11.rose.hp.com

Please enter the port of the BAC server [80] :

Do you wish to configure TransactionVision to access BAC using HTTPS? [n] :

TransactionVision has been configured to use the BAC server at: http://
ovresx1-vm11.rose.hp.com:80

Do you wish to register the TransactionVision UI with BAC? [y] :

Note: If your BAC server has Lightweight Single Sign On enabled, the hostnames of the TransactionVision UI and BAC servers must be a fully qualified domain name.

Enter TransactionVision UI host [ROS89891DUH.americas.hpqcorp.net] :

Do you wish to configure BAC to access TransactionVision using HTTPS? [n] :

Enter TransactionVision UI port [21000] :

Do you wish to configure BAC to access TransactionVision using HTTPS? [n] :

Enter TransactionVision UI port [21000] :

Do you wish to register a TransactionVision Analyzer for RUM Data Publishing with BAC? [y] :

Enter the TransactionVision Analyzer host for RUM Data Publishing
[ROS89891DUH.americas.hpqcorp.net] :

Enter RUM data publishing port [21113] :

User name for Basic Authentication to TransactionVision Analyzer used by Real User Monitor :

Password for Basic Authentication to TransactionVision Analyzer used by Real User Monitor :

The BAC server at `http://ovresx1-vm11.rose.hp.com:80` will be configured with:

TransactionVision Web Server: `http://`
ROS89891DUH.americas.hpqcorp.net:21000
TransactionVision URL for RUM: `http://`
ROS89891DUH.americas.hpqcorp.net:21113/tv_rum
TransactionVision User for RUM:
TransactionVision Password for RUM:

Please review the above settings and verify that they are correct.

Proceed with registration to BAC? [y] :

Registration with BAC was successful.

TransactionVision Info(TVisionSetupInfoSuccess): TVisionSetupInfo has completed successfully.

All program output and user input has been logged to
"C:/PROGRA~1/HP/TRANSA~1/logs/setup.log".

Please start HP Business Availability Center by running
%TVISION_HOME%\bin\SupervisorStart.bat

Informationen bei Beendigung der Ausführung

Bei Beendigung der Ausführung von **TVisionSetupInfo** wird die folgende Meldung angezeigt:

TransactionVision Info(TVisionSetupInfoSuccess): TVisionSetupInfo has completed successfully.

Neustart von Business Availability Center

Nach der Ausführung des Skripts **TVisionSetupInfo** kann ein Neustart von Business Availability Center notwendig sein. Wenn dies der Fall ist, wird der entsprechende Befehl von **TVisionSetupInfo** folgendermaßen angezeigt:

Unter Windows:

Please start HP Business Availability Center by running
%TVISION_HOME%\bin\SupervisorStart.bat

Unter UNIX:

Please start HP Business Availability Center by running \$TVISION_HOME/bin/
run_topaz start

Verwalten von UI/Job Server

Der UI/Job Server-Dienst wird vom HP Business Availability Center-Dienst verwaltet. Informationen hierzu finden Sie im *HP Handbuch zur Bereitstellung von HP Business Availability Center* (PDF).

13

Konfigurieren der UI/Job Server-Protokollierung

Dieses Kapitel umfasst die folgenden Themen:

- Protokolldateien auf Seite 139
- Umlaufprotokollierung auf Seite 139
- Ablaufprotokollierung auf Seite 141
- Verwenden von Windows- und UNIX-Systemprotokollen auf Seite 142

Protokolldateien

Standardmäßig werden alle Fehler- und Warnmeldungen von TransactionVision-Komponenten in den entsprechenden Protokolldateien protokolliert. Der Speicherort der Protokolldateien wird beim Ausführen von **TVisionSetupInfo** oder **SensorSetup** angegeben und in der Datei **Setup.properties** gespeichert.

Von UI/Job Server werden Fehlermeldungen in der Datei **ui.log** protokolliert.

Umlaufprotokollierung

In UI/Job Server wird standardmäßig eine Form der Umlaufprotokollierung verwendet. Wenn die Protokolldatei die konfigurierte Maximalgröße erreicht, wird sie als Sicherungsdatei umbenannt, und es wird eine neue, leere Protokolldatei erstellt. Die maximale Protokollgröße beträgt standardmäßig 10 MB und es gibt eine Sicherungsprotokolldatei.

Wenn die Standardeinstellungen verwendet werden und eine Protokolldatei (z. B. die UI/Job Server-Protokolldatei **ui.log**) die Größe von 10 MB erreicht, wird sie in **ui.log.1** umbenannt, und es wird eine neue Datei mit dem Namen **ui.log** erstellt. Wenn Sie die Konfiguration ändern, sodass zwei Sicherungsdateien verwendet werden, werden die folgenden Ereignisse ausgelöst, sobald **ui.log** eine Größe von 10 MB erreicht:

- ▶ Falls vorhanden wird **ui.log.2** entfernt.
- ▶ **ui.log.1** wird in **ui.log.2** umbenannt.
- ▶ **ui.log** wird in **ui.log.1** umbenannt.
- ▶ Eine neue Datei mit dem Namen **ui.log** wird erstellt.

Wenn Sie die Umlaufprotokollierung nicht verwenden möchten, können Sie die Konfiguration so ändern, dass die lineare Protokollierung verwendet wird. Hierbei wird nur eine einzige Protokolldatei erzeugt.

In den Dateien `<TVISION_HOME>/config/logging/*.Logging.xml` werden die Art der verwendeten Protokollierung, die maximale Größe der Protokolldatei und die Anzahl der Sicherungsprotokolldateien für jede Komponente angegeben. So gibt z. B. **Sensor.Logging.xml** die Konfiguration für die Servlet- und JMS-Sensoren an. Diese Datei enthält Einträge ähnlich den folgenden:

```
<appender class="tvision.org.apache.log4j.RollingFileAppender"
name="SENSOR_LOGFILE">
  <param name="File" value="C:\Program Files\ C:\Program
Files\HP\TransactionVision\logs\sensor.log"/>
  <param name="Append" value="true"/>
  <param name="MaxBackupIndex" value="2"/>
  <param name="MaxFileSize" value="10MB"/>
  <layout class="tvision.org.apache.log4j. PatternLayout">
    <param name="ConversionPattern" value="%d [%t] %-5p %c %x - %m%n"/>
  </layout>
</appender>
```

Maximale Größe der Protokolldatei

Wenn Sie die maximale Größe der Protokolldatei ändern möchten, ändern Sie den Wert des Parameters `MaxFileSize`, sodass er der gewünschten Dateigröße entspricht. Die angegebenen Werte sollten mit "MB" oder "KB" enden, um zwischen Mega- und Kilobytes zu unterscheiden.

Maximale Anzahl der Sicherungsprotokolldateien

Wenn Sie die Anzahl der Sicherungsdateien ändern möchten, ändern Sie den Wert des Parameters `MaxBackupIndex`, sodass er der gewünschten Anzahl an Sicherungsdateien entspricht.

Wechsel von der Umlauf- zur linearen Protokollierung

Führen Sie folgende Schritte aus, um die lineare Protokollierung anstelle der Umlaufprotokollierung zu verwenden:

- 1 Ändern Sie im `appender class`-Wert `RollingFileAppender` in `FileAppender`. Ändern Sie z. B. die erste Zeile des vorher aufgeführten Beispiels, sodass sie folgender Zeile entspricht:

```
<appender class="tvision.org.apache.log4j.FileAppender"  
name="SENSOR_LOGFILE">
```

- 2 Entfernen Sie die Einträge für die Parameter `MaxBackupIndex` und `MaxFileSize`.

Ablaufprotokollierung

Die Ablaufprotokollierung stellt ausführliche Informationen zu den internen Vorgängen bereit. Diese Form der Protokollierung wird in erster Linie für die Problembeseitigung verwendet und sollte nicht in Produktionsumgebungen zum Einsatz kommen.

Zum Aktivieren der Ablaufprotokollierung für die TransactionVision-Webbenutzeroberfläche legen Sie den Wert der Eigenschaft **trace** in der Datei `<TVISION_HOME>/config/ui/UI.properties` auf **on** fest. Nachdem Sie diese Konfigurationsdatei geändert haben, müssen Sie die Webbenutzeroberfläche neu starten, damit die Änderung wirksam wird. Zum Neustarten der TransactionVision-Webbenutzeroberfläche verwenden Sie die Verwaltungskonsole des Applikationsservers.

Verwenden von Windows- und UNIX-Systemprotokollen

Auf UNIX- und Windows-Plattformen können Sie TransactionVision so konfigurieren, dass die Ausgabe im Protokollierungsmechanismus für Systemereignisse – das Ereignisprotokoll unter Windows oder **syslog** unter UNIX – aufgezeichnet wird. Beispiele für die Protokollierungskonfigurationsdateien, die hierfür erforderlich sind, finden Sie in `TVISION_HOME/config/logging/system/*/Sensor.Logging.xml`.

Sowohl für Windows als auch für UNIX müssen Sie einen speziellen Ereignisappender definieren.

Windows-Ereignisappender

Im folgenden Beispiel wird gezeigt, wie der Windows-Ereignisappender so konfiguriert wird, dass das Ereignisprotokoll verwendet wird:

```
<appender name="NT_EVENT_LOG" class="tvision.org.apache.log4j.nt
.NTEventLogAppender">
  <layout class="tvision.org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%d [%t] - %m%n"/>
  </layout>
</appender>
```

Anschließend können Sie in einer Kategoriedefinition Ihrer Wahl auf NT_EVENT_LOG verweisen. Beispiel:

```
<category additivity="false" class="com.bristol.tvision.util. log.XCategory"
name="sensorLog">
  <priority class="com.bristol.tvision.util.log.XPriority" value="info"/>
  <appender-ref ref="NT_EVENT_LOG"/>
</category>
```

Unter Windows müssen Sie außerdem eine spezielle DLL zum Pfad hinzufügen. Diese DLL, **NTEventLogAppender.dll**, befindet sich im Verzeichnis **config\logging\system\bin**. Beispiel:

```
set path=%TVISION_HOME%\config\logging\system\bin;%PATH%
```

UNIX-Ereignisappender

Das folgende Beispiel zeigt, wie ein UNIX-Ereignisappender so konfiguriert wird, dass **syslog** verwendet wird:

```
<appender name="SYSLOG" class="tvision.org.apache.log4j.net. SyslogAppender">
  <param name="SyslogHost" value="localhost"/>
  <param name="Facility" value="local0"/>
  <layout class="tvision.org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="[%t] %-5p %c %x
- %m%n"/>
  </layout>
</appender>
```

Geben Sie für die Parameter **SyslogHost** und **Facility** die für Ihre Umgebung geeigneten Werte an.

Teil IV

Installation und Konfiguration von Sensoren und Agents

14

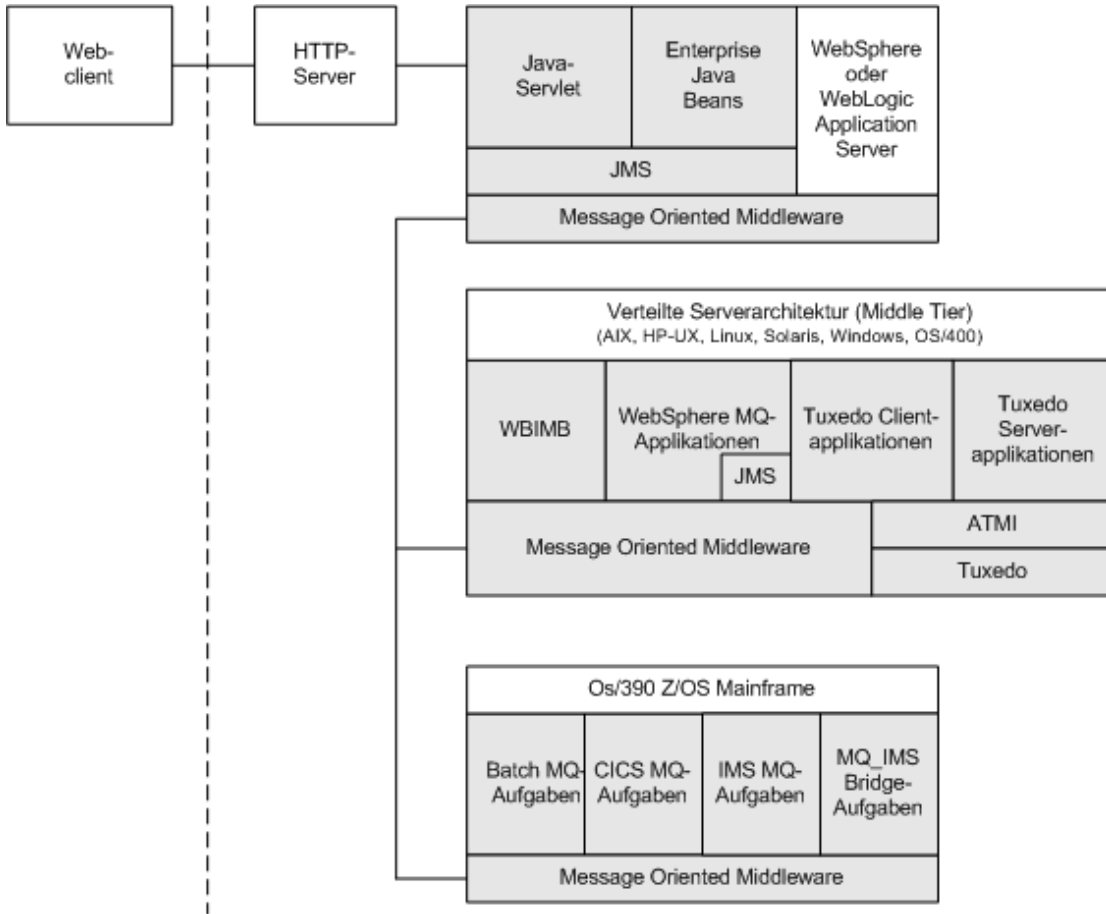
Vorbereiten der Installation von TransactionVision-Sensoren

Dieses Kapitel umfasst die folgenden Themen:

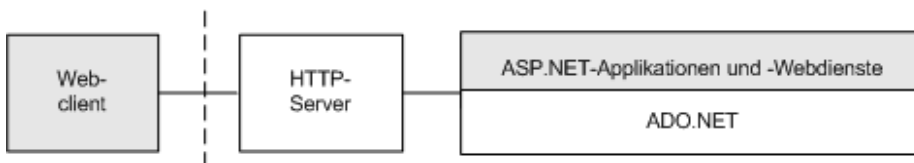
- Applikationen, die überwacht werden können auf Seite 148
- Verfügbare Sensor- und Agenttypen auf Seite 149

Applikationen, die überwacht werden können

Im folgenden Diagramm stellen grau hinterlegte Bereiche die Teile einer Webapplikation dar, für die TransactionVision Ereignisse verfolgen kann.



ASP.NET-Applikationen können ebenfalls überwacht werden:



.NET Remoting-Client- und -Serverapplikationen können ebenfalls überwacht werden.

NonStop TMF-Applikationen können ebenfalls überwacht werden:



Verfügbare Sensor- und Agenttypen

TransactionVision stellt die folgenden Sensor- und Agenttypen bereit:

- WebSphere MQ-Sensoren (WMQ)
- Java-Agent
 - Servlet-Sensor
 - JMS-Sensor
 - EJB-Sensor
 - JDBC-Sensor
- CICS-Sensor
- .NET-Agent
- BEA Tuxedo-Sensor
- NonStop TMF-Sensor

WebSphere MQ-Sensoren (WMQ)

Durch den **WebSphere MQ-Sensor** werden MQ-API-Aufrufe verfolgt. Diese API-Aufrufe umfassen den gesamten MQ-API-Satz mit den Haupt-APIs MQPUT, MQGET, MQCONN, MQDISC, MQOPEN, MQCLOSE usw. Es gibt zwei Arten von WebSphere MQ-Sensoren, die von TransactionVision auf verteilten Plattformen bereitgestellt werden: die WebSphere MQ-Sensorbibliothek und den Exit-Sensor für WebSphere MQ-APIs. Beide Sensortypen erfassen dieselben Informationen eines MQ-API-Aufrufs. Sie unterscheiden sich vor allem im Hinblick auf den Mechanismus, mit dem sie MQ-API-Aufrufe abfangen, hinsichtlich ihrer Verwendung und im Hinblick auf die Menge an Daten, die sie vom System erfassen.

- ▶ Die **WebSphere MQ-Sensorbibliothek** fängt einen WebSphere MQ-API-Aufruf mithilfe der Abfangmethode für gemeinsam genutzte Bibliotheken (DLLs) auf verteilten Plattformen ab. Hierzu ist es nötig, die TransactionVision-Sensorbibliotheken im Applikationsbibliothekspfad vor den WebSphere MQ-Bibliotheken zu platzieren. Diese Methode ist hilfreich, wenn Sie MQ-APIs für eine bestimmte Applikation verfolgen müssen.
- ▶ Der **Exit-Sensor für WebSphere MQ-APIs** nutzt die Unterstützung für WebSphere MQ-API-Exits, die in WebSphere MQ v5.3 und höher auf verteilten Plattformen verfügbar ist. Dieser Sensor wird als Exit für den Warteschlangen-Manager registriert und aufgerufen, wenn ein Programm, das eine Verbindung zum Warteschlangen-Manager herstellt, eine WebSphere MQ-API aufruft. Diese Methode empfiehlt sich, um MQ-Ereignisse aller Applikationen in einem Warteschlangen-Manager zu erfassen, insbesondere des Listeners und der Kanal-Agents.
- ▶ **z/OS WebSphere MQ-Sensoren** werden bereitgestellt, um MQ-API-Aufrufe in den CICS-, Batch- und IMS-Umgebungen des IBM-Systems z/OS zu verfolgen. In der CICS-Umgebung wird der vom CICS-Adapter für WebSphere MQ bereitgestellte API-Steuerübergabeexit verwendet, um MQ-API-Aufrufe abzufangen. In den Batch- und IMS-Umgebungen muss die Applikation erneut an den Sensor gebunden werden, um MQ-API-Aufrufe abzufangen.

Darüber hinaus sind die folgenden Sensoren für WebSphere MQ verfügbar:

- ▶ Der **Proxy-Sensor** korreliert Geschäftstransaktionen mit Prozessen, die nicht mithilfe der TransactionVision-Sensorbibliotheken überwacht werden (z. B. Ereignisse zwischen einer mit Sensoren überwachten Applikation und einer Applikation, die auf einem System ohne installierten Sensor ausgeführt wird, z. B. auf dem System eines externen Partners).
- ▶ Der **WebSphere Business Integration-Sensor** (früher als MQSI-Sensor bezeichnet) unterscheidet die verschiedenen Nachrichtenflüsse und identifiziert einzelne logische Transaktionspfade innerhalb von WBI. Bei diesem Sensor handelt es sich um ein WBI-Plugin. Dieses Plugin stellt einen Ablaufverfolgungsknoten bereit, der in den normalen Ausführungspfad eines Nachrichtenflusses eingefügt wird, sowie einen Fehlerknoten, der in den Fehlerpfad eines Nachrichtenflusses eingefügt wird. Durch diese Knoten wird ein MQSI2TRACE-Ereignis erzeugt, das die Verfolgung des Nachrichtenflusses in WBI ermöglicht.
- ▶ Der **WebSphere MQ IMS Bridge-Sensor** verfolgt WebSphere MQ IMS Bridge-Nachrichten anstelle der WebSphere MQ-API-Aufrufe, die von den aufrufenden Applikationen abgesetzt werden. MQ IMS Bridge ist eine Komponente, die es WebSphere MQ-Applikationen ermöglicht, IMS-Transaktionen aufzurufen und ihre Antwortnachrichten zu empfangen. Der MQ IMS Bridge-Sensor verfolgt MQ-Nachrichten, die an der Bridge ankommen, und korreliert sie mit der von IMS empfangenen Antwort. Für jede Nachricht, die die Bridge erreicht und die Bridge verlässt, werden die beiden Ereignisse MQIMS_BRIDGE_ENTRY und MQIMS_BRIDGE_EXIT erzeugt. Diese Ereignisse enthalten den MQ-Nachrichtenheader sowie Informationen über die aufgerufene IMS-Transaktion.

Java-Agent

- ▶ Der **Servlet-Sensor** verfolgt Servletmethoden in einem J2EE-Applikationsserver. Dieser Sensor verfolgt HTTP-Aufrufe wie HTTP_POST, HTTP_GET, HTTP_PUT usw., die zu Methodenaufrufen an den J2EE-Container führt. Der Servlet-Sensor verfolgt diese Methodenaufrufe, indem er das Servlet instrumentiert, um Ereignisse beim Eintritt und beim Verlassen jedes Aufrufs zu erfassen.

- ▶ Der **JMS-Sensor** verfolgt WebSphere MQ-JMS- (Java Message Service-) oder TIBCO EMS-Ereignisse von eigenständigen Java-Applikationen sowie von J2EE-Applikationsservern. Dieser Sensor verfolgt JMS-Schnittstellenmethoden wie **send**, **receive** usw. Diese Methoden werden verfolgt, indem die JMS-Bibliothek instrumentiert wird, um Ereignisse beim Eintritt und beim Verlassen jedes Aufrufs zu erfassen.
- ▶ Der **EJB-Sensor** verfolgt Transaktionen mittels Geschäftslogik innerhalb eines J2EE-Applikationsservers. Dieser Sensor verfolgt alle öffentlichen Geschäftsmethoden in einer Entity-Bean, Session-Bean oder nachrichtengesteuerten Bean. Zusätzlich zu den Geschäftsmethoden verfolgt dieser Sensor die Methoden **ejbCreate**, **ejbPostCreate**, **ejbRemove**, **ejbLoad**, **ejbStore** und **onMessage**. Diese Methoden werden vom Sensor instrumentiert, um Ereignisse beim Eintritt und beim Verlassen jedes Aufrufs zu erfassen.
- ▶ Der **JDBC-Sensor** ermöglicht es Benutzern, API- und Zeitablaufsinformationen zu SQL-Aufrufen und Transaktionen zu erfassen und zu analysieren, die über die JDBC-API für eine relationale Datenbank abgesetzt wurden.

Die Funktionalität der TransactionVision-Java-Sensoren (JMS, Servlet, EJB und JDBC) und der Diagnostics-Java-Probe werden zu einer einzigen Komponente, dem HP Diagnostics/TransactionVision-Java-Agent, zusammengefasst. Der Java-Agent instrumentiert und erfasst Ereignisse in Applikationen und sendet die Informationen an einen Diagnostics-Server und/oder an TransactionVision Analyzer. In dieser Version kann der Java-Agent so konfiguriert werden, dass er in einer Diagnostics-Umgebung als Java-Probe oder in einer TransactionVision-Umgebung als Java-Sensor fungiert. In integrierten Umgebungen kann der Agent gleichzeitig als Probe und als Sensor fungieren. Ausführliche Informationen finden Sie in Kapitel 15, "Installieren und Konfigurieren des Java-Agents".

CICS-Sensor

Der CICS-Sensor erfasst WebSphere MQ-fremde CICS-Ereignisse, um Transaktionen in einer Mainframe-Umgebung zu verfolgen. Der CICS-Sensor erfasst Daten für fünf Ereignistypen: Dateisteuerung, temporärer Speicher, transiente Daten, Intervallsteuerung und Programmsteuerung. Für alle Ereignistypen werden Informationen wie Transaktions-ID, Benutzer-ID, Terminal-ID und die SYSID erfasst. Ob weitere Informationen erfasst werden, hängt vom jeweiligen Ereignistyp ab.

.NET-Agent

Der .NET-Agent verfolgt Webdienste in der ASP.NET-Umgebung. Der .NET-Agent verfolgt diese Webdienst-Methodenaufrufe, indem er den .NET-Code instrumentiert, um Ereignisse beim Eintritt und beim Verlassen von Webdienstmethoden auf dem Server und ausgehende Webdienstaufrufe auf dem Client zu erfassen.

Der .NET-Agent verfolgt außerdem HTTP-Aufrufe wie HTTP_POST, HTTP_GET, HTTP_PUT in der ASP.NET-Umgebung sowie ADO- und Remoting-Ereignisse.

BEA Tuxedo-Sensor

Der BEA Tuxedo-Sensor überwacht Applikationen, die Tuxedo ATMI-Aufrufe in C- und C++-Umgebungen absetzen. Er fängt ATMI-Methoden ab und erfasst sie, wobei mindestens **tpenqueue**, **tpdequeue** und **tpcall** berücksichtigt werden.

Für die erfassten Methode werden zwei Erfassungsmodi unterstützt: API + Technologiedaten, API + Technologiedaten + Nutzdaten.

Dieser Sensor unterstützt außerdem das Filtern von Datensammlungen anhand von technologieübergreifenden Kriterien sowie anhand von Tuxedo ATMI-spezifischen Kriterien, einschließlich (jedoch nicht beschränkt auf) Tuxedo-Warteschlangenbereich, Warteschlangenname und Tuxedo-Dienstname.

NonStop TMF-Sensor

Der NonStop TMF-Sensor verfolgt den Zugriff auf ein überwachtes Enscribe-Dateisystem. Alle überwachten Transaktionen auf dem HP NonStop-System werden in TMF-Audit-Trails protokolliert. Da TMF alle überwachten Dateien im NonStop-System schützt, fungiert es als Repository aller Datenänderungen (Hinzufügungen, Löschungen, Änderungen) für das System als Ganzes.

Dieser Sensor liest die TMF-Audit-Trails und verfolgt den gesamten Zugriff auf Enscribe-Dateien, die die von einem Benutzer konfigurierten Filterbedingungen erfüllen.

15

Installieren und Konfigurieren des Java-Agents

In diesem Kapitel finden Sie Anweisungen zum Installieren und Konfigurieren des HP Diagnostics/TransactionVision-Java-Agents unter Windows und UNIX.

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Informationen zum Installieren und Konfigurieren des Java-Agents auf Seite 156
- ▶ Installieren und Konfigurieren des Java-Agents unter Windows auf Seite 157
- ▶ Installieren und Konfigurieren des Java-Agents unter UNIX auf Seite 172
- ▶ Unbeaufsichtigte Installation des Java-Agents auf Seite 183
- ▶ Ausführen des JRE-Instrumentierers auf Seite 184
- ▶ Konfigurieren der Applikationsserver auf Seite 195
- ▶ Konfigurieren von Nachrichtensystemprovidern auf Seite 195
- ▶ Konfigurieren von benutzerdefinierten Benutzerereignissen auf Seite 197

Informationen zum Installieren und Konfigurieren des Java-Agents

Mit dem Java-Agent wird die Funktionalität der TransactionVision-Java-Sensoren (JMS, Servlet, JDBC und EJB) der Diagnostics-Java-Probe in einer einzigen Komponente zusammengefasst. Der Java-Agent kann so konfiguriert werden, dass er in einer Diagnostics-Umgebung als Java Probe oder in einer TransactionVision-Umgebung als Java-Sensor fungiert. In integrierten Umgebungen kann der Agent auch gleichzeitig als Probe und als Sensor dienen.

Sie müssen folgende Operationen ausführen, um den Java-Agent als TransactionVision-Java-Sensor zu verwenden:

1 Installieren Sie den HP Diagnostics/TransactionVision-Java-Agent.

Der Java-Agent wird auf dem System installiert, das als Host für die zu überwachende Applikation dient. Informationen hierzu finden Sie unter "Installieren und Konfigurieren des Java-Agents unter Windows" auf Seite 157 und "Installieren und Konfigurieren des Java-Agents unter UNIX" auf Seite 172.

2 Konfigurieren Sie den Java-Agent.

Der Java-Agent wird so konfiguriert, dass er als TransactionVision-Java-Sensor, als J2EE-Probe oder beides fungiert. In diesem Handbuch finden Sie Anweisungen zum Konfigurieren des Java-Agents als TransactionVision-Java-Sensor. Informationen hierzu finden Sie unter "Konfigurieren des Java-Agents als TransactionVision-Java-Sensor unter Windows" auf Seite 160 und "Konfigurieren des Java-Agents als TransactionVision-Java-Sensor unter UNIX" auf Seite 175.

3 Konfigurieren Sie den Applikationsserver.

Damit der Java-Sensor eine Applikation überwachen kann, müssen Sie die JRE instrumentieren (Informationen hierzu finden Sie unter "Ausführen des JRE-Instrumentierers" auf Seite 184) und den Applikationsserver konfigurieren (Informationen hierzu finden Sie unter "Konfigurieren der Applikationsserver" auf Seite 195).

Installationsdateien

In der folgenden Tabelle sind die Namen der Installationsdateien des TransactionVision-Java-Agents für die einzelnen Plattformen aufgeführt.

Plattform	Dateien
Windows	JavaAgentSetup_win_8_00.exe
AIX	JavaAgentSetup_ibm_8_00.bin
Linux	JavaAgentSetup_linux_8_00.bin
Solaris	JavaAgentSetup_sol_8_00.bin

Installieren und Konfigurieren des Java-Agents unter Windows

In den folgenden Schritten finden Sie ausführliche Anweisungen zum Installieren des Java-Agents auf einem Windows-basierten Computer. Diese Anweisungen gelten auch, wenn Sie den Java-Agent auf einem UNIX-basierten Computer installieren und hierfür das grafische Installationsprogramm verwenden.

Wenn sich auf dem Hostcomputer eine ältere Installation des Java-Agents, der Legacy-J2EE-Probe oder der Legacysensoren von TransactionVision 5.0 befindet, müssen Sie sie deinstallieren, bevor Sie den Java-Agent installieren.

Dieser Abschnitt umfasst die folgenden Themen:

- Starten des Installationsprogramms unter Windows
- Ausführen der Installation unter Windows
- Konfigurieren des Java-Agents als TransactionVision-Java-Sensor unter Windows

Starten des Installationsprogramms unter Windows

Sie können das Installationsprogramm für den Java-Agent von der HP Software-Website, vom Diagnostics- oder TransactionVision-Produktträger oder von der Seite **Downloads** in Business Availability Center starten.

Sie müssen Mitglied der Gruppe **Administratoren** sein, um den Java-Agent installieren zu können.

So starten Sie das Installationsprogramm über die HP Software-Website:

- 1** Rufen Sie auf der HP Software-Website das Downloadcenter für HP BTO-Software auf.
- 2** Geben Sie **TransactionVision** in das Feld **Keyword** und **Trial Software** in das Feld **Refine Search by Resource Type** ein und klicken Sie auf **Search**.
- 3** Fahren Sie mit den Schritten unter "Ausführen der Installation unter Windows" auf Seite 159 fort.

So starten Sie das Installationsprogramm über Business Availability Center:

- 4** Wählen Sie **Admin > Platform** im Hauptmenü von Business Availability Center aus und klicken Sie auf die Registerkarte **Setup and Maintenance**.
- 5** Klicken Sie auf der Seite **Downloads** auf den entsprechenden Link, um das Java-Agent-Installationsprogramm für Windows herunterzuladen.
- 6** Fahren Sie mit den Schritten unter "Ausführen der Installation unter Windows" auf Seite 159 fort.

So starten Sie das Installationsprogramm über den Produkt-Installationsdatenträger für Business Availability Center:

- 1** Führen Sie die Datei **setup.exe** im Stammverzeichnis des Installationsdatenträgers aus. Das Diagnostics-Setupprogramm beginnt und zeigt die Seite mit dem Installationsmenü an.
- 2** Wählen Sie im Installationsmenü die Option **Diagnostics/TransactionVision Agent for Java** aus, um das Installationsprogramm zu starten.

So starten Sie das Installationsprogramm über die Produktinstallations-DVD für Business Availability Center:

- 3** Wählen Sie auf der Installations-DVD für HP Business Availability Center die ausführbare Datei für Ihre Plattform aus und führen Sie sie aus. Informationen hierzu finden Sie unter "Installationsdateien" auf Seite 157.
- 4** Fahren Sie mit den Schritten unter "Ausführen der Installation unter Windows" auf Seite 159 fort.

Ausführen der Installation unter Windows

Nachdem Sie das Installationsprogramm gestartet haben, wird die Softwarelizenzvereinbarung angezeigt. Sie können die Installation nun ausführen.

Hinweis: Schließen Sie alle Windows-Programme, die momentan auf dem Computer ausgeführt werden. Dies umfasst auch Programme für die automatische Sicherung. Antiviren-, Antispyware- und andere Schutzprogramme müssen nicht heruntergefahren werden.

So installieren Sie den Java-Agent auf einem Windows-basierten Computer:

- 1** Stimmen Sie der Endbenutzer-Lizenzvereinbarung zu.
Lesen Sie die Vereinbarung und wählen Sie **I accept the terms of the license agreement** aus.
Klicken Sie auf **Next**, um fortzufahren.
- 2** Geben Sie den Speicherort an, an dem der Agent installiert werden soll.
Übernehmen Sie das Standardverzeichnis oder wählen Sie einen anderen Speicherort aus, indem Sie entweder den Pfad zum Installationsverzeichnis in das Feld **Installation Directory Name** eingeben oder auf **Browse** klicken, um zum gewünschten Installationsverzeichnis zu navigieren.
Verzeichnisnamen dürfen nur Zeichen des lateinischen Alphabets enthalten.
Klicken Sie auf **Next**, um fortzufahren.

3 Überprüfen Sie die Zusammenfassungsinformationen.

Hier werden das Installationsverzeichnis und die Speicherplatzanforderungen aufgeführt.

Klicken Sie auf **Next**, um fortzufahren.

4 Überprüfen Sie die Zusammenfassungsinformationen zur Installation. Wenn die Zusammenfassungsinformationen keinen Fehler enthalten, klicken Sie auf **Next**, um fortzufahren.

Das Java-Agent-Setupmodul wird gestartet. Hierdurch wird die Konfiguration des Java-Agents eingeleitet.

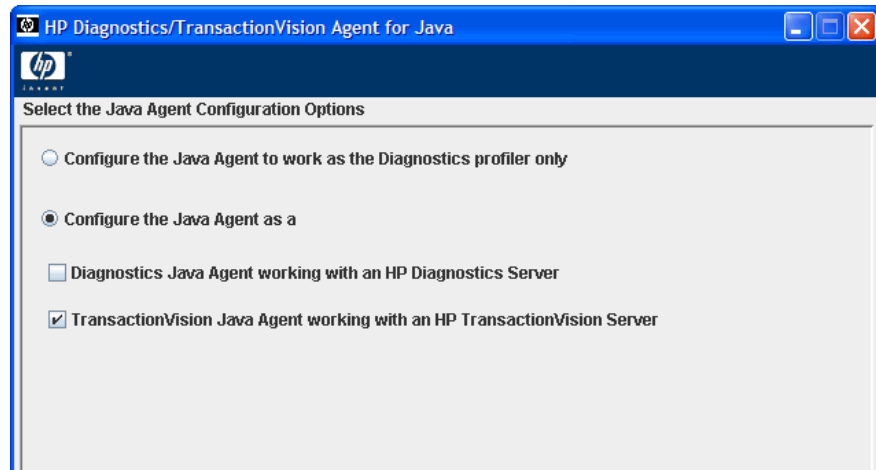
Konfigurieren des Java-Agents als TransactionVision-Java-Sensor unter Windows

In diesem Abschnitt finden Sie ausführliche Anweisungen, wie Sie den Java-Agent über die Benutzeroberfläche des Java-Agent-Setupmoduls als TransactionVision-Java-Sensor konfigurieren.

Das Java-Agent-Setupmodul wird automatisch am Ende der Installation des Java-Agents gestartet. Sie können das Setupmodul jederzeit starten, indem Sie **Start > Alle Programme > HP Java Agent > Setup Module** auswählen.

Führen Sie die folgenden Schritte aus, um den Java-Agent als TransactionVision-Java-Sensor zu konfigurieren:

- 1 Wählen Sie die Option **TransactionVision Java Agent working with an HP TransactionVision Server** aus.



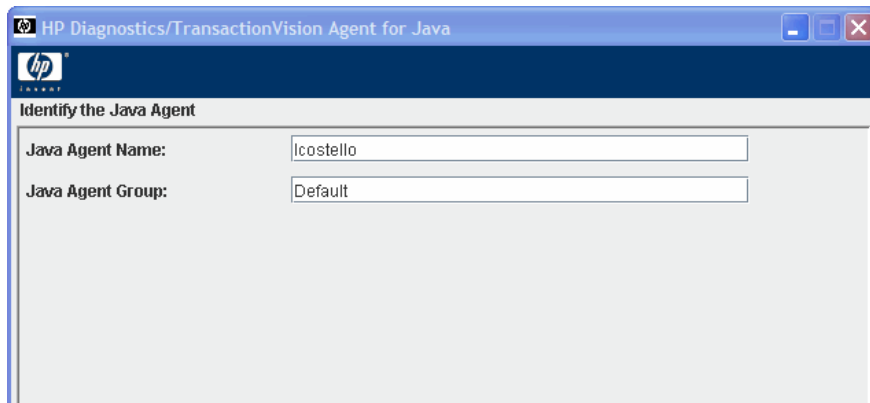
Ihnen stehen außerdem die folgenden Optionen zur Verfügung:

- **Configure the Java Agent to work as a Diagnostics Profiler only.** Wenn Sie den Java-Agent ausschließlich als Diagnostics-Profiler konfigurieren, finden Sie weitere Informationen unter "Configuring the Java Agent as a Profiler Only" im Handbuch *HP Diagnostics Installation and Configuration Guide*.
- **Configure the Java Agent as a Diagnostics Java Agent working with an HP Diagnostics Server.** Wenn Sie den Java-Agent als J2EE-Probe konfigurieren, um ihn mit einem Diagnostics-Server zu verwenden, finden Sie weitere Informationen unter "Configuring the Probe to Work with a Diagnostics Server" im Handbuch *HP Diagnostics Installation and Configuration Guide*.

- **Configure the Java Agent as both a Diagnostics Java Agent and a TransactionVision Java Agent.** Wenn Sie den Java-Agent als J2EE-Probe für den Einsatz mit einem Diagnostics-Server und gleichzeitig als TransactionVision-Java-Sensor konfigurieren, aktivieren Sie beide Kontrollkästchen und fahren dann mit Schritt 2 fort. Nach diesem Schritt müssen Sie den Java-Agent zuerst als J2EE-Probe (wie unter "Configuring the Java Agent to Work with a Diagnostics Server" im Handbuch *HP Diagnostics Installation and Configuration Guide* beschrieben) konfigurieren. Anschließend konfigurieren Sie den Java-Agent, beginnend mit Schritt 3, als J2EE-TransactionVision-Java-Sensor.

Klicken Sie auf **Next**, um fortzufahren.

- 2 Weisen Sie dem Java-Agent einen Namen zu und geben Sie die Gruppe an, der er angehört.



- Geben Sie einen Namen für den Java-Agent ein, der ihn eindeutig innerhalb von TransactionVision identifiziert. Die folgenden Zeichen dürfen im Namen verwendet werden: -, _ und alle alphanumerischen Zeichen. Der Agentname ist gleichzeitig der Name, der dem Java-Sensor zugewiesen wird.

Wenn Sie dem Agent einen Namen zuweisen, sollten Sie einen Namen auswählen, der Aufschluss über die Applikation, die der Agent überwacht, und über den Typ des Java-Sensors gibt.

- Geben Sie für den Java-Agent-Gruppennamen den Namen einer bestehenden Gruppe oder einer neu zu erstellenden Gruppe ein. Beim Agentgruppennamen wird die Groß- und Kleinschreibung beachtet.

Klicken Sie auf **Next**, um fortzufahren.

- 3 Wählen Sie den zu überwachenden Applikationsserver und sein Installationsverzeichnis aus.

Wählen Sie den Ereignistransportprovider aus und geben Sie die Anmeldeinformationen an.

HP Diagnostics/TransactionVision Agent for Java

Configure the TransactionVision Java Agent (page 1 of 2)

Supported Application Servers and Installation Path

WebSphere Application Server

WebLogic Application AppServer

None

Event Transport Provider

TransactionVision SonicMQ on Analyzer

Analyzer host:

WebSphere MQ

SonicMQ

TIBCO EMS

WebLogic JMS

Event Transport Provider Credentials

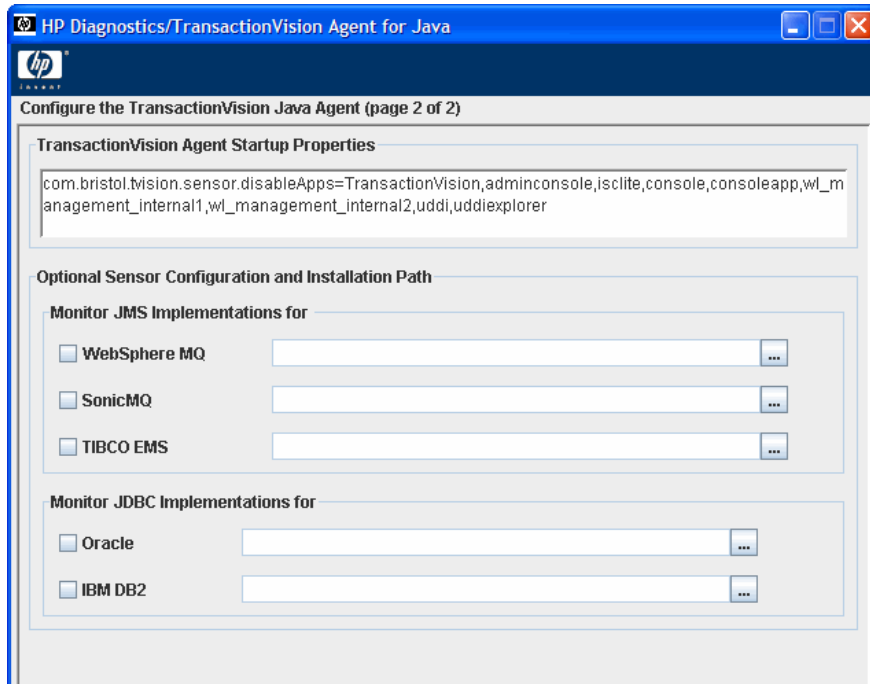
Configuration Queue:

Username (if required):

Password (if required):

- Sie können mit der rechten Maustaste in das Textfeld klicken, um ein Dateiauswahl-Dialogfeld zu öffnen.
- Wenn Sie den mit TransactionVision bereitgestellten SonicMQ-Ereignistransportprovider verwenden möchten, geben Sie den Hostnamen an, auf dem Analyzer ausgeführt wird.
- Ändern Sie den Namen der Konfigurationswarteschlange, falls TransactionVision Analyzer eine andere Warteschlange verwendet.

- ▶ Geben Sie den Benutzernamen und das Kennwort für den Ereignistransportprovider an, falls diese Angaben erforderlich sind.
 - ▶ Klicken Sie auf **Next**, um fortzufahren.
- 4 Konfigurieren Sie die zu überwachende JMS- und JDBC-Implementierung. Geben Sie den Pfad des Installationsverzeichnisses der entsprechenden Implementierung ein. Sie können mit der rechten Maustaste in das Textfeld klicken, um ein Dateiauswahl-Dialogfeld zu öffnen.



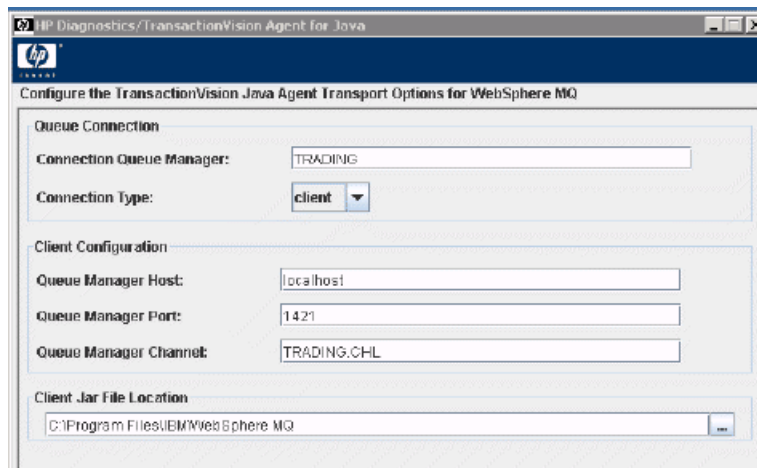
Das Feld **Transaction Agent Startup Properties** enthält Direktiven, die den Betrieb und die Instrumentierung des Agents betreffen. Falls Sie von der HP Kundenunterstützung keine anderen Anweisungen erhalten haben, sollten Sie hier normalerweise die Standardwerte übernehmen. Die Eigenschaften werden durch Semikolons getrennt.

Hinweis: BEA JMS für WebLogic 8.1.x wird automatisch überwacht; weitere Konfigurationsschritte sind nicht erforderlich, da die Schnittstelle auf WebLogic Application Server abgestimmt ist.

5 Konfigurieren Sie die JMS-Transporteinstellungen. Das nun angezeigte Dialogfeld hängt von der verwendeten Message Oriented Middleware ab:

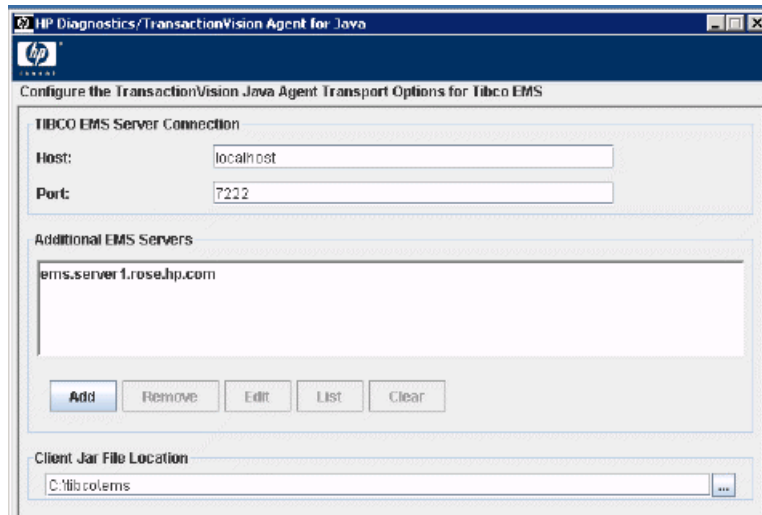
- ▶ Wenn Sie WebSphere MQ als Transport für die Kommunikationsverbindung auswählen, konfigurieren Sie die Transporteinstellungen, indem Sie als Verbindungstyp die WebSphere MQ-JMS-Serveranbindung oder die Clientverbindung angeben.

Hinweis: Unter 64-Bit-Windows mit WebSphere MQ 6.0 müssen Sie anstelle der Serververbindung die Clientverbindung auswählen, falls Sie eine 64-Bit-JVM überwachen möchten. WebSphere MQ 6.0 bietet keine Unterstützung für die Serveranbindung auf 64-Bit-Windows-Plattformen.



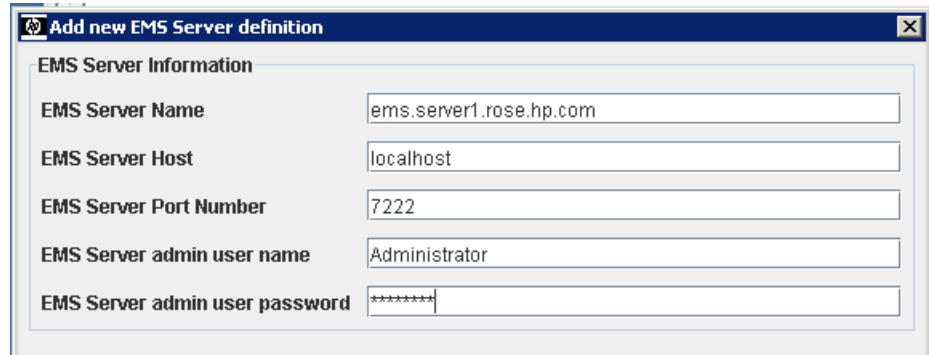
- ▶ Geben Sie den Namen des Konfigurationswarteschlangen-Managers ein.

- ▶ Wenn Sie die Clientverbindung ausgewählt haben, müssen Sie die clientseitigen Konfigurationsinformationen zum Warteschlangen-Manager angeben: Host, Port und Kanal.
- ▶ Geben Sie das Installationsverzeichnis für WebSphere MQ ein oder navigieren Sie zum gewünschten Verzeichnis.
- ▶ Wenn Sie TIBCO EMS als Transport für die Kommunikationsverbindung auswählen, müssen Sie die Transporteinstellungen für TIBCO EMS konfigurieren.



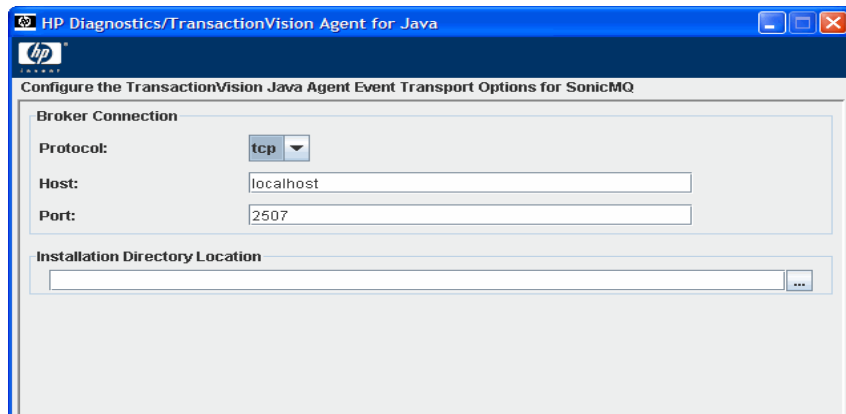
- ▶ Geben Sie den Hostnamen ein. Falls gewünscht, können Sie den Standardport ändern.
- ▶ Geben Sie das Installationsverzeichnis für TIBCO EMS ein oder navigieren Sie zum gewünschten Verzeichnis.

- Sie können EMS-Server hinzufügen und definieren. Klicken Sie auf **Add**, um ein Dialogfeld zum Hinzufügen von Servern zu öffnen.



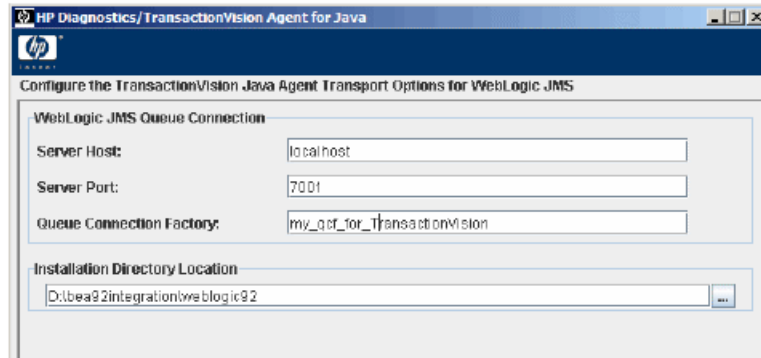
Geben Sie die Serverdefinitionen ein und klicken Sie auf **OK**. Zum Ändern der Definitionen wählen Sie den Server aus und klicken dann auf **Edit**. Klicken Sie auf **List**, um die Definitionen für einen ausgewählten Server anzuzeigen.

- Wenn Sie SonicMQ als Transport für die Kommunikationsverbindung auswählen, müssen Sie die Transporteinstellungen für SonicMQ konfigurieren.



- Verwenden Sie TCP (die Standardeinstellung) als Protokoll, sofern kein anderes Protokoll verwendet wird.
- Geben Sie den Hostnamen ein. Falls gewünscht, können Sie den Standardport ändern.

- Geben Sie das Installationsverzeichnis für SonicMQ ein oder navigieren Sie zum gewünschten Verzeichnis.
- Wenn Sie WebLogic JMS als Transport für die Kommunikationsverbindung auswählen, müssen Sie die Transporteinstellungen für WebLogic JMS konfigurieren.



- Geben Sie den Hostnamen ein. Falls gewünscht, können Sie den Standardport ändern.
- Geben Sie die Warteschlangen-Verbindungsfactory ein.
- Geben Sie das Installationsverzeichnis für WebLogic JMS ein oder navigieren Sie zum gewünschten Verzeichnis. Dieses Verzeichnis ist normalerweise mit dem Installationsverzeichnis des WebLogic-Applikationsservers identisch.

Sie können jederzeit jede beliebige Seite anzeigen, um Änderungen an der Konfiguration vorzunehmen.

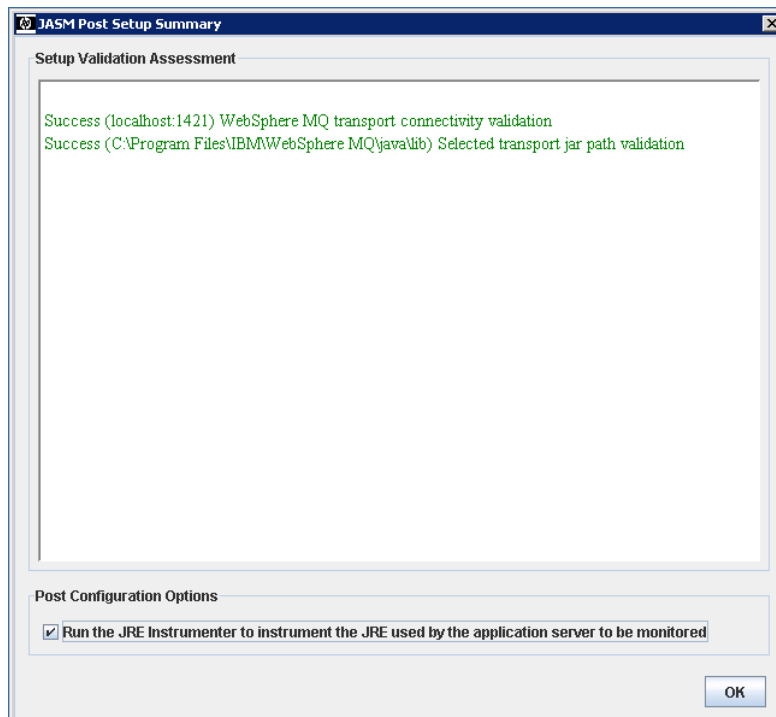
6 Speichern Sie die Konfiguration.

Wenn Sie alle gewünschten Einstellungen festgelegt haben, klicken Sie auf **Finish**, um die Konfiguration zu speichern. Hierdurch werden die Konfigurationsdateien von Diagnostics und TransactionVision geändert.

Optionen nach der Konfiguration

Nachdem die Konfigurationsdateien geändert wurden, erzeugt das Java-Agent-Setupmodul basierend auf der Version verschiedener auf dem System installierter Softwareprogramme automatisch eine Masterinstrumentierungsdatei. Dieser Vorgang nimmt einige Minuten in Anspruch. Währenddessen wird in einem Dialogfeld eine entsprechende Meldung angezeigt.

Sobald der Vorgang beendet ist, führt das Java-Agent-Setupmodul eine Reihe von Tests durch, um die Konfigurationseinstellungen zu überprüfen und zu testen:



Falls eine dieser Überprüfungen zu einem Fehler führt, müssen Sie die Transporteinstellungen überprüfen und sicherstellen, dass der JMS-Server oder der Warteschlangen-Manager mit den richtigen Einstellungen ausgeführt wird.

Sie haben die Möglichkeit, den JRE-Instrumentierer automatisch auszuführen, indem Sie **Run the JRE Instrumenter to instrument the JRE used by the application server to be monitored** aktivieren. Standardmäßig ist die Option für den JRE-Instrumentierer nicht ausgewählt. Wenn Sie eine JRE-Version verwenden, die älter als Version 1.5 ist, müssen Sie dieses Kontrollkästchen aktivieren oder den JRE-Instrumentierer manuell ausführen. Eine umfassende Beschreibung des JRE-Instrumentierers sowie Informationen zur manuellen Ausführung finden Sie unter "Ausführen des JRE-Instrumentierers" auf Seite 184.

Aktivieren des Java-Agents in Applikationen unter Windows

Für Java 1.5 und höher

- ▶ Wenn Sie den Java-Agent für die Überwachung einer Applikation aktivieren möchten, die in JRE 1.5 und höher ausgeführt wird, müssen Sie die folgende JVM-Option zur Java-Befehlszeile hinzufügen, mit der die Anwendung gestartet wird:

```
java -javaagent:<java_agent_install_verz>\DiagnosticsAgent\lib\probeagent.jar
```

Hierbei ist <java_agent_install_verz> der Pfad des Java-Agent-Installationsverzeichnisses. Der Standardpfad ist **C:\MercuryDiagnostics\JavaAgent**.

- ▶ Informationen zum Aktivieren des Java-Agents für Applikationsserver finden Sie unter "Konfigurieren der Applikationsserver" auf Seite 195.

Für Java 1.4

Für alle Applikationen oder Applikationsserver, die mit JRE, Version 1.4, ausgeführt werden (z. B. WebSphere 5.1, 6.0 oder WebLogic 8.1), müssen Sie den JRE-Instrumentierer des Java-Agents ausführen, um die JRE zu instrumentieren, die von der Applikation oder dem Applikationsserver verwendet wird. Umfassende Informationen hierzu finden Sie unter "Ausführen des JRE-Instrumentierers" auf Seite 184.

Aktivieren des Java-Agents in Applikationen unter UNIX

Für Java 1.5 und höher

- ▶ Wenn Sie den Java-Agent für die Überwachung einer Applikation aktivieren möchten, die in JRE 1.5 und höher ausgeführt wird, müssen Sie die folgende

JVM-Option zur Java-Befehlszeile hinzufügen, mit der die Anwendung gestartet wird:

```
java -javaagent:<java_agent_install_verz>/DiagnosticsAgent/lib/probeagent.jar
```

Hierbei ist <java_agent_install_verz> der Pfad des Java-Agent-Installationsverzeichnisses. Der Standardpfad ist **/opt/MercuryDiagnostics/JavaAgent**.

- Informationen zum Aktivieren des Java-Agents für Applikationsserver finden Sie unter "Konfigurieren der Applikationsserver" auf Seite 195.

Für Java 1.4

Für alle Applikationen oder Applikationsserver, die mit JRE, Version 1.4, ausgeführt werden (z. B. WebSphere 5.1, 6.0 oder WebLogic 8.1), müssen Sie den JRE-Instrumentierer des Java-Agents ausführen, um die JRE zu instrumentieren, die von der Applikation oder dem Applikationsserver verwendet wird. Umfassende Informationen hierzu finden Sie unter "Ausführen des JRE-Instrumentierers" auf Seite 184.

Überlegungen zu WebSphere MQ

Wenn Sie WebSphere MQ als Kommunikationstransport verwenden und bei der Konfiguration des Java-Agents **Server** (die Standardeinstellung) als Verbindungstyp ausgewählt haben, müssen Sie den Pfad zum WebSphere MQ-Verzeichnis **java/lib** zur Systemumgebungsvariablen für den Bibliothekspfad hinzufügen. Beispiel:

Fügen Sie unter AIX Folgendes hinzu:

```
set LIBPATH=$LIBPATH:/usr/mqm/java/lib
export LIBPATH
```

Fügen Sie unter Solaris oder Linux Folgendes hinzu:

```
set LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/mqm/java/lib
export LD_LIBRARY_PATH
```

Ersetzen Sie **lib** durch **lib64**, wenn Sie eine 64-Bit-JVM verwenden.

Unter Windows sind diese Einstellungen normalerweise nicht erforderlich, da der entsprechende Pfad bereits während der Installation von WebSphere MQ zur Umgebungsvariablen **PATH** hinzugefügt wurde.

Installieren und Konfigurieren des Java-Agents unter UNIX

Für verschiedene UNIX-Plattformen stehen Installationsprogramme für Java-Agents zur Verfügung. In den folgenden Anweisungen werden die Schritte beschrieben, die für die Installation des Java-Agents in den meisten UNIX-Umgebungen erforderlich sind. Die notwendigen Schritte werden sowohl für die Installation im grafischen Modus als auch für die Installation im Konsolenmodus beschrieben.

Die folgenden Anweisungen und Screenshots beziehen sich auf eine Agentinstallation auf einem AIX-basierten Computer. Die gleichen Anweisungen sollten auch für die anderen zertifizierten UNIX-Plattformen gelten.

Wenn sich auf dem Hostcomputer eine ältere Installation des Java-Agents, der Legacy-J2EE-Probe oder der Legacysensoren von TransactionVision 5.0 befindet, müssen Sie sie deinstallieren, bevor Sie den Java-Agent installieren.

Herunterladen des Installationsprogramms unter UNIX

Sie können das Installationsprogramm für den Java-Agent von der HP Software-Website, vom Diagnostics- oder TransactionVision-Produktträger oder von der Seite **Downloads** in Business Availability Center herunterladen.

Sie müssen als Root-Benutzer angemeldet sein, um den Java-Agent zu installieren.

So kopieren Sie das Installationsprogramm vom Produkt-Installationsdatenträger:

- 1 Kopieren Sie das Installationsprogramm **JavaAgentSetup<Plattform>_8_00.bin** vom Verzeichnis **<HP TransactionVision-Installationsdatenträger>/TransactionVision_Installers** auf den Computer, auf dem der TransactionVision-Server installiert werden soll.
- 2 Fahren Sie mit den Schritten unter "Ausführen der Installation unter UNIX" auf Seite 173 fort.

So laden Sie das Installationsprogramm von der Seite "Downloads" herunter (für Benutzer von Business Availability Center):

- 3 Wählen Sie **Admin > Platform** im Hauptmenü von Business Availability Center aus und klicken Sie auf die Registerkarte **Setup and Maintenance**.
- 4 Klicken Sie auf der Seite **Downloads** auf den Link zu dem für Ihre Umgebung geeigneten Installationsprogramm und speichern Sie das Installationsprogramm auf dem Computer, auf dem der Agent installiert werden soll.

Ausführen der Installation unter UNIX

Nachdem Sie das Installationsprogramm auf den Computer kopiert haben, auf dem der Java-Agent installiert werden soll, können Sie die Installation ausführen.

So installieren Sie den Java-Agent auf einem UNIX-basierten Computer:

- 1 Führen Sie das Installationsprogramm aus.

Ändern Sie, falls notwendig, den Modus der Installationsprogrammdatei, damit sie ausgeführt werden kann.

- Stellen Sie sicher, dass Sie als Root-Benutzer angemeldet sind.
- Geben Sie den folgenden Befehl an der UNIX-Eingabeaufforderung ein, um das Installationsprogramm im Konsolenmodus auszuführen:

```
./JavaAgentSetup_<Plattform>_8_00.bin -console
```

Das Installationsprogramm zeigt die Installationseingabeaufforderungen im Konsolenmodus an, wie in den folgenden Schritten gezeigt wird.

- Geben Sie den folgenden Befehl an der UNIX-Eingabeaufforderung ein, um das Installationsprogramm im grafischen Modus auszuführen:

```
xhost +          #ermöglicht das Anzeigen der Benutzeroberfläche an der  
Konsole
```

```
export DISPLAY=<Hostname>:0.0
```

```
./JavaAgentSetup_<Plattform>_7_50.bin
```

Die vom Installationsprogramm angezeigten Seiten sind mit den unter "Installieren und Konfigurieren des Java-Agents unter Windows" auf Seite 157 beschriebenen Seiten einer Installation unter Windows identisch.

2 Stimmen Sie der Endbenutzer-Lizenzvereinbarung zu.

Die Endbenutzer-Softwarelizenzvereinbarung wird angezeigt.

Lesen Sie die Vereinbarung. Beim Lesen können Sie die EINGABETASTE drücken, um zur nächsten Textseite zu blättern, oder **q** eingeben, um zum Ende der Lizenzvereinbarung zu springen.

Stimmen Sie den Bedingungen der Vereinbarung zu, indem Sie die Zahl **1** eingeben und dann die EINGABETASTE drücken.

Geben Sie **0** (null) ein und drücken Sie die EINGABETASTE. Geben Sie dann **1** ein und drücken Sie erneut die EINGABETASTE, um mit der Installation fortzufahren.

3 Geben Sie den Speicherort an, an dem der Agent installiert werden soll.

Übernehmen Sie an der Eingabeaufforderung **Installation Directory Name** das Standardinstallationsverzeichnis (in Klammern angezeigt) oder geben Sie den Pfad zu einem anderen Speicherort ein.

Geben Sie die Zahl **1** ein und drücken Sie die EINGABETASTE, um mit der Installation fortzufahren.

4 Überprüfen Sie das Installationsverzeichnis.

Das Installationsverzeichnis und die voraussichtliche Größe der Installation werden aufgeführt.

Wenn diese Angaben richtig bzw. vertretbar sind, geben Sie **1** ein und drücken Sie die EINGABETASTE, um die Installation zu starten.

Die Installation kann einige Minuten in Anspruch nehmen.

Das Java-Agent-Setupmodul wird gestartet.

Konfigurieren des Java-Agents als TransactionVision-Java-Sensor unter UNIX

In den folgenden Anweisungen werden die Schritte beschrieben, die für die Konfiguration des Java-Agents als TransactionVision-Java-Sensor in den meisten UNIX-Umgebungen erforderlich sind. Die notwendigen Schritte mithilfe des Java-Agent-Setupmoduls werden sowohl für den grafischen Modus als auch für den Konsolenmodus beschrieben.

Das Java-Agent-Setupmodul wird automatisch nach Beendigung des Installationsprogramms gestartet. Sie können das Java-Agent-Setupmodul jederzeit starten, indem Sie folgenden Befehl ausführen:

```
<java_agent_install_verz>/DiagnosticsAgent/bin/setupModule.sh
```

Hierbei ist <java_agent_install_verz> der Pfad des Java-Agent-Installationsverzeichnisses. Der Standardpfad ist `/opt/MercuryDiagnostics/JavaAgent`.

Konfigurieren des Java-Agents unter UNIX im grafischen Modus

So verwenden Sie den grafischen Modus, um den Java-Agent als Java-Sensor zu konfigurieren:

- 1 Richten Sie die grafische Schnittstelle ein.

Exportieren Sie die Anzeige zurück zum Terminal.

xhost + #ermöglicht das Anzeigen der Benutzeroberfläche an der Konsole

```
export DISPLAY=<Hostname>:0.0
```

- 2 Führen Sie das Java-Agent-Setupmodul aus.

```
<java_agent_install_verz>/DiagnosticsAgent/bin/setupModule.sh
```

Die vom Java-Agent-Setupmodul angezeigten Seiten sind mit den unter "Konfigurieren des Java-Agents als TransactionVision-Java-Sensor unter Windows" auf Seite 160 beschriebenen Seiten identisch, die bei einer Windows-basierten Installation mit dem Java-Agent-Setupmodul angezeigt werden.

Konfigurieren des Java-Agents unter UNIX im Konsolenmodus

So verwenden Sie den Konsolenmodus, um den Java-Agent als Java-Sensor zu konfigurieren:

- 3 Wählen Sie die Option **TransactionVision Java Agent working with an HP TransactionVision Server** aus, indem Sie **X** für diese Option eingeben.

```

*****
INFORMATION-> [Welcome]: Welcome to the Java Agent Setup Module (JASM)
INFORMATION-> [Init]: Initializing...
INFORMATION-> [Init]: Initializing... complete
INFORMATION-> [Options]:
At any input prompt you may enter:
'Enter' for default value,
'-' for Prev Step,
'+' for Next Step,
'c' to Cancel, or
'f' to Finish
*****
HP Diagnostics/TransactionVision Agent for Java
Focus: Select the Java Agent Configuration Options
Progress: 1 of 9
*****
PLEASE INPUT (X:Yes, 0:No)-> Diagnostics Profiler Only [0]:0
PLEASE INPUT (X:Yes, 0:No)-> Diagnostics Java Agent working with an HP Diagnosti
cs Server [0]:0
PLEASE INPUT (X:Yes, 0:No)-> TransactionVision Java Agent working with an HP Tra
nsactionVision Server [X]:X

```

- Geben Sie **O** (als Großbuchstaben) ein, um die Option **Diagnostics Profile Only** zu überspringen, und geben Sie erneut **O** ein, um auch die Option **Diagnostics Java Agent working with an HP Diagnostics Server** zu überspringen.
- Wenn Sie den Java-Agent so konfigurieren möchten, dass er als J2EE-Probe für einen Diagnostics-Server und auch als TransactionVision-Java-Sensor fungiert, geben Sie für beide Optionen **O** ein und fahren dann mit Schritt 2 fort.

Drücken Sie die EINGABETASTE, um fortzufahren.

- 4 Weisen Sie dem Java-Agent einen Namen zu und geben Sie die Gruppe an, der er angehört.

```

*****
HP Diagnostics/TransactionVision Agent for Java
Focus: Identify the Java Agent
Progress: 2 of 4
*****
PLEASE INPUT-> Java Agent Name [system_A rose.hp.com]:systemArose.hp.com
PLEASE INPUT-> Java Agent Group [Default]:
*****

```


- Geben Sie einen Namen für den Java-Agent ein, der ihn eindeutig innerhalb von TransactionVision identifiziert. Die folgenden Zeichen dürfen im Namen verwendet werden: -, _ und alle alphanumerischen Zeichen. Der Agentname ist gleichzeitig der Name, der dem Java-Sensor zugewiesen wird.

Wenn Sie dem Agent einen Namen zuweisen, sollten Sie einen Namen auswählen, der Aufschluss über die Applikation, die der Agent überwacht, und über den Typ des Java-Sensors gibt.

- Geben Sie für den Java-Agent-Gruppennamen den Namen einer bestehenden Gruppe oder einer neu zu erstellenden Gruppe ein. Beim Agentgruppennamen wird die Groß- und Kleinschreibung beachtet.

Drücken Sie die EINGABETASTE, um fortzufahren.

- 5 Legen Sie den zu überwachenden Applikationsserver fest und geben Sie das zugehörige Installationsverzeichnis ein. Wählen Sie den JMS-Anbieter aus, der als Kommunikationsverbindungstransport verwendet werden soll.

```
Focus: Configure the TransactionVision Java Agent <page 1 of 2>
Progress: 3 of 4
*****
PLEASE MAKE SELECTION-> Please select Application Server:
Selection 1. WebSphere
Selection 2. WebLogic
Selection 3. None
SELECT-> Please type in corresponding Number or
SELECT-> press Enter for default [WebSphere]:
1
PLEASE INPUT-> Please installation type in path for AppServer WebSphere [/usr/We
bSphere60/AppServer]:/usr/WebSphere60/AppServer
PLEASE MAKE SELECTION-> Please select Analyzer Communication Transport:
Selection 1. WebSphere MQ
Selection 2. Sonic MQ
Selection 3. TIBCO EMS
Selection 4. WebLogic JMS
SELECT-> Please type in corresponding Number or
SELECT-> press Enter for default [Sonic MQ]:
1
PLEASE INPUT-> Configuration Queue [TVISION.CONFIGURATION.QUEUE]:
PLEASE INPUT-> Username <if required> []:
PLEASE INPUT-> Password <if required> []:*
*****
```

- Ändern Sie den Namen der Konfigurationswarteschlange, falls Sie eine andere Warteschlange verwenden.
- Geben Sie den Benutzernamen und das Kennwort für den JMS-Provider an, falls diese Angaben erforderlich sind.
- Drücken Sie die EINGABETASTE, um fortzufahren.

6 Konfigurieren Sie die JMS- und JDBC-Implementierung, die Sie überwachen möchten.

```

*****
HP Diagnostics/TransactionVision Agent for Java
Focus: Configure the TransactionVision Java Agent (page 2 of 2)
Progress: 4 of 5
*****
PLEASE INPUT-> Additional TransactionVision Properties: [com.bristol.tvision.sen
sor.disableApps=TransactionVision,adminconsole,isclite,console,consoleapp,wl_man
agement_internal1,wl_management_internal2,uddi,uddiexplorer]
PLEASE INPUT (X:Yes, O:No)-> Monitor JMS Implementations for WebSphere MQ [O1:X
PLEASE INPUT-> Installation path for JMS transport WebSphere MQ [/usr/mqm]:/usr/
mqm
PLEASE INPUT (X:Yes, O:No)-> Monitor JMS Implementations for Sonic MQ [O1:O
PLEASE INPUT (X:Yes, O:No)-> Monitor JMS Implementations for TIBCO EMS [O1:O
PLEASE INPUT (X:Yes, O:No)-> Monitor JDBC Implementations for Oracle [O1:O
PLEASE INPUT (X:Yes, O:No)-> Monitor JDBC Implementations for IBM DB2 [O1:X
PLEASE INPUT-> Installation path for JDBC transport IBM DB2 [/usr/opt/db2_08_01]
:/usr/opt/db2_08_01
*****

```

Hinweis: Das Feld **Transaction Agent Startup Properties** enthält Direktiven, die den Betrieb und die Instrumentierung des Agents betreffen. Falls Sie von der HP Kundenunterstützung keine anderen Anweisungen erhalten haben, sollten Sie hier normalerweise die Standardwerte übernehmen. Die Eigenschaften werden durch Semikolons getrennt.

- ▶ Drücken Sie an der Eingabeaufforderung **Additional TransactionVision Properties** die EINGABETASTE.
- ▶ Wählen Sie einen oder mehrere JMS-Transporte aus, indem Sie neben jedem Transport **X** (als Großbuchstaben) für "Ja" oder **O** (als Großbuchstaben) für "Nein" eingeben. (Falls gewünscht können Sie "Nein" für alle Eigenschaften eingeben.)
- ▶ Wenn Sie für einen Transport "Ja" angeben, müssen Sie den Installationspfad für diesen Transport eingeben.
- ▶ Wählen Sie die JDBC-Datenbank aus, indem Sie neben jeder Datenbank **X** (als Großbuchstaben) für "Ja" oder **O** (als Großbuchstaben) für "Nein" eingeben.
- ▶ BEA JMS für WebLogic 8.1.x wird automatisch überwacht; weitere Konfigurationsschritte sind nicht erforderlich, da die Schnittstelle auf WebLogic Application Server abgestimmt ist.

7 Konfigurieren Sie die JMS-Transporteinstellungen.

- Wenn Sie WebSphere MQ als Transport für die Kommunikationsverbindung auswählen, konfigurieren Sie die Transporteinstellungen, indem Sie als Verbindungstyp die WebSphere MQ-JMS-Serveranbindung oder die Clientverbindung angeben.

```

*****
HP Diagnostics/TransactionVision Agent for Java
Focus: Configure the TransactionVision Java Agent Transport Options for WebSphere MQ
Progress: 5 of 5
*****
PLEASE INPUT-> Connection Queue Manager [configuration_qm_name]:TRADING
PLEASE MAKE SELECTION-> Please select Connection Type:
Selection 1. server
Selection 2. client
SELECT-> Please type in corresponding Number or
SELECT->   press Enter for default [server]:
2
PLEASE INPUT-> Host []:localhost
PLEASE INPUT-> Port []:1421
PLEASE INPUT-> Channel []:TRADING.CHL
PLEASE INPUT-> Install Path [/usr/mqm]:/usr/mqm
PLEASE INPUT->
CONFIRM-> Would you like to save your changes now? Enter Y or N: [Y]:N
*****

```

- Geben Sie den Namen des Konfigurationswarteschlangen-Managers ein und drücken Sie die EINGABETASTE.
- Wenn Sie die Clientverbindung ausgewählt haben, müssen Sie die clientseitigen Konfigurationsinformationen zum Warteschlangen-Manager angeben: Host, Port und Kanal.
- Geben Sie das WebSphere MQ-Installationsverzeichnis ein und drücken Sie die EINGABETASTE.
- Wenn Sie SonicMQ als Transport für die Kommunikationsverbindung auswählen, müssen Sie die Transporteinstellungen für SonicMQ konfigurieren.

```

*****
HP Diagnostics/TransactionVision Agent for Java
Focus: Configure the TransactionVision Java Agent Transport Options for Sonic MQ
Progress: 5 of 5
*****
PLEASE MAKE SELECTION-> Please select Broker Protocol Type:
Selection 1. tcp
SELECT-> Please type in corresponding Number or
SELECT->   press Enter for default [tcp]:
1
PLEASE INPUT-> Host [localhost]:
PLEASE INPUT-> Port [2506]:
PLEASE INPUT-> Install Path [/opt/Sonic75/MQ7.5]:/opt/Sonic75/MQ7.5
PLEASE INPUT->
CONFIRM-> Would you like to save your changes now? Enter Y or N: [Y]:

```

- ▶ Geben Sie den Namen des Protokolls ein und drücken Sie die EINGABETASTE.
- ▶ Geben Sie den Hostnamen ein und drücken Sie die EINGABETASTE.
- ▶ Drücken Sie die EINGABETASTE, um den Standardport auszuwählen, oder geben Sie einen anderen Port ein und drücken Sie dann die EINGABETASTE.
- ▶ Geben Sie das SonicMQ-Installationsverzeichnis ein und drücken Sie die EINGABETASTE.
- ▶ Wenn Sie TIBCO EMS als Transport für die Kommunikationsverbindung auswählen, müssen Sie die Transporteinstellungen für TIBCO EMS konfigurieren.

```
*****
HP Diagnostics/TransactionVision Agent for Java
Focus: Configure the TransactionVision Java Agent Transport Options for Tibco EMS
Progress: 5 of 5
*****
PLEASE INPUT-> Host [l:localhost
PLEASE INPUT-> Port [7222]:
PLEASE INPUT-> Install Path [l:/opt/tibco/ems
INFORMATION-> [Existing TIBCO EMS Server Definitions]:
PLEASE INPUT-> Option? <A:Add, Rn: Remove, En: Edit, or 0: Exit> [0]:A
PLEASE INPUT-> EMS Server Name [l:ems_server1.rose.hp.com
PLEASE INPUT-> EMS Server Host [l:localhost
PLEASE INPUT-> EMS Server Port Number [l:7222
PLEASE INPUT-> EMS Server admin user name [l:Administrator
PLEASE INPUT-> EMS Server admin user password [l:*****
INFORMATION-> [Existing TIBCO EMS Server Definitions]:
1: ems_server1.rose.hp.com
   URL: tcp://ems_server1.rose.hp.com:7222/
   Username: Administrator
   Password: *****
PLEASE INPUT-> Option? <A:Add, Rn: Remove, En: Edit, or 0: Exit> [0]:0
```

- ▶ Geben Sie den Hostnamen ein und drücken Sie die EINGABETASTE.
- ▶ Drücken Sie die EINGABETASTE, um den Standardport auszuwählen, oder geben Sie einen anderen Port ein und drücken Sie dann die EINGABETASTE.
- ▶ Geben Sie das TIBCO EMS-Installationsverzeichnis ein.
- ▶ Sie können EMS-Server hinzufügen und definieren:
Geben Sie **A** ein und drücken Sie die EINGABETASTE. Geben Sie die einzelnen Serverdefinitionen ein und drücken Sie nach jeder eingegebenen Definition die EINGABETASTE.

- Wenn Sie WebLogic JMS als Transport für die Kommunikationsverbindung auswählen, müssen Sie die Transporteinstellungen für WebLogic JMS konfigurieren.

```
*****
HP Diagnostics/TransactionVision Agent for Java
Focus: Configure the TransactionVision Java Agent Transport Options for WebLogic
JMS
Progress: 5 of 5
*****
PLEASE INPUT-> Server Host []:localhost
PLEASE INPUT-> Server Port [7001]:
PLEASE INPUT-> Queue Connection Factory []:myQueueFactory
PLEASE INPUT-> Install Path []:/opt/boa/weblogic81
```

- Geben Sie den Hostnamen ein und drücken Sie die EINGABETASTE.
 - Drücken Sie die EINGABETASTE, um den Standardport auszuwählen, oder geben Sie einen anderen Port ein und drücken Sie dann die EINGABETASTE.
 - Geben Sie die Warteschlangen-Verbindungsfactory und drücken Sie die EINGABETASTE.
 - Geben Sie das WebLogic JMS-Installationsverzeichnis ein und drücken Sie die EINGABETASTE. Dieses Verzeichnis ist normalerweise mit dem Installationsverzeichnis des WebLogic-Applikationsservers identisch.
- 8** Geben Sie **Y** ein, wenn Sie aufgefordert werden, die Änderungen im Java-Agent-Setup modul zu speichern.

9 Instrumentieren Sie die JRE.

```

CONFIRM-> Would you like to save your changes now? Enter Y or N: [Y]:Y
INFORMATION-> [Save]: Saving Dialog Select the Java Agent Configuration Options
INFORMATION-> [Save]: Saving Dialog Identify the Java Agent
INFORMATION-> [Save]: Saving Dialog Configure the Diagnostics Java Agent
INFORMATION-> [Save]: Saving Dialog Configure the TransactionVision Java Agent (
page 1 of 2)
INFORMATION-> [Save]: Saving Dialog Configure the TransactionVision Java Agent (
page 2 of 2)
INFORMATION-> [Save]: Saving Dialog Configure the TransactionVision Java Agent T
ransport Options for WebSphere MQ
INFORMATION-> [Save]: Saving Dialog Configure the TransactionVision Java Agent T
ransport Options for Sonic MQ
INFORMATION-> [Save]: Saving Dialog Configure the TransactionVision Java Agent T
ransport Options for Tibco EMS
INFORMATION-> [Save]: Saving Dialog Configure the TransactionVision Java Agent T
ransport Options for WebLogic JMS
INFORMATION-> [Save]: Saving Diagnostics Agent property files
INFORMATION-> [Save]: Saving TransactionVision Agent property files
INFORMATION-> [Merge]: Merging TransactionVision Rules files...please wait
INFORMATION-> [JASM Post Setup Summary]:
Success (localhost:2506) Sonic MQ transport connectivity validation
Success (/opt/Sonic75/MQ7.5/lib) Selected transport jar path validation
INFORMATION-> [Currently Instrumented VMs]:
IBM 1.5.0 (/usr/java5/jre)
IBM 1.4.2 (/usr/java14/jre)
IBMJ9 1.4.2 (/usr/java14/jre/lib/jc1SC14)
IBM 1.5.0 (/usr/java/jre)
PLEASE INPUT-> Option? (<'Command', H: Help, or 0: Exit) [0]:

```

Geben Sie die Instrumentierungsbefehle wie unter "Ausführen des JRE-Instrumentierers auf einem UNIX-basierten Computer" auf Seite 191 beschrieben ein.

10 Geben Sie 0 (null) ein, um das Setup abzuschließen.

Nachdem Sie den Agent installiert, ihn als Java-Probe konfiguriert und die JRE instrumentiert haben, müssen Sie bestimmte Aufgaben ausführen, die sich an die Installation anschließen.

11 Ändern Sie das Startskript für den Applikationsservers, sodass die Probe zusammen mit der überwachten Applikation gestartet wird.

Ausführliche Anweisungen finden Sie unter "Konfigurieren der Applikationsserver" auf Seite 195.

12 Überprüfen Sie die Installation des Java-Agents.

Unbeaufsichtigte Installation des Java-Agents

Eine *unbeaufsichtigte Installation* ist eine Installation, die automatisch ausgeführt wird. Benutzerinteraktionen sind hierbei nicht erforderlich. Anstelle von Benutzereingaben werden bei der unbeaufsichtigten Installation für jeden Installationsschritt Eingaben aus einer Antwortdatei übernommen.

Angenommen, ein Systemadministrator muss eine Komponente auf mehreren Computern bereitstellen. Dieser Administrator könnte eine Antwortdatei erstellen, die alle erforderlichen Konfigurationsinformationen enthält, und dann eine unbeaufsichtigte Installation auf mehreren Computern ausführen. Auf diese Weise entfällt die Notwendigkeit manueller Eingaben während des Installationsverfahrens.

Bevor Sie eine unbeaufsichtigte Installation auf mehreren Computern ausführen, müssen Sie eine Antwortdatei erzeugen, die während des Installationsverfahrens die notwendigen Eingaben bereitstellt. Diese Antwortdatei kann für alle unbeaufsichtigten Installationen verwendet werden, für die während der Installation dieselben Eingaben erforderlich sind.

Bei der unbeaufsichtigten Installation werden zwei Antwortdateien verwendet: eine für die Installation des Java-Agents und eine für das Java-Agent-Setupmodul:

So erzeugen Sie eine Antwortdatei für die Installation des Java-Agents:

Führen Sie eine reguläre Installation mit der folgenden Befehlszeilenoption durch:

```
<installationsprogramm> -options-record <install_antwortdateiname>
```

Hierdurch wird eine Antwortdatei erstellt, die alle während der Installation übermittelten Informationen einschließt.

So erzeugen Sie eine Antwortdatei für das Java-Agent-Setupprogramm:

Führen Sie das Java-Agent-Setupprogramm mit der folgenden Befehlszeilenoption aus.

- Unter Windows:

```
<java_agent_install_verz>\bin\setupModule.cmd -createBackups -console  
-recordFile <JASM_antwortdateiname>
```

- Unter UNIX:

```
<java_agent_install_verz>/bin/setupModule.sh -createBackups -console  
-recordFile <JASM_antwortdateiname>
```

Durch beide Befehle wird eine Antwortdatei erstellt, die alle während der Installation übermittelten Informationen einschließt.

So führen Sie eine unbeaufsichtigte Installation oder Konfiguration durch:

Führen Sie mithilfe der entsprechenden Antwortdatei eine unbeaufsichtigte Installation oder Konfiguration durch.

Geben Sie Folgendes ein, um eine Umgebungsvariable festzulegen und die Befehlszeilenoption `-silent` zu verwenden:

```
set HP_JAVA_AGENT_SETUP=-DoNotRun  
<installationsprogramm> -options <install_antwortdateiname> -silent
```

Geben Sie dann Folgendes ein:

```
set HP_JAVA_AGENT_SETUP=  
cd <setupmodul> -createBackups -console -installFile  
<JASM_antwortdateiname>
```

Beachten Sie, dass Sie auf UNIX-Systemen `"-DoNotRun"` in Anführungszeichen einschließen müssen.

Ausführen des JRE-Instrumentierers

Der JRE-Instrumentierer instrumentiert die Klasse **ClassLoader** für die JVM, die von der Applikation verwendet wird, und legt das instrumentierte Klassenladeprogramm in einem Ordner unter dem Verzeichnis `<java_agent_install_verz>/DiagnosticsAgent/classes` ab. Außerdem stellt der Instrumentierer den JVM-Parameter bereit, den Sie beim Starten einer Applikation oder eines Applikationsservers verwenden müssen, damit der Applikationsserver das instrumentierte Klassenladeprogramm verwendet.

Wenn sich das JDK (die ausführbare Datei **java.exe**) ändert, das vom Applikationsserver verwendet wird, müssen Sie den JRE-Instrumentierer erneut ausführen, damit der Java-Agent die Verarbeitung weiter überwachen kann.

Hinweise:

- ▶ Wenn Sie die IMB-JRE 1.4.2 J9 instrumentieren möchten, müssen Sie das richtige Klassenladeprogramm instrumentieren und dann die Option **-Xj9** in der Befehlszeile der Applikation hinzufügen. Das richtige Klassenladeprogramm befindet sich im Verzeichnis `<java_verz>\jre\lib\jclSC14` (z. B. `jreinstrumenter.sh -i \usr\java14_64\jre\lib\jclSC14`).
- ▶ Wenn der Java-Agent zum Überwachen mehrerer JVMs verwendet wird, muss der JRE-Instrumentierer einmal für jede JVM ausgeführt werden. Auf diese Weise kann der Java-Agent vorbereitet werden, um die auf jeder JVM ausgeführten Applikationen zu instrumentieren. Ausführliche Informationen finden Sie unter "Configuring the Probes for Multiple Application Server JVM Instances" im Handbuch *HP Diagnostics Installation and Configuration Guide*.

Verarbeitungsschritte des JRE-Instrumentierers

Der JRE-Instrumentierer führt die folgenden Aufgaben aus:

- ▶ Identifizieren der JVMs, die für die Instrumentierung verfügbar sind.
- ▶ Suchen nach weiteren JVMs in den von Ihnen angegebenen Verzeichnissen.
- ▶ Instrumentieren der von Ihnen angegebenen JVMs und Bereitstellen des Parameters, den Sie dem Startskript für die JVM hinzufügen müssen, damit es auf den Speicherort der instrumentierten **ClassLoader**-Klasse zeigt.

Manuelles Ausführen des JRE-Instrumentierers

Im Folgenden finden Sie Anweisungen zum Ausführen des JRE-Instrumentierers in einer Windows-Umgebung und im Konsolenmodus in einer UNIX-Umgebung.

Dieser Abschnitt umfasst die folgenden Themen:

- "Ausführen des JRE-Instrumentierers auf einem Windows-basierten Computer" auf Seite 186
- "Ausführen des JRE-Instrumentierers auf einem UNIX-basierten Computer" auf Seite 191

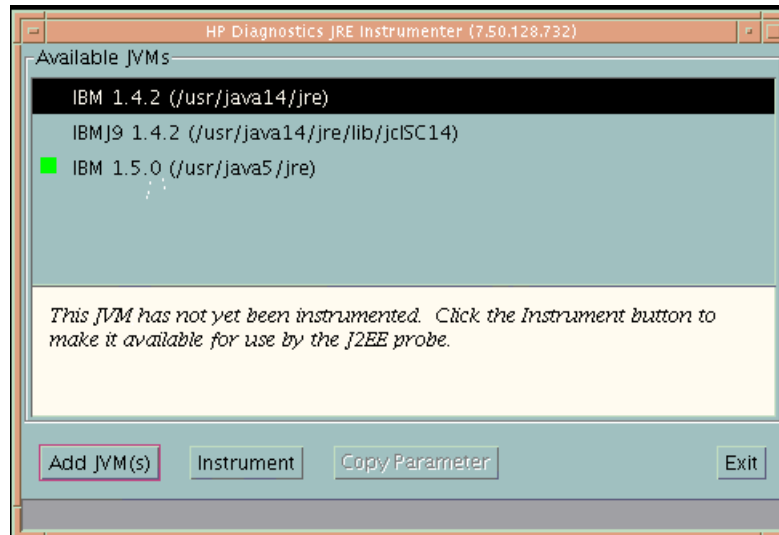
Ausführen des JRE-Instrumentierers auf einem Windows-basierten Computer

Wenn der JRE-Instrumentierer in einer Windows-Umgebung ausgeführt wird, zeigt der Instrumentierer die Dialogfelder seiner grafischen Benutzeroberfläche an. Die gleichen Dialogfelder werden angezeigt, wenn das Installationsprogramm auf einem UNIX-basierten Computer ausgeführt wird und der Instrumentierer im grafischen Modus ausgeführt wird.

Starten des JRE-Instrumentierers auf einem Windows-basierten Computer

- 1** Navigieren Sie zum Verzeichnis `<java_agent_install_verz>\DiagnosticsAgent\bin`, um die ausführbare Datei des JRE-Instrumentierers zu suchen.
- 2** Führen Sie den folgenden Befehl aus:
`jreinstrumenter.cmd`

Wenn der Instrumentierer gestartet wird, wird das Dialogfeld des JRE-Instrumentierungstools angezeigt.



Der Instrumentierer listet die JVMs auf, die von ihm ermittelt wurden und die für die Instrumentierung verfügbar sind. Die bereits instrumentierten JVMs werden mit einem grünen Quadrat vor dem Namen der JVM aufgeführt.

Ausgehend von diesem Dialogfeld können Sie die folgenden Aufgaben ausführen:

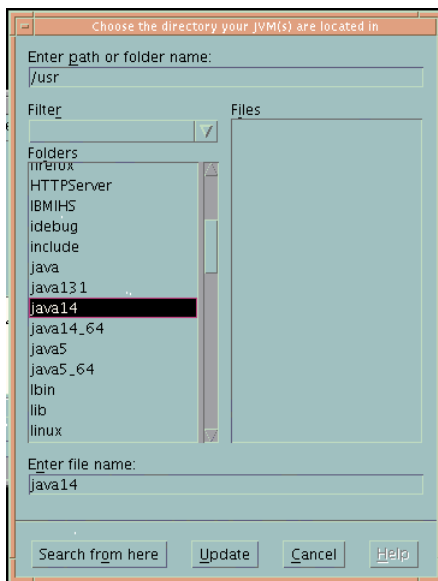
- ▶ Wenn die JVM, die Sie instrumentieren möchten, nicht in der Liste der verfügbaren JVMs aufgeführt wird, können Sie die gewünschten JVMs wie unter "Hinzufügen von JVMs zur Liste der verfügbaren JVMs" auf Seite 188 beschrieben zur Liste hinzufügen.
- ▶ Wenn die JVM, die Sie instrumentieren möchten, aufgeführt wird, jedoch noch nicht instrumentiert wurde, können Sie die JVM wie unter "Instrumentieren einer ausgewählten JVM" auf Seite 190 beschrieben instrumentieren.

- ▶ Wenn die JVM, die Sie instrumentieren möchten, aufgeführt wird und bereits instrumentiert wurde, können Sie den JVM-Parameter kopieren, um ihn wie unter "Einbinden des JVM-Parameters in das Startskript des Applikationsservers" auf Seite 190 beschrieben in das Startskript für die JVM einzufügen und damit die Überwachung durch die Probe zu aktivieren.
- ▶ Wenn Sie alle gewünschten Aufgaben im JRE-Instrumentierer ausgeführt haben, können Sie auf **Exit** klicken, um das Dialogfeld des JRE-Instrumentierungstools zu schließen.

Hinzufügen von JVMs zur Liste der verfügbaren JVMs

- 1 Klicken Sie im Dialogfeld des JRE-Instrumentierungstools auf **Add JVM(s)**, um nach weiteren JVMs zu suchen und diese zur Liste der verfügbaren JVMs hinzuzufügen.

Der Instrumentierer zeigt das Dialogfeld **Choose the Directory** an.

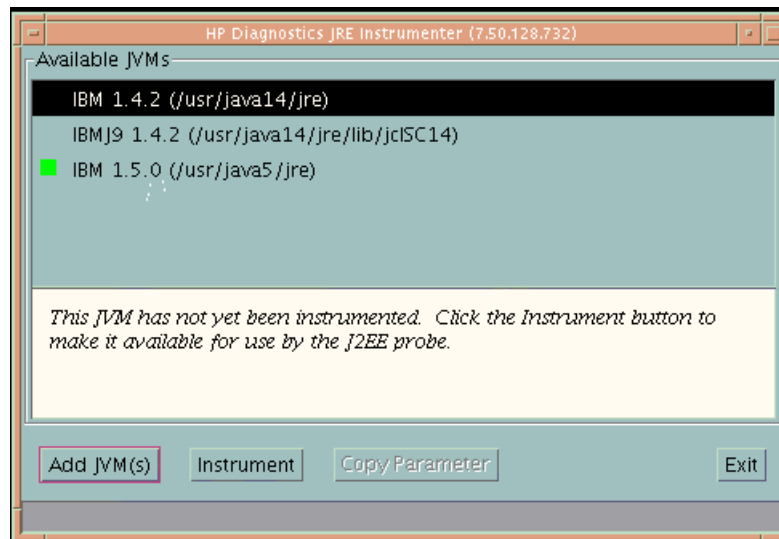


- 2 Geben Sie den Verzeichnispfad an, an dem der Instrumentierer mit der Suche nach JVMs beginnen soll.
- 3 Klicken Sie auf **Update**, um alle Ordner in diesem Verzeichnis in der Liste **Folders** aufzulisten.

- 4 Wählen Sie den Ordner aus, in dem Sie mit der Suche beginnen möchten, sodass der Name im Feld **File name** angezeigt wird.
- 5 Klicken Sie auf **Search from here**, um mit der Suche nach JVMs zu beginnen.

Der Instrumentierer schließt das Dialogfeld und zeigt erneut das Dialogfeld des JRE-Instrumentierungstools an. Während der Instrumentierer nach JVMs sucht, sind die Befehlsschaltflächen im Dialogfeld deaktiviert. Am unteren Rand des Dialogfelds zeigt eine Statusleiste an, dass der Instrumentierer nach JVMs sucht und wie weit der Suchvorgang vorangeschritten ist.

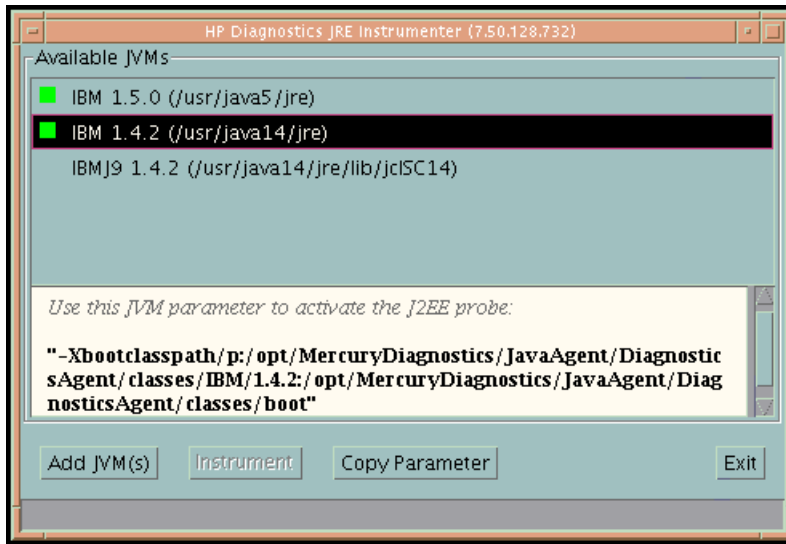
Sobald JVMs gefunden werden, werden diese in der Liste **Available JVMs** aufgeführt.



Wenn die Suche durch den Instrumentierer abgeschlossen ist, werden die Befehlsschaltflächen im Dialogfeld wieder aktiviert. Wenn es sich bei einer ausgewählten Zeile um eine bereits instrumentierte JVM handelt, ist die Schaltfläche **Instrument** deaktiviert. Das grüne Quadrat zeigt an, dass die JVM bereits instrumentiert wurde.

Instrumentieren einer ausgewählten JVM

Wählen Sie in der Liste **Available JVMs** eine JVM aus, die noch nicht instrumentiert wurde, und klicken Sie auf **Instrument**.



Der JRE-Instrumentierer zeigt im Feld unterhalb der Liste **Available JVMs** den JVM-Parameter an, der beim Starten des Applikationsservers verwendet werden muss.

Einbinden des JVM-Parameters in das Startskript des Applikationsservers

Wenn eine JVM durch den JRE-Instrumentierer instrumentiert wird, wird außerdem der JVM-Parameter erstellt, den Sie in das Startskript für den Applikationsserver aufnehmen müssen. Hierdurch ist sichergestellt, dass die Applikation das instrumentierte Klassenladeprogramm verwendet. Wenn Sie eine instrumentierte JVM aus der Liste **Available JVMs** auswählen, wird der JVM-Parameter unterhalb der Liste angezeigt. Sie können eine oder mehrere instrumentierte JVMs (nacheinander) kopieren und an einen geeigneten Speicherort kopieren, damit der Applikationsserver darauf zugreifen kann.

Klicken Sie auf **Copy Parameter**, um den in diesem Feld angezeigten JVM-Parameter in die Zwischenablage zu kopieren. Der JVM-Parameter wird in die Zwischenablage kopiert. Sie können ihn nun an einen geeigneten Speicherort kopieren, damit der Applikationsserver beim Start darauf zugreifen kann.

Hinweis: Wenn alle JVM-Parameter kopiert und am geeigneten Speicherort für den Applikationsserver eingefügt wurden, müssen Sie den Applikationsserver anhalten und neu starten, damit die Einstellungsänderungen wirksam werden. Wenn das Kopieren nicht möglich ist, können Sie die Parameter auch manuell eingeben. Die WebLogic-JVM befindet sich normalerweise im Verzeichnis `%bea_home%`. Die WebSphere-JVM befindet sich normalerweise im Verzeichnis `%WebSphere_home%\java`.

Ausführen des JRE-Instrumentierers auf einem UNIX-basierten Computer

In den folgenden Anweisungen werden die Schritte beschrieben, die für die Ausführung des JRE-Instrumentierers erforderlich sind. Die notwendigen Schritte werden sowohl für die Installation im grafischen Modus als auch für die Installation im Konsolenmodus beschrieben.

Die im grafischen Modus angezeigten Seiten des JRE-Instrumentierers sind mit den unter "Ausführen des JRE-Instrumentierers auf einem Windows-basierten Computer" auf Seite 186 beschriebenen Seiten für eine Installation unter Windows identisch.

Starten des JRE-Instrumentierers auf einem UNIX-basierten Computer

Öffnen Sie das Verzeichnis `<java_agent_install_verz>/DiagnosticsAgent/bin`, um nach der ausführbaren Datei des JRE-Instrumentierers zu suchen. Führen Sie den folgenden Befehl aus:

```
./jreinstrumenter.sh -console
```

Wenn der Instrumentierer startet, wird eine Liste der verfügbaren Verarbeitungsoptionen angezeigt. Die verfügbaren Optionen sind in der folgenden Tabelle aufgeführt:

Instrumentierer-Funktion	Beschreibung
jreinstrumenter -l	Anzeigen der Liste der bekannten JVMs. Ausführliche Informationen finden Sie unter "Anzeigen der Liste instrumentierter JVMs" auf Seite 192.
jreinstrumenter -a DIR	Suchen von JVMs unterhalb des Verzeichnisses DIR. Ausführliche Informationen finden Sie unter "Hinzufügen von JVMs zur Liste der verfügbaren JVMs" auf Seite 193.
jreinstrumenter -i JVM_DIR	Instrumentieren der JVM in JVM_DIR. Ausführliche Informationen finden Sie unter "Instrumentieren einer aufgelisteten JVM" auf Seite 193.
jreinstrumenter -b JVM_DIR	Instrumentieren der JVM in JVM_DIR und Platzieren des Klassenladeprogramms in <code><java_agent_install_verz>/DiagnosticsAgent/classes/boot</code> . Ausführliche Informationen finden Sie unter "Instrumentieren einer aufgelisteten JVM" auf Seite 193.

Sie können die Liste der Optionen erneut anzeigen, indem Sie beim Ausführen des Befehls `jreinstrumenter.sh` die Option `-x` angeben:

```
./jreinstrumenter.sh -x
```

Anzeigen der Liste instrumentierter JVMs

Geben Sie den folgenden Befehl ein, um eine Liste der dem JRE-Instrumentierer bekannten JVMs anzuzeigen:

```
./jreinstrumenter.sh -l
```

Vom Instrumentierer werden die ihm bekannten JVMs zeilenweise angezeigt. Jede Zeile gibt den JVM-Anbieter, die JVM-Version und den Speicherort der JVM an.

Hinzufügen von JVMs zur Liste der verfügbaren JVMs

Geben Sie den folgenden Befehl ein, um innerhalb eines bestimmten Verzeichnisses nach JVMs zu suchen und die gefundenen JVMs zur Liste der JVMs hinzuzufügen, die dem JRE-Instrumentierer bekannt sind:

```
./jreinstrumenter.sh -a DIR
```

Ersetzen Sie DIR durch den Pfad zu dem Speicherort, an dem der Instrumentierer mit der Suche beginnen soll.

Der Instrumentierer durchsucht die Verzeichnisse ausgehend vom angegebenen Speicherort einschließlich der zugehörigen Unterverzeichnisse. Sobald die Suche abgeschlossen ist, wird eine aktualisierte Liste der verfügbaren JVMs angezeigt.

Instrumentieren einer aufgelisteten JVM

Verwenden Sie einen der beiden folgenden Befehle, um eine JVM zu instrumentieren, die in der Liste der verfügbaren JVMs aufgeführt ist:

- Expliziter Pfad zum Klassenladeprogramm

```
./jreinstrumenter.sh -i JVM_DIR
```

Ersetzen Sie JVM_DIR durch den Pfad zum Speicherort der JVM, wie er in der Liste der verfügbaren JVMs angegeben ist.

Durch diesen Befehl wird der JRE-Instrumentierer angewiesen, die **ClassLoader**-Klasse für die ausgewählte JVM zu instrumentieren und das instrumentierte Klassenladeprogramm in einem Ordner unterhalb des Verzeichnisses `<java_agent_install_verz>/DiagnosticsAgent/classes/<JVM_anbieter>/<JVM_version>` abzulegen.

Diesen Befehl sollten Sie normalerweise verwenden; verwenden Sie ihn insbesondere dann, wenn Sie mehrere JVMs instrumentieren möchten, die durch einen einzigen Java-Agent überwacht werden sollen.

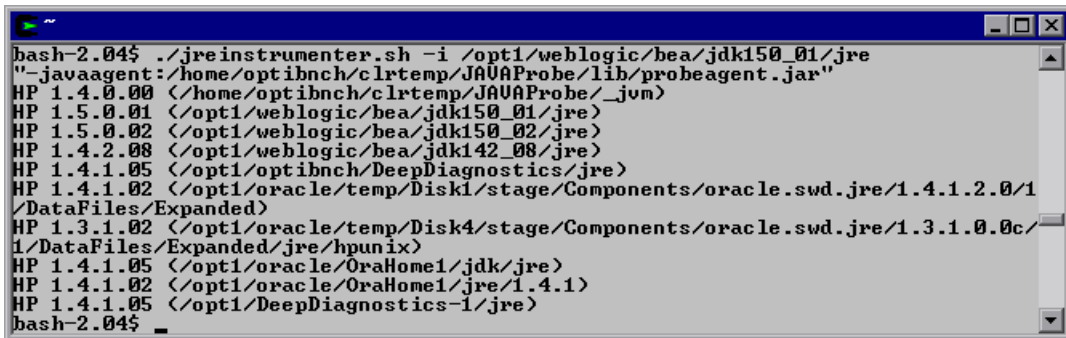
- Generischer Pfad zum Klassenladeprogramm

```
./jreinstrumenter.sh -b JVM_DIR
```

Ersetzen Sie JVM_DIR durch den Pfad zum Speicherort der JVM, wie er in der Liste der verfügbaren JVMs angegeben ist.

Verwenden Sie diesen Befehl nur, wenn Sie eine einzige JVM mit dem Java-Agent überwachen und Sie den expliziteren Pfad, der bei Verwendung der Befehlsoption `-i` erzeugt wird, aus bestimmten Gründen nicht verwenden möchten.

Wenn die Instrumentierung der JVM durch den Instrumentierer abgeschlossen ist, wird der JVM-Parameter angezeigt, den Sie verwenden müssen, um die Instrumentierung zu aktivieren und die Überwachung der Applikation durch den Java-Agent zu aktivieren. Im Anschluss an den JVM-Parameter wird erneut die Liste der verfügbaren JVMs aufgeführt, wie im folgenden Beispiel gezeigt wird:



```
bash-2.04$ ./jreinstrumenter.sh -i /opt1/weblogic/bea/jdk150_01/jre
"-javaagent:/home/optibnch/clrtemp/JAUAProbe/lib/probeagent.jar"
HP 1.4.0.00 </home/optibnch/clrtemp/JAUAProbe/_jvm>
HP 1.5.0.01 </opt1/weblogic/bea/jdk150_01/jre>
HP 1.5.0.02 </opt1/weblogic/bea/jdk150_02/jre>
HP 1.4.2.08 </opt1/weblogic/bea/jdk142_08/jre>
HP 1.4.1.05 </opt1/optibnch/DeepDiagnostics/jre>
HP 1.4.1.02 </opt1/oracle/temp/Disk1/stage/Components/oracle.swd.jre/1.4.1.2.0/1
/DataFiles/Expanded>
HP 1.3.1.02 </opt1/oracle/temp/Disk4/stage/Components/oracle.swd.jre/1.3.1.0.0c/
1/DataFiles/Expanded/jre/hpunix>
HP 1.4.1.05 </opt1/oracle/OraHome1/jdk/jre>
HP 1.4.1.02 </opt1/oracle/OraHome1/jre/1.4.1>
HP 1.4.1.05 </opt1/DeepDiagnostics-1/jre>
bash-2.04$ _
```

Einbinden des JVM-Parameters in das Startskript des Applikationsservers

Wenn eine JVM durch den JRE-Instrumentierer instrumentiert wird, wird außerdem der JVM-Parameter erstellt, den Sie in das Startskript für den Applikationsserver aufnehmen müssen. Hierdurch ist sichergestellt, dass die Applikation das instrumentierte Klassenladeprogramm verwendet. Wenn die Instrumentierung der JVM durch den Instrumentierer abgeschlossen ist, wird der JVM-Parameter angezeigt.

Kopieren Sie den JVM-Parameter in die Zwischenablage und fügen Sie ihn dann an einem geeigneten Speicherort ein, damit der Applikationsserver beim Start darauf zugreifen kann.

Konfigurieren der Applikationsserver

Nachdem Sie den JRE-Instrumentierer für den Java-Agent ausgeführt haben, müssen Sie das Startskript für die Applikation ändern, damit der Java-Agent, der die Applikation überwachen soll, zusammen mit der Applikation gestartet wird.

Sie können die Applikationsserver konfigurieren, indem Sie die Applikationsserver-Startskripts manuell aktualisieren. In den folgenden Abschnitten finden Sie Anweisungen zum manuellen Aktualisieren der Applikationsserver.

Konfigurieren von Nachrichtensystem Providern

In TransactionVision werden Warteschlangen für die Kommunikation zwischen Analyzer und den Sensoren verwendet. Mindestens drei Warteschlangen sind erforderlich: die Konfigurationswarteschlange, die Ereigniswarteschlange und die Ausnahmewarteschlange. Die Standardnamen für diese Warteschlangen lauten `TVISION.CONFIGURATION.QUEUE`, `TVISION.EVENT.QUEUE` und `TVISION.EXCEPTION.QUEUE`. Sie müssen diese Warteschlangen im Nachrichtensystemprovider gemäß den Vorgaben des Anbieters konfigurieren. Informationen hierzu finden Sie unter "Managing Communication Links" im Handbuch *Verwenden von TransactionVision*.

IBM WebSphere MQ

Sie können die Warteschlangen im Warteschlangen-Manager mithilfe des IBM WebSphere MQ-Dienstprogramms **runmqsc** oder mithilfe der grafischen Benutzeroberfläche von MQ Explorer erstellen, die unter Windows und Linux verfügbar ist. Wenn Sie **Client** als Verbindungstyp verwenden, müssen Sie im Warteschlangen-Manager außerdem einen Serververbindungskanal und einen Listener definieren. Ausführliche Informationen finden Sie in der Dokumentation des Anbieters.

TIBCO EMS

Sie können die Warteschlangen mit dem Verwaltungswerkzeug von TIBCO EMS erstellen. Ausführliche Informationen finden Sie in der Dokumentation des Anbieters.

Progress SonicMQ

Sie können die Warteschlangen in den SonicMQ-Brokern mithilfe der SonicMQ-Verwaltungskonsole erstellen. Ausführliche Informationen finden Sie in der Dokumentation des Anbieters.

BEA WebLogic JMS

Sie können die WebLogic-Verwaltungskonsole verwenden, um einen JMS-Server und die erforderlichen Warteschlangen zu konfigurieren. Im Folgenden finden Sie eine Zusammenfassung der für WebLogic 8.1 erforderlichen Schritte. Ausführliche Informationen finden Sie in der Dokumentation des Anbieters.

1 Erstellen Sie eine temporäre Vorlage:

Wählen Sie **Services > JMS > Templates** aus und klicken Sie dann auf **Configure a new JMS Template...**

2 Erstellen Sie einen persistenten JMS-Speicher:

Wählen Sie **Services > JMS > Store** aus und klicken Sie dann auf **Configure a new JMS Store...**

Sie müssen einen Verzeichnispfad für den Speicher (z. B. `/tmp`) angeben.

Sie können stattdessen auch einen JDBC-Speicher verwenden.

3 Erstellen Sie einen JMS-Server:

Wählen Sie **Services > JMS > Server** aus und klicken Sie dann auf **Configure a new JMS Server...**

Auf dieser Seite müssen Sie den persistenten Speicher und die temporäre Vorlage, den bzw. die Sie zuvor erstellt haben, angeben.

4 Erstellen Sie die Warteschlangen:

Wählen Sie **Services > JMS > Servers > <ihr_servername>**, klicken Sie auf **Configure Destinations...** und klicken Sie dann auf **Create a new JMS Queue...**

5 Erstellen Sie eine Verbindungsfactory:

Wählen Sie **Services > JMS > ConnectionFactories** aus und klicken Sie dann auf **Create an new JMS Connection Factory...**

Konfigurieren von benutzerdefinierten Benutzerereignissen

Sie können benutzerdefinierte Methoden als Teil des Transaktionspfades aufnehmen, damit sie als Ereignisse dargestellt werden. Diese Methoden werden nicht standardmäßig in den Ereignisverlauf aufgenommen, es sei denn sie sind Teil des standardmäßigen Java-Applikationsframeworks, wie z. B. JMS, EJB, Servlets, JSP usw.

Bearbeiten Sie die Datei **auto_detect.points** und erstellen Sie entweder einen reinen TransactionVision-Punkt oder bearbeiten Sie einen bestehenden Diagnostics-Punkt so, dass er ein TransactionVision-Benutzerereignis einschließt, indem Sie das Tag **tv:user_event** in der Detailzeile angeben.

Beispiel:

```
#
# GENERIC EVENT
#
[GenericEvent]
class    = !com.company.importantclass
method   = !.*
signature = !.*
detail   = tv:user_event
```

Weitere Informationen zum Erstellen und Ändern bestehender Instrumentierungspunkte finden Sie unter "Custom Instrumentation" im Handbuch *HP Diagnostics Installation and Configuration Guide*.

16

Installieren von WebSphere MQ- und User Event-Sensoren unter Windows

Dieses Kapitel umfasst die folgenden Themen:

- Starten des Installationsprogramms unter Windows auf Seite 199
- Erstinstallation auf Seite 200
- Upgrade-Installation auf Seite 201
- Bearbeiten der Installation auf Seite 203
- Deinstallieren von Sensoren auf Seite 205

Starten des Installationsprogramms unter Windows

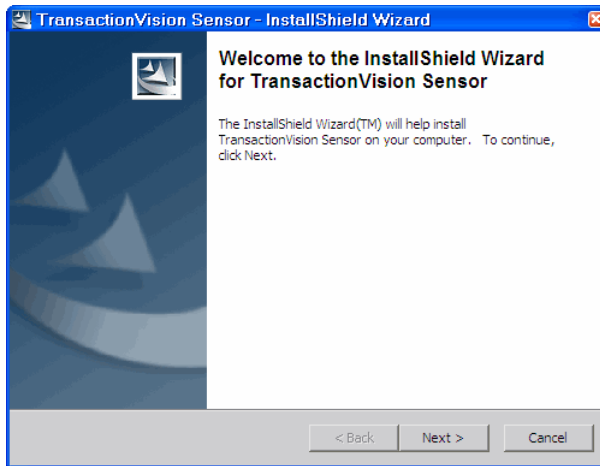
Unter Windows werden die TransactionVision-Sensoren für WebSphere MQ und User Event in Form eines einzigen Pakets installiert. In diesem Kapitel finden Sie Anweisungen zum Installieren dieser Sensoren. (Informationen zum Installieren von TransactionVision-Sensoren für Java-Applikationen und Applikationsservern finden Sie in Kapitel 15, "Installieren und Konfigurieren des Java-Agents".)

Beachten Sie, dass Sie entweder als Administrator oder als Benutzer mit Administratorberechtigungen am Zielsystem angemeldet sein müssen.

Führen Sie die folgenden Schritte aus, um dieses Paket zu installieren:

- 1** Schließen Sie alle Windows-Programme, die momentan auf dem Computer ausgeführt werden. Dies umfasst auch Programme für die automatische Sicherung. Antiviren-, Antispyware- und andere Schutzprogramme müssen nicht heruntergefahren werden.

- 2 Doppelklicken Sie in Windows-Explorer auf **tvwmque_800_win.exe**. Die InstallShield-Willkommenseite wird angezeigt.



- 3 Klicken Sie auf **Next>**, um die InstallShield-Seite **Save Files** anzuzeigen.
- 4 Klicken Sie auf **Next>**, um den Standardordner zum Extrahieren der Installationsdateien zu verwenden. Wenn Sie einen anderen Ordner verwenden möchten, klicken Sie auf **Change**. Wählen Sie den gewünschten Ordner aus, und klicken Sie dann auf **Next>**. Die Installationsdateien werden von InstallShield extrahiert.

Wenn TransactionVision-Sensoren erstmalig auf diesem Computer installiert werden, fahren Sie gemäß der Beschreibung unter "Erstinstallation" fort. Falls eine frühere Version von TransactionVision-Sensoren auf dem Computer installiert ist, fahren Sie gemäß der Beschreibung unter "Upgrade-Installation" fort.

Erstinstallation

Bei einer Erstinstallation wird die Willkommenseite des Setupprogramms angezeigt.

- 1 Klicken Sie auf der Willkommenseite des Setupprogramms auf **Next>**, um die Lizenzvereinbarung für TransactionVision anzuzeigen.

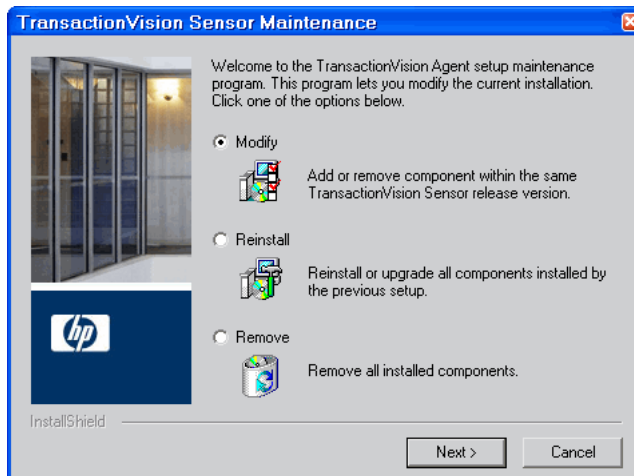
- 2 Klicken Sie auf **Yes**, um der Lizenzvereinbarung zuzustimmen. Die Seite **User Information** wird angezeigt.
- 3 Geben Sie Ihren Namen und den Namen Ihres Unternehmens ein und klicken Sie dann auf **Next>**. Die Seite **Destination Location** wird angezeigt.
- 4 Zum Installieren der WebSphere MQ- und User Event-Sensoren wählen Sie **Complete** aus und klicken dann auf **Next>**. Wenn Sie nur einige Sensoren installieren möchten, wählen Sie **Custom** aus, klicken Sie auf **Next>**, wählen Sie die gewünschten Sensoren aus und klicken Sie auf **Next>**.

Die ausgewählten Sensoren werden am angegebenen Speicherort installiert. Die Seite **Setup Complete** wird angezeigt.

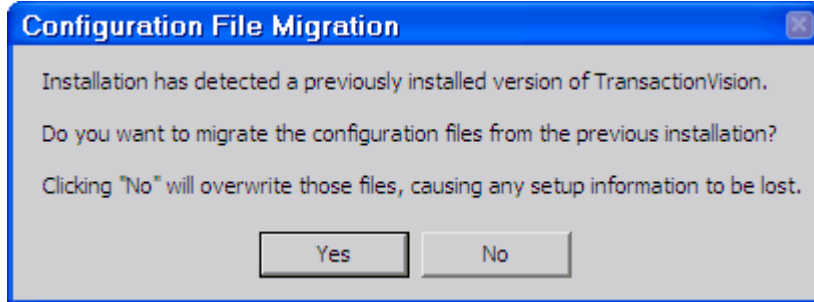
- 5 Klicken Sie auf **Finish**, um die Installation abzuschließen.

Upgrade-Installation

Wenn Sie eine Upgrade-Installation durchführen, doppelklicken Sie auf **twwmque_800_win.exe**. Klicken Sie auf der InstallShield-Willkommenseite auf **Next>**, um das Wartungsmenü für die Sensorinstallation anzuzeigen:

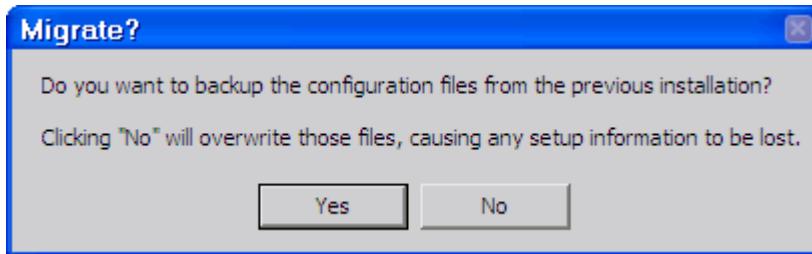


- 1 Wählen Sie eine der folgenden Optionen aus, um die TransactionVision-Installation zu aktualisieren (Informationen zum Ändern der Installation finden Sie unter "Bearbeiten der Installation" auf Seite 203):
 - ▶ Wenn Sie für diese Installation der TransactionVision-Sensoren andere Einstellungen als in der vorherigen Installation verwenden möchten, wählen Sie **Remove** aus. Klicken Sie auf **Next>**, um die vorherige Installation zu deinstallieren, und beginnen Sie dann erneut mit dem Installationsverfahren.
 - ▶ Wenn Sie ein Upgrade für eine frühere Version durchführen, wählen Sie **Reinstall** aus. Klicken Sie dann auf **Next>**, um TransactionVision-Sensoren mit den Einstellungen der vorherigen Installation zu installieren. Das Dialogfeld **Configuration File Migration** wird angezeigt:



- 2 Klicken Sie auf **Ja**, um die Konfigurationsinformationen aus der vorherigen Installation zu übernehmen. Der Installations-Assistent erstellt eine Sicherungskopie der vorhandenen Konfigurationsdateien, installiert die neue Version von TransactionVision und öffnet ein MS-DOS-Fenster, um vorhandene Konfigurationsdateien zur neuen Version zu migrieren. Wenn dieser Vorgang abgeschlossen ist, wird die Seite **Setup Complete** angezeigt.

Klicken Sie auf **No**, um vorhandene Konfigurationsdateien zu überschreiben. In einem Meldungsfeld des Installations-Assistenten werden Sie aufgefordert, anzugeben, ob Sie eine Sicherungskopie vorhandener Konfigurationsdateien erstellen möchten, bevor Sie mit der Installation fortfahren.



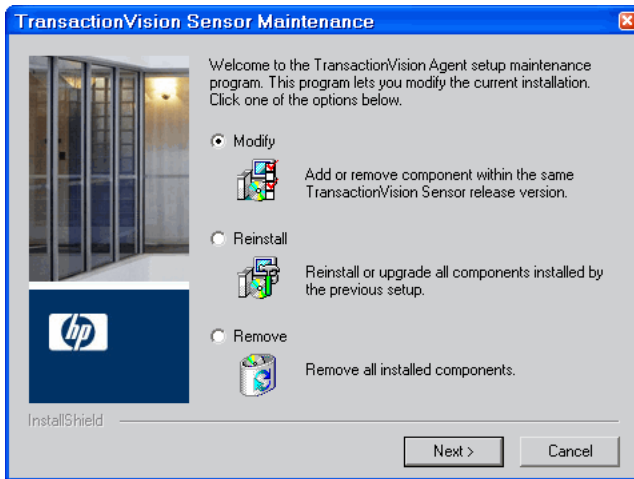
Klicken Sie auf **Yes**, um eine Sicherungskopie zu erstellen, oder auf **No**, um mit der Installation fortzufahren, ohne die Konfigurationsdateien zu sichern. Der Installations-Assistent installiert nun die neue Version von TransactionVision (wobei vorhandene Konfigurationsdateien überschrieben werden) und zeigt danach die Seite **Setup Complete** an.

- 3 Der Installations-Assistent installiert die neue TransactionVision-Version und zeigt die Seite **Setup Complete** an.
- 4 Klicken Sie auf **Finish**, um die Installation abzuschließen.

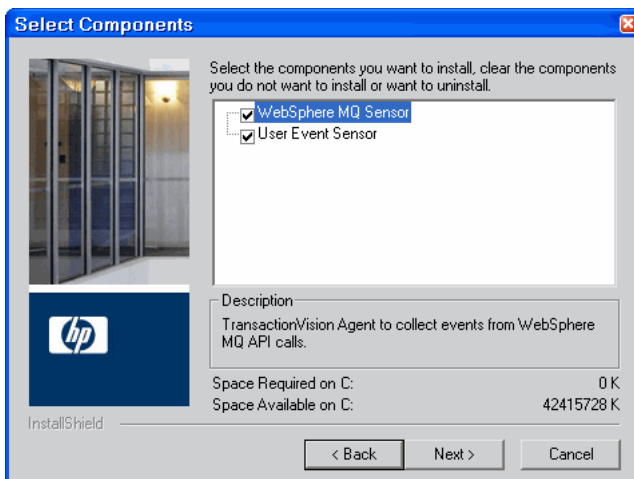
Bearbeiten der Installation

Nachdem Sie TransactionVision-Sensoren auf einem Host installiert haben, kann es u. U. notwendig sein, die Installation zu bearbeiten. Nehmen Sie beispielsweise an, Sie installieren den WebSphere MQ-Sensor erstmalig auf einem Host und entscheiden sich später dafür, den User Event-Sensor ebenfalls zu installieren. Führen Sie die folgenden Schritte aus, um die Installation zu bearbeiten:

- 1 Doppelklicken Sie auf **twwmque_800.exe** und klicken Sie auf der InstallShield-Willkommensseite auf **Next>**, um die Seite **TransactionVision Sensor Maintenance** anzuzeigen:



- 2 Zum Installieren eines weiteren Sensors oder zum Entfernen eines installierten Sensors wählen Sie **Modify** aus und klicken dann auf **Next>**, um die Seite **Select Components** anzuzeigen.



- 3 Wählen Sie die Sensoren aus, die Sie installieren möchten, und klicken Sie auf **Next>**, um die geänderte Installation auszuführen.

- 4 Klicken Sie auf **Finish**, um die Installation abzuschließen.

Deinstallieren von Sensoren

Führen Sie die folgenden Schritte aus, um TransactionVision-Komponenten zu deinstallieren:

- 1 Wählen Sie im Startmenü **Einstellungen** > **Systemsteuerung** aus.
- 2 Doppelklicken Sie auf **Software**.
- 3 Wählen Sie das HP TransactionVision-Sensorpaket aus und klicken Sie auf **Ändern/Entfernen**. Die Seite mit dem Wartungsmenü wird angezeigt.
- 4 Wählen Sie **Remove** aus und klicken Sie auf **Next>**, um TransactionVision-Komponenten zu entfernen.
- 5 Klicken Sie auf **OK**, um die Deinstallation des angegebenen Pakets zu bestätigen. Das angegebene Paket wird deinstalliert. Die folgenden Dateien werden nicht gelöscht:

- Alle Dateien, die nach der Installation hinzugefügt wurden.
- Alle gemeinsam genutzten Dateien, die Paketen zugeordnet sind, die noch installiert sind.

Wenn es den Anschein hat, dass gemeinsam genutzte Dateien keinem installierten Paket zugeordnet sind (z. B. wenn alle anderen TransactionVision-Pakete deinstalliert wurden), wird die Seite **Shared File Detected** angezeigt.

- Wenn die gemeinsam genutzten Dateien installiert bleiben sollen, aktivieren Sie **Don't display this message again** und klicken Sie dann auf **No**.
- Wenn Sie die aktuelle Datei beibehalten möchten, diese Meldung jedoch für jede weitere gemeinsam genutzte Datei angezeigt werden soll, klicken Sie auf **No**.
- Zum Löschen der gemeinsam genutzten Datei klicken Sie auf **Yes**.

Die Seite **Uninstallation Complete** wird angezeigt.

- 6 Klicken Sie auf **Finish**, um die Deinstallation abzuschließen.

17

Installieren von WebSphere MQ- und User Event-Sensoren auf UNIX-Plattformen

In diesem Kapitel finden Sie Anweisungen zum Installieren von TransactionVision-Sensoren für WebSphere MQ und User Event auf UNIX-Plattformen. Informationen zum Installieren von TransactionVision-Sensoren für Java-Applikationen und Applikationsservern finden Sie in Kapitel 15, "Installieren und Konfigurieren des Java-Agents".

Beachten Sie, dass TransactionVision für jeden Sensor eigene Pakete bereitstellt.

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Installieren von Sensoren auf Seite 207
- ▶ Deinstallieren von Sensoren auf Seite 210

Installieren von Sensoren

Installationsdateien

In der folgenden Tabelle sind die Namen der Installationsdateien für den WebSphere MQ-Sensor aufgeführt.

Plattform	Dateien
AIX	tvwmq_800_aix_power.tar.gz
HP-UX	tvwmq_800_hpux_parisc.tar.gz tvwmq_800_hpux_ia64.tar.gz

Plattform	Dateien
Linux	tvwmq_800_linux_x86.tar.gz
Solaris	tvwmq_800_sol_sparc.tar.gz
IBM i5/OS	tvmq_800_i5os_iseries.savf

In der folgenden Tabelle sind die Namen der Installationsdateien für den User Event-Sensor aufgeführt. Beachten Sie, dass bei der Installation des User Event-Sensors das allgemeine Paket automatisch installiert wird, da der Sensor das Vorhandensein dieses Pakets voraussetzt.

Plattform	Dateien
AIX	tvue_800_aix_power.tar.gz
Linux	tvue_800_linux_x86.tar.gz
Solaris	tvue_800_sol_sparc.tar.gz

Installationsschritte

- 1 Wechseln Sie zum Verzeichnis, in dem sich die TransactionVision-Installationsdateien befinden (entweder ein DVD-Laufwerk oder ein Downloadverzeichnis).

Hinweis: Unter Solaris und HP-UX müssen Sie die Installationsdateien vom DVD-Laufwerk in ein temporäres Verzeichnis auf der Festplatte des Hostsystems kopieren.

- 2 Melden Sie sich als Superuser an:

su

- 3 Geben Sie den folgenden Befehl ein, um das Installationsverfahren zu starten:

./tvinstall_800_unix.sh

Das folgende Menü wird angezeigt:

The following TransactionVision packages are available for installation:

1. TransactionVision Analyzer
2. TransactionVision Web
3. TransactionVision WebSphere MQ Agent
4. TransactionVision User Event Agent

99. All of above

q. Quit install

Please specify your choices (separated by,) by number/letter:

Hinweis: Die tatsächlich angezeigten Optionen und Nummern hängen von den auf dem Computer verfügbaren Installationsdateien ab.

- 4** Zum Installieren einer einzigen Komponente geben Sie die Nummer ein, die dem Paket der TransactionVision-Komponente zugeordnet ist, und drücken dann die EINGABETASTE.

Zum Installieren mehrerer, jedoch nicht aller Komponenten geben Sie die Nummern (durch Kommas getrennt) ein, die den zu installierenden Komponenten zugeordnet sind, und drücken dann die EINGABETASTE. Wenn Sie beispielsweise alle Sensoren installieren möchten, die im oben beschriebenen Menü aufgeführt sind, geben Sie Folgendes ein und drücken dann die EINGABETASTE.

3,4

Zum Installieren aller verfügbaren Komponenten geben Sie **99** ein und drücken dann die EINGABETASTE.

Das oder die angegebenen Pakete werden durch das Installationskript installiert. Anschließend wird erneut das Menü angezeigt.

- 5** Geben Sie **q** ein und drücken Sie die EINGABETASTE, um das Installationsverfahren zu beenden. Informationen zum Installieren weiterer Komponenten finden Sie in den Installationsanweisungen für die betreffenden Komponenten.

Erneutes Binden des WebSphere MQ-Sensors unter AIX

Bei der Installation des WebSphere MQ-Sensors auf der AIX-Plattform wird das Skript `rebind_sensor` aufgerufen, um die Sensorbibliothek erneut zu binden. Wenn Sie ein WebSphere MQ-Support-Pack installieren, durch das WebSphere MQ-Bibliotheken (`libmqm.a`, `libmqic.a`, `libmqm_r.a`, `libmqic_r.a`) verändert werden, müssen Sie dieses Skript erneut ausführen, damit überwachte Applikationen ordnungsgemäß ausgeführt werden. Weitere Informationen zu diesem Skript finden Sie in Anhang A, "Referenz der Dienstprogramme".

Deinstallieren von Sensoren

Führen Sie die folgenden Schritte aus, um TransactionVision-Komponenten zu deinstallieren:

- 1 Melden Sie sich als Superuser an:

```
su
```

- 2 Geben Sie den folgenden Befehl ein:

```
./tvinstall_800_unix.sh -u
```

Das folgende Menü wird angezeigt (beachten Sie, dass die tatsächlich angezeigten Optionen von den auf dem Computer installierten TransactionVision-Paketen abhängen):

The following TransactionVision packages are installed on the system:

1. TransactionVision Web
2. TransactionVision Analyzer
3. TransactionVision WebSphere MQ Agent
4. TransactionVision User Event Agent

99. All of above
- q. Quit uninstall

Please specify your choices (separated by,) by number/letter:

- 3 Geben Sie die Nummer des TransactionVision-Pakets ein, das Sie deinstallieren möchten, und drücken Sie die EINGABETASTE.

Zum Deinstallieren aller TransactionVision-Komponenten geben Sie **99** ein und drücken dann die EINGABETASTE.

Das angegebene Paket wird durch das Installationskript deinstalliert. Anschließend wird erneut das Menü angezeigt.

- 4** Geben Sie **q** ein und drücken Sie die EINGABETASTE, um die Deinstallation zu beenden. Wenn das allgemeine Paket das einzige noch installierte TransactionVision-Paket ist, wird es automatisch deinstalliert.

18

Installieren von Sensoren unter i5/OS

Dieses Kapitel umfasst die folgenden Themen:

- Starten des Installationsprogramms unter i5/OS auf Seite 213

Starten des Installationsprogramms unter i5/OS

Führen Sie die folgenden Schritte aus, um den Sensor auf der i5/OS-Plattform zu installieren:

- 1** Falls eine frühere Version des TransactionVision-Sensors installiert ist, verwenden Sie den folgenden Befehl, um den Sensor zu deinstallieren:

```
DLTLICPGM LICPGM(3RBB9ES)
```

- 2** Suchen Sie auf einem i5/OS-basierten Computer nach einer vorhandenen Bibliothek oder erstellen Sie eine neue Bibliothek, um die Installationsdatei zu kopieren (z. B. TVTMP).

- 3** Übertragen Sie auf einem PC die Sensorinstallationsdatei **tvwmq_800_i5os_ieries.savf** mit FTP von der CD-ROM in die in Schritt 2 erstellte Bibliothek auf dem i5/OS-basierten Computer und benennen Sie sie in **sensor800.savf** um. Legen Sie in jedem Fall folgendermaßen den binären Übertragungsmodus fest:

```
ftp> bin  
ftp> cd /qsys.lib/tvtmp.lib  
ftp> put tvwmq_800_i5os_ieries.savf sensor800.savf
```

- 4** Führen Sie auf dem i5/OS-basierten Computer den folgenden Befehl aus, um den Sensor zu installieren. Beachten Sie, dass es notwendig sein kann, TVTMP im Befehl durch den Namen der Bibliothek zu ersetzen, in der sich das Paket **sensor800.savf** befindet.

```
RSTLICPGM LICPGM(3RBB9ES) DEV(*SAVF) SAVF(TVTMP/  
SENSOR750)
```

- 5** Führen Sie den folgenden Befehl aus, um die Installation zu überprüfen:
DSPSFWRSC
- 6** Zum Verwenden des C-Sensors binden Sie die Programme an TVSENSOR/
LIBMQM.
- 7** Wenn Sie in Schritt 2 eine neue temporäre Bibliothek erstellt haben, kann sie nun problemlos gelöscht werden.

19

Installieren und Konfigurieren von Sensoren unter z/OS

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Informationen zu Sensoren in der z/OS-Umgebung auf Seite 215
- ▶ Übersicht über die Basiskomponenteninstallation auf Seite 216
- ▶ Basiskomponenteninstallation – Verfahrensbeschreibung auf Seite 217
- ▶ Konfigurieren der SLD-Sensorkomponenten unter z/OS: CICS, WebSphere MQ Batch und WebSphere MQ IMS auf Seite 224
- ▶ Konfigurieren der SLM-Sensorkomponenten unter z/OS: WebSphere MQ CICS Bridge und WebSphere MQ IMS Bridge auf Seite 226
- ▶ Hintergrundinformationen: WebSphere MS-Sensor für CICS auf Seite 229
- ▶ Konfigurieren von SLMC für CICS auf Seite 230
- ▶ Hintergrundinformationen: WebSphere MQ IMS Bridge-Sensor auf Seite 231

Informationen zu Sensoren in der z/OS-Umgebung

Es gibt zwei TransactionVision-Sensortypen, die in einer IBM z/OS-Umgebung bereitgestellt werden können. Jeder Sensortyp hat einen eigenen, aus drei Zeichen bestehenden Produktcode:

- ▶ Mit den SLD-Sensorkomponenten können CICS-, WebSphere MQ (WMQ) Batch- und WMQ IMS-Ereignisse überwacht werden.
- ▶ Mit den SLM-Sensorkomponenten können WMQ CICS Bridge- und WMQ IMS Bridge-Ereignisse überwacht werden.

In diesem Kapitel finden Sie Anweisungen zum Ausführen der Basiskomponenteninstallation und der im Anschluss an die Installation notwendigen Konfigurationsaufgaben für jeden Sensortyp. Die Basisinstallationen sind im Prinzip identisch und werden daher nur einmal beschrieben. Die Aufgaben im Rahmen der Sensorkonfiguration unterscheiden sich erheblich und werden daher separat erläutert.

Informationen zu weiteren externen Systemanforderungen unter z/OS finden Sie in Anhang D, "Zusätzliche z/OS-Einstellungen"

Übersicht über die Basiskomponenteninstallation

- 1 Übertragen Sie die Installationsdateien von den Installationsmedien auf die zSeries-Plattform. Hierfür gibt es zwei Möglichkeiten:
 - ▶ Verwenden Sie das auf den Installationsmedien bereitgestellte Skript **tvinstall_800_zos_zseries.bat**, um die Dateien zu übertragen (empfohlen).
 - ▶ Übertragen Sie die Installationsdateien manuell mit FTP.
- 2 Führen Sie den TSO-Befehl RECEIVE für die übertragenen Dateien aus, um Produktdatasets zu erstellen. Hierfür gibt es zwei Möglichkeiten:
 - ▶ Verwenden Sie das in Schritt 1 durch das Skript **tvinstall_800_zos_zseries.bat** erstellte REXX-Skript (TSORECV), um die TSO-RECEIVE-Befehle auszuführen (empfohlen).
 - ▶ Führen Sie die erforderlichen TSO-RECEIVE-Befehle manuell aus.
- 3 Bestimmen Sie den besten Installationsansatz für Ihre Umgebung. Hierfür gibt es zwei Möglichkeiten:
 - ▶ SMP/E (empfohlen). Sie können entweder eine neue globale SMP/E-Zone erstellen (empfohlen) oder eine bestehende globale SMP/E-Zone verwenden.
 - ▶ Nicht-SMP/E
- 4 Passen Sie die Installationsjobs an und führen Sie sie aus. Der Grad der erforderlichen Anpassung variiert in Abhängigkeit vom Installationsansatz (SMP/E oder Nicht-SMP/E).

5 Überprüfen Sie die Installationsbibliotheken.

Basiskomponenteninstallation – Verfahrensbeschreibung

Schritt 1: Übertragen Sie die Dateien von den Installationsmedien auf den zSeries-Host

Hierfür gibt es zwei Möglichkeiten:

- ▶ Führen Sie das Skript `tvinstall_800_zos_zseries.bat` aus.
- ▶ Übertragen Sie die Installationsdateien manuell mit FTP.

Dateinamenkonventionen

Die Dateinamen auf den Installationsmedien entsprechen dem folgenden Muster:

`tv<prodcode><dateiqual>_<ver>_zos_zseries.xmit`

Die Dateinamen der FTP-Ausgabedatasets entsprechen dem folgenden Muster:

`<&hlq>.<prodcode>|<ver>.<dateiqual>`

Hierbei gilt Folgendes:

`<dateiqual>` = f1 | f2 | f3 | mcs

`<prodcode>` = SLD | SLM

`<ver>` = 800

`<&hlq>` = ein kundendefiniertes übergeordnetes

Dataset-Qualifikationsmerkmal für die Ausgabe des FTP-Unterbefehls PUT auf dem z/OS-Zielsystem.

Ein Dateiname auf den Installationsmedien lautet beispielsweise folgendermaßen:

`tvslrdf1_800_zos_zseries.xmit`

Der Name eines Datasets für die FTP-Ausgabe könnte folgendermaßen lauten:

`TVISION.SLD800.F1`

Möglichkeit 1: Führen Sie das Skript "tvisntall_800_zos_zseries.bat" aus

- 1** Melden Sie sich an einer Arbeitsstation unter Microsoft Windows an, von der mit FTP eine TCP/IP-Verbindung zum z/OS-Zielsystem hergestellt werden kann.
- 2** Starten Sie unter Windows eine Eingabeaufforderungssitzung (**Start > Alle Programme > Zubehör > Eingabeaufforderung**). Passen Sie die Fenstereigenschaften an, indem Sie Größe und Puffer nach Bedarf verändern.
- 3** Navigieren Sie mithilfe des Befehls **change directory (cd)** zu dem Ordner mit den Installationsmedien.
- 4** Überprüfen Sie die Informationen, die vom Skript erfasst werden indem Sie Folgendes eingeben:

```
tvininstall_800_zos_zseries.bat /h | more
```
- 5** Rufen Sie **tvininstall_800_zos_zseries.bat** ohne Parameter auf, und reagieren Sie auf die Eingabeaufforderungen.

Hinweis: Wenn Sie mit den eingegebenen Werten nicht zufrieden sind, können Sie **tvininstall_800_zos_zseries.bat** beenden, ohne eine Dateiübertragung zu initiieren. Darüber hinaus können Sie die Skriptausführung jederzeit beenden, indem Sie wiederholt STRG+C drücken.

- 6** Prüfen Sie sorgfältig die Ergebnisse der FTP-Befehlsausgabe, die von **tvininstall_800_zos_zseries.bat** generiert werden.
 - Es sollten keine Fehler aufgeführt werden.
 - Prüfen Sie, ob Installations-RELFILES mit dem angegebenen übergeordneten Qualifikationsmerkmal auf dem z/OS-Zielsystem vorhanden sind.
 - Prüfen Sie, ob das REXX-Skript (TSORECV) in der angegebenen PDS-Bibliothek vorhanden ist.
 - Wenn Datasets fehlen oder FTP-Fehler festgestellt wurde, sollten Sie diese Fehler untersuchen und korrigieren und **tvininstall_800_zos_zseries.bat** erneut ausführen.

Möglichkeit 2: Übertragen Sie die Installationsdateien manuell mit FTP

Führen Sie diesen Schritt nicht aus, wenn Sie die Dateien mit dem Skript `tvinstall_800_zos_zseries.bat` installiert haben.

- 1** Melden Sie sich an einer Arbeitsstation unter Microsoft Windows an, von der mit FTP eine TCP/IP-Verbindung zum z/OS-Zielsystem hergestellt werden kann.
- 2** Starten Sie unter Windows eine Eingabeaufforderungssitzung (**Start > Alle Programme > Zubehör > Eingabeaufforderung**). Passen Sie die Fenstereigenschaften an, indem Sie Größe und Puffer nach Bedarf verändern.
- 3** Navigieren Sie mithilfe des Befehls **change directory (cd)** zu dem Ordner mit den Installationsmedien.
- 4** Verwenden Sie ein z/OS-Konto, das über ausreichende Berechtigungen zum Erstellen von Datasets mit dem gewünschten übergeordneten Qualifikationsmerkmal verfügt, und starten Sie eine FTP-Sitzung. Beispiel:

```
> ftp hostname
```

- 5** Setzen Sie die folgenden FTP-Unterbefehle ab:

a `ftp> quote site fixrecfm 80 lrecl=80 recfm=fb blksize=3120 vol=&custvol u=&custunit pri=30 sec=5 tr`

Hierbei ist `&custvol` eine gültige Datenträgerfolgenummer und `&custunit` ein gültiger Datenträger-Einheitentyp oder ein privater Einheitenname

b `ftp> bin`

c `ftp> put tv<prodcode><dateiqual>_<ver>_zos_zseries.xmit '<&hlq>.<prodcode><|ver>.<dateiqual>'`

Hierbei gilt Folgendes:

`<prodcode>` = slm | sld

`<&hlq>` = kundendefiniertes übergeordnetes Qualifikationsmerkmal

`<dateiqual>` = f1 | f2 | f3 | mcs

`<ver>` = 800

Beispiel:

```
put tvsldf1_800_zos_zseries.xmit 'TVISION.SLD800.F1'
```

- d** Setzen Sie FTP-Befehle des Typs **put** für alle verbleibenden Produktinstallationsdateien mit dem Dateinamensuffix **fn** oder **mcs** ab.
- e** Es ist wichtig, die FTP-Befehlsausgabe sorgfältig im Hinblick auf Fehler zu untersuchen. Falls Fehler gefunden werden, kann es notwendig sein, das Zieldataset auf dem z/OS-System zu löschen und den FTP-Befehl **put** erneut auszuführen.
- f** `ftp> quit`

Schritt 2: Führen Sie den TSO-Befehl RECEIVE für die übertragenen Dateien aus, um Produktdatasets zu erstellen

Hierfür gibt es zwei Möglichkeiten:

- Führen Sie das in "Schritt 1: Übertragen Sie die Dateien von den Installationsmedien auf den zSeries-Host" auf Seite 217 durch das Skript `tvinstall_800_zos_zseries.bat` erstellte REXX-Skript (TSORECV) aus.
- Führen Sie die erforderlichen TSO-RECEIVE-Befehle manuell aus.

Möglichkeit 1: Führen Sie das REXX-Skript (TSORECV) aus

Führen Sie an einer TSO- oder TSO/ISPF-Befehlszeile das in "Schritt 1: Übertragen Sie die Dateien von den Installationsmedien auf den zSeries-Host" auf Seite 217 erstellte REXX-Skript (TSORECV) aus. Dies bewirkt, dass TSO-RECEIVE-Befehle für alle angegebenen z/OS-Zieldatasets ausgeführt werden.

Die empfangenen Datasets werden mithilfe desselben übergeordneten Qualifikationsmerkmals zugeordnet, wobei jedoch der Buchstabe "A" als Präfix zum Produktcode hinzugefügt wird. Wenn das Eingabemuster beispielsweise `TVISION.SLD800.filequal` entspricht, würde das Ausgabemuster `TVISION.ASLD800.filequal` sein.

Möglichkeit 2: Führen Sie die erforderlichen TSO-RECEIVE-Befehle manuell aus

Führen Sie an einer TSO- oder TSO/ISPF-Befehlszeile RECEIVE-Befehle, falls notwendig mit dem Präfix TSO, wie in der folgenden Beispielstabelle gezeigt aus. Geben Sie als Antwort auf die Aufforderung "INMR906A Enter restore parameters or 'DELETE' or 'END'" **DSN(&hlq.A<prodcode>800.dateiqual)** ein.

Befehl	Dateiname
RECEIVE INDSNAME('&hlq.SL_800.F1')	DSN('&hlq.ASL_800.F1')
RECEIVE INDSNAME('&hlq.SL_800.F2')	DSN('&hlq.ASL_800.F2')
RECEIVE INDSNAME('&hlq.SL_800.F3')	DSN('&hlq.ASL_800.F3')
RECEIVE INDSNAME('&hlq.SL_800.MCS')	DSN('&hlq.ASL_800.SMPMCS')

Schritt 3: Bestimmen Sie den besten Installationsansatz

Hierfür gibt es zwei Möglichkeiten:

- SMP/E (empfohlen). Sie können entweder eine neue globale SMP/E-Zone (empfohlen) erstellen oder eine bestehende globale SMP/E-Zone verwenden.
- Nicht-SMP/E

Schritt 4: Passen Sie die Installationsjobs an und führen Sie sie aus

Hierfür gibt es zwei Möglichkeiten: SMP/E oder Nicht-SMP/E.

Möglichkeit 1: SMP/E-Installationsjobs

Passen Sie die Jobs an und führen Sie sie in der angegebenen Reihenfolge aus. Die Jobs befinden sich im Dataset **&hlq.A<prodcode>800.F3**. Ersetzen Sie den Unterstrich ("_") in der folgenden Liste durch das dritte Zeichen des Produktcodes und fahren Sie dann mit der Anpassung der einzelnen Elemente gemäß den Anforderungen Ihres Standorts fort. Lesen Sie die Kommentare am Anfang jedes Jobs und passen Sie die Jobs entsprechend an.

Prüfen Sie im Anschluss an die Ausführung sorgfältig alle Ausgaben, um sicherzustellen, dass alle Schritte erfolgreich waren. Fahren Sie erst dann mit dem nächsten Job fort. Falls die Installation in einer bestehenden globalen SMP/E-Zone erfolgt, sollten SMP/E-Schritte, Parameter und Eingaben sorgfältig geprüft werden.

SMP/E-Installationsjobs:

Reihenfolge	Job	Beschreibung
1.	SL_ALLOC	Ordnet Ziel- und Verteilungsbibliotheken zu
2.	SL_GZON	Definiert die globale SMP/E-Zone (nicht erforderlich, wenn eine bestehende globale Zone verwendet wird)
3.	SL_DZON	Definiert die SMP/E-Verteilerzone
4.	SL_TZON	Definiert die SMP/E-Zielzone
5.	SL_DDDEF	Definiert SMP/E-DDDEF-Einträge
6.	SL_RECV	SMP/E RECEIVE
7.	SL_APPLY	SMP/E APPLY
8.	SL_IVP	Installationsüberprüfung
9.	SL_ACCPT	SMP/E ACCEPT Wichtig! Führen Sie diesen Job erst aus, wenn die Installation konfiguriert und vollständig getestet wurde. Sobald die SMP/E-Funktion ACCEPT ausgeführt wurde, ist es nicht möglich, das Produkt aus Ziel- oder Verteilungsbibliotheken zu entfernen.

Möglichkeit 2: Nicht-SMP/E-Installationsjobs

Passen Sie die folgenden Jobs an und führen Sie sie in der angegebenen Reihenfolge aus, um eine Nicht-SMP/E-Installation durchzuführen. Die Jobs befinden sich im Dataset **&hlq.A<prodcode>800.F3**. Lesen Sie die Kommentare am Anfang jedes Jobs und passen Sie die Jobs entsprechend an. Prüfen Sie sämtliche Ausgaben nach der Ausführung und stellen Sie sicher, dass alle Schritte erfolgreich abgeschlossen wurden.

Nicht-SMP/E-Installationsjobs:

Reihenfolge	Job	Beschreibung
1.	SL_INSTL	
2.	SL_IVP	Installationsüberprüfung

Schritt 5: Überprüfen Sie die Installationsbibliotheken

DS-Nummer	Datasetname	Beschreibung
1	TVISION.SSL_AUTH	Sensormodul (APF-Autorisierung)
2	TVISION.SSL_INST	Beispiele für Sensorinstallations-JCL
3	TVISION.SSL_LOAD	Sensormodul
4	TVISION.SSL_PROC	Sensor-Beispiel-JCL
5	TVISION.SL_800.F1	SMP/E JCLIN
6	TVISION.SL_800.F2	Installations-RELFILE - Lademodul
7	TVISION.SL_800.F3	Installations-RELFILE - Installations-JCL
8	TVISION.SL_800.SMPMCS	Installations-RELFILE - SMP/E MCS-Anweisungen

- Die SMP/E-Verteilungsbibliotheken (DLIBS) für die Installation sind in der vorherigen Tabelle nicht aufgeführt. Diese Bibliotheken werden erst bei Ausführung der SMP/E-Funktion ACCEPT aufgefüllt.
- TVISION ist das standardmäßige übergeordnete Qualifikationsmerkmal; das von Ihnen verwendete Qualifikationsmerkmal kann davon abweichen.

Schritt 6: Wiederholen Sie nach Bedarf das Verfahren zur Basiskomponenteninstallation (Schritte 3 bis 5)

In Abhängigkeit von den Mainframe-Sensorkomponenten, die für Ihre Installation lizenziert sind, kann es notwendig sein, die Schritte 3 bis 5 der Basiskomponenteninstallation mit einem anderen Produktcode (SLM oder SLD) zu wiederholen.

Konfigurieren der SLD-Sensorkomponenten unter z/OS: CICS, WebSphere MQ Batch und WebSphere MQ IMS

Weitere Informationen im Hinblick auf Sensorbetrieb, -verhalten und -konfiguration finden Sie in Kapitel 20, "Konfigurieren des CICS-, WMQ Batch- und WMQ IMS Bridge-Sensors". Diese Informationen helfen Ihnen, den Betrieb des TransactionVision-Sensors sowie seine Interaktion mit anderen Systemkomponenten besser zu verstehen.

Führen Sie die folgenden Schritte aus, um die SLD-Sensorkomponenten (CICS, WebSphere MQ Batch und WebSphere MQ IMS) unter z/OS zu konfigurieren:

- 1** Aktualisieren Sie die CICS-CSD-Datei mit den erforderlichen Ressourcendefinitionen für den SLD-Sensor, indem Sie den Job SLDCICSD anpassen und ausführen, den Sie in DS 2 (Beispiele für Sensorinstallations-JCL) finden. Wenn die Umgebung aus mehreren CICS-Regionen mit separaten CSDs oder verschiedenen Startlisten besteht, müssen Sie diesen Schritt für jede vom Sensor überwachte CICS-Region wiederholen.
- 2** Platzieren Sie die Programmlademodule des CICS-Sensors in einer Bibliothek in Ihrer DFHRPL-Verkettung. Kopieren Sie die Module entweder in eine bestehende Bibliothek, die sich bereits in der DFHRPL-Verkettung befindet, oder fügen Sie die Bibliothek SSLDLOAD zur DFHRPL-Verkettung hinzu. Wenn Sie mehrere CICS-Regionen mit separater Start-JCL ausführen, müssen Sie diesen Schritt für jede vom Sensor überwachte CICS-Region wiederholen. Es handelt sich um folgende CICS-Module:

SLDPCCX

SLDPCMX

SLDPCPX

SLDPCSX

SLDPDSX

SLDPFCX

SLDPICX

SLDPPCX

SLDPPSX

SLDPTCX

SLDPTDX

SLDPTSX

- 3** Optional. Wenn Sie die CICS-Exitprogramme des Sensors zur CICS-Startzeit automatisch aktivieren möchten, müssen Sie SLDPCSX als PLTPI-Programm der zweiten Phase definieren. Informationen hierzu finden Sie unter "DFHPLT" im Handbuch **CICS Resource Definition Guide**. Fügen Sie die Definition zur vorhandenen DFHPLTxx-Tabelle hinzu. Wenn eine neue Programmistentabelle (Program List Table, PLT) definiert wird, fügen Sie der CICS-Systeminitialisierungstabelle (System Initialization Table, SIT) Verweise auf diese Programmistentabelle hinzu. Informationen hierzu finden Sie unter "Specifying CICS system initialization parameters" im Handbuch **CICS System Definition Guide**. Wiederholen Sie diesen Schritt für jede CICS-Region, die vom Sensor überwacht werden soll. (Hinweis: Wenn dieser optionale Schritt nicht ausgeführt wird, kann die CICS-Transaktion SLDS ausgeführt werden, um Sensoren zu aktivieren.)

SIT-Beispieleintrag:

```
...
PLTPI=BI,
...
```

Beispieldefinition der PLT mit Suffixeintrag:

```
//DFHPLTPI EXEC DFHAUPLE
//ASSEM.SYSUT1 DD *
DFHPLT TYPE=INITIAL,SUFFIX=BI
DFHPLT TYPE=ENTRY,PROGRAM=DFHDELIM
DFHPLT TYPE=ENTRY,PROGRAM=SLDPCSX
DFHPLT TYPE=FINAL
END
/*
```

- 4** Richten Sie die APF-Autorisierung für die TransactionVision-Bibliothek **&hlq.SSLDAUTH** ein. Ausführliche Informationen finden Sie in der Beschreibung der PROGxx- und IEAAPFxx-Systemparameter im Handbuch **MVS Initialization and Tuning Reference**. (Hinweis: Für die Bibliothek **&hlq.SSLDLOAD** sollte die APF-Autorisierung NICHT eingerichtet werden.)

- 5 Erstellen Sie geeignete Sensorkonfigurations- und -ereigniswarteschlangen im WebSphere MQ-Warteschlangen-Manager, der für die Kommunikation zwischen der Treiberkomponente des CICS-Sensors und TransactionVision Analyzer verwendet wird. Passen Sie den Job SLDCRTQS in DS 2 (-Beispiele für Sensorinstallations-JCL) an und führen Sie ihn aus.
- 6 Passen Sie die als Beispiel bereitgestellten Sensorstartprozeduren TVISION und TVISIONC an, die Sie in DS 4 (Sensor-Beispiel-JCL) finden, und kopieren Sie sie in eine für Ihren Standort geeignete Prozedurbibliothek. Eine ausführliche Beschreibung der Schlüsselwortparameter der Startprozedur und ihrer Bedeutung finden Sie in Kapitel 20, "Konfigurieren des CICS-, WMQ Batch- und WMQ IMS Bridge-Sensors".

Konfigurieren der SLM-Sensorkomponenten unter z/OS: WebSphere MQ CICS Bridge und WebSphere MQ IMS Bridge

Weitere Informationen im Hinblick auf Sensorbetrieb und -verhalten finden Sie in Kapitel 20, "Konfigurieren des CICS-, WMQ Batch- und WMQ IMS Bridge-Sensors". Diese Informationen helfen Ihnen, den Betrieb des TransactionVision-Sensors sowie seine Interaktion mit anderen Systemkomponenten besser zu verstehen.

- ▶ Einige der folgenden Konfigurationsschritte gelten nur für den WebSphere MQ-Sensor für CICS. Diese Schritte sind mit "nur CICS" gekennzeichnet. Überspringen Sie diese Schritte, wenn Sie den WebSphere MQ-Sensor für CICS nicht verwenden möchten.
- ▶ Einige der folgenden Konfigurationsschritte gelten nur für den WebSphere MQ-Sensor für IMS. Diese Schritte sind mit "nur IMS" gekennzeichnet. Überspringen Sie diese Schritte, wenn Sie den WebSphere MQ-Sensor für IMS nicht verwenden möchten.
- ▶ Schritte, die nicht mit "nur CICS" oder "nur IMS" gekennzeichnet sind, gelten für beide Sensortypen und sollten unabhängig vom Sensortyp ausgeführt werden.

Weitere Informationen im Hinblick auf Sensorkonfiguration und -betrieb finden Sie außerdem in Kapitel 24, "Konfigurieren von WebSphere MQ-Sensoren".

- 1** (nur CICS) Lesen Sie den Abschnitt "Hintergrundinformationen: WebSphere MQ-Sensor für CICS" am Ende dieser Konfigurationsbeschreibung.
- 2** (nur CICS) Aktualisieren Sie die CICS-CSD-Datei mit den erforderlichen Ressourcendefinitionen für den SLM-Sensor, indem Sie den Job SLMCICSD anpassen und ausführen, den Sie in DS 2 (Beispiele für Sensorinstallations-JCL) finden. Wenn die Umgebung aus mehreren CICS-Regionen mit separaten CSDs oder verschiedenen Startlisten besteht, müssen Sie diesen Schritt für jede vom Sensor überwachte CICS-Region wiederholen.

Durch diesen Jobschritt wird eine Programmressourcendefinition für CSQCAPX in Ihrer CICS-Region definiert. Beachten Sie Folgendes: Wenn in Ihrer Region bereits eine Programmressource für CSQCAPX definiert und ein CSQCAPX-Programm installiert ist, müssen diese älteren Programmladmodule aus der DFHRPL-Verkettung entfernt werden, bevor das sensorspezifische CSQCAPX-Modul installiert und verwendet werden kann. Dies ist aufgrund eines starren Benennungsmechanismus erforderlich, der von WebSphere MQ für den API-Steuerübergabeexit für CICS/ WebSphere MQ verwendet wird (es kann nur ein Steuerübergabeexit installiert sein, und dieser muss den Namen CSQCAPX aufweisen).

Durch diesen Jobschritt werden außerdem Programm- und Transaktionsdefinitionen für SLMC definiert.

- 3** (nur CICS) Platzieren Sie die Lademoduls CSQCAPX und SLMC in einer Bibliothek innerhalb Ihrer DFHRPL-Verkettung. Sie können die Module entweder in eine bestehende Bibliothek kopieren, die bereits in der DFHRPL-Verkettung vorhanden ist, oder die Bibliothek SSLDLOAD zur DFHRPL-Verkettung hinzufügen.
- 4** (nur CICS) Aktivieren Sie mithilfe der Transaktion CKQC den WebSphere MQ-API-Steuerübergabeexit in Ihrer CICS-Region. Dies kann über die Menüelemente **Connection > Modify > Enable API Exit** in der CKQC-Anzeige oder durch Angabe von Argumenten beim Aufrufen von CKQC erfolgen. Sie könnten beispielsweise Folgendes eingeben:
 - a** CKQC MODIFY N E - aktiviert den Steuerübergabeexit (E für "enable").
 - b** CKQC MODIFY N D - deaktiviert den Steuerübergabeexit (D für "disable").

- 5 Richten Sie die APF-Autorisierung für die TransactionVision-Bibliothek **&hlq.SSLMAUTH** ein. Ausführliche Informationen finden Sie in der Beschreibung der PROGxx- und IEAAPFxx-Systemparameter im Handbuch **MVS Initialization and Tuning Reference**. (Hinweis: Für die Bibliothek **&hlq.SSLMLOAD** sollte die APF-Autorisierung NICHT eingerichtet werden.)
- 6 Erstellen Sie geeignete Sensorkonfigurations- und -ereigniswarteschlangen im WebSphere MQ-Warteschlangen-Manager, der für die Kommunikation zwischen der Treiberkomponente des Sensors und TransactionVision Analyzer verwendet wird. Passen Sie den Job SLDMRTQS in DS 4 (Sensor-Beispiel-JCL) an und führen Sie ihn aus.

Hinweis: Dieser Schritt ist eventuell nicht erforderlich, wenn zuvor der Job SLDCATQS ausgeführt wurde.

- 7 (nur CICS) Konfigurieren Sie SLMC für CICS. Dieser Schritt ist nicht erforderlich, wird jedoch nachdrücklich empfohlen, da hierdurch die Leistung von WMQ-Applikationsprogrammen in Ihrer CICS-Region erheblich verbessert werden kann, wenn die Region nicht von TransactionVision Analyzer überwacht wird. Ausführliche Informationen finden Sie im Abschnitt "Konfigurieren von SLMC für CICS" am Ende dieses Kapitels.
- 8 (nur IMS) Lesen Sie den Abschnitt "Hintergrundinformationen: WebSphere MQ IMS Bridge-Sensor" am Ende dieser Konfigurationsbeschreibung.
- 9 (nur IMS) Für TVISIONB ist ein Systemverbindungsindex (Linkage Index, LX) erforderlich, der beim erstmaligen Start nach dem einleitenden Programm laden reserviert und danach wiederverwendet wird. Informationen hierzu finden Sie im Abschnitt zum IEASYSxx-Systemparameter NSYSLX im Handbuch **MVS Initialization and Tuning Reference** für IBM z/OS.

Hintergrundinformationen: WebSphere MQ-Sensor für CICS

Der WebSphere MQ-Sensor für z/OS CICS besteht aus zwei Komponentenprogrammen. Das Programm, das die WebSphere MQ-API einer Applikation aufruft, ist ein WebSphere MQ-API-Steuerübergabeexit mit dem Namen CSQCAPX. Dieses Programm muss in der CICS-Region installiert werden, damit der Sensor Ereignisse erfassen kann. Ein zweites Programm, SLMC (mit einer zugeordneten CICS-Transaktion desselben Namens), ist optional. Dieses Programm kann ausgeführt werden, um den sensorspezifische Steuerübergabeexit in der CICS-Region je nach Sensorbedarf zu aktivieren und zu deaktivieren. Sie müssen SLMC nicht ausführen, damit der Sensor ordnungsgemäß funktioniert. Es empfiehlt sich jedoch, um die Leistung der WebSphere MQ-Applikation in Ihrer Region zu verbessern, wenn die Region nicht von Analyzer überwacht wird. Wenn Sie sich gegen die Ausführung von SLMC entscheiden, müssen Sie sicherstellen, dass der Steuerübergabeexit in der CICS-Region aktiviert bleibt.

WebSphere MQ-API-Steuerübergabeexits für CICS müssen den Programmnamen CSQCAPX aufweisen. Daher kann zu einem bestimmten Zeitpunkt immer nur ein WebSphere MQ-API-Steuerübergabeexit in einer bestimmten CICS-Region installiert sein. Wenn in der CICS-Zielregion bereits ein Programm mit diesem Namen installiert ist, müssen Sie vor der Sensorinstallation das vorhandene CSQCAPX-Lademodul aus der DFHRPL-Verkettung entfernen und die bestehenden CICS-Programmdefinitionen für CSQCAPX löschen oder deaktivieren.

Der WebSphere MQ-Sensor für z/OS CICS erfordert die Unterstützung für z/OS Language Environment. Vergewissern Sie sich vor der Installation des Sensors, dass die LE-Unterstützung in der CICS-Zielregion aktiviert ist. Die Verwendung des Sensors setzt weiterhin voraus, dass die CICS-Region LE-Programme unterstützt, die aus C kompiliert wurden. Das Verfahren zum Aktivieren der LE-Unterstützung wird in Kapitel "Installing Application Programs" im **CICS System Definition Guide** der CICS TS-Dokumentation beschrieben. Umfassende Informationen finden Sie in der Dokumentation zur Ihrer CICS-Version.

Konfigurieren von SLMC für CICS

SLMC überwacht die TransactionVision-Konfigurationswarteschlange im Hinblick auf Analyzer-Nachrichten, untersucht die dort gefundenen Nachrichten und aktiviert oder deaktiviert je nach Analyzer-Anforderung automatisch den WebSphere MQ-Steuerübergabeexit. Durch das Deaktivieren des Steuerübergabeexits, wenn die Region nicht aktiv von Analyzer überwacht wird (wenn keine Konfigurationsnachrichten gefunden werden oder wenn sich die Konfigurationsnachrichten nicht auf die CICS-Region oder den Host beziehen), kann die Applikationsleistung verbessert werden.

Um SLMC zu verwenden, müssen Sie die CICS-Region so konfigurieren, dass die Transaktion SLMC ausgeführt wird, die wiederum das Programm SLMC ausführt. Da SLMC das WebSphere MQ-Programm CSQCRST verwendet, das mit zugeordnetem Terminal ausgeführt werden muss, um den Steuerübergabeexit zu aktivieren und zu deaktivieren, muss SLMC ebenfalls mit zugeordnetem Terminal ausgeführt werden. Zur Vereinfachung des Betriebs empfiehlt es sich, SLMC mit einem sequenziellen Terminal zu verwenden. Auf diese Weise kann das Programm im Hintergrund gestartet werden, während die CICS-Region aufgerufen wird. Falls erforderlich können Sie SLMC jedoch auch direkt von einem herkömmlichen Terminal aufrufen.

Eine sequenziellen Terminaldefinition definiert Eingabe- und Ausgabedatasets, die von CICS für die Bereitstellung von Terminaleingaben und -ausgaben verwendet werden. Das Eingabedataset enthält die Namen der CICS-Transaktionen, die Sie ausführen möchten (in diesem Fall SLMC), und das Ausgabedataset empfängt die Terminalausgabe von der oder den Transaktionen. Um eine sequenzielle Terminaldefinition zu erstellen, ist Folgendes erforderlich: Assemblieren Sie Makrodefinitionen für CICS-Terminalressourcen mittels DFHAUPLE, legen Sie den Parameter TCT in der SIT so fest, dass das Lesen der TCT möglich ist, in der die assemblierten Makros abgelegt wurden, und fügen Sie DD-Namen zur Start-JCL der CICS-Region hinzu, um die Eingabe- und Ausgabedatasets für das Terminal zu definieren.

Um ein sequenzielles Terminal zu erstellen, das SLMC ausführt, verwenden Sie die Beispielfinitionen für sequenzielle Terminals, die von IBM in den CICS-SDFHSAMP-Elementen DFHTCT5\$ und DFH\$TCTS bereitgestellt werden. Umfassende Informationen zu diesem Beispiel und zum Einrichten eines sequenziellen Terminals finden Sie im Handbuch **CICS System Definition Guide** und im Handbuch **CICS Resource Definition Guide** für Ihre CICS-Version.

Um die Transaktion SLMC mit dem sequenziellen Terminal auszuführen, muss das Eingabedataset des sequenziellen Terminals (z. B. CARDIN DDNAME in dem von IBM bereitgestellten Beispiel) die beiden folgenden Zeilen enthalten:

```
SLMC\  
CESF LOGOFF\  

```

Hinweis: "\" ist das standardmäßige Datenendezeichen in CICS. Wenn Sie das Datenendezeichen über den Systeminitialisierungsparameter EODI neu definiert haben, geben Sie hier stattdessen das von Ihnen festgelegte Datenendezeichen an.

Hintergrundinformationen: WebSphere MQ IMS Bridge-Sensor

Der WebSphere MQ IMS Bridge-Sensor wird in Form von drei Komponenten implementiert, die mittels speicherübergreifender Programmaufrufe miteinander kommunizieren:

- ▶ Eine Exitroutine zur Bearbeitung von OTMA-Eingaben/Ausgaben mit der Bezeichnung DFSYIOEO, die in der IMS-Steuerregion ausgeführt wird.
- ▶ Eine Steuerfunktion mit der Bezeichnung TVISIONB, die als gestarteter Task in einem separaten Adressraum ausgeführt wird.
- ▶ Eine Ereignisverteilerfunktion mit der Bezeichnung TVISIOND, die als gestarteter Task in einem weiteren separaten Adressraum ausgeführt wird.

Anweisungen zum Abschließen der Sensorkonfiguration und zum Betrieb des WebSphere MQ IMS Bridge-Sensors finden Sie in Kapitel 24, "Konfigurieren von WebSphere MQ-Sensoren."

20

Konfigurieren des CICS-, WMQ Batch- und WMQ IMS Bridge-Sensors

Nachdem Sie die die TransactionVision-CICS-Sensoren installiert haben, müssen Sie sie konfigurieren und starten.

Dieses Kapitel umfasst die folgenden Themen:

- Übersicht auf Seite 233
- Allgemeine Sensorkomponenten auf Seite 234
- Befehle für den CICS-Sensor auf Seite 236
- Sensorbetrieb auf Seite 243
- Überlegungen zur Pufferwarteschlange auf Seite 245
- Startprozedur für TransactionVision Manager auf Seite 247
- Startprozedur für den Sensortreiber auf Seite 248

Übersicht

Alle TransactionVision-Sensoren für z/OS werden als Kombination aus einer gemeinsamen Basiskomponente – TransactionVision Manager – einer umgebungs-/subsystemspezifischen Sensor-Managerkomponente und dem Sensor selbst – WMQ Batch-Sensor oder WMQ IMS-Sensor – implementiert.

TransactionVision Manager (TVM)

TVM ist ein autorisiertes Programm, das als gestarteter Task ausgeführt wird. Das Programm steuert die anderen Komponenten und stellt gemeinsame Routinen und Strukturen bereit, die von allen Sensoren verwendet werden. Nach dem Start von TVM setzen Sie MVS-Befehle des Typs **modify** an den Jobnamen des gestarteten Tasks TVM ab, um des Starten, Stoppen, Abfragen oder Steuern von Sensoren anzufordern. Die verfügbaren Befehle sind unter "Befehle für den CICS-Sensor" auf Seite 236 aufgeführt. Setzen Sie den MVS-Befehl **stop** ab, um TVM zu beenden. In diesem Fall werden alle von TVM gesteuerten Sensoren ebenfalls beendet. Weitere Informationen finden Sie unter "Startprozedur für TransactionVision Manager" auf Seite 247.

Allgemeine Sensorkomponenten

Sensor-Manager

Es gibt zwei Sensor-Manager, einen für WebSphere MQ-Sensoren und einer für den CICS-Sensor, die jedoch beide einem allgemeinem Modell folgen und allgemeine Funktionen erfüllen, die von allen Sensoren benötigt werden. Der Sensor-Manager wird gestartet, wenn der Befehl **start <Sensor>** an TVM abgesetzt wird, und wird als Subtask von TVM ausgeführt. Durch Startparameter werden der Sensortyp und die zu überwachende Zielapplikationsgruppe identifiziert. Der Typ des Sensor-Managers ist MQBATCH, WMQ-IMS oder CICS.

Für die Sensortypen WMQ-IMS und CICS müssen Sie außerdem die IMSID oder CICS-SYSID des zu überwachenden IMS- oder CICS-Systems angeben. Sie müssen für jedes WMQ IMS- oder CICS-System, das überwacht werden soll, einen eigenen Sensor-Manager starten.

Der Sensor-Manager erstellt einen Pufferwarteschlangen-Datenraum zum Speichern von WebSphere MQ- oder CICS-Ereignissen und startet die Treiberkomponente des Sensors. Ein einziger Sensor verarbeitet alle Ereignisse, die von allen Applikationen in allen abhängigen Regionen für ein WMQ IMS-System erzeugt werden.

Sensortreiber

Es gibt zwei Sensortreiber, einen für WebSphere MQ-Sensoren und einen für den CICS-Sensor, die jedoch beide einem allgemeinen Modell folgen und allgemeine Funktionen erfüllen, die von allen Sensoren benötigt werden. Der Sensortreiber ist ein nicht autorisiertes Programm, das als gestarteter Task in seinem eigenen Adressraum ausgeführt wird.

Der Sensortreiber wird automatisch vom beim Start des Sensor-Managers gestartet und beendet, wenn der Sensor-Manager heruntergefahren wird. Durch einen Parameter im Befehl **start <Sensor>** wird die Prozedur angegeben, die zum Starten des Treibers verwendet wird. Weitere Informationen finden Sie unter "Startprozedur für den Sensortreiber" auf Seite 248.

Die Kommunikation zwischen dem Sensortreiber und TransactionVision Analyzer erfolgt über WebSphere MQ. Als Antwort auf Konfigurationsnachrichten von Analyzer weist der Treiber die Sensoren an, welche Ereignistypen erfasst und welche Filterkriterien angewendet werden sollen. Der Treiber ruft außerdem die erfassten Ereignisse aus dem Pufferwarteschlangen-Datenraum ab und sendet sie, vorausgesetzt die Filterkriterien sind erfüllt, an Analyzer.

Sensorkomponenten für die Ereigniserfassung

Jeder Sensor verfügt über eine eigene Komponente für die Ereigniserfassung, die in der Applikationsumgebung ausgeführt wird, um Transaktionsaktivitäten abzufangen und die zugeordneten Ereignisdaten in der Pufferwarteschlange zu speichern.

WebSphere MQ-Sensoren

Batch- und IMS-Programme, die überwacht werden sollen, müssen an ein TransactionVision-Stubprogramm gebunden werden, das den WebSphere MQ-Stub folgendermaßen ersetzt:

WebSphere MQ-Stub	Ersatzstub	Beschreibung
CSQBSTUB	SLDPSTBB	Batch
CSQBRSI	SLDPSTBB	Batch-RRS

WebSphere MQ-Stub	Ersatzstub	Beschreibung
CSQBRSTB	SLDPSTBB	Batch-RRS
CSQQSTUB	SLDPSTBB	IMS

Befehle für den CICS-Sensor

Die Befehle zum Starten und Stoppen von TransactionVision Manager sind MVS-Standardbefehle des Typs **start** und **stop**.

Start <TransactionVision Manager>

```
START prozname[.jobname],TVID=tvid
```

Hierbei gilt Folgendes:

- ▶ **prozname** ist der Name einer katalogisierten Prozedur zum Starten von TVM.
- ▶ **jobname** ist der MVS-Jobname, der dem gestarteten Task zugewiesen werden soll. Wird dieser Parameter nicht angegeben, entspricht der Jobname standardmäßig dem Prozedurnamen.
- ▶ **tvid** ist die eindeutige System-ID dieser TVM-Instanz, die aus maximal vier Zeichen besteht. Beachten Sie, dass **tvid** je nach Prozedurdefinition ein optionaler Parameter sein kann. Informationen hierzu finden Sie unter "Startprozedur für TransactionVision Manager" auf Seite 247.

Beispiel:

```
S TVISION, TVID=TV01
```

Die TransactionVision Manager-Komponente wird gestartet und die folgenden Meldungen werden angezeigt.

```
SLDS434I TVISION Bristol TransactionVision for z/OS – V5.0.0
SLDS400I TVISION TransactionVision Manager startup in progress.
SLDS401I TVISION TV01 TransactionVision Manager startup complete.
```

Stop <TransactionVision Manager>

```
STOP jobname
```

Hierbei gilt Folgendes:

- **jobname** ist der MVS-Jobname, der im **start**-Befehl für den zu stoppenden TVM-Job angegeben oder standardmäßig verwendet wurde.

Beispiel:

```
P TVISION
```

Die TransactionVision Manager-Komponente wird gestoppt. Alle im Rahmen der beendeten TVM-Komponente gestarteten Sensoren werden ebenfalls gestoppt. Die folgenden Meldungen werden angezeigt:

```
SLDS402I TVISION TV01 STOP command received.  
SLDS404I TVISION TV01 TransactionVision Manager termination in progress.  
SLDS405I TVISION TV01 TransactionVision Manager termination complete.
```

Wenn Sensoren innerhalb von TVM ausgeführt werden, werden die zugehörigen Beendigungsmeldungen ebenfalls angezeigt.

Hinweis: Alle anderen Sensorbefehle werden in Form standardmäßiger MVS-Befehle des Typs **modify** abgesetzt. Die meisten Befehls- und Operandennamen können auf gerade so wenig Zeichen abgekürzt werden, wie notwendig sind, um die Namen eindeutig zu identifizieren. Falls die Abkürzung nicht eindeutig ist, wird der bei alphabetischer Sortierung zuerst vorkommende Befehls- oder Operandenname, der zu dieser Abkürzung passt, verwendet. Einige Namen sind jedoch für die zukünftige Verwendung reserviert.

Start <Sensor>

```
MODIFY tvm_jobname,START sensortyp [SYSID(sysid) | IMSID(imsid)]  
[DRVRPROC(trbproz)]  
[QMGR(ws_mgr)] [CONFIGQ(konfig_ws)]  
[QBLKSIZE(ws_blkgroesse)] [MAXQBLKS(max_wsblks)]
```

Hierbei gilt Folgendes:

- ▶ **tvm_jobname** ist der MVS-Jobname, der im **start**-Befehl für den TVM-Job, der den gestarteten Sensor steuern wird, angegeben oder standardmäßig verwendet wurde.
- ▶ **START** ist der Befehlsname. Diese Angabe ist erforderlich.
- ▶ **sensortyp** ist der Sensortyp. Diese Angabe ist erforderlich. Geben Sie MQBATCH, MQIMS oder CICS an.

Wenn Sie für **sensortyp** der Wert MQBATCH angeben, werden die Parameter IMSID und SYSID nicht angegeben.

- ▶ **sysid** ist erforderlich, wenn für den Sensortyp der Wert CICS angegeben wird. **sysid** muss der SYSID der zu überwachenden CICS-Region entsprechen.
- ▶ **imsid** ist erforderlich, wenn für den Sensortyp der Wert MQIMS angegeben wird. **imsid** muss der IMSID des zu überwachenden IMS-Systems entsprechen.
- ▶ Die Angabe von **trbproz** ist optional. **trbproz** gibt den Namen der katalogisierten Prozedur zum Starten des Sensortreibers an. Weitere Informationen finden Sie unter "Startprozedur für den Sensortreiber" auf Seite 248. Der Standardname ist TVISIONC.
- ▶ Die Angabe von **ws_mgr** ist optional. **ws_mgr** gibt den Namen des WebSphere MQ-Warteschlangen-Managers an, über den die Kommunikation zwischen der Treiberkomponente des Sensors und TransactionVision Analyzer erfolgt. Der Standardwert wird als Parameter in der Startprozedur für den Sensortreiber angegeben. Weitere Informationen zur Beziehung zwischen diesem Parameter und der Startprozedur des Sensortreibers finden Sie unter "Startprozedur für den Sensortreiber" auf Seite 248.

- Die Angabe von **konfig_ws** ist optional. **konfig_ws** gibt den Namen der WebSphere MQ-Warteschlange an, von der die Treiberkomponente des Sensors TransactionVision Analyzer-Konfigurationsnachrichten empfängt. Der Standardwert wird als Parameter in der Startprozedur für den Sensortreiber angegeben. Weitere Informationen zur Beziehung zwischen diesem Parameter und der Startprozedur des Sensortreibers finden Sie unter "Startprozedur für den Sensortreiber" auf Seite 248.
- Die Angabe von **ws_blkgroesse** ist optional. **ws_blkgroesse** gibt die Größe (in MB) jedes Blocks im Pufferwarteschlangen-Datenraum an. Der Mindestwert für **ws_blkgroesse** ist **1**, der Höchstwert ist **100** und der Standardwert ist **3**. Weitere Informationen finden Sie unter "Überlegungen zur Pufferwarteschlange" auf Seite 245.
- Die Angabe von **max_wsblks** ist optional. **max_wsblks** gibt die maximale Anzahl von Pufferwarteschlangenblöcken an, die der Sensor in seinem Datenraum zuordnen darf. Jeder Warteschlangenblock weist die durch den Parameter **ws_blkgroesse** angegebene oder standardmäßig festgelegte Größe auf. Der Mindestwert für **max_wsblks** ist **3**, der Höchstwert ist **2046** und der Standardwert ist **5**. Weitere Informationen finden Sie unter "Überlegungen zur Pufferwarteschlange" auf Seite 245.

Beispiel 1:

```
F TVISION,S MQB DR(TVISIONM) QM(CSQ1)
CO(TVISION.CONFIGURATION.QUEUE)
```

Der WebSphere MQ Batch-Sensor wird gestartet: ein Sensor-Manager-Subtask von TVM wird gestartet und ein Sensortreibertask wird in einem eigenen Adressraum gestartet. Die folgenden Meldungen werden angezeigt:

```
SLDS400I TVISION MQBATCH TransactionVision sensor startup in progress.
SLDS401I TVISION MQBATCH TransactionVision sensor startup complete.
IEF403I TVISIONM - STARTED [...and other MVS messages issued when starting a
job]
+SLMS278I : TransactionVision Sensor: WMQ **** Sensor Driver startup completed.
QMGR=CSQ1, CONFIGQ=TVISION.CONFIGURATION.QUEUE.
```

Beispiel 2:

```
F TVISION,S MQI IMSID(IVP1) DR(TVISIONM) QM(CSQ1)
CO(TVISION.CONFIGURATION.QUEUE)
```

Der WebSphere MQIMS-Sensor wird gestartet: ein Sensor-Manager-Subtask von TVM wird gestartet und ein Sensortreibertask wird in einem eigenen Adressraum gestartet. Die folgenden Meldungen werden angezeigt:

```
SLDS400I TVISION MQIMS IVP1 TransactionVision sensor startup in progress.
SLDS401I TVISION MQIMS IVP1 TransactionVision sensor startup complete.

IEF403I TVISIONM - STARTED [...and other MVS messages issued when starting a
job]

+SLMS278I : TransactionVision Sensor: WMQ IVP1 Sensor Driver startup completed.
QMGR=CSQ1, CONFIGQ=TVISION.CONFIGURATION.QUEUE.
```

Beim Starten des CICS-Sensors kann der folgende TransactionVision-Fehler des Typs SLDS499S auftreten:

```
SLDS499S TVISION CICS xxxx TVISION system error. Diagnostic data follow:
SLDS499S 01 07050224 19472077 0000000C 00000034
SLDS499S 02 00E00080 00000000 0000000C 6C000611
```

Wenn die beiden letzten Werte in der dritten Zeile den in dieser Meldung aufgeführten Werten entsprechen, ist der Fehler darauf zurückzuführen, dass die im aktuellen System verfügbare Anzahl der Datenräume, für die SCOPE=COMMON gilt, nicht ausreicht. Erhöhen Sie den Wert des Parameters MAXCAD im entsprechenden **IEASYSnn**-Element von SYS1.PARMLIB und starten Sie dann das MVS-Image über IPL. Eine ausführliche Beschreibung zur Optimierung des Parameters MAXCAD finden Sie im IBM-Handbuch *MVS Initialization and Tuning Reference*.

Stop <Sensor>

```
MODIFY tvm_jobname,STOP sensortyp [SYSID(sysid) | IMSID(imsid)]
```

Hierbei gilt Folgendes:

- ▶ **tvm_jobname** ist der MVS-Jobname, der im **start**-Befehl für den TVM-Job, der den zu stoppenden Sensor steuert, angegeben oder standardmäßig verwendet wurde.
- ▶ STOP ist der Befehlsname. Diese Angabe ist erforderlich.
- ▶ **sensortyp** ist der Sensortyp. Diese Angabe ist erforderlich. Geben Sie MQBATCH, MQIMS oder CICS an.
 - ▶ Wenn Sie für **sensortyp** der Wert MQBATCH angeben, werden die Parameter SYSID und IMSID nicht angegeben.
- ▶ **sysid** ist erforderlich, wenn für den Sensortyp der Wert CICS angegeben wird. Mit **sysid** muss dieselbe SYSID angegeben werden, die im **start**-Befehl für diesen Sensor angegeben wurde.
- ▶ **imsid** ist erforderlich, wenn für den Sensortyp der Wert MQIMS angegeben wird. Mit **imsid** muss dieselbe IMSID angegeben werden, die im **start**-Befehl für diesen Sensor angegeben wurde.

Beispiel 1:

```
F TVISION,STO MQB
```

Der Sensor-Manager und der Sensortreiber des WebSphere MQ Batch-Sensors werden gestoppt. Die folgenden Meldungen werden angezeigt:

```
SLDS404I TVISION MQBATCH TransactionVision sensor termination in progress.
SLDS443I TVISION MQBATCH sensor quiescing: 14 events in buffer queue.
...
SLDS445I TVISION MQBATCH Sensor quiesce completed.
SLDS448I TVISION MQBATCH Sensor statistics:
  Events in queue:          0
  Events collected         474
  Events dispatched       474
  Events_lost              0

+SLDS27BI : TransactionVision Sensor: MQBATCH Sensor Driver is ending
IEF404I TVISIONM - ENDED [...and other MVS messages issued when stopping a job]
SLDS405I TVISION MQBATCH TransactionVision sensor termination complete.
```

Beispiel 2:

```
F TVISION,STO MQI IMSID(IVP1)
```

Der Sensor-Manager und der Sensortreiber des WebSphere MQ Batch-Sensors werden gestoppt. Die folgenden Meldungen werden angezeigt:

```
SLDS404I TVISION MQIMS IVP1 TransactionVision sensor termination in progress.
SLDS443I TVISION MQIMS IVP1 sensor quiescing: 14 events in buffer queue.
...
SLDS445I TVISION MQIMS IVP1 Sensor quiesce completed.
SLDS448I TVISION MQIMS IVP1 Sensor statistics:
Events in queue:          0
Events collected         474
Events dispatched       474
Events_lost              0

+SLDS27BI : TransactionVision Sensor: MQIMS IVP1 Sensor Driver is ending
IEF404I TVISIONM - ENDED [...and other MVS messages issued when stopping a job]
SLDS405I TVISION MQIMS IVP1 TransactionVision sensor termination complete.
```

Inquire

```
MODIFY tvn_jobname,INQUIRE STATISTICS [sensortyp] [SYSID(sysid) |
IMSID(imsid)]
```

Hierbei gilt Folgendes:

- ▶ **tvn_jobname** ist der MVS-Jobname, der im **start**-Befehl für den TVM-Job, an den die Abfrage gerichtet ist, angegeben oder standardmäßig verwendet wurde.
- ▶ **INQUIRE** ist der Befehlsname. Diese Angabe ist erforderlich.
- ▶ **sensortyp** ist der Sensortyp. Diese Angabe ist erforderlich. Geben Sie MQBATCH, MQIMS oder CICS an. Wenn der Sensortyp nicht angegeben wird, werden alle Sensortypen abgefragt.
- ▶ Sie können **imsid** oder **sysid** angeben, um den MQIMS- oder CICS-Sensor zu identifizieren, den Sie abfragen möchten. Wird die IMSID oder die SYSID nicht angegeben, werden alle Sensoren des angegebenen Typs abgefragt.

Beispiel:

```
F TVISION,I ST
```

Durch den Befehl **inquire statistics** wird Folgendes angezeigt:

```
SLDS448I TVISION MQBATCH Sensor statistics:
Events in queue:          56
Events collected:        530
Events dispatched:      474
Events_lost:             0
SLDS448I TVISION MQIMS IVP1 Sensor statistics:
Events in queue:         175
Events collected:       1068
Events dispatched:     893
Events_lost:            0
```

Sensorbetrieb

- 1** TransactionVision Manager muss gestartet werden, bevor ein beliebiger Sensor gestartet wird.
- 2** TransactionVision Manager wird mit einem TVID-Parameter gestartet, der die Instanz eindeutig identifiziert. Es können mehrere TransactionVision Manager-Instanzen gleichzeitig gestartet werden, was z. B. in Test- oder Produktionsumgebungen wünschenswert sein kann. Innerhalb derselben Umgebung können Sie Ressourcen jedoch am effizientesten nutzen und den Betrieb vereinfachen, wenn die Ausführung mehrerer Sensoren durch eine Instanz von TransactionVision Manager gesteuert wird.
- 3** Eine bestimmte IMSID oder SYSID kann nicht von mehreren Sensoren gleichzeitig verwendet werden.

- 4** Das Verbot der mehrfachen Verwendung einer bestimmten IMSID/SYSID gilt übergreifend für alle TransactionVision Manager-Instanzen. Es ist daher nicht möglich, einen Sensor für die Überwachung eines bestimmten IMS-Systems oder einer bestimmten CICS-Region zu starten, wenn dieses IMS-System oder diese CICS-Region bereits von einem Sensor überwacht wird, und zwar unabhängig davon, welche TransactionVision Manager-Instanzen aufgerufen wurden. Außerdem ist es nicht möglich, WebSphere MQ Batch-Jobs gleichzeitig durch mehrere TransactionVision Manager-Instanzen überwachen zu lassen.
- 5** Die TVID darf nicht denselben Wert wie die IMSID oder SYSID eines beliebigen Sensors aufweisen. Es wird nachdrücklich empfohlen, von vornherein eine TVID zu verwenden, die im Hinblick auf alle TransactionVision Manager-TVIDs, CICS-SYSIDs, IMSIDs und die IDs aller MVS-Subsysteme an Ihrem Standort eindeutig ist.
- 6** Durch das Stoppen einer TransactionVision Manager-Instanz werden automatisch alle von dieser Instanz gesteuerten Sensoren gestoppt.
- 7** Wenn ein Sensor gestoppt wird, teilt der Sensor-Manager der zugehörigen Sensorkomponente in der Applikationsumgebung mit, dass die weitere Datenerfassung eingestellt werden soll, um dann in den Wartemodus zu wechseln, bis der Sensortreiber alle Ereignisse in der Pufferwarteschlange abgerufen hat. Sobald alle Ereignisse abgerufen wurden (was eine Weile dauern kann), wird der Sensortreiber beendet. Anschließend wird der Sensor-Manager beendet.
- 8** Die Ausführung von TransactionVision Manager wird erst beendet, wenn alle von dieser Instanz gesteuerten Sensoren beendet wurden.
- 9** Wenn Sie TransactionVision Manager abbrechen, werden alle von dieser Instanz gesteuerten Sensoren unverzüglich beendet. Alle Ereignisse, die sich noch in der Pufferwarteschlange befinden, werden verworfen. Wird die Ausführung von TransactionVision Manager abgebrochen, bleiben außerdem einige gemeinsame Speichersteuerblöcke übrig. Dies führt dazu, dass der erste nachfolgende WebSphere MQ-Aufruf von einer aktiven Batch- oder IMS-Applikation, die an einen TransactionVision WebSphere MQ-Stub gebunden ist, die Anzeige einer Sensorfehlermeldung auslöst. Die Applikation selbst bleibt ansonsten unbeeinträchtigt; der WebSphere MQ-Aufruf wird wie gewohnt verarbeitet, so wie es ohne Sensor der Fall gewesen wäre. Alle nachfolgenden Aufrufe an den Stub, die innerhalb desselben Jobs erfolgen, führen nicht zu einer erneuten Anzeige der

Fehlermeldung. Der Sensor deaktiviert sich selbst und leitet alle Aufrufe an WebSphere MQ weiter. Werden nachfolgend weitere Jobs ausgeführt oder die IMS-abhängigen Regionen neu gestartet, ohne das Problem zuvor zu beheben, führt dies erneut zum Anzeigen einer Fehlermeldung, sobald der erste Aufruf durch eine Applikation erfolgt. Ein Neustart von TransactionVision Manager bewirkt das Bereinigen der übrig gebliebenen Steuerblöcke und verhindert nachfolgende Fehler in den Sensoren. Sie können stattdessen auch eine **reset**-Prozedur ausführen, durch die die Steuerblöcke bereinigt werden, ohne TransactionVision Manager neu starten zu müssen. Eine Beispiel für eine entsprechende **reset**-Prozedur finden Sie im Element TVISIONR der Bibliothek **hlq.SSLDPROC**.

Überlegungen zur Pufferwarteschlange

Um die Systembelastung, die durch Sensorkomponenten in den Applikationsumgebungen erzeugt wird, möglichst gering zu halten, speichern die Sensoren alle erfassten Transaktionsereignisse in einem Datenraum, der Pufferwarteschlange genannt wird. Jeder Sensor kann einen Datenraum exklusiv nutzen.

Die Pufferwarteschlange wird in Form einer definierten Anzahl von Warteschlangenblöcken mit einer bestimmten Größe konfiguriert. Beide Größen werden durch Parameter im Befehl **start <Sensor>** angegeben. Mit **MAXQBLKS** wird die maximale Anzahl der Warteschlangenblöcke angegeben und mit **QBLKSIZE** wird die Größe jedes Warteschlangenblocks in MB festgelegt. Die maximale Größe (in MB) des verwendeten Datenraums ist das Produkt aus **MAXQBLKS** und **QBLKSIZE** und darf 2 GB nicht überschreiten. **MAXQBLKS** hat standardmäßig den Wert **5** und muss mindestens auf **3** festgelegt werden. **QBLKSIZE** hat standardmäßig den Wert **3** und muss mindestens auf **1** festgelegt werden. Der Datenraum muss daher mindestens 3 MB groß sein – ein Wert, der jedoch lediglich für Tests mit geringem Transaktionsaufkommen ausreicht.

Die richtige Größe der Pufferwarteschlange variiert beträchtlich in Abhängigkeit von folgenden Faktoren: Transaktionsdurchsatz der überwachten Applikationsumgebung, Anzahl, Typ und Größe der erfassten Ereignisse, Durchsatz des Sensortreibers und Reaktionsfähigkeit der Pufferwarteschlangen-Verwaltungsfunktionen im Sensor-Manager. Diese Variablen werden weiter unten noch ausführlicher behandelt. Bevor Sie jedoch aufwändige Berechnungen durchführen und sich dabei auf Parameter stützen, die möglicherweise kaum mehr als Mutmaßungen sind, sollten Sie versuchen, die geeigneten Werte durch Ausprobieren zu ermitteln. Eine großzügigere Auslegung zu Beginn kann einen Mangel an Genauigkeit bei der Schätzung des Ereignisaufkommens ausgleichen, ohne die Systembelastung zusätzlich zu erhöhen.

Beim Sensorstart werden drei Warteschlangenblöcke zugeordnet. Wenn die TransactionVision-Applikationsumgebungskomponenten ausgeführt werden, werden die erfassten Ereignisse zuerst im ersten Warteschlangenblock gespeichert, dann im nächsten usw. Parallel dazu ruft der Sensortreiber Ereignisse aus der Pufferwarteschlange ab. Dies erfolgt in derselben Reihenfolge, in der die Ereignisse gespeichert wurden (FIFO, First In, First Out). Jede Hundertstel Sekunde werden die Pufferwarteschlangen-Verwaltungsfunktionen aufgerufen, um den Status der Pufferwarteschlange zu überprüfen. Wenn es sich bei dem Warteschlangenblock, der momentan Ereignisse empfängt, um den letzten zugeordneten Block handelt, wird ein weiterer Block zugeordnet, vorausgesetzt die Gesamtzahl der zugeordneten Blöcke überschreitet nicht den mit MAXQBLKS festgelegten Wert. Jeder Warteschlangenblock, aus dem alle Ereignisse abgerufen wurden, wird wieder freigegeben, wodurch sich die Anzahl der zugeordneten Blöcke reduziert. Aus diesem Grund nimmt die Größe der Pufferwarteschlange in Abhängigkeit vom Datenverkehr zu und ab. Unabhängig vom zugelassenen Maximalwert wird jenseits der Mindestgröße von drei Warteschlangenblöcken nie mehr Speicherplatz verwendet, als tatsächlich benötigt wird. Eine großzügigere Auslegung zu Beginn kann einen Mangel an Genauigkeit bei der Schätzung des Ereignisaufkommens ausgleichen, ohne die Systembelastung zusätzlich zu erhöhen.

Startprozedur für TransactionVision Manager

Wenn Sie TransactionVision Manager (TVM) starten, müssen Sie der Instanz eine TVID zuweisen, die im Hinblick auf alle parallel ausgeführten TVM-Instanzen eindeutig ist. Die TVID wird als einziger Parameter an das Hauptprogramm, SLDPTVM, übergeben. Im Folgenden sehen Sie die Beispielprozedur TVISION zum Starten von TVM:

```
//TVISION PROC TVID=TV01,TVHLQ=TVISION
//TVISION EXEC PGM=SLDPTVM,TIME=1440,PARM='&TVID'
//STEPLIB DD DISP=SHR,DSN=&TVHLQ..SSLDAUTH
//      PEND
```

Für TVM ist ein Parameter erforderlich. Hierbei handelt es sich um einen aus vier Zeichen bestehenden TVID-Wert, der im Hinblick auf alle parallel ausgeführten TVM-Instanzen eindeutig ist. Die TVID muss sich auch von den IDs unterscheiden, die den parallel ausgeführten Sensoren zugewiesen sind. Es wird nachdrücklich empfohlen, von vornherein eine TVID zu verwenden, die im Hinblick auf alle TVM-Instanzen, CICS-SYSIDs, IMSIDs und die IDs aller MVS-Subsysteme an Ihrem Standort eindeutig ist.

Die TVID ist als Schlüsselwortparameter der Prozedur definiert; der Standardwert ist TV01. Durch den Operatorbefehl START TVISION wird TVM mit dem TVID-Wert TV01 gestartet. Durch den Operatorbefehl START TVISION,TVID=TV02 wird TVM mit dem TVID-Wert TV02 gestartet. Wenn beide Befehle abgesetzt werden, werden zwei TVM-Instanzen gestartet, wobei beide Instanzen denselben Jobnamen, TVISION, verwenden.

Zum Stoppen von TVM oder zum Starten und Stoppen von Sensoren, die von TVM gesteuert werden, müssen STOP- bzw. MODIFY-Befehle abgesetzt werden, in denen der Jobname der TVM-Instanz angegeben wird, auf die der Befehl abzielt. Wenn mehrere TVM-Instanzen mit demselben Jobnamen gestartet werden, gilt der Befehl für alle fraglichen Instanzen. Sie können dieses Problem vermeiden, indem Sie im Befehl START einen eindeutigen Jobnamen angeben. Es empfiehlt sich, die TVID als Jobnamen zu verwenden. Beispiel:

```
START TVISION,JOBNAME=TV01,TVID=TV01
START TVISION,JOBNAME=TV02,TVID=TV02
```

Startprozedur für den Sensortreiber

Wenn Sie den Befehl **start <Sensor>** an TVM absetzen, wird die Sensortreiberkomponente in einem eigenen Adressraum gestartet. Der Sensortreiber wird automatisch durch den Sensor-Manager gestartet, wobei die Prozedur zum Einsatz kommt, die im Befehl **start <Sensor>** angegeben ist (siehe "Start <Sensor>" auf Seite 238). Die folgenden Parameter aus dem Befehl **start <Sensor>** werden an die Prozedur für den Sensortreiber übergeben: SYSID, QMGR und CONFIGQ.

Durch den Parameter DRVRPROC im Befehl **start <Sensor>** wird die Prozedur angegeben, die zum Starten des Sensortreibers aufgerufen wird. Die Namen der Standardprozeduren sind TVISIONM für WebSphere MQ-Sensoren und TVISIONC für den CICS-Sensor. Folgende Beispiele für diese Prozeduren finden Sie in der Bibliothek **hlq,SSLDPROC**:

```
//TVISIONM PROC SYSID=,QMGR=CSQ1,
//          CONFIGQ=TVISION.CONFIGURATION.QUEUE,
//          TVHLQ=TVISION,MQHLQ=CSQ520
//TVISIONM EXEC PGM=SLDPMDR,TIME=1440,
//          PARM='&SYSID &QMGR &CONFIGQ'
//STEPLIB DD DISP=SHR,DSN=&TVHLQ..SSLDLOAD
//         DD DISP=SHR,DSN=&MQHLQ..SCSQAUTH
//         PEND
//TVISIONC PROC SYSID=,QMGR=CSQ1,
//          CONFIGQ=TVISION.CONFIGURATION.QUEUE,
//          TVHLQ=TVISION,MQHLQ=CSQ520
//TVISIONC EXEC PGM=SLDPcdr,TIME=1440,
//          PARM='&SYSID &QMGR &CONFIGQ'
//STEPLIB DD DISP=SHR,DSN=&TVHLQ..SSLDLOAD
//         DD DISP=SHR,DSN=&MQHLQ..SCSQAUTH
//         PEND
```

SYSID, QMGR und CONFIGQ sind als Schlüsselwortparameter der Prozedur mit den angegebenen Standardwerten definiert. Bedenken Sie, dass diese Prozedur automatisch durch den Sensor-Manager aufgerufen wird. Sie sollten diese Prozedur nicht auf anderem Wege aufrufen. Sie sollten beispielsweise nicht den Befehl **START TVISIONM** absetzen.

Je nach Kommunikationsverbindungen zwischen den Sensortreibern und TransactionVision Analyzer können Sie diese Prozedur anpassen oder mehrere Prozeduren erstellen.

Durch die Parameter für den Websphere MQ-Warteschlangen-Manager (QMGR) und die Konfigurationswarteschlange (CONFIGQ) wird bestimmt, welche Analyzer-Instanzen mit den Sensortreibern kommunizieren. In der Beispielprozedur können die Parameter QMGR und CONFIGQ außer Kraft gesetzt werden, indem die gleichen Parameter im Befehl **start <Sensor>** angegeben werden. Der Parameter SYSID muss im Befehl **start <Sensor>** bereitgestellt werden. Die Parameter SYSID, QMGR und CONFIGQ werden als durch Leerzeichen getrennte Positionsparameter an das Hauptprogramm des Sensortreibers, SLDPMDR oder SLDPCCR, übergeben. Daher darf nur der letzte Parameter, CONFIGQ, weggelassen werden. Wenn Sie diesen Parameter nicht angeben, wird er standardmäßig auf TVISION.CONFIGURATION.QUEUE festgelegt. Beim Anpassen dieser Prozeduren oder beim Erstellen eigener Prozeduren für den Sensortreiber sollten Sie folgende Konventionen einhalten:

- Geben Sie immer alle drei Parameter – SYSID, QMGR und CONFIGQ – als Parameter der Prozedur an.
- Geben Sie für SYSID immer den Standardwert **null** an.
- Geben Sie für QMGR immer einen Standardwert ungleich null an.
- Übergeben Sie im Feld PARM immer zumindest den SYSID- und den QMGR-Wert an SLDPCCR.

Hinweis: Der Parameter SYSID in den Sensortreiberprozeduren wird generisch verwendet und steht entweder für eine SYSID oder eine IMSID.

21

Installieren und Konfigurieren von Sensoren in BEA Tuxedo

Dieses Kapitel umfasst die folgenden Themen:

- Vorbereiten der Installation auf Seite 251
- Ausführen der Installation auf Seite 252
- Erneutes Binden des Tuxedo-Sensors auf Seite 253
- Deinstallieren von Sensoren auf Seite 253

Vorbereiten der Installation

Die Sensorinstallation wird vom Produkt-Installationsdatenträger für Business Availability Center oder von der Seite **Downloads** in Business Availability Center gestartet.

In der folgenden Tabelle sind die Namen der Installationsdateien der BEA Tuxedo-Pakete für die verschiedenen unterstützten Plattformen aufgeführt.:

Plattform	Dateien
AIX	tvtux_800_aix_power.tar.gz
HP-UX	tvtux_800_hpux_parisc.tar.gz
Solaris	tvtux_800_sol_sparc.tar.gz

Ausführen der Installation

- 1 Wechseln Sie zum Verzeichnis, in dem sich die TransactionVision-Installationsdateien befinden (entweder ein DVD-Laufwerk oder ein Downloadverzeichnis). HINWEIS: Unter Solaris und HP-UX müssen Sie die Installationsdateien stattdessen vom DVD-Laufwerk in ein temporäres Verzeichnis auf der Festplatte des Hostsystems kopieren.

- 2 Melden Sie sich als Superuser an:

```
su
```

- 3 Entpacken Sie die Pakete für Ihre Plattform; weitere Informationen finden Sie unter "Vorbereiten der Installation" auf Seite 251. Beispiel:

```
gunzip tvtux_800_hpux_parisc_tar.gz
```

- 4 Geben Sie den folgenden Befehl ein, um das Installationsverfahren zu starten:

```
./tvinstall_800_unix.sh
```

Nachdem das Paket entzippt wurde, wird ein Menü angezeigt, in dem die verfügbaren Komponenten aufgeführt sind. Beispiel:

```
The following TransactionVision packages are available for installation:
```

1. TransactionVision Analyzer
2. TransactionVision UI/Job Server
3. TransactionVision WebSphere MQ Agent
4. TransactionVision BEA Tuxedo Agent
5. TransactionVision User Event Agent

```
99. All of above  
q. Quit install
```

```
Please specify your choices (separated by,) by number/letter:
```

- 5 Zum Installieren einer einzigen Komponente geben Sie die Nummer ein, die dem Paket der TransactionVision-Komponente zugeordnet ist, und drücken dann die EINGABETASTE.

Zum Installieren mehrerer, jedoch nicht aller Komponenten geben Sie die Nummern (durch Kommas getrennt) ein, die den zu installierenden Komponenten zugeordnet sind, und drücken dann die EINGABETASTE. Wenn Sie beispielsweise alle Sensoren installieren möchten, die im oben beschriebenen Menü aufgeführt sind, geben Sie Folgendes ein und drücken dann die EINGABETASTE.

3,4

Zum Installieren aller verfügbaren Komponenten geben Sie **99** ein und drücken dann die EINGABETASTE.

Das oder die angegebenen Pakete werden durch das Installationskript installiert. Anschließend wird erneut das Menü angezeigt.

- 6 Geben Sie **q** ein und drücken Sie die EINGABETASTE, um das Installationsverfahren zu beenden. Informationen zum Installieren weiterer Komponenten finden Sie in den Installationsanweisungen für die betreffenden Komponenten.

Erneutes Binden des Tuxedo-Sensors

Anders als beim WMQ-Sensor wird bei der Installation des Tuxedo-Sensors nicht automatisch das Skript zum erneuten Binden aufgerufen.

Führen Sie das Skript `rebind_tux_sensor` aus, um die Sensorbibliothek erneut zu binden. Weitere Informationen zu diesem Skript finden Sie in Anhang A, "Referenz der Dienstprogramme".

Deinstallieren von Sensoren

Führen Sie die folgenden Schritte aus, um den Sensor oder andere TransactionVision-Komponenten auf dem Host zu deinstallieren:

- 1 Melden Sie sich als Superuser an:

```
su
```

- 2 Geben Sie den folgenden Befehl ein:

```
./tvinstall_800_unix.sh -u
```

Das folgende Menü wird angezeigt (beachten Sie, dass die tatsächlich angezeigten Optionen von den auf dem Computer installierten TransactionVision-Paketen abhängen):

The following TransactionVision packages are available for installation:

1. TransactionVision Analyzer
2. TransactionVision UI/Job Server
3. TransactionVision WebSphere MQ Agent
4. TransactionVision BEA Tuxedo Agent
5. TransactionVision User Event Agent

99. All of above
q. Quit install

Please specify your choices (separated by,) by number/letter:

- 3** Geben Sie die Nummer des TransactionVision-Pakets ein, das Sie deinstallieren möchten, und drücken Sie die EINGABETASTE.

Zum Deinstallieren aller TransactionVision-Komponenten geben Sie **99** ein und drücken dann die EINGABETASTE.

Das angegebene Paket wird durch das Installationskript deinstalliert. Anschließend wird erneut das Menü angezeigt.

- 4** Geben Sie **q** ein und drücken Sie die EINGABETASTE, um die Deinstallation zu beenden. Wenn das allgemeine Paket das einzige noch installierte TransactionVision-Paket ist, wird es automatisch deinstalliert.

Konfigurieren von BEA Tuxedo-Sensoren

Bevor Sie TransactionVision verwenden können, um Ereignisdaten für eine Applikation aufzuzeichnen, müssen Sie die Applikationsumgebung so konfigurieren, dass die Sensorbibliothek anstelle der BEA Tuxedo-Standardbibliothek geladen wird.

Die erforderliche Konfiguration besteht aus den folgenden Schritten:

- ▶ Konfigurieren der Applikationsumgebung, sodass die Sensorbibliothek geladen wird.

- Festlegen der Dienst-URL für die Analyzer-Konfiguration.

Der folgende Konfigurationsschritt ist optional:

- Filtern sensibler Daten.

Konfigurieren der Applikationsbibliothek, sodass die Sensorbibliothek geladen wird.

Die BEA Tuxedo-Sensorbibliothek wird dynamisch zur Laufzeit geladen, indem ihr Bibliothekssuchpfad vor dem Suchpfad für die BEA Tuxedo-Standardbibliothek angegeben wird. Die BEA Tuxedo-Standardbibliothek wird dynamisch durch die Sensorbibliothek geladen.

Fügen Sie den Verzeichnispfad der Sensorbibliothek zu einer Umgebungsvariableneinstellung auf dem Computer hinzu, auf dem die überwachte Applikation ausgeführt wird. Dieser Schritt muss vor dem Start der Applikation erfolgen.

In der folgenden Tabelle sind die geeignete Umgebungsvariable und der Verzeichnispfad aufgeführt, die für die verschiedenen Plattformen festgelegt werden sollten, wenn Sie 32-Bit-Applikationen verwenden. Wenn ein Pfad, der die BEA Tuxedo-Standardbibliothek einschließt, bereits in der Umgebungsvariablen angegeben ist, muss der Sensoreintrag vor diesem Pfad hinzugefügt werden, um TransactionVision verwenden zu können.

Plattform	Umgebungsvariable	Standardverzeichnis
Sun Solaris	LD_LIBRARY_PATH	/opt/HP/TransactionVision/lib
HP-UX	SHLIB_PATH	/opt//HP/TransactionVision/lib
IBM AIX	LIBPATH	/usr/lpp/HP/TransactionVision/lib

Alle Verzeichnispfade in dieser Tabelle entsprechen den Standardinstallationsverzeichnissen des Sensors. Wenn der Sensor in einem anderen als dem Standardverzeichnis installiert wurde, müssen Sie den Verzeichnispfad zur ausführbaren Datei des Sensors angeben.

Wenn Sie 64-Bit-Applikationen verwenden, legen Sie die Bibliothekspfade wie in der folgenden Tabelle aufgeführt fest:

Plattform	Umgebungsvariable	Standardverzeichnis
Sun Solaris	LD_LIBRARY_PATH	/opt/HP/TransactionVision/lib64:/opt/mqm/lib64
HP-UX	SHLIB_PATH	/opt/HP/TransactionVision/lib64:/opt/mqm/lib64
IBM AIX	LIBPATH	/usr/lpp/HP/TransactionVision/lib64:/usr/lpp/mqm/lib64

Auf der AIX-Plattform müssen Sie `/usr/sbin/slibclean` ausführen, um die ursprüngliche gemeinsam genutzte Bibliothek aus dem Speicher zu entfernen, bevor eine neue Bibliothek geladen werden kann, die denselben Namen wie die bestehende Bibliothek aufweist.

Auf der HP-UX-Plattform müssen Sie eventuell den Befehl `chatr` verwenden, um der Applikation das Laden der Sensorbibliothek zu ermöglichen, falls die Verwendung der Umgebungsvariablen `SHLIB_PATH` nicht aktiviert ist oder falls diese Variable nicht als erste im Suchpfad für dynamische Bibliotheken der Applikation aufgeführt ist. Sie können den Befehl `chatr` verwenden, um die aktuellen Einstellungen der Applikation zu überprüfen. Stellen Sie in jedem Fall sicher, dass `SHLIB_PATH` aktiviert und vor dem Pfad für die BEA Tuxedo-Standardbibliothek angegeben ist. Weitere Informationen finden Sie unter "Troubleshooting" im Handbuch *Verwenden von TransactionVision*.

Festlegen der Sensorverbindungs-URL

Sie müssen die Eigenschaftendatei des BEA Tuxedo-Sensors (`TVISION_HOME/config/TuxedoSensor.properties`) bearbeiten und die Option `transport` im Abschnitt `static configuration` auf die Sensorverbindungs-URL festlegen.

Die Sensorverbindungs-URL wird im Rahmen der Definition der Kommunikationsverbindung definiert. Sie erhalten die URL, indem Sie auf die Vorlage für Kommunikationsverbindungen zugreifen und die Angabe im Feld **Connection URL** kopieren:

The screenshot shows the 'Business Availability Center - TransactionVision Administration' web interface. The breadcrumb path is 'Administration > Communication Link Templates'. The current page is 'Edit Communication Link Template' for 'TransactionVision HTTP (15.43.248.176:21111)'. The 'Communication Link Name' field contains 'TransactionVision HTTP (15.43.248.176:21111)'. Under the 'Sensor Connection' section, the 'Event Connection' sub-section shows the 'Connection URL' field with the value 'http://ros83853tst.ovrtest.adapps.hp.com:'.

Filtern sensibler Daten

Der BEA Tuxedo-Sensor ist standardmäßig so konfiguriert, dass Nutzdaten (Benutzerdaten) aus überwachten Aufrufen erfasst werden. Falls die Benutzerdaten sensible Informationen enthalten, kann es möglicherweise gewünscht sein, diese Funktionalität zu deaktivieren. Hierfür müssen Sie die Eigenschaftendatei des BEA Tuxedo-Sensors (**TVISION_HOME/config/TuxedoSensor.properties**) bearbeiten und die Option **collectUserData** im Abschnitt **static configuration** auf **no** festlegen.

22

Installieren und Konfigurieren des .NET-Agents

Mit dem .NET-Agent wird die Funktionalität der .NET-Probe von Diagnostics und des .NET-Sensors von TransactionVision in einer einzigen Komponente zusammengefasst. Auf einem .NET-Host kann der .NET-Agent gleichzeitig als TransactionVision-Sensor und Diagnostics-Probe fungieren.

Mit dem .NET-Agent steht eine ressourcensparende Lösung zur Ereigniserfassung zur Verfügung, die in Kombination mit Business Availability Center-Applikationen von HP verwendet werden kann. Der .NET-Agent erfasst Ereignisse aus einer .NET-Applikation und sendet die Ereignismessdaten an TransactionVision Analyzer, an den Diagnostics-Server oder an beide.

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Informationen zum Installationsprogramm des .NET-Agents auf Seite 260
- ▶ Installieren des .NET-Agents auf Seite 260
- ▶ Konfigurieren des .NET-Agent auf Seite 267
- ▶ Erneutes Starten von IIS auf Seite 275
- ▶ Bestimmen der Version des .NET-Agent auf Seite 276
- ▶ Deinstallieren des .NET-Agents auf Seite 276
- ▶ SSL-Konfiguration für .NET-Agents auf Seite 277

Informationen zum Installationsprogramm des .NET-Agents

Während der Installation des .NET-Agents werden die folgenden Installations- und Konfigurationsschritte für Sie ausgeführt:

- ▶ Discovery von ASP.NET-Applikationen. Das Installationsprogramm versucht, automatisch die ASP.NET-Applikationen auf dem System zu erkennen, auf dem der Agent installiert ist.
- ▶ Agentstandardkonfiguration
 - ▶ Durch das Installationsprogramm wird der Agent so konfiguriert, dass für jede erkannte ASP.NET-Applikation grundlegende ASP.NET- ADO- und WCF-Arbeitslastdaten erfasst werden. Die Agentkonfiguration wird mithilfe der Datei **probe_config.xml** gesteuert. Informationen hierzu finden Sie unter "Konfigurieren des .NET-Agent" auf Seite 267.
 - ▶ Standardmäßige .NET-Applikationsinstrumentierung. Die Standarddateien **ASP.NET.points** und **ADO.NET.points** werden installiert und aktiviert. Diese Dateien stellen die Standardinstrumentierung bereit und ermöglichen es Ihnen, mit der Überwachung von ASP.NET-Applikationen zu beginnen. Mit den POINTS-Dateien wird die Arbeitslast gesteuert, die der Agent für die Applikation erfasst.

Die standardmäßigen POINTS-Dateien **ASP.NET.points** und **ADO.NET.points** müssen aktiviert sein, um TransactionVision-Ereignisse zu erzeugen. Sie sind standardmäßig aktiviert.

Zum Erzeugen von .NET Remoting-Ereignissen benötigen Sie außerdem **Remoting.points** und müssen die Applikation für die Instrumentierung einrichten. Weitere Informationen hierzu finden Sie in der Remoting-Dokumentation.

- ▶ Optionale Konfiguration. Bearbeiten Sie die Agentkonfiguration in der Datei **probe_config.xml**. Informationen hierzu finden Sie unter "Konfigurieren des .NET-Agent" auf Seite 267.

Installieren des .NET-Agents

Im folgenden Abschnitt finden Sie ausführliche Anweisungen zum Installieren des .NET-Agents.

Dieser Abschnitt umfasst die folgenden Themen:

- "Vorbereiten der Installation" auf Seite 261
- "Starten der Installation" auf Seite 261
- "Ausführen der Installation" auf Seite 262
- "Installieren des Agents für den Einsatz in einer TransactionVision-Umgebung" auf Seite 265

Vorbereiten der Installation

Der .NET-Agent wird auf dem Hostcomputer der Applikation installiert, die Sie überwachen möchten. Die Systemlast, die durch den Sensor für .NET (.NET-Probe) auf dem zu überwachenden System erzeugt wird, ist ausgesprochen niedrig.

Im Folgenden sind die empfohlenen Werte im Hinblick auf Arbeitsspeicher- und Speicherplatzausstattung aufgeführt, um den Sensorbetrieb zu unterstützen:

Plattform	Alle Plattformen
Arbeitsspeicher	60 MB zusätzlicher Arbeitsspeicher
Freier Speicherplatz	200 MB zusätzlicher Speicherplatz
.NET Framework	1.1 oder höher

Wichtig: Der .NET-Agent schließt einen SOAP-Erweiterungshandler ein. Die Installation des .NET-Agents kann dazu führen, dass vorhandene Webapplikationen, die SOAP verwenden, neu gestartet werden.

Starten der Installation

Das Installationsprogramm wird vom Produkt-Installationsdatenträger für Business Availability Center oder von der Seite **Downloads** in Business Availability Center gestartet.

So starten Sie die Installation vom Produkt-Installationsdatenträger:

- 1 Suchen Sie die für Ihre Plattform geeignete Installationspaketdatei:

Plattform	Dateien
32-Bit-.NET	DotNetAgentSetup_x86_8_00.msi
64-Bit-.NET	DotNetAgentSetup_x64_8_00.msi

- 2 Führen Sie die Datei im Stammverzeichnis des Installationsdatenträgers aus. Fahren Sie mit den Schritten unter "Ausführen der Installation" auf Seite 262 fort.

So starten Sie die Installation vom Downloadcenter für HP BTO-Software:

- 1 Geben Sie **TransactionVision** in das Feld **Keyword** und **Trial software** in das Feld **Refine Search by resource type** ein.
- 2 Klicken Sie auf den entsprechenden Link, um das .NET-Agent-Installationsprogramm für Ihre Plattform herunterzuladen. Fahren Sie mit den Schritten unter "Ausführen der Installation" auf Seite 262 fort.

Ausführen der Installation

Nachdem Sie das Installationsprogramm gestartet haben, können Sie mit der Hauptinstallation beginnen.

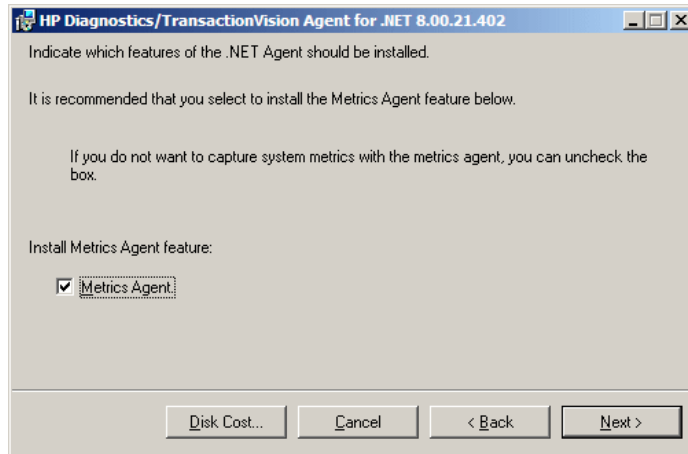
So installieren Sie den .NET-Agent auf einem Windows-basierten Computer:

- 1 Stimmen Sie der Endbenutzer-Lizenzvereinbarung zu.
Lesen Sie die Vereinbarung und wählen Sie **I accept the terms of the license agreement** aus.
Klicken Sie auf **Next**, um fortzufahren.
- 2 Geben Sie den Speicherort an, an dem der Agent installiert werden soll.
Standardmäßig wird der Agent unter **C:\MercuryDiagnostics\.NET Probe** installiert.

Übernehmen Sie das Standardverzeichnis oder wählen Sie einen anderen Speicherort aus, indem Sie entweder den Pfad zum Installationsverzeichnis in das Feld **Installation HP Diagnostics/TransactionVision Agent for .NET to** eingeben oder auf **Browse** klicken, um zum gewünschten Installationsverzeichnis zu navigieren.

Klicken Sie auf **Next**, um fortzufahren.

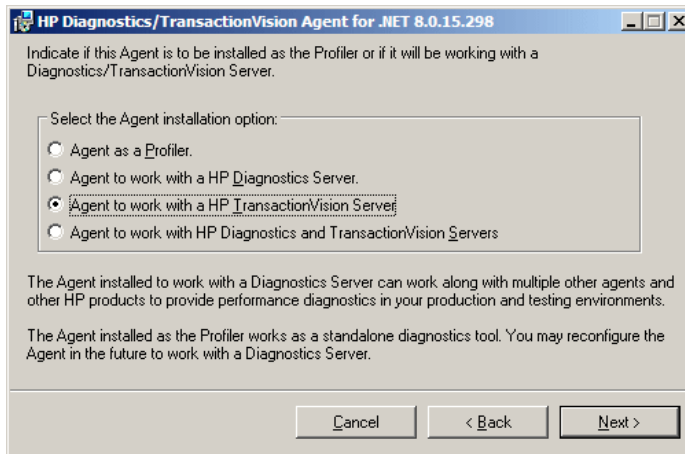
- 3 Wählen Sie die Features des .NET-Agents aus, die Sie installieren möchten.



Wählen Sie **Metrics Agent** aus, wenn Sie nur Systemmessdaten auf dem Hostcomputer erfassen möchten, ohne eine der vorhandenen Applikationen zu überwachen.

Klicken Sie auf **Disk Cost**, um den Umfang des Speicherplatzes zu überprüfen, der auf den Laufwerken des Hostsystems verfügbar ist. Mit dieser Funktionalität können Sie sicherstellen, dass ausreichend Speicherplatz für die Agentinstallation vorhanden ist.

- 4 Geben Sie an, ob der Agent ausschließlich als Diagnostics-Profiler, als Probe, die Daten an einen Diagnostics Server meldet, oder als Agent, der Daten an TransactionVision Analyzer meldet, installiert werden soll.



- Wählen Sie die Option **Agent to work with an HP TransactionVision Server** aus.

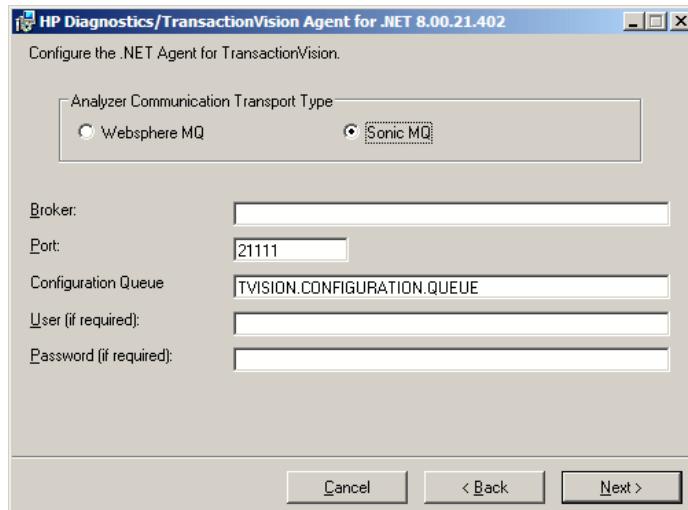
Klicken Sie auf **Next**, um fortzufahren.

Hinweis: Wenn Sie den Agent als Diagnostics-Profiler oder für den Einsatz mit einem Diagnostics-Server installieren, finden Sie weitere Informationen im Handbuch *HP Diagnostics Installation and Configuration Guide*.

Installieren des Agents für den Einsatz in einer TransactionVision-Umgebung

Fahren Sie mit dem folgenden Verfahren fort, wenn Sie den Agent installieren, um ihn in einer TransactionVision-Umgebung zu verwenden.

- 1 Das Dialogfeld **Configure the .NET Agent for TransactionVision** wird angezeigt.



- 2 Wählen Sie den Provider für Message Oriented Middleware aus. Folgende Optionen sind verfügbar: WebSphere MQ und SonicMQ.

SonicMQ wird mit dem .NET-Agent bereitgestellt. Wenn Sie diese Option auswählen, wird der .NET-Client von SonicMQ (**Sonic.Client.dll**, Progress SonicMQ .NET Client, Version 7.6.0.112) im Rahmen der Agentinstallation installiert.

Sie können stattdessen jedoch auch eine separate WebSphere MQ-Installation verwenden. In diesem Fall müssen Sie den .NET-Client für MQSeries (**amqmdnet.dll**, WebSphere MQ Classes for .NET, Version 1.0.0.3) auf dem zu überwachenden Host installieren.

SonicMQ ist standardmäßig ausgewählt.

- 3 Geben Sie für SonicMQ Folgendes ein:

Broker. Der Name des Hosts, auf dem der Sonic-Broker ausgeführt wird. Normalerweise ist dies der Name des Analyzer-Hosts.

Port. Der Port, über den der Broker kommuniziert. Standardmäßig handelt es sich um Port 21111.

Konfigurationswarteschlange. Der Name der Konfigurationswarteschlange. Standardmäßig ist dies TVISION.CONFIGURATION.QUEUE.

User. Die Benutzer-ID, falls diese von der SonicMQ-Installation zum Herstellen der Verbindung benötigt wird. Standardmäßig ist kein Benutzername erforderlich.

Password. Das Kennwort, falls es von der SonicMQ-Installation zum Herstellen der Verbindung benötigt wird. Dieses Kennwort wird in verschleierter Form mit dem Dienstprogramm **PassGen** erstellt. Standardmäßig ist kein Kennwort erforderlich. Weitere Informationen zu **PassGen** finden Sie unter "Referenz der Dienstprogramme" auf Seite 359.

4 Geben Sie für WebSphere MQ Folgendes ein:

Host. Der Host, auf dem sich der WebSphere MQ-Warteschlangen-Manager befindet.

Port. Die Portnummer für den WebSphere MQ-Warteschlangen-Manager.

Konfigurationswarteschlange. Der Name der Konfigurationswarteschlange.

User. Die Benutzer-ID, falls diese von der WebSphere MQ-Installation zum Herstellen der Verbindung benötigt wird.

Password. Das Kennwort, falls es von der WebSphere MQ-Installation zum Herstellen der Verbindung benötigt wird. Dieses Kennwort wird in verschleierter Form mit dem Dienstprogramm **PassGen** erstellt. Weitere Informationen zu **PassGen** finden Sie unter "Referenz der Dienstprogramme" auf Seite 359.

Websphere MQ-Kanal. Der Kanalname für den WebSphere MQ-Warteschlangen-Manager.

Websphere MQ Q Manager. Die CCSID für WebSphere.

Klicken Sie auf **Next**, um fortzufahren.

5 Das Dialogfeld mit einer Zusammenfassung der ausgewählten Installationsoptionen wird angezeigt.

Klicken Sie auf **Install**, um mit der Installation fortzufahren.

Nach Abschluss der Installation müssen Sie IIS neu starten (Informationen hierzu finden Sie unter "Erneutes Starten von IIS" auf Seite 275). Sie können nun nach Wunsch beliebige benutzerdefinierte Konfigurationen vornehmen, wie im folgenden Abschnitt beschrieben wird.

Konfigurieren des .NET-Agent

Die Standardkonfiguration des .NET-Agent ermöglicht es Ihnen, mit der Erfassung von Leistungsmessdaten für die überwachte Applikation zu beginnen. Die Konfiguration des Agents kann angepasst werden, sodass sie zur Konfiguration Ihrer Umgebung und zu den Leistungsproblemen passt, die Sie untersuchen möchten.

Mithilfe der Datei `<sensor_install_verz>/etc/probe_config` können Sie die Standardkonfiguration außer Kraft setzen.

Verwenden Sie die folgenden Elemente:

► `<tv>`, `<time skew>`, `<transport>`

Diese Elemente werden nur von TransactionVision unterstützt.

► `<logging>`, `<modes>`

Diese Elemente werden von TransactionVision und Diagnostics unterstützt.

Umfassende Informationen zur Datei `probe_config` finden Sie im Handbuch *HP Diagnostics Installation and Configuration Guide*.

Das Element <tv>

Zweck

Konfiguriert den .NET-Agent für den Einsatz mit TransactionVision.

Attribute

Attribute	Gültige Werte	Standardwert	Beschreibung
eventthreads	Zahl	3	(Wird beim Start gelesen) Die Anzahl der Threads, die vom Agent generiert werden, um Ereignisse an Analyzer zu senden.
eventthreadsleep	Zahl	100	(Dynamisch) Der Zeitraum (in Millisekunden), für den der Ereignisthread inaktiv ist, nachdem eine Nachricht (ein Ereignispaket) gesendet wurde.
eventmemorythreshold	Zahl	250.000.000	(Dynamisch) Der Arbeitsspeicher, der vom internen Puffer (Warteschlange) beansprucht wird, bevor der Agent versucht, die Nachricht über den Applikationsthread zu senden.
configthreadsleep	Zahl	10.000	(Dynamisch) Der Zeitraum (in Millisekunden), für den der Ereignisthread inaktiv ist, nachdem die Konfigurationswarteschlange durchsucht wurde.

Elemente

Verwendungshäufigkeit	1 (ein Mal)
Übergeordnete Elemente	ProbeConfig
Untergeordnete Elemente	transport, timeskew

Beispiel

```
<tv eventthreads="3" eventthreadsleep="80"
eventmemorythreshold="25000000" configthreadsleep="10000" >
  <timeskew historysize="24" checkinterval="300000" latencythreshold="100"
  retrythreshold="8"/>
  <transport type="sonicmq"
  connectionstring="broker=myhost.mydomain.com;
  port=21111; user=; password=;
  configurationqueue=TVISION.CONFIGURATION.QUEUE"/>
</tv>
```

Das Element <timeskew>

Zweck

Berechnet den Zeitunterschied zwischen dem Zeitserver und dem Host, auf dem der .NET-Agent ausgeführt wird. Die Häufigkeit des Abgleichs mit dem Zeitserver kann konfiguriert werden.

Attribute

Attribute	Gültige Werte	Standardwert	Beschreibung
historysize	Zahl	24	(Wird beim Start gelesen) Die Anzahl der Zeitabweichungsstichproben, die gespeichert und im Hinblick auf die beste Stichprobe verglichen werden.
checkinterval	Zahl	300.000 ms.	(Dynamisch) Der Zeitraum (in Millisekunden), der verstreicht, bevor der Zeitserver zur Berechnung der Zeitabweichung überprüft wird.
latencythreshold	Zahl	100 ms.	(Dynamisch) Die Zeit (in Millisekunden), die eine Antwort von einem Zeitserver maximal benötigen darf, um einen gültigen Zeitabweichungswert zu ermitteln.
retrythreshold	Zahl	8	(Dynamisch) Die Anzahl der Wiederholungsversuche, wenn eine Anfrage an einen Zeitserver nicht erfolgreich ist.

Elemente

Verwendungshäufigkeit	1 (ein Mal)
Übergeordnete Elemente	tv
Untergeordnete Elemente	Keine

Beispiel

```
<timeskew historysize="24" checkinterval="300000" latencythreshold="100"
retrythreshold="8"/>
```

Das Element <transport>

Zweck

Konfiguriert den von TransactionVision verwendeten Ereigniskanal.

Attribute

Attribute	Gültige Werte	Standardwert	Beschreibung
type	mqseries sonicmq	sonicmq	Der Ereignistransportprovider, der vom Agent verwendet wird.
connectionString	Siehe unten.		Die Verbindungsinformationen für den Ereignistransportprovider.

connectionString-Syntax bei type=sonicmq

```
broker = <broker>; port = <port>; user = <benutzer>; password =<kennwort>;
configurationQueue = <konfigurations_ws>
```

Platzhalter:	Bedeutung:
broker	Der Name des Hosts, auf dem der Sonic-Broker ausgeführt wird. Normalerweise ist dies der Name des Analyzer-Hosts.
port	Der Port, über den der Broker kommuniziert. Standardmäßig handelt es sich um Port 21111.
benutzer	Die Benutzer-ID, falls diese von der SonicMQ-Installation zum Herstellen der Verbindung benötigt wird. Standardmäßig ist kein Benutzername erforderlich.
kennwort	Das Kennwort, falls es von der SonicMQ-Installation zum Herstellen der Verbindung benötigt wird. Dieses Kennwort wird in verschleierter Form mit dem Dienstprogramm PassGen erstellt. Standardmäßig ist kein Kennwort erforderlich. Weitere Informationen zu PassGen finden Sie unter "Referenz der Dienstprogramme" auf Seite 359.
konfigurations_ws	Der Name der Warteschlange, in der sich die Konfigurationsnachrichten für den .NET-Agent von TransactionVision befinden.

connectionString-Syntax bei type=mqseries

```
host= <host>; queuemanager=<ws_manager>; port= <port>; channel= <kanal>
configurationQueue = <konfigurations_ws>
```

Platzhalter:	Bedeutung:
host	Der Host, auf dem sich die TransactionVision-Konfigurationswarteschlange befindet.
ws_manager	Der Name des Warteschlangen-Managers.
port	Der MQSeries-Port, über den der Warteschlangen-Manager kommuniziert.
kanal	Der MQSeries-Kanal, der für die Kommunikation verwendet wird.
konfigurations_ws	Der Name der Warteschlange, in der sich die Konfigurationsnachrichten für den .NET-Sensor von TransactionVision befinden.

Elemente

Verwendungshäufigkeit	1 (ein Mal)
Übergeordnete Elemente	tv
Untergeordnete Elemente	Keine

Beispiel

Für SonicMQ:

```
<transport type="sonicmq" connectionString="broker=brokerHost;
port=21111; user=; password=;
configurationqueue=TVISION.CONFIGURATION.QUEUE"/>
```

Für MQSeries:

```
<transport type="mqseries" connectionString="host=mqHost;
queuemanager=; port=1414; channel=TRADING.CHL;
configurationqueue=TVISION.CONFIGURATION.QUEUE"/>
```

Das Element <logging>

Das Tag TVDEBUG kann angegeben werden, um TransactionVision-spezifischen Code im .NET-Agent zu verfolgen.

Beispiel:

```
<logging level="TVDEBUG"/>
```

Weitere Informationen zum Element <<logging> finden Sie im Handbuch *HP Diagnostics Installation and Configuration Guide*.

Das Element <modes>

Mit dem Tag **tv** von <modes> geben Sie den TransactionVision-Modus für den Agent an.

Beispiel:

```
<modes tv="true"/>
```

Wenn dies der einzige angegebene Modus ist, wird der Agent im Modus "nur TV" ausgeführt, d. h., der Diagnostics-Profilierer und die Diagnostics-Probe sind deaktiviert und es werden ausschließlich TransactionVision-Ereignisse erzeugt.

Wenn weitere Modi angegeben sind, werden sowohl TransactionVision- als auch Diagnostics-Ereignisse erzeugt.

Weitere Informationen zum Element <modes> finden Sie im Handbuch *HP Diagnostics Installation and Configuration Guide*.

.NET Remoting-Client- und Serverapplikationen

Das folgende Beispiel veranschaulicht die Konfigurationsänderungen, die für den TransactionVision-Transport in einer .NET Remoting-Umgebung erforderlich sind.

Änderungen an **probe_config.xml**:

```
<tv eventthreads="5" eventthreadsleep="100"  
eventmemorythreshold="25000000" configthreadsleep="60000" >
```

```

<timeskew historysize="20" checkinterval="300000" latencythreshold="100"
  retrythreshold="6"/>
<transport type="sonicmq"
  connectionstring="broker=myhost.mydomain.com;
  port=21111; user=; password=;
  configurationqueue=TVISION.CONFIGURATION.QUEUE"/>
</tv>
<process name="TVRemotingClient">
  <points file="Remoting.points"/>
  <points file="TVRemotingClient.points"/>
  <modes enterprise="true" pro="true" tv="true"/>
</process>

```

Aktivieren der Korrelierung von .NET-Ereignissen

Standardmäßig werden .NET-Ereignisse nicht korreliert. Nehmen Sie die folgenden Änderungen an der Konfiguration auf dem Analyzer-Host vor, um die Benutzerkorrelierungsbean zu aktivieren:

- ▶ Entfernen Sie die Kommentarzeichen im .NET-Agent-Abschnitt in der Datei **EventCorrelationDefinition.xml**.
- ▶ Entfernen Sie die Kommentarzeichen für das folgende Tag in **Beans.xml**:

```

<Attribute name="UserCorrelationBean"
  value="com.bristol.tvision.services.analysis.eventanalysis.XMLRuleCorrelationBean"/>

```

Weitere Informationen zur Datei **EventCorrelationDefinition.xml** finden Sie im Handbuch *TransactionVision Advanced Customization Guide*.

Erneutes Starten von IIS

Nachdem Sie den .NET-Agent installiert und nach Bedarf die Konfiguration geändert oder eine benutzerdefinierte Instrumentierung erstellt haben, müssen Sie IIS neu starten, bevor Sie den .NET-Agent mit ASP.NET-Applikationen verwenden können.

Geben Sie Folgendes ein, um IIS über die Befehlszeile oder über **Start > Ausführen** neu zu starten:

iisreset

Durch diesen Befehl wird der Webpublishingdienst neu gestartet; er führt jedoch nicht zu einem unmittelbaren Start des Sensors. Sobald das nächste Mal eine Webseite in der Applikation angefordert wird, wird der Sensor gestartet, die Applikationen werden instrumentiert und der Sensor liest die von Analyzer gesendeten Nachrichten in der Konfigurationswarteschlange.

Hinweis: In bestimmten Situationen löst ASP.NET den automatischen Neustart von Applikationen aus. Dies trifft u. a. zu, wenn festgestellt wurde, dass Applikationen erneut bereitgestellt wurden, oder wenn Applikationen den konfigurierten Ressourcenschwellenwert überschreiten.

Bestimmen der Version des .NET-Agent

Wenn Sie Unterstützung anfordern, ist es hilfreich zu wissen, welche Versionen der verschiedenen Komponenten installiert sind.

So bestimmen Sie die Version des .NET-Agents:

- Klicken Sie mit der rechten Maustaste auf die Datei `<.net_agent_install_verz>\bin\HP.Profiler.dll` und zeigen Sie die Versionsinformationen der Komponente an, indem Sie im angezeigten Menü die Option **Eigenschaften** auswählen.

Deinstallieren des .NET-Agents

Führen Sie die folgenden Schritte aus, um den .NET-Agent zu deinstallieren:

- 1 Beenden Sie alle Webapplikationen, die SOAP verwenden.
- 2 Verwenden Sie die Option **Software** in Windows.
Entfernen Sie das Programm **HP TransactionVision/Diagnostics Agent for .NET**.
- 3 Starten Sie die Webapplikationen erneut.

SSL-Konfiguration für .NET-Agents

Wenn der .NET-Agent SonicMQ als Message Oriented Middleware verwendet, kann SSL aktiviert werden. Informationen hierzu finden Sie im Handbuch *HP Business Availability Center Hardening Guide* (PDF).

23

Installieren und Konfigurieren von Sensoren in NonStop TMF

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zum NonStop TMF-Sensor auf Seite 279
- Vorbereiten der Installation auf Seite 280
- Installieren des NonStop TMF-Sensors auf Seite 280
- Starten/Beenden auf Seite 282
- Konfigurieren des NonStop TMF-Sensors auf Seite 283

Informationen zum NonStop TMF-Sensor

Alle überwachten Transaktionen auf HP NonStop Guardian-Systemen werden in Audit-Trail-Dateien protokolliert. Dies schließt jede Form des Datenbankzugriffs ein, angefangen bei indizierten Dateien (B-Baum) bis hin zu SQL-Datenbanken. Das Produkt, durch das die Daten im Audit-Trail erzeugt werden, heißt Transaction Monitoring Facility (TMF). Da TMF alle überwachten Dateien auf dem NonStop-System schützt, fungiert es als Repository für sämtliche Änderungen. Zur Unterstützung von OLTP-Applikationen (Online Transaction Processing,

Onlinetransaktionsverarbeitung) können mit TMF Tausende komplexer Transaktionen überwacht werden, die von Hunderten Benutzern an die Audit-Trail-Datenbank gesendet werden. TMF bietet darüber hinaus Funktionalität für Transaktionsschutz, Datenbankkonsistenz und Datenbankwiederherstellung, was im Rahmen der Verarbeitung großer Transaktionsmengen von enormer Bedeutung ist.

Mit dem NonStop TMF-Sensor wird die Audit-Trail-Datenbank im Hinblick auf Änderungen (Hinzufügungen, Löschungen und Bearbeitungen) überwacht, die an Enscribe-Datenbanken vorgenommen wurden. Durch den NonStop TMF-Sensor werden keine Änderungen an relationalen SQL-Datenbanken überwacht; es wird nur die Überwachung von Enscribe-Dateien unterstützt.

Da TMF alle überwachten Änderungen in der Audit-Trail-Datenbank aufzeichnet, kann der NonStop TMF-Sensor die relevanten Audit-Trail-Datensätze lesen und TransactionVision-Benutzerereignisse erzeugen, die zur Analyse an TransactionVision Analyzer weitergeleitet werden.

Vorbereiten der Installation

Die Sensorinstallation wird vom Produkt-Installationsdatenträger für Business Availability Center oder von der Seite **Downloads** in Business Availability Center gestartet.

In der folgenden Tabelle sind die Namen der Installationsdateien der NonStop TMF-Pakete für die verschiedenen unterstützten Plattformen aufgeführt.:

Plattform	Dateien
Guardian	tvtmf_800_nonstop_guardian.zip

Installieren des NonStop TMF-Sensors

- 1 Entzippen Sie das Installationspaket in ein temporäres Verzeichnis.

2 Öffnen Sie die Datei **FTPSCRIPT.txt** und ändern Sie sie folgendermaßen:

- ▶ <NonStop Username> in einen gültigen NonStop-Benutzer.
- ▶ <NonStop Userpassword> in das Kennwort des betreffenden Benutzers.
- ▶ <NonStop destination volume.subvolume> in das Zielvolume und das Untervolume.

3 Übertragen Sie die Dateien mit FTP, und verwenden Sie hierfür das bearbeitete FTP-Skript.

Beispiel:

```
ftp -s:FTPSCRIPT.txt <NonStop host DNS/IP>
```

4 Melden Sie sich als **super.super** am NonStop-Host an und wechseln Sie dort zur temporären Installation **\$volume.subvol**.

5 Führen Sie das TACL-Makro **INSTALL** aus und geben Sie das Zielinstallationsvolume und -untervolume sowie den DNS-Namen oder die IP-Adresse des Hosts an. Beispiel:

```
135> run install
This program collects configuration information in order to set up the TransactionVision
environment. This includes:
- Installation volume and subvolume where to install the sensor
- Location of the TransactionVision Analyzer Configuration Service

You will be prompted to input required configuration parameters.
If the previous value is provided in (), pressing <Enter> will set
the parameter to the previous value.

Enter the installation volume (): $data02.tvision

Enter the analyzer configuration service DNS or IP address (): 15.178.196.101

Sensor installation complete!
```

Starten/Beenden

Der NonStop TMF-Sensor verfügt über drei TACL-Makros, die zum Starten und Beenden verwendet werden:

- **COLDMON**. Startet den Sensor zum ersten Mal.
- **STRTMON**. Startet den Sensor, nachdem er gestoppt wurde.
- **STOPMON**. Stoppt den Sensor.

COLDMON

Durch dieses Makro werden alle temporären Daten, die vom Sensor verwendet werden (Ereignisse, die in der Warteschlange platziert wurden), gelöscht und es wird die erste Audit-Datei erstellt, um mit der Suche nach Ereignissen zu beginnen, die die User Event-Filterkriterien erfüllen.

Starten Sie das Makro für den erstmaligen Start, indem Sie zu dem Volumen bzw. Untervolumen wechseln, in dem der NonStop-Sensor installiert ist und dort das Makro ausführen (mit anderen Worten: Führen Sie COLDMON aus).

Nach dem Starten von COLDMON werden Sie aufgefordert, die Audit-Trail-Datei einzugeben, um mit der Suche nach Ereignissen zu beginnen.

Beispiel:

```
Select TMF Audittrail file to commence scan from:
File 1: $AUDIT.ZTMFAT.AA000006
File 2: $AUDIT.ZTMFAT.AA000007
File 3: $AUDIT.ZTMFAT.AA000008
Select number from list above :
```

Wenn Sie sicher wissen, dass der Sensor mit der Suche in einer bestimmten Audit-Trail-Datei beginnen soll, können Sie diese Datei eingeben. Beginnen Sie andernfalls einfach mit der ersten Datei, die angezeigt wird (wählen Sie **1** aus).

Der Sensor beginnt mit der Suche nach Ereignissen, die die Filterkriterien erfüllen. Diese Suche schließt alle Ereignisse bis zum aktuellen Zeitpunkt ein. Wenn der Sensor zum ersten Mal gestartet wird, gibt es keine zu erfassenden Ereignisse. Sie müssen erst einen Filter konfigurieren. Der Sensor überprüft die Audit-Dateien und wartet auf neue Audit-Trail-Datensätze.

STRTMON

Wenn der Sensor gestoppt wird, speichert er die letzte Position des momentan gelesenen Datensatzes in der Audit-Trail-Datei. Auf diese Weise kann er beim Neustart an der Position mit der Suche nach Ereignissen fortfahren, an der die Suche unterbrochen wurde. Führen Sie das TACL-Makro STRTMON aus, um den NonStop TMF-Sensor nach dem Stoppen erneut zu starten. Beispiel:

```
run STRTMON
```

STOPMON

Führen Sie das TACL-Makro STOPMON aus, um den NonStop-Sensor zu stoppen. Beispiel:

```
run STOPMON
```

Konfigurieren des NonStop TMF-Sensors

Der NonStop TMF-Sensor wird mithilfe zweier Quellen konfiguriert:

- ▶ Mit dem TCL-Makro INSTALL, durch das Werte in den Startmakros (COLDMON, STRTMON) festgelegt werden.

- ▶ Mit den Daten, die von Analyzer empfangen werden.

Durch das TACL-Makro INSTALL werden alle erforderlichen Werte in den Makros COLDMON und STRTMON festgelegt, sodass es in den meisten Fällen nicht mehr notwendig ist, dass Benutzer Änderungen an den Startmakros vornehmen. Nach der Installation des Sensors werden bei der Ausführung des Sensors sämtliche Werte verwendet, die durch das Makro INSTALL festgelegt wurden.

Der NonStop-Sensor liest den Analyzer-Konfigurationsdienst, um u. a. Transportdetails zum Senden von Ereignissen zu bestimmen und zu ermitteln, wie Datensammlungen gefiltert werden sollen. In Analyzer sind die Transportdetails Teil der Konfiguration der Kommunikationsverbindung. Die Informationen zur Filterung der Datensammlung stammen aus der Konfiguration des Datensammlungsfilters. Informationen hierzu finden Sie unter "Communication Links" im Handbuch *Verwenden von TransactionVision*.

24

Konfigurieren von WebSphere MQ-Sensoren

In TransactionVision werden zwei Sensortypen für WebSphere MQ bereitgestellt:

- WebSphere MQ-Sensorbibliothek
- WebSphere MQ-API-Exit

Dieses Kapitel umfasst die folgenden Themen:

- Konfigurieren der WebSphere MQ-Sensorbibliothek auf Seite 286
- Konfigurieren des Exit-Sensors für WebSphere MQ-APIs auf Seite 298
- WebSphere MQ-Sensoren und FASTPATH_BINDING auf Seite 306
- Verwenden von Sensoren mit WebSphere MQ-Beispielen auf Seite 306
- Überwachen von WebSphere MQ-Clientapplikationen auf Seite 307
- Verwenden des WebSphere MQ IMS Bridge-Sensors auf Seite 313
- Verwenden des WebSphere Business Integration-Sensors auf Seite 320

Konfigurieren der WebSphere MQ-Sensorbibliothek

Die WebSphere MQ-Sensorbibliothek wird dynamisch zur Laufzeit geladen, indem ihr Bibliothekssuchpfad vor dem Suchpfad für die WebSphere MQ-Standardbibliothek angegeben wird. Die WebSphere MQ-Standardbibliothek wird dynamisch durch die Sensorbibliothek geladen. Bevor Sie TransactionVision verwenden können, um Ereignisdaten für eine Applikation aufzuzeichnen, müssen Sie die Applikationsumgebung so konfigurieren, dass die Sensorbibliothek anstelle der WebSphere MQ-Standardbibliothek geladen wird.

Wichtig: Wenn Sie die Sensorbibliothek mit 64-Bit-Applikationen, einschließlich 64-Bit-Java, verwenden, kann es zu Bibliothekskonflikten zwischen den 32-Bit-Bibliotheken von WebSphere MQ und den 64-Bit-Bibliotheken kommen. Informationen zum Beheben von Problemen, die bei dem Versuch, die Sensorbibliothek für 64-Bit-Applikationen zu verwenden, auftreten können, finden Sie unter "Implications of a 64-bit queue manager" im Handbuch *WebSphere MQ Quick Beginnings*.

Verteilte Plattformen

Zu den verteilten Plattformen zählen alle Plattformen außer z/OS und i5/OS. Auf verteilten Plattformen müssen Sie den Verzeichnispfad der Sensorbibliothek zu einer Umgebungsvariableneinstellung auf dem Computer hinzufügen, auf dem die überwachte Applikation ausgeführt wird. Dieser Schritt muss vor dem Start der Applikation erfolgen.

In der folgenden Tabelle sind die geeignete Umgebungsvariable und der Verzeichnispfad aufgeführt, die für die verschiedenen Plattformen festgelegt werden sollten, wenn Sie 32-Bit-Applikationen verwenden. Wenn ein Pfad, der die WebSphere MQ-Standardbibliothek einschließt, bereits in der Umgebungsvariablen angegeben ist, muss der Sensoreintrag vor diesem Pfad hinzugefügt werden, um TransactionVision verwenden zu können.

Plattform	Umgebungsvariable	Standardverzeichnis
Windows	PATH	C:\Programme\HP\TransactionVision\lib
Sun Solaris	LD_LIBRARY_PATH	/opt/HP/TransactionVision/lib
HP-UX	SHLIB_PATH	/opt//HP/TransactionVision/lib
IBM AIX	LIBPATH	/usr/lpp/HP/TransactionVision/lib
RedHat Linux	LD_LIBRARY_PATH	/opt/HP/TransactionVision/lib

Alle Verzeichnispfade in dieser Tabelle entsprechen den Standardinstallationsverzeichnissen des Sensors. Wenn der Sensor in einem anderen als dem Standardverzeichnis installiert wurde, müssen Sie den Verzeichnispfad zur ausführbaren Datei des Sensors angeben.

Wenn Sie 64-Bit-Applikationen verwenden, legen Sie die Bibliothekspfade wie in der folgenden Tabelle aufgeführt fest:

Plattform	Umgebungsvariable	Standardverzeichnis
Sun Solaris	LD_LIBRARY_PATH	/opt/HP/TransactionVision/lib64:/opt/mqm/lib64
HP-UX	SHLIB_PATH	/opt/HP/TransactionVision/lib64:/opt/mqm/lib64
IBM AIX	LIBPATH	/usr/lpp/HP/TransactionVision/lib64:/usr/lpp/mqm/lib64
RedHat Linux x86-64	LD_LIBRARY_PATH	/opt/HP/TransactionVision/lib64:/opt/mqm/lib64

Auf der AIX-Plattform müssen Sie `/usr/sbin/slibclean` ausführen, um die ursprüngliche gemeinsam genutzte Bibliothek aus dem Speicher zu entfernen, bevor eine neue Bibliothek geladen werden kann, die denselben Namen wie die bestehende Bibliothek trägt.

Auf der HP-UX-Plattform müssen Sie eventuell den Befehl `chatr` verwenden, um der Applikation das Laden der Sensorbibliothek zu ermöglichen, falls die Verwendung der Umgebungsvariablen `SHLIB_PATH` nicht aktiviert ist oder falls diese Variable nicht als erste im Suchpfad für dynamische Bibliotheken der Applikation aufgeführt ist. Sie können den Befehl `chatr` verwenden, um die aktuellen Einstellungen der Applikation zu überprüfen. Stellen Sie in jedem Fall sicher, dass `SHLIB_PATH` aktiviert und vor dem Pfad für die WebSphere MQ-Standardbibliothek angegeben ist. Ausführliche Informationen finden Sie unter "Troubleshooting" im Handbuch *Using TransactionVision*.

Wichtig: Bei der Verwendung von Multithread-Applikationen mit WebSphere MQ auf UNIX-Systemen müssen Sie sicherstellen, dass die Applikationen über genug Stapelspeicher für die Threads verfügen. Von IBM wird eine Stapelgröße von mindestens 256 KB empfohlen, wenn Multithread-Applikationen MQI-Aufrufe absetzen. Wird der TransactionVision WebSphere MQ-Sensor verwendet, kann sogar noch mehr Stapelspeicher erforderlich sein; die empfohlene Stapelgröße liegt hier bei mindestens 512 KB. Weitere Informationen finden Sie in Kapitel 7, "Connecting to and disconnecting from a queue manager", im Handbuch *WebSphere MQ Application Programming Guide*.

Führen Sie das mit TransactionVision bereitgestellte Skript `wmqsensor` folgendermaßen aus, um Applikationen, die durch Sensoren überwacht werden sollen, auszuführen, ohne die Bibliotheksumgebung global festzulegen:

Auf UNIX-Plattformen:

```
installationsverzeichnis/wmqsensor applikationsbefehlszeile
```

Auf Windows-Plattformen:

installationsverzeichnis\wmqsensor applikationsbefehlszeile

Verwenden Sie folgenden Befehl, wenn Sie die WMQ-Applikation beispielsweise unter dem Namen amqsput ausführen:

installationsverzeichnis\wmqsensor amqsput...

z/OS Batch, IMS und WebSphere MQ IMS Bridge

Auf den z/OS Batch- und IMS-Plattformen enthält das Element SLMLKSTB der Beispielprozedurbibliothek **thlqual.SSLMPROC** einen Beispieljob, um den Sensor an ein WebSphere MQ-Programm zu binden. Der Abschnitt für den WebSphere MQ-Batchstub – CSQBSTUB, CSQBRRSI oder CSQBRSTB für Batch oder CSQQSTUB für IMS – wird im Applikationslademodul durch den entsprechenden Sensorbatch oder IMS-Stub – SLMBSTUB, SLMBRRSI, SLMBRSTB oder SLMQSTUB – ersetzt. Nach diesem Binden ruft die Applikation den Sensor und nicht mehr direkt WebSphere MQ auf.

Beachten Sie, dass **thlqual** dem übergeordneten Qualifikationsmerkmal entspricht, das der Systemadministrator bei der Installation von TransactionVision ausgewählt hat. Wenn das übergeordnete Qualifikationsmerkmal beispielsweise TVISION lautet, ist TVISION.SSLMPROC der Name der Beispielprozedurbibliothek.

Hinweis: Wenn die aktuellen Applikationen z/OS Resource Recovery Services (RRS) im Batchbetrieb verwenden, werden die Aufrufe von SRRCMIT und SRRBACK nicht vom Sensor erfasst, sondern einfach an WebSphere MQ weitergeleitet.

Stellen Sie beim Ausführen der Applikation sicher, dass die Bibliothek, die den Sensor enthält, in der Verkettung LNKLST, STEPLIB oder JOBLIB angegeben ist. Definieren Sie in UNIX System Services die Umgebungsvariable STEPLIB, um die Bibliothek anzugeben, die den Sensor enthält.

z/OS CICS

Auf der z/OS CICS-Plattform verwenden Sensoren den Mechanismus für API-Steuerübergabeexits, der durch den CICS-Adapter von WebSphere MQ für z/OS bereitgestellt wird.

i5/OS

Auf der i5/OS-Plattform trägt das Sensor-Hauptserviceprogramm denselben Namen wie das WebSphere MQ-Serviceprogramm: LIBMQM für nicht threadbasierte Programme und LIBMQM_R für Threadprogramme. Die beiden TransactionVision-Serviceprogramme haben die gleiche Signatur und die gleichen exportierten Symbole (in derselben Reihenfolge) wie ihre Gegenstücke in WebSphere MQ.

TransactionVision stellt außerdem zwei Diensts-serviceprogramme bereit:

- ▶ MQMUTL5 stellt eine Bindung zu QMQM/LIBMQM her.
- ▶ MQMUTL5_R stellt eine Bindung zu QMQM/LIBMQM_R her.

Das Sensor-Hauptserviceprogramm stellt eine Bindung zu einem dieser drei Serviceprogramme her, sodass ein MQI-Aufruf eines Programms an das WebSphere MQ-Serviceprogramm weitergegeben wird.

Verwenden Sie eine der folgenden Methoden, um das Sensorserviceprogramm zu verwenden:

- ▶ Das Benutzerprogramm wird durch CRTPGM mit dem Parameter BNDSRVPGM(QMQM/LIBMQM), BNDSRVPGM(QMQM/LIBMQM_R oder BNDSRVPGM(QMQM/AMQZSTUB) erstellt.

Geben Sie die Parameter ALWUPD(*YES) und ALWLIBUPD(*YES) für CRTPGM an. Die Standardwerte sind *YES für ALWUPD und *NO für ALWLIBUPD. Wenn das Benutzerprogramm ohne diese Parameter erstellt wird, binden Sie es mit einer dieser Methoden erneut.

Nach dem Binden des Programms können Sie UPDPGM verwenden, um zwischen den von WebSphere MQ und dem Sensor bereitgestellten Serviceprogrammen zu wechseln. Der Parameter für diesen Befehl ist SRVPGMLIB. Sie können diesen Parameter auf QMQM oder TVSENSOR festlegen.

- Das Benutzerprogramm wird durch CRTPGM mit dem Parameter BNDSRVPGM(*LIBL/LIBMQM) oder BNDSRVPGM(*LIBL/LIBMQM_R) erstellt.

Verwenden Sie nach dem Binden des Programms ADDLIBLE oder CHGLIBL, um zwischen der WebSphere MQ-Bibliothek QMQM und der Sensorbibliothek TVSENSOR zu wechseln. Die Sensorbibliothek muss in der Bibliotheksliste vor der WebSphere MQ-Bibliothek QMQM angegeben werden, um TransactionVision verwenden zu können.

Hinweis: Beachten Sie, dass ein RPG-Programm, das durch den ILE RPG-Compiler erstellt wird, das gleiche WebSphere MQ-Serviceprogramm wie das C-Programm verwendet, das durch den ILE C-Compiler erstellt wird. TransactionVision wurde in den folgenden Szenarios getestet, wenn MQI in einem ILE RPG-Programm verwendet wird.

- Verwenden von MQI über einen Aufruf von MQM
- Verwenden von mittels Prototyp getesteter Aufrufe der MQI

Konfigurieren der Sensorprotokollierung

Bei einigen Betriebssystemen sind keine weiteren Schritte erforderlich, um die Fehler- und Ablaufprotokollierung durch die WebSphere MQ-Sensoren zu aktivieren. Auf UNIX-Plattformen kann es jedoch notwendig sein, **syslogd** so zu konfigurieren, dass Protokollierung mit dem von den WebSphere MQ-Sensoren verwendeten Protokollierungstyp erfolgt. Ausführliche Informationen zur WebSphere MQ-Sensorprotokollierung auf UNIX-Plattformen finden Sie in Kapitel 26, "Konfigurieren der Agent- und Sensorprotokollierung".

Festlegen des Namens der Konfigurationswarteschlange

Standardmäßig suchen Sensoren im Warteschlangen-Manager, der im WebSphere MQ-API-Aufruf angegeben ist, nach einer Konfigurationswarteschlange mit dem Namen TVISION.CONFIGURATION.QUEUE. Sie können jedoch einen anderen Namen für die Konfigurationswarteschlange angeben, wenn Sie eine Kommunikationsverbindung erstellen. Wenn Sie eine Kommunikationsverbindung verwenden, die einen anderen Namen als den Standardnamen für Konfigurationswarteschlangen angibt, müssen Sie den Sensor so konfigurieren, dass Konfigurationsnachrichten in dieser Warteschlange und nicht in TVISION.CONFIGURATION.QUEUE gesucht werden.

UNIX, Windows und i5/OS

Legen Sie auf UNIX-, Windows- und i5/OS-Plattformen die Umgebungsvariable TVISION_CONFIGURATION_QUEUE auf die Sensor-Konfigurationswarteschlange fest, die Sie in der Kommunikationsverbindung für alle Prozesse angegeben haben, die den Sensor verwenden.

IBM z/OS Batch, IMS und WebSphere MQ IMS Bridge

In IBM z/OS Batch, IMS und WebSphere MQ IMS Bridge kann ein vom Benutzer installierbares Lademodul mit dem Namen SLMBCNFG erzeugt werden, um zu steuern, welche Warteschlange vom Sensor gelesen wird, um Analyzer-Konfigurationsnachrichten abzurufen.

- 1** Bearbeiten Sie `thlqual.SSLMPROC(SLMBCFGQ)` und nehmen Sie folgende Änderungen vor:
 - a** Ändern Sie den Wert der Anweisung `SET CONFIGQ`, sodass er dem Namen der gewünschten Konfigurationswarteschlange entspricht.
 - b** Ändern Sie die Anweisung `SET THLQUAL`, sodass das sie das übergeordnete Qualifikationsmerkmal der TransactionVision-Sensorbibliotheken angibt.
 - c** Wenn die von IBM bereitgestellte Prozedur `HLASMCL` auf Ihrem System nicht verfügbar ist, ändern Sie die JCL nach Bedarf, um den angegebenen Quellcode zu assemblieren und zu binden. Nehmen Sie keine Änderung am Quellcode vor.

- 2** Senden Sie `thlqual.SSLMPROC(SLMBCFGQ)` ab, damit die Änderung wirksam wird.

Weitere Informationen zum WebSphere MQ IMS Bridge-Sensor finden Sie unter "Verwenden des WebSphere MQ IMS Bridge-Sensors" auf Seite 313.

In IBM z/OS CICS kann ein vom Benutzer installierbares Programm mit dem Namen `SLMCNFQ` erzeugt werden, um zu steuern, welche Warteschlangen vom Sensor gelesen werden, um Analyzer-Konfigurationsnachrichten abzurufen.

Wenn das Programm `SLMCNFQ` vom Sensor über den Befehl `EXEC CICS LINK` ausgeführt werden kann, führt der Sensor das Programm aus und übergibt einen CICS-Kommunikationsbereich an `SLMCNFQ`, der groß genug ist, um den Namen einer Konfigurationswarteschlange aufzunehmen (48 Bytes). Der Kommunikationsbereich enthält anfangs den Standardnamen der Konfigurationswarteschlange (`TVISION.CONFIGURATION.QUEUE`). Wenn `SLMCNFQ` durch den Sensor ausgeführt wird, sollte der gewünschte Name der Konfigurationswarteschlange in den übergebenen Kommunikationsbereich geschrieben werden. Anschließend wird die Steuerung an den Sensor zurückgegeben. Anschließend liest der Sensor den Kommunikationsbereich, um den richtigen Namen der Konfigurationswarteschlange abzurufen.

Hinweis: Durch das Installationsverfahren für den z/OS CICS-Sensor wird KEIN Programm mit dem Namen `SLMCNFQ` erstellt. Anweisungen zum Schreiben dieses Programms finden Sie weiter unten im Abschnitt "Schreiben von `SLMCNFQ`".

Wenn der Versuch, `SLMCNFQ` auszuführen, nicht erfolgreich ist, versucht der WebSphere MQ-Sensor für z/OS CICS, das Programm `SLMBCNFG` zu laden, und ruft den Namen der Konfigurationswarteschlange ab. Anweisungen zum Erzeugen dieses Programms finden Sie unter "Erzeugen von `SLMBCNFG`" auf Seite 294.

Wenn der Versuch, `SLMBCNFG` zu laden, nicht erfolgreich ist, liest der Sensor den Inhalt von `TVISION.CONFIGURATION.QUEUE`.

Hinweis: Informationen zur Verwendung eines anderen Konfigurationswarteschlangennamens für jede CICS-Region finden Sie unter "Verwenden verschiedener Konfigurationswarteschlangen für jede CICS-Region" auf Seite 295.

Schreiben von SLMCNFQ

Sie können das Programm SLMCNFQ in jeder Sprache schreiben, die von Ihrer CICS-Region unterstützt wird. Folgendes ist ein Codebeispiel in C, in dem der Name der Konfigurationswarteschlange auf MY.CONFIGURATION.QUEUE festgelegt wird:

```
#include <cmqc.h>
#include <string.h>
#include <stdlib.h>
int main(int argc, char * argv[])
{
    void *pCommArea = NULL;
    EXEC CICS ADDRESS COMMAREA(pCommArea) EIB(dfheiptr);
    if (pCommArea && (dfheiprt->eibcalen >= sizeof(MQCHAR48)))
    {
        memset(pCommArea, 0, sizeof(MQCHAR48));
        strcpy(pCommArea, "MY.CONFIGURATION.QUEUE");
    }
    EXEC CICS RETURN;
}
```

Erzeugen von SLMBCNFG

Die Verfahren zum Erzeugen von SLMBCNFG sind für z/OS Batch und IMS identisch.

- 1** Bearbeiten Sie **thlqual.SSLMPROC(SLMBCFGQ)** und nehmen Sie folgende Änderungen vor:
 - a** Ändern Sie den Wert der Anweisung SET CONFIGQ, sodass er dem Namen der gewünschten Konfigurationswarteschlange entspricht.
 - b** Ändern Sie die Anweisung SET THLQUAL, sodass das sie das übergeordnete Qualifikationsmerkmal der TransactionVision-Sensorbibliotheken angibt.

- c Wenn die von IBM bereitgestellte Prozedur HLASMCL auf Ihrem System nicht verfügbar ist, ändern Sie die JCL nach Bedarf, um den angegebenen Quellcode zu assemblieren und zu binden. Nehmen Sie keine Änderung am Quellcode vor.
- 2 Senden Sie `thlqual.SSLMPROC(SLMBCFGQ)` ab, damit die Änderung wirksam wird.

Verwenden verschiedener Konfigurationswarteschlangen für jede CICS-Region

Wenn Sie mehrere CICS-Regionen verwenden, kann es gewünscht sein, für jede CICS-Region einen anderen Konfigurationswarteschlangennamen zu verwenden, während gleichzeitig die Sensorbibliothek von allen CICS-Regionen gemeinsam genutzt wird. Führen Sie die folgenden Schritte aus, um dies zu erreichen:

- 1 Fügen Sie die Tabelle `SLMBCNFG` zur CSD-Definition hinzu, falls dies noch nicht erfolgt ist. Im Folgenden ist die Definition von `SLMBCNFG` aufgeführt:

```
DEFINE PROGRAM(SLMBCNFG) GROUP(BTITV)
  DESCRIPTION(TVISION SENSOR CONFIGQ TABLE)
  LANGUAGE(ASSEMBLER) RELOAD(NO) RESIDENT(YES) USAGE(NORMAL)
  USELPACOPY(NO) STATUS(ENABLED) DATALOCATION(ANY)
```

- 2 Erstellen Sie in `&TVISION.SSLMSAMP` ein Element mit dem Namen `CONFIGQ`. Das Element hat folgenden Inhalt, der ein Auszug aus dem bereitgestellten Beispiel `TVISION.SSLMPROC(SLMBCFGQ)` ist:

```
. *-----
. * SLMCONFIG - TVision configuration macro - PLEASE DO NOT CHANGE
. *-----
  MACRO
    SLMCONFIG &CONFIGQ=TVISION.CONFIGURATION.QUEUE
  SLMBCNFG  AMODE 31
  SLMBCNFG  RMODE ANY
  SLMBCNFG  SECT
  CONFIGQ  DC  CL48'&CONFIGQ'
  MEND
*
  SLMCONFIG CONFIGQ=&SYSPARM
  END
```

3 Nehmen Sie an der Start-JCL der CICS-Region folgende Aktualisierung vor:

Fügen Sie den Parameter CONFIGQ zur Anweisung PROC von DFHSTART hinzu. Beispiel:

```
//DFHSTART,PROC,START=AUTO,
// INDEX1='CICSTS22',
...
// SIP=0,
// CONFIGQ='TVISION.CONFIGURATION.QUEUE',
...
```

Fügen Sie einen weiteren Schritt vor dem eigentlichen CICS-Ausführungsschritt hinzu:

```
/*-----
// EXEC HLASMCL,
// PARM.C='NORLD,NOXREF,NORXREF,SYSPARM(&CONFIGQ)',
// PARM.L='MAP,REFR'
//C.SYSIN DD DISP=SHR,DSN=TVISION.SSLMSAMP(CONFIGQ)
//L.SYSLMOD DD
DSN=&&CONFIGQ(SLMBCNFG),DISP=(,PASS),SPACE=(TRK,(1,,1))
/*-----
```

Fügen Sie der DFHRPL-Verkettung vor der Bibliothek TVISION eine DD-Anweisung hinzu, die auf das zuvor erstellte temporäre PDS verweist. Beispiel:

```
...
// DD DISP=SHR,DSN=&&CONFIGQ
// DD DISP=SHR,DSN=TVISION.SSLMLOAD
...
```

4 Nachdem Sie diese Änderungen vorgenommen haben, können Sie jede CICS-Region mit einem anderen Konfigurationswarteschlangennamen starten, indem Sie einfach einen eindeutigen CONFIGQ-Parameter übergeben.. Beispiel:

```
S CICS.CICS1,START=COLD,...,CONFIGQ='TV.CONFIG.Q1'
S CICS.CICS2,START=COLD,...,CONFIGQ='TV.CONFIG.Q2'
```


Festlegen des Prüfintervalls für die Konfigurationswarteschlange

Standardmäßig überprüft der Sensor alle fünf Sekunden, ob die Konfigurationswarteschlange neue Konfigurationsnachrichten enthält. Auf UNIX-, Windows- und i5/OS-Plattformen können Sie jedoch ein anderes Prüfintervall für die Konfigurationswarteschlange angeben. Wenn Sie für einen Sensor ein anderes als das standardmäßig definierte Konfigurationswarteschlangen-Prüfintervall angeben möchten, müssen Sie die Umgebungsvariable `TVISION_CONFIG_CHECK_INTERVAL` auf das gewünschte Intervall (in Millisekunden) festlegen.

Konfigurieren des WebSphere MQ-Nachrichtensystemproviders

In TransactionVision werden Warteschlangen für die Kommunikation zwischen Analyzer und den Sensoren verwendet. Mindestens drei Warteschlangen sind erforderlich: die Konfigurationswarteschlange, die Ereigniswarteschlange und die Ausnahmewarteschlange. Die Standardnamen für diese Warteschlangen lauten `TVISION.CONFIGURATION.QUEUE`, `TVISION.EVENT.QUEUE` und `TVISION.EXCEPTION.QUEUE`. Sie müssen diese Warteschlangen im Nachrichtensystemprovider gemäß den Vorgaben des Anbieters konfigurieren. Informationen hierzu finden Sie unter "Communication Links" im Handbuch *Using TransactionVision*.

Sie können die Warteschlangen im Warteschlangen-Manager mithilfe des IBM WebSphere MQ-Dienstprogramms `runmqsc` oder mithilfe der grafischen Benutzeroberfläche von MQ Explorer erstellen, die unter Windows und Linux verfügbar ist. Wenn Sie **Client** als Verbindungstyp verwenden, müssen Sie im Warteschlangen-Manager außerdem einen Serververbindungskanal und einen Listener definieren. Ausführliche Informationen finden Sie in der Dokumentation des Anbieters.

Konfigurieren des Exit-Sensors für WebSphere MQ-APIs

Der Exit-Sensor für WebSphere MQ-APIs ist ein Exitprogramm, das alle MQI-Aufrufe untersucht, die in Bezug auf den zugeordneten Warteschlangen-Manager abgesetzt werden. Durch das Exitprogramm werden Funktionen registriert, die vor und nach einem MQI-Aufruf aufgerufen werden sollen. Es wird in den folgenden gemeinsam genutzten Objekten/DLLs implementiert:

- tvisionapiexit
- tvisionapiexit_r

Obwohl der Sensor in Bezug auf den Warteschlangen-Manager registriert wird, wird er tatsächlich innerhalb des Adressraums der Applikation geladen und ausgeführt, die die MQI-Aufrufe absetzt. Der Sensor wird beispielsweise im Adressraum des Programms **amqspu**t, nicht im Adressraum des Warteschlangen-Managers ausgeführt.

Mit dem Exit-Sensor für WebSphere MQ-APIs können Sie jede WebSphere MQ-Serverapplikation überwachen. Sie können Clientapplikationen indirekt überwachen, indem Sie die entsprechenden MQI-Aufrufe in den Kanal-Agents der Serververbindung (Listener) erfassen.

Der Exit-Sensor für WebSphere MQ-APIs unterscheidet sich folgendermaßen von der WebSphere MQ-Sensorbibliothek:

- Es ist nicht notwendig, FASTPATH_BINDING zu deaktivieren (weitere Informationen finden Sie im Kapitel "WebSphere MQ-Sensoren und FASTPATH_BINDING" auf Seite 306).
- Die Fertigstellungs- und der Ursachencodes für MQDISC-Aufrufe werden vom API-Exit-Sensor nicht erfasst und werden auf die festen Werte MQCC_OK bzw. MQRC_NONE festgelegt. Als Ereigniszeit für MQDISC-Ereignisse wird der Zeitpunkt vor Aufruf der Funktion MQDISC festgelegt.
- Der API-Exit-Sensor erfasst einige TransactionVision-interne Ereignisse, die von WebSphere MQ-Aufrufen (WMQ) erzeugt werden, die von Analyzer abgesetzt wurden, und außerdem interne WMQ-Ereignisse von Java-Sensoren, die eine Clientverbindung zum Listener verwenden.

Hinweis: Wenn der API-Exit-Sensor und die WebSphere MQ-Sensorbibliothek zur gleichen Zeit aktiv sind, protokolliert der API-Exit-Sensor eine Warnmeldung, bleibt jedoch inaktiv und registriert keine MQI-Exits. Die WebSphere MQ-Sensorbibliothek fährt mit der Verarbeitung von Ereignissen fort.

Konfigurieren des API-Exit-Sensors auf verteilten und i5/OS-Plattformen

Führen Sie die folgenden Schritte aus, um den Exit-Sensor für WebSphere MQ-APIs auf verteilten Plattformen oder unter i5/OS zu verwenden:

- 1** Binden Sie das für Ihre Umgebung geeignete gemeinsam genutzte Objekt bzw. die DLL für den Exit-Sensor für WebSphere MQ-APIs (nur verteilte Plattformen).
- 2** Fügen Sie die erforderlichen Zeilengruppen zu den Dateien **mqs.ini** und **qm.ini** hinzu.

Binden des Exit-Sensors für WebSphere MQ-APIs

Da TransactionVision sowohl die 32-Bit- als auch die 64-Bit-Version von WebSphere MQ unterstützt, müssen Sie das richtige gemeinsam genutzte Objekt bzw. die richtige DLL binden.

Verwenden Sie für WebSphere MQ V5.3 und V6.0 (32-Bit) die folgenden Befehle, um den Sensor zu binden:

```
In -s <TransactionVision_Installationsverzeichnis>/lib/tvisionapiexit /var/mqm/exits/tvisionapiexit
```

```
In -s <TransactionVision_Installationsverzeichnis>/lib/tvisionapiexit_r /var/mqm/exits/tvisionapiexit_r (für Solaris nicht erforderlich)
```

Verwenden Sie für WebSphere MQ V6.0 (64-Bit) die folgenden Befehle, um den Sensor zu binden:

```
In -s <TransactionVision_Installationsverzeichnis>/lib64/tvisionapiexit /var/mqm/exits64/tvisionapiexit
```

In `–s <TransactionVision_Installationsverzeichnis>/lib64/tvisionapiexit_r /var/mqm/exits64/tvisionapiexit_r` (für Solaris nicht erforderlich)

Stellen Sie sicher, dass die Datei **qm.ini** die folgende Zeilengruppe enthält:

ExitPath:

```
ExitsDefaultPath=/var/mqm/exits/  
ExitsDefaultPath64=/var/mqm/exits64
```

Beachten Sie Folgendes: Wenn die Parameterwerte für **ExitsDefaultPath** und/oder **ExitsDefaultPath64** der Zeilengruppe **ExitPath:** in der Datei **qm.ini** geändert werden, müssen Sie den Verzeichnisnamen **/var/mqm/exits** und/oder **/var/mqm/exits64**, der in den Bindungsbefehlen angegeben ist, entsprechend ändern.

Neue Zeilengruppen

Sie müssen den API-Exit-Sensor in neuen Zeilengruppen in der Datei **mqs.ini** definieren, die Definitionen für die gesamte WebSphere MQ-Umgebung enthält, und in der Datei **qm.ini**, die für einzelne Warteschlangen-Manager gilt. Die Datei **mqs.ini** befindet sich üblicherweise im Verzeichnis **/var/mqm**. Die Datei **qm.ini** befindet sich normalerweise im Verzeichnis **/var/mqm/qmgrs/<wsmgr_name>**. Eine Zeilengruppe besteht aus einer Abschnittsüberschrift, gefolgt von einem Doppelpunkt, auf den wiederum Zeilen mit Attribut/Wert-Paaren folgen, die durch das Zeichen "=" getrennt sind. Beachten Sie, dass in den Dateien **mqs.ini** oder **qm.ini** die gleichen Attribute verwendet werden können.

Fügen Sie die folgenden Zeilengruppen zu **mqs.ini** hinzu:

➤ **ApiExitCommon**

Die Attribute in dieser Zeilengruppe werden beim Start des Warteschlangen-Managers gelesen und dann von den in **qm.ini** definierten API-Exits überschrieben.

➤ **ApiExitTemplate**

Wenn ein Warteschlangen-Manager erstellt wird, werden die Attribute in dieser Zeilengruppe in die neu erstellte Datei **qm.ini** unter der Zeilengruppe **ApiExitLocal** kopiert.

Fügen Sie die folgende Zeilengruppen zu **mq.ini** hinzu:

► ApiExitLocal

Wenn der Warteschlangen-Manager gestartet wird, setzen die hier definierten API-Exits die in **mqs.ini** definierten Standardwerte außer Kraft.

Zeilengruppenattribute und -werte

Alle aufgeführten Zeilengruppen weisen die folgenden Attribute und Werte auf:

► Name=TransactionVisionWMQSensor

Der beschreibende Name des API-Exits, der im Feld **ExitInfoName** der Struktur MQAXP übergeben wird. Dieses Attribut sollte auf die Zeichenfolge **TransactionVisionWMQSensor** festgelegt werden.

► Function=TVisionEntryPoint

Der Name des Funktionseinstiegspunkts in das Modul, das den API-Exitcode enthält. Dieser Einstiegspunkt ist die Funktion MQ_INIT_EXIT. Dieses Attribut sollte auf die Zeichenfolge **TVisionEnryPoint** festgelegt werden.

► Module=tvisionapiexit

Das Modul, das den API-Exitcode enthält. Legen Sie dieses Attribut auf die Binärdatei des TransactionVision-Exit-Sensors für WebSphere MQ-APIs fest. Für Plattformen, die getrennte threadbasierte Bibliotheken unterstützen (AIX, HP-UX und Linux) wird dieses Attribut auf den Pfad für die nicht threadbasierte Version des Sensormoduls festgelegt. Die threadbasierte Version des WebSphere MQ-Applikationsstubs fügt implizit **_r** zum angegebenen Modulnamen hinzu, bevor das Modul geladen wird.

Wichtig: Geben Sie keinen Pfad an; der Modulpfad wird durch den Bindungsbefehl bestimmt, den Sie zum Binden des richtigen Moduls verwendet haben (Informationen hierzu finden Sie unter "Binden des Exit-Sensors für WebSphere MQ-APIs" auf Seite 299). Der Speicherort hängt davon ab, ob Sie die 32-Bit- oder 64-Bit-Bibliotheken für WebSphere MQ verwenden. Das 32-Bit-Modul befindet sich in `<TransactionVision_Installationsverzeichnis>/lib`, während sich das 64-Bit-Modul in `<TransactionVision_Installationsverzeichnis>/lib64` befindet.

► `Data=TVQ=warteschlangenname`

Legen Sie das Attribut TVQ auf den Objektnamen oder auf einen Teil des Objektnamens mit Platzhalterzeichen fest, um die Namen der Warteschlangenobjekte festzulegen, für die der Sensor WMQ-Ereignisse ignorieren soll. Wenn der Abschnitt **Data** nicht angegeben ist, werden Ereignisse für Objekte, die TVISION* entsprechen, vom Sensor ignoriert. Geben Sie eine leere Zeichenfolge für TVQ (`Data=TVQ=`) an, um dieses Feature vollständig zu deaktivieren.

Das Datenfeld kann maximal 24 Zeichen umfassen. Aus diesem Grund kann der Objektnamenwert für TVQ aus bis zu 20 Zeichen bestehen und kann das Platzhalterzeichen * am Anfang und/oder Ende der Zeichenfolge enthalten. Für das Attribut TVQ kann nur eine Warteschlangenzeichenfolge angegeben werden. Weitere Informationen finden Sie unter "Verwerfen von WebSphere MQ-Ereignissen in TransactionVision-Warteschlangen" auf Seite 305.

► `Sequence=Folgenummer`

Die Reihenfolge, in der das TransactionVision-Exit-Sensormodul für WebSphere MQ-APIs in Relation zu anderen API-Exits aufgerufen wird. Ein Exit mit einer niedrigen Folgenummer wird vor einem Exit mit einer höheren Folgenummer aufgerufen. Die Folgenummern von Exits müssen nicht fortlaufend sein. Die Reihenfolge 1, 2, 3 führt zu dem gleichen Ergebnis wie die Reihenfolge 7, 42, 1096. Wenn zwei Exits dieselbe Folgenummer aufweisen, legt der Warteschlangen-Manager fest, welcher Exit zuerst aufgerufen wird. Dieses Attribut ist ein numerischer Wert ohne Vorzeichen.

Folgendes ist ein Beispiel, um die Sensorkonfiguration über den Warteschlangen-Manager (**qm.ini**) zu veranschaulichen.

```
ApiExitLocal:
  Name=TransactionVisionWMQSensor
  Sequence=100
  Function=TVisionEntryPoint
  Module=tvisionapiexit
```

Konfigurieren des API-Exit-Sensors auf Windows-Plattformen

Zum Konfigurieren des Exit-Sensors für WebSphere MQ-APIs auf Windows-Plattformen verwenden Sie das Snap-In WebSphere MQ Services oder den Befehl **amqmdain**, um die Windows-Registrierung zu aktualisieren.

Auf der neuen Eigenschaftenseite **API Exits** für den Knoten **WebSphere MQ Services** werden zwei API-Exittypen beschrieben, die von diesem Knoten verwaltet werden: **ApiExitCommon** und **ApiExitTemplate**. Auf der Eigenschaftenseite **Exits** für einzelne Warteschlangen-Manager können Sie die Zeilengruppe **ApiExitLocal** aktualisieren. Mit den Schaltflächen **Configure...** starten Sie ein Dialogfeld, um die Einträge innerhalb der einzelnen Zeilengruppen zu verwalten. Das Dialogfeld besteht aus einer mehrspaltigen Liste aller API-Exits, die bereits in der geeigneten Zeilengruppe definiert sind, sowie aus Schaltflächen zum Hinzufügen und Anzeigen von Exits, zum Ändern der Exiteigenschaften und zum Entfernen von Exits. Eine Beschreibung der erforderlichen Zeilengruppen und Attributwerte finden Sie unter "Konfigurieren des API-Exit-Sensors auf verteilten und i5/OS-Plattformen" auf Seite 299.

Wenn Sie mit dem Definieren oder Ändern eines Exits fertig sind, klicken Sie auf OK, um die Registrierung zu aktualisieren, oder auf Cancel, um die Änderungen zu verwerfen.

Identifizieren der zu überwachenden Programme

Der Exit-Sensor für WebSphere MQ-APIs verwendet zwei Dateien, um die zu überwachenden Programme zu identifizieren:

- ▶ Mit **exit_sensor.allow** werden die zu überwachenden Programme definiert. Alle anderen Programme werden NICHT überwacht. Beachten Sie, dass keine Programme überwacht werden, falls diese Datei leer ist. Unter i5/OS trägt diese Datei den Namen ALLOW.MBR.
- ▶ Mit **exit_sensor.deny** werden die Programme definiert, die nicht überwacht werden. Alle anderen Programme werden überwacht. Unter i5/OS trägt diese Datei den Namen DENY.MBR. Diese Datei wird mit dem WebSphere MQ-Sensor bereitgestellt und enthält einige Standardprogramme, für die der API-Exit-Sensor keine Ereignisse erfasst, z. B. den WebSphere MQ-Befehlsserver.

Diese Dateien befinden sich auf der obersten Ebene des TransactionVision-Installationsverzeichnis. Wenn TransactionVision unter Solaris beispielsweise unter **/opt/HP/TransactionVision** installiert wird, befinden sich die beiden Dateien im Verzeichnis **/opt/HP/TransactionVision**. Unter i5/OS befinden sich diese Dateien in **/qsys.lib/tvsensor.lib/EXITSENSOR.FILE**.

In diesen Dateien beginnen Kommentarzeilen mit einem #-Zeichen. Sie können das Platzhalterzeichen * am Anfang und/oder Ende von Programmnamen verwenden.

Wenn sowohl **exit_sensor.allow** als auch **exit_sensor.deny** vorhanden sind, wird **exit_sensor.deny** vom Sensor ignoriert.

Die meisten WebSphere MQ-Befehle (Programme) werden zurückgewiesen, und der API-Exit wird für diese Programme nicht registriert. Weitere Programme können vom Benutzer zurückgewiesen werden, indem die entsprechenden Namen in **exit_sensor.deny** angegeben werden.

Hinweis: Wenn Sie den Exit-Sensor für WebSphere MQ-APIs verwenden und die Datei **exit_sensor.allow** vorhanden, aber leer ist, werden keine Programme überwacht.

Folgendes ist ein Beispiel für die Datei `exit_sensor.allow`, mit dem ausschließlich Ereignisse von WebSphere MQ-Listenern erfasst werden:

```
# File: exit_sensor.allow
# Description: Only collect from WebSphere MQ Listeners
amqcrsta
amqrmppa
runmqtsr
```

Folgendes ist ein Beispiel für die Datei `exit_sensor.deny`. Hiermit werden Ereignisse aus allen Programmen mit Ausnahme von Programmen erfasst, die mit **amq** beginnen:

```
# File: exit_sensor.deny
# Description: Collect any program except those that
# start with "amq"
amq*
```

Verwerfen von WebSphere MQ-Ereignissen in TransactionVision-Warteschlangen

Standardmäßig verwirft der Sensor jeglichen WebSphere MQ-Datenverkehr, der mit einem Warteschlangenobjekt mit dem Namenspräfix TVISION verbunden ist. Um einen anderen Namen für Warteschlangenobjekte anzugeben, müssen Sie im Attribut **Data** für TVQ die entsprechende Objektnamenszeichenfolge festlegen. Verwenden Sie, wie in den folgenden Beispielen, das Platzhalterzeichen *, um anzugeben, wo innerhalb des Objektnamens die angegebenen Zeichen auftreten können:

- **HP_TV***
HP_TV ist das Präfix für alle TransactionVision-Warteschlangenobjekte.
- ***HP_TV**
HP_TV ist das Suffix für alle TransactionVision-Warteschlangenobjekte.
- ***HP_TV***
Alle TransactionVision-Warteschlangenobjekte enthalten die Zeichenfolge HP_TV.

Hinweis: Platzhalter können vor und/oder nach dem TVQ-Wert, jedoch nicht innerhalb des Werts verwendet werden. So ist beispielsweise der Wert T*VISION ungültig.

Wenn Sie präziser steuern müssen, welche Warteschlangenobjekte verfolgt werden sollen, sollten Sie stattdessen einen Datenerfassungsfilter verwenden. Anweisungen zur Verwendung von Datenerfassungsfiltern finden Sie unter "Data Collection Filters" im Handbuch *Using TransactionVision*.

WebSphere MQ-Sensoren und FASTPATH_BINDING

Wenn die standardmäßigen WebSphere MQ-Sensorbibliothek auf verteilter Plattform verwendet wird und FASTPATH_BINDING für die überwachte Applikation festgelegt ist, wird die Applikation an denselben Adressraum wie der Warteschlangen-Manager gebunden, und es wird versucht, eine sekundäre DLL zu laden, die an die WebSphere MQ-Standardbibliothek gebunden ist. Da diese Umgebung so konfiguriert ist, dass die Sensorbibliothek anstelle von WebSphere MQ geladen wird, versucht die sekundäre DLL, interne Symbole aufzurufen, die nicht verfügbar sind.

Um dieses potenzielle Problem zu vermeiden, werden alle FASTPATH_BINDING-Einstellungen durch die Sensoren deaktiviert, indem die Umgebungsvariable MQ_CONNECT_TYPE auf den Wert STANDARD festgelegt wird, sobald die überwachte Applikation MQCONNX aufruft.

Verwenden von Sensoren mit WebSphere MQ-Beispielen

Wenn Sie Sensoren zum Überwachen von WebSphere MQ-Beispielapplikationen verwenden möchten, müssen Sie die folgenden Einschränkungen berücksichtigen:

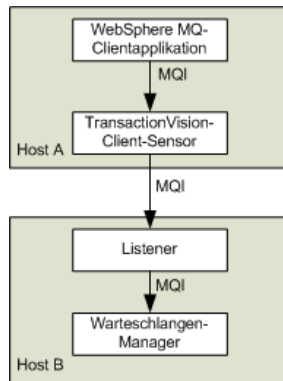
- ▶ Unter Windows gibt es zwei Speicherorte für WebSphere MQ-Beispiele. Wenn Sie die Beispiele unter WebSphere MQ\bin ausführen, müssen Sie die Beispielprogrammdateien (**amqspat**, **amqsget** usw.) in ein anderes Verzeichnis kopieren, um ihnen das Laden der Sensorbibliothek anstelle der WebSphere MQ-Standardbibliothek zu ermöglichen. Dies ist notwendig, da sich die WMQ-Bibliotheken in demselben Ordner befinden und Vorrang vor den Sensorbibliotheken haben, und zwar auch dann, wenn PATH ordnungsgemäß festgelegt wurde. Bei den Beispielen unter WebSphere MQ\TOOLS\c\samples\bin gibt es dieses Problem nicht.
- ▶ Auf HP-UX- und Linux-Plattformen weisen die Beispielprogrammdateien einen hartcodierten Pfad zur WebSphere MQ-Bibliothek auf, sodass die Sensorbibliothek nicht geladen wird.
- ▶ Wenn Sie das WebSphere MQ-Beispiel **amqsgbr** verwenden, dürfen Sie die Sensorereigniswarteschlange nicht als ersten Parameter verwenden.

Überwachen von WebSphere MQ-Clientapplikationen

Bei Applikationen, die WebSphere MQ-Clientbindungen verwenden, ist es TransactionVision möglich, die Nachrichtenaktivitäten dieser Applikationen im verteilten oder zentralisierten Modus zu überwachen und zu verfolgen.

Verteilte Überwachung

Das folgende Diagramm veranschaulicht die Funktionsweise von TransactionVision in verteilten Überwachungsumgebungen:



Im Allgemeinen kommunizieren Applikationen, die die WebSphere MQ-Clientbindung verwenden, mit einem "Listener"-Prozess (auch Kanalresponder genannt), der auf demselben Host wie der adressierte Warteschlangen-Manager ausgeführt wird. Alle WebSphere MQ-Aktivitäten (d. h. alle MQI-Aufrufe) werden an das Listenerprogramm weitergeleitet und von diesem verarbeitet. Das Listenerprogramm seinerseits setzt dann anstelle der Clientapplikationen die geeigneten MQI-Aufrufe an die entsprechenden Warteschlangen-Manager ab.

Im verteilten Überwachungsmodus wird eine Instanz des TransactionVision-Client-Sensors auf demselben Host installiert, auf dem die Clientapplikationen ausgeführt werden. Der Sensor fängt die von der Clientapplikation abgesetzten MQI-Aufrufe ab und überwacht sie, erzeugt entsprechende Ablaufverfolgungsinformationen und ruft die entsprechenden MQI-Einstiegspunkte in der regulären WebSphere MQ-Clientbindung auf.

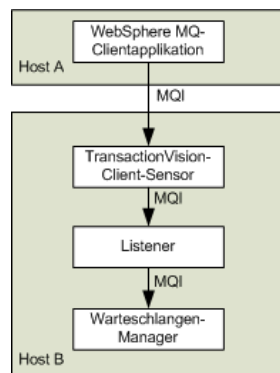
Die erzeugten Ablaufverfolgungsinformationen basieren auf dem Kontext der Clientapplikation. Das bedeutet, dass Informationen wie Programmname, Programminstanz und Hostname direkt mit der Clientapplikation in Beziehung stehen.

Für dieses Überwachungsschema ist es erforderlich, dass ein Client-Sensor auf jedem System installiert ist, auf dem WebSphere MQ-Clientapplikationen ausgeführt werden. Außerdem kann der Client-Sensor alle Applikationen überwachen, die die C-Laufzeitbindung des WebSphere MQ-Clients nutzen. Anders ausgedrückt: Der Client-Sensor unterstützt Applikationen, die in C und C++ geschrieben sind. Der WebSphere MQ-Java-Client setzt die C-Laufzeitbindung allerdings nicht ein. Dieser Ansatz ist daher nicht auf WebSphere MQ-Java-Clientapplikationen oder -applets übertragbar. (Beachten Sie, dass WebSphere MQ-Java-Serverapplikationen durch den TransactionVision-Server-Sensor für C unterstützt werden).

Dieser Ansatz wird für Clientapplikationen unter Windows, Solaris, HP-UX, AIX und Linux unterstützt.

Zentralisierte Überwachung

Die zentralisierte Überwachung des WebSphere MQ-Listenerprogramms wird nur bei Verwendung des API-Exit-Sensors unterstützt. Die Sensorbibliothek bietet diese Unterstützung nicht. Das folgende Diagramm veranschaulicht die Funktionsweise des Sensors in einer zentralisierten Überwachungs Umgebung:



In diesem Fall wird der Sensor auf dem Host bereitgestellt, auf dem sich der Listenerprozess und der Warteschlangen-Manager befinden. Anstelle der direkten Überwachung der Clientapplikation überwacht der Sensor das Listenerprogramm. Beachten Sie, dass der TransactionVision-Server-Sensor bereitgestellt wurde. Der Sensor fängt alle MQI-Aufrufe, die vom Listenerprogramm abgesetzt wurden, ab und meldet sie. Anders ausgedrückt: Das Listenerprogramm führt dieselben MQI-Aufrufe aus, die die Clientapplikation absetzt (mit einigen wenigen Ausnahmen, wie weiter unten genauer erläutert wird). Anhand der Aufzeichnungen zu MQI-Aufrufen des Listenerprogramms können sich Benutzer von TransactionVision einen guten Eindruck von den Nachrichtenaktivitäten verschaffen, die von den Clientapplikationen ausgehen.

Ein Vorteil dieses Ansatzes besteht darin, dass die Sensorbereitstellung zentral auf den Hostsystemen erfolgen kann, auf denen der Warteschlangen-Manager ausgeführt wird. Anders als beim verteilten Ansatz sind auf den Clientsystemen keine Client-Sensoren erforderlich. Hierdurch kann der Installations- und Verwaltungsaufwand in Umgebungen erheblich reduziert werden, in denen Clientapplikationen auf Tausenden von Systemen ausgeführt werden, die sich in unterschiedlichen physischen Einrichtungen befinden.

Weiterhin ist hervorzuheben, dass dieser Ansatz die Überwachung von WebSphere MQ-Java-Clientapplikationen/-applets unterstützen kann. Diese Form der Überwachung wird im verteilten Modus nicht unterstützt.

Wenn Sie sich für diese Form der Überwachung entscheiden, sollten Sie Folgendes bedenken:

- ▶ Da der Sensor das Listenerprogramm und nicht die Clientapplikation direkt überwacht, beziehen sich bestimmte gemeldete Kontextinformationen, z. B. der Programmname, die Programminstanz-IDs, der Hostname usw. auf das Listenerprogramm. Da jede Clientverbindung jedoch in einer bestimmten Thread- oder Prozessinstanz des Listenerprogramms verarbeitet wird, werden MQI-Aufrufe aus derselben Clientapplikation und derselben Verbindung unter derselben Listenerprogramminstanz aufgeführt.
- ▶ Das Listenerprogramm setzt die MQCONN- oder MQCONNX-Aufrufe in Clientverbindungsanforderungen nicht ab. Daher gibt es keine entsprechenden TransactionVision-Ablaufverfolgungsinformationen, die für derartige Verbindungsereignisse gemeldet werden.

- Das Listenerprogramm kann zusätzliche MQI-Aufrufe anstelle der Clientapplikation absetzen. Wenn beispielsweise eine neue Verbindung verarbeitet wird, wird die Aufruffolge MQOPEN-MQINQ-MQCLOSE abgesetzt, um Informationen zum Warteschlangen-Manager abzufragen. Darüber hinaus wird die Aufruffolge MQCMIT-MQBACK abgesetzt, wenn eine Clientanforderung zum Trennen der Verbindung verarbeitet wird.
- Zwischen den MQPUT/MQPUT1/MQGET-Aufrufen, die die Clientapplikationen und das Listenerprogramm absetzen, gibt es eine 1:1-Entsprechung. Die Nachrichtenaktivitäten des Listeners sollten daher die Nachrichtenaktivitäten des zugehörigen Clients exakt widerspiegeln.
- Wie zuvor erläutert beziehen sich die gemeldeten Kontextinformationen auf das Listenerprogramm. Von Clientapplikationen stammende Kontextinformationen können jedoch indirekt über die Nachrichtenheaderstruktur (MQMD) abgerufen werden, die über die Aufrufparameter in die Rückmeldedaten für MQPUT/MQPUT1/MQGET eingebettet wird.
- Wenn Sie die Servlet-, JMS- oder EJB-Sensoren für eine über einen Listener hergestellte Clientverbindung zu einem Warteschlangen-Manager verwenden und der Listener mit dem TransactionVision WebSphere MQ-Sensor überwacht wird, werden interne TransactionVision-Ereignisse erfasst. Es empfiehlt sich, entweder einen separaten nicht überwachten Listener für die Servlet-, JMS- oder EJB-Sensoren zu verwenden oder Serververbindungsverbindungen dieser Sensoren zu verwenden. Ist dies nicht möglich, sollten Sie den Datenerfassungsfilter so ändern, dass die Konfigurations- und Ereigniswarteschlangen ausgeschlossen werden.

In der folgenden Tabelle werden die Besonderheiten der beiden Ansätze zusammengefasst:

Verteilte Überwachung	Zentralisierte Überwachung
Direkte Client-MQI-Ablaufverfolgung	Indirekte Ablaufverfolgung über Listener-MQI-Aufrufe
Direkter Zugriff auf Clientkontextinformationen	Clientkontextinformationen über Aufrufparameter
Client-Sensor auf jedem Clienthost	Server-Sensor je Warteschlangen-Manager-Host

Verteilte Überwachung	Zentralisierte Überwachung
Überwacht Applikationen mittels C-Bindung von WebSphere MQ	Server-Sensor je Warteschlangen-Manager-Host
Unterstützt Clientapplikationen unter Windows, Solaris, HP-UX, AIX und Linux	Unterstützt Clientverbindungen zu Warteschlangen-Manager unter Windows, Solaris, HP-UX und AIX

Überlegungen zur Installation und Konfiguration

Für die verteilte Überwachung müssen Sie Client-Sensoren auf jedem Host installieren, auf dem WebSphere MQ-Clientapplikationen ausgeführt werden. Befolgen Sie die Standardanweisungen zur Sensorinstallation unter "Installieren und Konfigurieren des Java-Agents unter Windows" auf Seite 157.

Für die zentralisierte Überwachung müssen Sie Server-Sensoren auf jedem Host installieren, auf dem mindestens ein Listenerprogramm überwacht werden soll.

Hinweis: Die Mechanismen für die zentralisierte Überwachung und für die verteilte Überwachung schließen sich gegenseitig aus. Im Allgemeinen meldet der Server-Sensor, der das Listenerprogramm überwacht, die Aktivitäten aller zugeordneten Clients, einschließlich derjenigen Clients, die eventuell von Client-Sensoren auf dem Clienthost überwacht werden. Um redundante Meldungen zu vermeiden, sollten Sie bei Verwendung der zentralisierten Überwachung sicherstellen, dass auf jedem Host, auf dem Clients eine Verbindung zum Warteschlangen-Manager herstellen, dessen Listener überwacht wird, der Client-Sensor deaktiviert ist.

Verwenden des WebSphere MQ IMS Bridge-Sensors

WebSphere MQ IMS Bridge ist eine WebSphere MQ-Komponente, die es WebSphere MQ-Applikationen ermöglicht, IMS-Transaktionen aufzurufen und ihre Antwortnachrichten zu empfangen. Die Applikation setzt einen MQPUT-Aufruf an eine WebSphere MQ IMS Bridge-Eingabewarteschlange ab und verwendet hierfür eine Nachricht, die aus einem IMS-Transaktionscode gefolgt von Transaktionsdaten besteht. Sie empfängt die IMS-Ausgabenachricht, indem sie einen MQGET-Aufruf an die Empfangswarteschlange für Antworten absetzt, die im Nachrichtendeskriptor des MQPUT-Aufrufs angegeben war. Eine Änderung der IMS-Transaktion ist nicht notwendig, um diese Schnittstelle aufzunehmen.

Der TransactionVision WebSphere MQ IMS Bridge-Sensor überwacht WebSphere MQ IMS Bridge-Nachrichten anstelle der WebSphere MQ-API-Aufrufe, die von den aufrufenden Applikationen abgesetzt werden.

Sensorkonfiguration

Führen Sie die folgenden Konfigurationsaufgaben aus, bevor Sie den WebSphere MQ IMS Bridge-Sensor verwenden:

- 1 Passen Sie die Beispielstartprozedur TVISIONB in **thlqual.SSLMPROC** an und kopieren Sie sie in eine geeignete Prozedurbibliothek. Für TVISIONB sind vier Startparameter erforderlich, die in der Prozedur oder im Befehl START angegeben werden können.
 - Mit dem Parameter QMGR wird der Name des WebSphere MQ-Warteschlangen-Managers angegeben, mit dem TVISIONB eine Verbindung herstellen muss, um auf die zugehörigen Konfigurations- und Ereigniswarteschlangen zuzugreifen. Bei diesem Warteschlangen-Manager handelt es sich um den Warteschlangen-Manager, mit dem Analyzer eine Verbindung herstellt, wenn eine Kommunikationsverbindung mit dem Sensor aufgebaut wird, und nicht notwendiger um den oder die Warteschlangen-Manager, mit dem oder denen WebSphere MQ IMS Bridge verbunden ist. Es muss sich um denselben Warteschlangen-Manager handeln, der während der Installation beim Definieren der Konfigurations- und Ereigniswarteschlangen verwendet wurde (siehe den Beispieljob in **thlqual.SSLMINST(SLMCRTQS)**).

- ▶ Mit dem Parameter MAXQ wird der maximale Umfang des Speichers (in MB) angegeben, der von TVISIONB für die zugehörige Pufferwarteschlange zugeordnet wird. Weitere Informationen finden Sie unter "Die TVISIONB-Pufferwarteschlange" auf Seite 316.
 - ▶ Mit dem Parameter EDPROC wird der Name der Prozedur zum Starten des TVISIOND-Adressraums angegeben.
 - ▶ Mit dem Parameter IMSJOB wird der Jobname der IMS-Steuerregion für das zu überwachende IMS-System angegeben.
- 2** Schließen Sie die Bibliothek **thlqual.SSLMAUTH** in die Verkettung STEPLIB für jede IMS-Steuerregion ein, für die die WebSphere MQ IMS Bridge-Überwachung mittels TransactionVision erforderlich ist, oder kopieren Sie das Modul DFSYIOE0 in eine bestehende qualifizierende Bibliothek.

WebSphere MQ IMS Bridge-Sensorbetrieb

Führen Sie die folgenden Schritte aus, um den WebSphere MQ IMS Bridge-Sensor auszuführen:

- 1** Stellen Sie sicher, dass die IMS-Steuerregion mit der TransactionVision-spezifischen DFSYIOE0-Exitroutine gestartet wird, die über den zugehörigen STEPLIB-Wert zugänglich ist.
- 2** Starten Sie den TVISIONB-Adressraum über die Konsole des Systemoperators und geben Sie hierbei alle Parameter an, die in der Startprozedur außer Kraft gesetzt werden sollen. Beispiel:

```
S TVISIONB[.jobname],IMSJOB=IMS71CR1,QMGR=CSQ1, MAXQ=10
```

Wenn Sie mehrere Instanzen des Sensors ausführen, sollten Sie für jede Instanz einen eindeutigen Jobnamen angeben. Andernfalls wird für den Jobnamen standardmäßig der Prozedurname verwendet und alle MVS-Befehle des Typs **modify** und **stop** gelten für alle Instanzen. Sie können stattdessen separate, eindeutig benannte Startprozeduren für jedes zu überwachende IMS-System erstellen.

Wird IMSJOB nicht angegeben (indem z. B. der Wert auf **null** festgelegt wird), überwacht die gestartete Sensorinstanz jedes IMS-System, in dem die Exitroutine DFSYIOE0 ausgelöst wird und das nicht explizit durch eine andere Instanz des Sensors überwacht wird. Wenn ein IMS-systemspezifischer Sensor gestartet wird, während ein Sensor, der alle Systeme überwacht, ausgeführt wird, wird im Rahmen der Überwachung des ausgewählten IMS-Systems ein Wechsel zur konkreten Sensorinstanz vorgenommen. Umgekehrt gilt: Wenn ein spezifischer Sensor gestoppt wird, wird im Rahmen der Überwachung des gewünschten IMS-Systems ein Wechsel zum Sensor vorgenommen, der alle Systeme überwacht (sofern er ausgeführt wird). Um ein Durcheinander zu vermeiden, empfiehlt es sich, entweder nur spezifische Sensoren auszuführen oder ausschließlich einen Sensor auszuführen, der alle Systeme überwacht, und keine spezifischen Sensoren auszuführen. Es ist nur ein Sensor, der alle Systeme überwacht, zulässig, und für jedes individuelle IMS-System ist nur die Überwachung durch einen spezifischen Sensor zulässig.

TVISIOND wird automatisch durch TVISIONB gestartet.

- 3** Verwenden Sie auf einer verbundenen Arbeitsstation die UI/Job Server-Komponente von TransactionVision, um die Bridgeüberwachung anzufordern. Weitere Informationen finden Sie im Handbuch *TransactionVision User's Guide*.
- 4** Die Aktivitäten des Bridge-Sensors werden normalerweise über TransactionVision UI/Job Server gesteuert. Sie können den Sensor jedoch auch mit dem Befehl MODIFY über die Systemkonsole deaktivieren. F TVISIONB,DISABLE MQIMSBGD. Nach der Deaktivierung wird die TransactionVision-Exitroutine DFSYIOE0 weiterhin in der IMS-Steuerregion ausgeführt, sendet jedoch keine Ereignisse an die TVISIONB-Serverkomponente. Sie können den Sensor folgendermaßen reaktivieren: F TVISIONB,ENABLE MQIMSBGD.
- 5** Stoppen Sie den TVISIONB-Adressraum folgendermaßen: P TVISIONB. Hierdurch wird der Sensor implizit deaktiviert. Die Exitroutine wird weiterhin ausgeführt, versucht jedoch nicht mehr, Ereignisse an TVISIONB zu senden. TVISIOND wird automatisch gestoppt.

Vor Abschluss der Beendigung werden alle Ereignisse in der Pufferwarteschlange an die Ereignisverteilerkomponente gesendet. Um dieses Verhalten zu vermeiden, können Sie die sofortige Beendigung durch P TVISIONB IMMED anfordern. In diesem Fall werden alle Ereignisse in der Pufferwarteschlange verworfen.

Die TVISIONB-Pufferwarteschlange

Von der Sensorserverkomponente wird eine speicherinterne Warteschlange unterhalten, um Ereignisse zu puffern, die von der Exitroutine über TVISIONB an TVISIOND gesendet werden. Es ist wahrscheinlich, dass die Rate, mit der Ereignisse von der Exitroutine gesendet werden, um ein Mehrfaches höher liegt als die Rate, mit der Ereignisse durch TVISIOND verteilt werden. Die Warteschlange wird auf der Basis dieser Ereignisflüsse erweitert und geschrumpft. Die maximale Größe der Warteschlange kann ungeachtet des REGION-Werts gesteuert werden.

Geben Sie im Startbefehl für TVISIONB oder in der Startprozedur **MAXQ=nn** an. Hierbei entspricht **nn** der maximalen Warteschlangengröße (in MB). Die Mindestgröße beträgt 3 MB. Der maximal zulässige Wert ist 2046; hiermit wird es TVISIONB ermöglicht, den gesamten Adressraum von 2 GB zu verwenden.

Speicherblöcke werden von TVISIONB in 1-MB-Blöcken zugeordnet und freigegeben. Wenn ein weiterer Block erforderlich ist, dieser jedoch durch TVISIONB aufgrund der Einschränkung durch MAXQ oder der REGION-Größenbeschränkung nicht zugeordnet werden kann, wird eine Warnmeldung ausgegeben. Sobald der aktuelle Block vollständig belegt ist, werden alle neuen Ereignisse verworfen, bis es möglich ist, einen neuen Block zuzuordnen. Ereignisse, die bereits in der Warteschlange platziert wurden, werden weiterhin erfasst.

Sie müssen ein wenig herumexperimentieren, um den für Ihre Umgebung optimalen Wert für MAXQ zu definieren. Eine großzügige Definition, die sich als unnötig erweist, führt jedoch nicht zu zusätzlichen Kosten, da die Warteschlange ungeachtet der MAXQ-Einstellung bis auf ungefähr 3 MB schrumpft, wenn der überschüssige Platz nicht benötigt wird.

Ereignisdaten

Vom WebSphere MQ IMS Bridge-Sensor werden für jedes WebSphere MQ IMS Bridge-Ereignis die folgenden Ereignisdaten erfasst:

- Ein-/Ausgabekennzeichen
- Segmentfolgenindikator
- Transaktionscode
- IMS-Nachricht (oder Nachrichtensegment)
- Benutzer-ID
- XCF-Elementname (Cross-Systems Coupling Facility) des Warteschlangen-Managers
- Der Nachrichtendeskriptor (MQMD), der im MQPUT-Aufruf der Ursprungsapplikation angegeben wurde.

Damit der Sensor das Warteschlange-Manager- und das Warteschlangenobjekt zu den Ereignisdaten des WebSphere MQ IMS Bridge-Eintrittsereignisses hinzufügt, ist in Analyzer eine Ereignismodifikatorbean erforderlich. Die mit TransactionVision bereitgestellte Bean stellt einen einfachen Ansatz zur Verfügung. Sie definiert die WebSphere MQ-Warteschlangen-Manager- und Warteschlangenobjekte in getrennten XML-Konfigurationsdateien und definiert einen besonderen Ereignismodifikator, um die Definition zu lesen und in WebSphere MQ IMS Bridge-Eintrittsereignisse einzufügen. Die beiden folgenden Dateien im Verzeichnis `<TVISION_HOME>/config/services` werden verwendet, um einen Ereignismodifikator für das WebSphere MQ IMS Bridge-Eintrittsereignis einzurichten:

- Beans.xml
- IMSBridgeObject.xml

Beans.xml

Durch diese Datei wird das Gerüst für die Ereignisanalyse festgelegt, indem eine Abfolge von Verarbeitungsbeans definiert wird. Standardmäßig ist **com.bristol.tvision.services.analysis.event-modifier.IMSBridgeEntryModifier Bean** bereits unter **EventModifierCtx** definiert. Diese Bean liest eine Objektdefinition in **IMSBridgeObject.xml** und platziert die Definition (im Format eines XML-Dokumentfragments) im Ereignis-XML-Dokument, falls dieses Ereignis ein WebSphere MQ IMS Bridge-Eintrittsereignis ist.

```
<Module type="Context" name="EventModifierCtx">
<!--
  This bean read MQObject definition for IMS bridge
  entry event from $TVISION_HOME/config/service/
  IMSBridgeObject.xml
-->
<Module type="Bean" class="com.bristol.tvision.services.analysis.eventmodifier
.IMSBridgeEntryModifierBean"/>
</Module>
```

IMSBridgeObject.xml

Diese Datei definiert, wie im folgenden Beispiel gezeigt, die WebSphere MQ-Warteschlangenobjekte, die die WebSphere MQ IMS Bridge-Ereignisse erzeugen:

```
<?xml version="1.0" encoding="UTF-8"?>
<IMSBridgeMQObject>
  <MQObject objectName="IMS.BRIDGE.QUEUE" queueManager="MQS1"
  objectType="Q_LOCAL"/>
</IMSBridgeMQObject>
```

Attribut	Beschreibung
objectName	Definiert den Namen der WebSphere MQ-Warteschlange.
queueManager	Definiert den Namen des Warteschlangen-Managers.
objectType	Definiert den Typ der Warteschlange. Gültige Werte sind Q_LOCAL, Q_ALIAS, Q_REMOTE, Q_CLUSTER, Q_LOCAL_CLUSTER, Q_ALIAS_CLUSTER, Q_REMOTE_CLUSTER und DISTRIBUTION_LIST.

Beachten Sie, dass nur ein Element des Typs MQOBJECT unter dem Stammelement **IMSBridgeMQObject** definiert wird. Wenn mehrere **MQObject**-Elemente definiert werden, liest die Ereignismodifikatorbean lediglich die erste Definition.

Je nach Objekttyp kann die Struktur durch das XML-Dokument erweitert werden, um ausführlichere Informationen bereitzustellen. Mit folgendem XML-Code wird beispielsweise ein Remotewarteschlangenobjekt definiert:

```
?xml version="1.0" encoding="UTF-8"?>
<IMSBridgeMQObject>
  <MQObject objectName="REMOTE.BRIDGE.QUEUE" queueManager="MQS1"
objectType="Q_REMOTE">
  <MQObject objectName="IMS.BRIDGE.QUEUE" queueManager="MQS2"
objectType="Q_LOCAL">
< /MQObject>
</IMSBridgeMQObject>
```

Das XML-Schema befindet sich in **<TVISION_HOME>/config/xmlschema/IMSBridgeObj.xsd**.

Datenerfassungsfilter und Abfragen

Die Möglichkeit zum Filtern (entweder mit einem Datenerfassungsfilter oder einer Abfrage) wird für einige Ereignisattribute wie Benutzername, IMS-PSB-Name, IMS-Regionstyp, IMS-ID, Programm, Eintrittsereignis-Warteschlange und Warteschlangen-Manager oder Rückgabecode nicht bereitgestellt.

Zum Filtern anhand der WebSphere MQ IMS Bridge-Eintritts- oder Austrittsereignisse müssen Sie die entsprechende API entweder auf der Kriterienseite für WebSphere MQ-APIs (Abfragen) oder auf der Kriterienseite für MQ IMS Bridge-APIs (Datenerfassungsfilter) auswählen):

API	Beschreibung
MQIMS_BRIDGE_ENTRY	WebSphere MQ IMS Bridge-Eintrittsereignis
MQIMS_BRIDGE_EXIT	WebSphere MQ IMS Bridge-Austrittsereignis

Verwenden des WebSphere Business Integration-Sensors

In TransactionVision wird ein WebSphere Business Integration-Sensor (WBI) bereitgestellt, der es TransactionVision ermöglicht, die verschiedenen Nachrichtenflüsse zu unterscheiden und einzelne logische Transaktionspfade innerhalb von WBI zu identifizieren. Der WBI-Sensor ist ein WBI-Plugin, das Ablaufverfolgungsknoten (**TransactionVisionTrace**) unterstützt, die in normale Ausführungspfade eingefügt werden, sowie Fehlerknoten (**TransactionVisionFailure**), die in Fehlerpfade eingefügt werden.

In einen bestehenden Nachrichtenfluss können beliebig viele Verarbeitungsknoten an den gewünschten Punkten eingefügt werden. Jeder Verarbeitungsknoten stellt einen Prüfpunkt dar, der den Status des aktuellen Nachrichtenflusses erfasst und an Analyzer meldet. Das gemeldete Ereignis stellt Informationen wie den Brokernamen, den Namen des Nachrichtenflusses, Nachrichtendaten usw. bereit. Jedem Knoten kann eine eindeutige Bezeichnung hinzugefügt werden, die ebenfalls im TransactionVision-Ereignis gemeldet wird, das der Knoteninstanz zugeordnet ist.

Zum Installieren und Konfigurieren des WBI-Sensors müssen Sie folgendermaßen vorgehen:

- ▶ Einbinden des TransactionVision-Plugins in das Message Brokers Toolkit for WebSphere Studio.
- ▶ Installieren des TransactionVision-WBI-Sensors auf der WBIMB-Plattform.

Integration in das Message Brokers Toolkit for WebSphere Studio

Die folgenden Schritte basieren auf den Standardinstallationsverfahren für Plugins für das Message Brokers Toolkit und Eclipse-Technologien:

- 1 Stellen Sie sicher, dass das Message Brokers Toolkit nicht ausgeführt wird.
- 2 Entzippen Sie `sensor_installationsverzeichnis\mqsi\TransactionVisionWBIPlugin.zip` in das Verzeichnis `WBIMB_installationsverzeichnis\eclipse\plugins`.

Wenn das Message Brokers Toolkit gestartet wird, sind die TransactionVision-Ablaufverfolgungsknoten zusammen mit anderen integrierten Knoten sichtbar, wenn ein Nachrichtenfluss bearbeitet wird.

Installation benutzerdefinierter TransactionVision-Knoten für WBIMB

Führen Sie die folgenden Schritte aus, um den TransactionVision-WBI-Sensor auf der WBIMB-Plattform zu installieren:

- 1** Stoppen Sie den oder die WBI-Nachrichtenbroker.
- 2** Kopieren Sie die benutzerdefinierte Knotenbibliothek des WBI-Sensors in das entsprechende Unterverzeichnis des WBIMB-Installationsverzeichnisses.

Windows:

Kopieren Sie die Bibliothek

`<sensor_installationsverzeichnis>\mqsi\tvisiontrace.lil` in das Verzeichnis `<WBIMB_installationsverzeichnis>\bin`.

UNIX:

Kopieren Sie die Bibliothek in das Verzeichnis

`<WBIMB_installationsverzeichnis>/lil`.

- 3** Starten Sie den oder die WBI-Nachrichtenbroker erneut.

Einfügen von Knoten

Mithilfe des Message Brokers Toolkits können Sie nun in einen beliebigen Nachrichtenfluss eine beliebige Anzahl von Ablaufverfolgungs- und Fehlerknoten für den TransactionVision-Sensor einfügen. Bedenken Sie, dass alle Änderungen am Konfigurationsrepository in den geeigneten Brokern bereitgestellt werden müssen.

Informationen zur Verwendung des WBI-Sensors im Rahmen der TransactionVision-Datenerfassung und -analyse finden Sie im Handbuch *TransactionVision User's Guide*.

25

Konfigurieren des Proxy-Sensors

Dieses Kapitel umfasst die folgenden Themen:

- Informationen zum Proxy-Sensor auf Seite 323
- Applikationsanforderungen auf Seite 324
- Aktivieren des Proxy-Sensors auf Seite 324
- Konfigurieren der Proxydefinitionsdatei auf Seite 325
- Konfigurieren der Benutzeroberfläche auf Seite 328

Informationen zum Proxy-Sensor

Der TransactionVision-Proxy-Sensor ermöglicht es TransactionVision, auf allgemeiner Ebene eine Korrelation zwischen Geschäftstransaktion und Prozessen herzustellen, die nicht mithilfe von TransactionVision-Sensoren überwacht werden. Im Folgenden sind einige Beispiele für den geeigneten Einsatz des Proxy-Sensors aufgeführt:

- Transaktionen, in denen eine überwachte Applikation eine Anforderungsnachricht in einer Warteschlange platziert. Anschließend ruft eine Applikation, die auf einer nicht vom TransactionVision-Sensor unterstützten Plattform (z. B. Tandem) ausgeführt wird, die Nachricht ab, verarbeitet sie und platziert eine Antwort in einer Warteschlange, damit sie dort von der überwachten Applikation abgerufen wird.

- ▶ Transaktionen, in denen eine überwachte Applikation eine Anforderungsnachricht in einer Warteschlange platziert. Anschließend ruft eine Applikation am Standort eines Geschäftspartners (an dem TransactionVision nicht installiert ist) die Nachricht ab, verarbeitet sie und platziert eine Antwort in einer Warteschlange, damit sie dort von der überwachten Applikation abgerufen wird.

In diesen Szenarios, in denen einige nicht überwachte Applikationen an den Geschäftstransaktionen teilnehmen, wird es TransactionVision durch den Proxy-Sensor ermöglicht, in begrenztem Umfang Informationen zur gesamten Geschäftstransaktion bereitzustellen.

Anders als der TransactionVision-Sensor ist der Proxy-Sensor eine Java-Bean, die innerhalb von Analyzer ausgeführt wird. Sie erkennt Transaktionen, die an nicht überwachte Applikationen gesendet werden, und erstellt spezielle Proxy-Objekte, um die nicht überwachten Applikationen darzustellen, die an der Transaktion beteiligt sind.

Applikationsanforderungen

Damit der Proxy-Sensor Geschäftstransaktionen, an denen nicht überwachte Applikationen beteiligt sind, korrelieren kann, müssen die Applikationen die folgenden Anforderungen erfüllen:

- ▶ Die vom Sensor überwachte Applikation muss die Nachrichten-ID und Korrelations-IDs im MQMD führen.
- ▶ Die vom Sensor überwachte Applikation muss eine Empfangswarteschlange für Antworten in der Anforderung angeben.
- ▶ Die nicht überwachte Applikation muss für Antwortereignisse einen aussagekräftigen Programmnamen im MQMD bereitstellen.

Aktivieren des Proxy-Sensors

Der Proxy-Sensor wird durch den TransactionVision-Lizenzcode aktiviert.

Konfigurieren der Proxydefinitionsdatei

Analyzer erzeugt Proxy-Objekte, wenn WebSphere MQ-Ereignisse aus bestimmten Warteschlangen stammen und zu einem Anforderung/Antwort-Paar (MQPUT-MQGET) mit übereinstimmenden Nachrichten- und Korrelations-IDs gehören. Bei der Proxydefinitionsdatei handelt es sich um eine XML-Datei, in der die Attribute von Proxy-Objekten definiert werden. Sie befindet sich in `<TVISION_HOME>/config/services/ProxySensorDef.xml`.

Sie müssen ein **proxy**-Element für jede nicht überwachte Applikation definieren, die Sie in die TransactionVision-Analyse einbeziehen möchten.

Hinweis: Sobald Sie diese Datei bearbeitet haben, müssen Sie Analyzer neu starten, damit die Änderungen wirksam werden.

Im folgenden Beispiel wird ein **proxy**-Element für das Programm P2 definiert:

```
<ProxySensor>
  <Proxy matchMsgIdToCorrelId="true">
    <Request queue="Q1" queueManager="QM1"/>
    <Reply queue="Q2" queueManager="QM2" />
    <Retrieve queue="INPUT_LQ" queueManager="DWMQI1"/>
    <Program name="P2" path="/usr/local/bin/P2path" />
    <Host name="P2_host_name" os="SOLARIS" />
  </Proxy>
</ProxySensor>
```

Unterelemente

Geben Sie für jedes **proxy**-Element die folgenden Unterelemente an:

Element	Erforderlich?	Attribute	Beschreibung
Request	Ja	queue queueManager	Die WebSphere MQ-Warteschlange und der Warteschlangen-Manager, von der bzw. dem das Proxyprogramm die Ereignisse erhält.
Reply	Ja	queue queueManager	Die WebSphere MQ-Warteschlange und der Warteschlangen-Manager, in der bzw. dem das Proxyprogramm das Antwortereignis ablegt, das dieselbe Nachrichten-ID und Korrelations-ID wie das Anforderungsereignis aufweist.
Program	Ja	name path	Der Name und der Pfad des Proxyprogramms.
Host	Ja	name os	Der Name und das Betriebssystem des Hosts, auf dem das Proxyprogramm ausgeführt wird.
Retrieve	Nein	queue	Veranlasst, dass der Proxy-Sensor das angegebene Warteschlangenobjekt anhand seiner Definition überprüft und nicht das in <Reply> definierte Objekt. Dieses Element ermöglicht es, den Sensor mit loser Koppelung auszuführen.
z/OSBatch	Nein	jobID jobName stepName tcbAddr	Wenn es sich bei dem Proxyprogramm um einen z/OS Batch-Job handelt, geben Sie hiermit die Job-ID, den Jobnamen, den Schrittnamen und die TCB-Adresse an.

Element	Erforderlich?	Attribute	Beschreibung
z/OSCICS	Nein	regionName transactionID taskNum	Wenn es sich bei dem Proxyprogramm um einen z/OS CICS-Task handelt, geben Sie hiermit den Namen der Region, die Transaktions-ID und die Tasknummer an.
z/OSIMS	Nein	psbName transactionName regionID jobName imsID imsType	Wenn es sich bei dem Proxyprogramm um einen z/OS IMS-Job handelt, geben Sie hiermit den PSB-Namen, den Transaktionsnamen, die Regions-ID, den Jobnamen, die IMS-ID und den IMS-Typ an.

Optionale Attribute für das Element "proxy"

Zusätzlich zu den oben aufgeführten Unterelementen können Sie die folgenden optionalen Attribute für jedes **proxy**-Element angeben:

Attribut	Beschreibung
matchMsgIdToCorrelId	Veranlasst, dass der Proxy-Sensor die Nachrichten-ID aus dem MQPUT-Ereignis mit der Korrelations-ID aus dem MQGET-Ereignis abgleicht.
matchCorrelIdToMsgId	Veranlasst, dass der Proxy-Sensor die Korrelations-ID aus dem MQPUT-Ereignis mit der Nachrichten-ID aus dem MQGET-Ereignis abgleicht.
swapMsgCorrelID	Bei true tauscht TransactionVision die Nachrichten-ID und die Korrelations-ID für MQPUT-/MQPUT1-Ereignisse, wenn der Suchschlüssel erzeugt wird. Dieses Attribut kann nicht mit matchMsgIdToCorrelId oder matchCorrelIdToMsgId verwendet werden.

Konfigurieren der Benutzeroberfläche

Standardmäßig werden in der Ansicht **Component Topology Analysis** keine proxybezogenen Links im dynamischen Modus angezeigt. Legen Sie das Attribut **hasProxySensor** in der Datei **UI.properties** auf **true** fest, um den Proxyknoten in dieser Ansicht zu aktivieren. Weitere Informationen zum Ändern dieser Konfigurationsdatei finden Sie in Anhang B, "Konfigurationsdateien"

26

Konfigurieren der Agent- und Sensorprotokollierung

Dieses Kapitel umfasst die folgenden Themen:

- Protokolldateien auf Seite 329
- Umlaufprotokollierung auf Seite 330
- Ablaufprotokollierung auf Seite 332
- Konfigurieren getrennter Protokolldateien für mehrere Sensorinstanzen auf Seite 333
- Verwenden von Windows- und UNIX-Systemprotokollen auf Seite 334

Protokolldateien

Java-Agents

Standardmäßig werden Fehler- und Warnmeldungen von den TransactionVision-Java-Agents in dem Protokollverzeichnis protokolliert, in dem der Java-Agent installiert wurde. Dieses Verzeichnis wird in der Datei **Sensor.Logging.xml** gespeichert.

Sie können den Java-Sensoren die Bannerausgabe, sofern aktiviert, ermöglichen, indem Sie die Java-Systemeigenschaft **com.bristol.tvision.sensor.banner** auf **true** festlegen. Die Bannerausgabe erfolgt über die Standardausgabe.

WebSphere MQ-Sensoren

Standardmäßig werden Fehler-, Warn- und Ablaufverfolgungsnachrichten von den WebSphere MQ-Sensoren als Typ **local0** im UNIX-Systemprotokoll (**syslogd**), im Windows-Ereignisprotokoll, im z/OS-Operatorkonsolenprotokoll oder im i5/OS-Benutzerjobprotokoll protokolliert.

Auf UNIX-Plattformen können Sie den Protokollierungstyp angeben, indem Sie die Umgebungsvariable `TVISION_SYSLOG` auf einen der folgenden Werte festlegen: **user**, **local0**, **local1**, **local2**, **local3**, **local4**, **local5**, **local6** oder **local7**. Wenn `TVISION_SYSLOG` nicht festgelegt ist oder keiner der oben aufgeführten Werte festgelegt wurde, wird in TransactionVision der Wert **local0** verwendet. Die Zielprotokolldatei muss bereits vorhanden sein, damit von **syslogd** Protokolleinträge in die Datei geschrieben werden können. Setzen Sie sich ggf. mit Ihrem Systemadministrator in Verbindung, um den Systemprotokollierungstyp einzurichten.

Auf UNIX- und Windows-Plattformen legen Sie die Umgebungsvariable `TVISION_BANNER` fest und starten anschließend die Applikation. Daraufhin sollte ein Banner angezeigt werden, in dem gemeldet wird, dass der Sensor von der Applikation geladen wird. Heben Sie die Festlegung von `TVISION_BANNER` auf, um dieses Verhalten zu deaktivieren. Diese Umgebungsvariable kann auf jeden beliebigen Wert festgelegt werden. Unter Windows darf sie nicht auf eine leere Zeichenfolge festgelegt werden. Unter i5/OS wird von `TVISION_BANNER`, anders als unter UNIX, der Bibliothekspfad nicht angezeigt.

Umlaufprotokollierung

Die Java-Agents verwenden standardmäßig eine Form der Umlaufprotokollierung. Wenn die Protokolldatei die konfigurierte Maximalgröße erreicht, wird sie als Sicherungsdatei umbenannt, und es wird eine neue, leere Protokolldatei erstellt. Die maximale Protokollgröße beträgt standardmäßig 10 MB und es gibt eine Sicherungsprotokolldatei.

Wenn die Standardeinstellungen verwendet werden und eine Sensorprotokolldatei, z. B. **sensor.log**, die Größe von 10 MB erreicht, wird sie **sensor.log.1** umbenannt, und es wird eine neue Datei mit dem Namen **sensor.log** erstellt. Wenn Sie die Konfiguration ändern, sodass zwei Sicherungsdateien verwendet werden, werden die folgenden Ereignisse ausgelöst, sobald **sensor.log** eine Größe von 10 MB erreicht:

- Falls vorhanden wird **sensor.log.2** entfernt.
- **sensor.log.1** wird in **sensor.log.2** umbenannt.
- **sensor.log** wird in **sensor.log.1** umbenannt.
- Eine neue Datei mit dem Namen **sensor.log** wird erstellt.

Wenn Sie die Umlaufprotokollierung *nicht* verwenden möchten, können Sie die Konfiguration so ändern, dass die lineare Protokollierung verwendet wird. Hierbei wird nur eine einzige Protokolldatei erzeugt.

In den Dateien `<TVISION_HOME>/config/logging/*.Logging.xml` werden die Art der verwendeten Protokollierung, die maximale Größe der Protokolldatei und die Anzahl der Sicherungsprotokolldateien für jede Komponente angegeben. So gibt z. B. **Sensor.Logging.xml** die Konfiguration für die Servlet- und JMS-Sensoren an. Diese Datei enthält Einträge ähnlich den folgenden:

```
<appender class="tvision.org.apache.log4j.RollingFileAppender"
name="SENSOR_LOGFILE">
  <param name="File" value="c:/Program Files/HP/TransactionVision/logs/sensor.log"/>
  <param name="Append" value="true"/>
  <param name="MaxBackupindex" value="2"/>
  <param name="MaxFileSize" value="10MB"/>
  <layout class="tvision.org.apache.log4j. PatternLayout">
  <param name="ConversionPattern" value="%d [%t] %-5p %c %x - %m%n"/>
  </layout>
</appender>
```

Maximale Größe der Protokolldatei

Wenn Sie die maximale Größe der Protokolldatei ändern möchten, ändern Sie den Wert des Parameters **MaxFileSize**, sodass er der gewünschten Dateigröße entspricht. Die angegebenen Werte sollten mit "MB" oder "KB" enden, um zwischen Mega- und Kilobytes zu unterscheiden.

Maximale Anzahl der Sicherungsprotokolldateien

Wenn Sie die Anzahl der Sicherungsdateien ändern möchten, ändern Sie den Wert des Parameters **MaxBackupIndex**, sodass er der gewünschten Anzahl an Sicherungsdateien entspricht.

Wechsel von der Umlauf- zur linearen Protokollierung

Führen Sie folgende Schritte aus, um die lineare Protokollierung anstelle der Umlaufprotokollierung zu verwenden:

- 1 Ändern Sie im **appender class**-Wert **RollingFileAppender** in **FileAppender**. Ändern Sie z. B. die erste Zeile des vorher aufgeführten Beispiels, sodass sie folgender Zeile entspricht:

```
<appender class="tvision.org.apache.log4j.FileAppender"  
name="SENSOR_LOGFILE">
```

- 2 Entfernen Sie die Einträge für die Parameter **MaxBackupIndex** und **MaxFileSize**.

Ablaufprotokollierung

Die Ablaufprotokollierung stellt ausführliche Informationen zu den internen Vorgängen bereit. Diese Form der Protokollierung wird in erster Linie für die Problembeseitigung verwendet und sollte **nicht** in Produktionsumgebungen zum Einsatz kommen.

Sie können die Ablaufprotokollierung in TransactionVision-Sensoren aktivieren, um Probleme mit der Sensorkonfiguration zu debuggen. Ablaufverfolgungsinformationen für den WebSphere MQ-Sensor werden im UNIX-Systemprotokoll, im Windows-Ereignisprotokoll, im z/OS-Operatorkonsolenprotokoll oder im Jobprotokoll eines i5/OS-Benutzers protokolliert. Bei den Java-Agents werden die Ablaufverfolgungsnachrichten in das Ablaufprotokoll **log4j** des Agents gesendet.

Informationen zum Aktivieren oder Deaktivieren der Sensorablaufprotokollierung in der Kommunikationsverbindung finden Sie unter "Communication Links" im Handbuch *Using TransactionVision*.

Konfigurieren getrennter Protokolldateien für mehrere Sensorinstanzen

Wenn der Agent für mehrere Applikationsserver oder JVM-Prozesse auf demselben System aktiviert ist, muss der Agent so eingerichtet werden, dass die Protokollierung entweder im Windows- oder UNIX-Systemprotokoll oder in einer eigenen Protokolldatei für jeden Applikationsserver bzw. jede JVM erfolgt. Geschieht dies nicht, werden Protokolldateieinträge beschädigt oder überschrieben.

Führen Sie die folgenden Schritte aus, um für jede Sensorinstanz die Protokollierung in einer anderen Protokolldatei einzurichten:

- 1 Kopieren Sie die Datei `<TVISION_HOME>/config/sensor/Sensor.properties` unter einem anderen Namen in das Verzeichnis `<TVISION_HOME>/config/sensor`. Wenn Sie den Sensor beispielsweise für die beiden Server **ServerA** und **ServerB** konfigurieren, kopieren Sie **Sensor.properties** mit den Namen **Sensor_serverA.properties** und **Sensor_serverB.properties**.
- 2 Kopieren Sie die Datei `<TVISION_HOME>/config/logging/Sensor.logging.xml` unter einem anderen Namen in das Verzeichnis `<TVISION_HOME>/config/logging`. Kopieren Sie die Datei beispielsweise für die beiden Server aus dem vorherigen Beispiel (**ServerA** und **ServerB**) mit dem Namen **Sensor.Logging.serverA.xml** und **Sensor.Logging.serverB.xml**.
- 3 Bearbeiten Sie die Eigenschaft **logging_xml** in den einzelnen **Sensor.properties**-Dateien. Ändern Sie beispielsweise in **Sensor_serverA.properties** die Eigenschaftenzeilen in **logging_xml=Sensor.Logging.serverA.xml**. Ändern Sie gleichermaßen in **Sensor_serverB.properties** die Eigenschaftenzeilen in **logging_xml=Sensor.Logging.serverB.xml**.
- 4 Legen Sie die JVM-Eigenschaft **com.bristol.tvision.sensor.properties** auf die geeignete **Sensor.property**-Datei fest. Legen Sie die JVM-Eigenschaft für **ServerA** beispielsweise folgendermaßen fest:

```
com.bristol.tvision.sensor.properties=sensor/Sensor_serverA.properties
```

Legen Sie die JVM-Eigenschaft für **ServerB** folgendermaßen fest:

```
com.bristol.tvision.sensor.properties=sensor/Sensor_serverB.properties
```

Für eigenständige Programme wird diese JVM-Eigenschaft beim Aufrufen von JVM folgendermaßen über die Befehlszeile festgelegt:

```
java -Dcom.bristol.tvision.sensor.properties=sensor/Sensor_serverA.properties
```

Für WebSphere legen Sie diese JVM-Eigenschaft über die Verwaltungskonsole für den jeweiligen Applikationsserver fest. Wählen Sie hierzu **Servers > Application Servers > Process Definition > Java Virtual Machine > Custom Properties** aus.

Für WebLogic legen Sie diese JVM-Eigenschaft mithilfe des WebLogic-Startskripts fest. Öffnen Sie einen beliebigen Texteditor, um das Skript zu bearbeiten, beispielsweise **startWebLogic.cmd**. Legen Sie die Umgebungsvariable **JAVA_OPTIONS** folgendermaßen fest, sodass sie **com.bristol.tvision.sensor.properties** einschließt:

```
SET JAVA_OPTIONS=%JAVA_OPTIONS% -Dcom.bristol.tvision.  
sensor.properties=<TVISION_HOME>/config/sensor/  
<benutzerdef_eigenschaften_datei>
```

Verwenden von Windows- und UNIX-Systemprotokollen

Auf UNIX- und Windows-Plattformen können Sie TransactionVision so konfigurieren, dass die Ausgabe im Protokollierungsmechanismus für Systemereignisse – das Ereignisprotokoll unter Windows oder **syslog** unter UNIX – aufgezeichnet wird. Beispiele für die Protokollierungskonfigurationsdateien, die hierfür erforderlich sind, finden Sie in **TVISION_HOME/config/logging/system/*/Sensor.Logging.xml**.

Sowohl für Windows als auch für UNIX müssen Sie einen speziellen Ereignisappender definieren.

Windows-Ereignisappender

Im folgenden Beispiel wird gezeigt, wie der Windows-Ereignisappender so konfiguriert wird, dass das Ereignisprotokoll verwendet wird:

```
<appender name="NT_EVENT_LOG" class="tvision.org.apache.log4j.nt
.NTEventLogAppender">
  <layout class="tvision.org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="%d [%t] - %m%n"/>
  </layout>
</appender>
```

Anschließend können Sie in einer Kategoriedefinition Ihrer Wahl auf NT_EVENT_LOG verweisen. Beispiel:

```
<category additivity="false" class="com.bristol.tvision.util.log.XCategory"
name="sensorLog">
  <priority class="com.bristol.tvision.util.log.XPriority" value="info"/>
  appender-ref ref="NT_EVENT_LOG"/>
</category>
```

Unter Windows müssen Sie außerdem eine spezielle DLL zum Pfad hinzufügen. Diese DLL, **NTEventLogAppender.dll**, befindet sich im Verzeichnis **config\logging\system\bin**. Beispiel:

```
set path=%TVISION_HOME%\config\logging\system\bin;%PATH%
```

UNIX-Ereignisappender

Das folgende Beispiel zeigt, wie ein UNIX-Ereignisappender so konfiguriert wird, dass **syslog** verwendet wird:

```
<appender name="SYSLOG" class="tvision.org.apache.log4j.net.SyslogAppender">
  <param name="SyslogHost" value="localhost"/>
  <param name="Facility" value="local0"/>
  <layout class="tvision.org.apache.log4j.PatternLayout">
    <param name="ConversionPattern" value="[%t] %-5p %c %x
    - %m%n"/>
  </layout>
</appender>
```

Geben Sie für die Parameter **SyslogHost** und **Facility** die für Ihre Umgebung geeigneten Werte an.

Teil V

Sicherheit

27

Sicherheit

In diesem Kapitel werden die Verfahren beschrieben, die Sie ausführen sollten, um Sicherheitseinstellungen in TransactionVision zu konfigurieren.

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Informationen zur Sicherheit in TransactionVision auf Seite 339
- ▶ SSL-Konfiguration für TransactionVision auf Seite 340
- ▶ Verwaltung von TransactionVision-Benutzerrechten auf Seite 342
- ▶ Standardrollen auf Seite 346
- ▶ TransactionVision-Authentifizierung und Lightweight Single Sign-On auf Seite 347
- ▶ Schützen der TransactionVision-Konfigurationsdateien auf Seite 348
- ▶ Schützen von TransactionVision Analyzer auf Seite 351
- ▶ Schützen der TransactionVision-Datenbank auf Seite 353

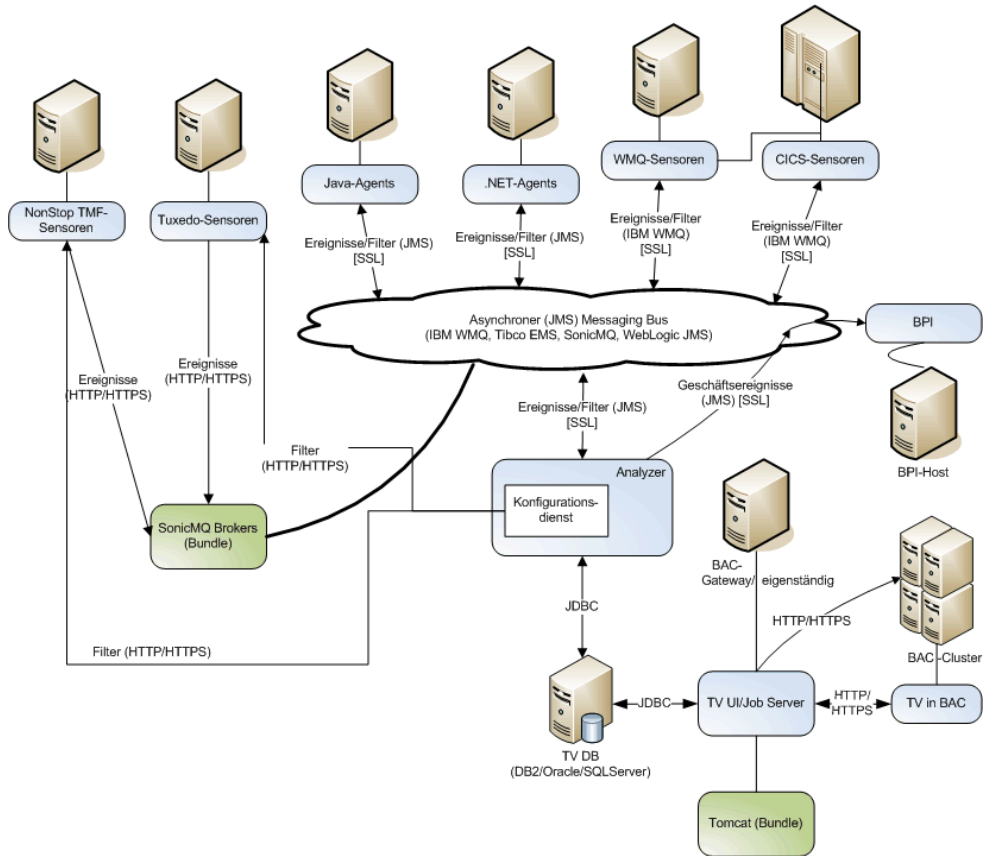
Informationen zur Sicherheit in TransactionVision

Sicherheitseinstellungen in TransactionVision werden über die folgenden Aufgaben verwaltet:

- ▶ Aktivieren von SSL zwischen TransactionVision-Komponenten.
- ▶ Steuern von Benutzerrechten.
- ▶ Schützen von Analyzer, Konfigurationsdateien und Datenbank.

SSL-Konfiguration für TransactionVision

Das folgende Diagramm zeigt die Pfade in der TransactionVision-Bereitstellungsumgebung, für die sich die Verwendung von SSL anbietet.



Das allgemeine Verfahren zum Aktivieren von SSL wird im Folgenden zusammengefasst.

Schritt 1: Erzeugen Sie die erforderlichen Zertifikate zum Verschlüsseln des Datenverkehrs

Für jede der Komponenten TransactionVision Analyzer, UI/Job Server und SonicMQ Server ist je ein privater Schlüssel erforderlich, der als Ergebnis der Erzeugung der erforderlichen Zertifikate erstellt wird. Wenn sich mehrere dieser Komponenten auf demselben Host befinden, können sie denselben privaten Schlüssel gemeinsam nutzen.

Schritt 2: Konfigurieren Sie TransactionVision SonicMQ Server

TransactionVision Analyzer wird im Paket mit Progress SonicMQ 7.5 zur Verfügung gestellt. Letzteres wird während der Analyzer-Installation installiert.

Konfigurieren Sie TransactionVision SonicMQ Server, sodass eines der in Schritt 1 erzeugten Zertifikate verwendet wird.

Schritt 3: Konfigurieren Sie Analyzer

- a** Wenn der HTTPS-Konfigurationsnachrichtendienst verwendet wird (der zusammen mit NonStop TMF-Sensoren eingesetzt wird), konfigurieren Sie das in Schritt 1 erzeugte Zertifikat.
- b** Für die Kommunikation mit TransactionVision SonicMQ Server konfigurieren Sie Analyzer so, dass das in Schritt 2 eingerichtete Zertifikat verwendet wird.

Schritt 4: Konfigurieren Sie UI/Job Server

- a** Zur Unterstützung eingehender SSL-Anforderungen konfigurieren Sie TransactionVision UI/Job Server mit dem in Schritt 1 erzeugten Zertifikat.
- b** Zur Unterstützung ausgehender SSL-Verbindungen zu Business Availability Center konfigurieren Sie TransactionVision UI/Job Server so, dass ein öffentliches Business Availability Center-Zertifikat verwendet wird, oder legen Sie fest, dass HTTPS für die Kommunikation mit Business Availability Center verwendet wird.

Schritt 5: Konfigurieren Sie die Sensoren/Agents

- a** Für die direkte Kommunikation mit TransactionVision SonicMQ Server konfigurieren Sie die Sensoren/Agents so, dass das in Schritt 2 eingerichtete Zertifikat verwendet wird.

- b** Wenn der HTTPS-Konfigurationsnachrichtendienst verwendet wird (für NonStop TMF-Sensoren), konfigurieren Sie den Sensor so, dass das in Schritt 3a erzeugte Zertifikat verwendet wird.

Ausführliche Informationen zu den einzelnen Schritten finden Sie im Handbuch *HP Business Availability Center Hardening Guide* (PDF).

Verwaltung von TransactionVision-Benutzerrechten

Mithilfe von Benutzerrechten wird gesteuert, welche TransactionVision-Seiten ein bestimmter Benutzer anzeigen kann und ob dieser Benutzer administrative Änderungen vornehmen oder bestimmte Daten anzeigen darf. Die Benutzerrechte in einer bestimmten TransactionVision-Sitzung werden anhand der Anmelde- und Authentifizierungsinformationen des Benutzers in Business Availability Center bestimmt. Weitere Informationen finden Sie im Handbuch *HP Business Availability Center Hardening Guide* (PDF).

Festlegen von Benutzerrechten

Das Festlegen von Benutzer- oder Gruppenberechtigungen für TransactionVision-Ressourcen erfolgt durch Auswahl des TransactionVision-Kontextes auf der Business Availability Center-Registerkarte **Users and Permissions**. Benutzer- oder Gruppenberechtigungen für Ressourcen können festgelegt werden, indem eine vordefinierte Rolle zugewiesen oder individuelle Vorgänge für Ressourcen zugelassen werden. Durch das Erteilen von Berechtigungen für den Ressourcencontainer **Projects** oder **User-created Reports** werden den untergeordneten Ressourceninstanzen und allen nachfolgenden Instanzen, die automatisch hinzugefügt werden, die gleichen Berechtigungen erteilt.

Standardmäßig werden Änderungen an TransactionVision-Berechtigungen wirksam, wenn sich der betreffende Benutzer das nächste Mal anmeldet. Damit Berechtigungen bereits während der aktuellen Anmeldesitzung eines Benutzers in Kraft treten, muss die TransactionVision-Eigenschaft **PermissionsCacheMaxAge** in `<TVISION_HOME>/config/ui/UI.properties` aktualisiert und auf die Anzahl der Minuten festgelegt werden, die verstreichen sollen, bis TransactionVision Berechtigungsaktualisierungen von Business Availability Center abrufen.

Hinweis: Das Festlegen dieses Wertes bewirkt, dass TransactionVision die Benutzerberechtigungen erneut bei Business Availability Center abfragt. Wenn TransactionVision UI/Job Server auf einem separaten Computer installiert ist (insbesondere, wenn sich diese Komponente in einem anderen Netzwerk als der Business Availability Center-Server befindet), kann es zu Verzögerungen beim Antwortverhalten der TransactionVision-Applikation in Business Availability Center kommen, falls der Wert so festgelegt ist, dass die Berechtigungen zu oft neu geladen werden.

Innerhalb des TransactionVision-Kontextes sind die folgenden Ressourcen verfügbar:

Hinweis: Projects und User-created Reports sind Container, die dynamisch auf der Grundlage der in TransactionVision verfügbaren Projekte und/oder benutzererstellten Reports aufgefüllt werden. Untergeordnete Projektressourceninstanzen werden der TAS-Struktur hinzugefügt oder aus ihr gelöscht, wenn neue Projekte mit dem Projekt-Assistenten hinzugefügt bzw. gelöscht werden. Der im Projekt-Assistenten angegebene Projektname wird in der TAS-Struktur angezeigt und dem Ersteller des Projekts wird die Berechtigung FULLCONTROL (Vollzugriff) für die neue Projektressourceninstanz erteilt.

Untergeordnete Ressourceinstanzen für benutzererstellte Reports werden der TAS-Struktur hinzugefügt oder aus ihr gelöscht, sobald die Datei <TVISION_HOME>/config/ui/Reports.xml mit benutzerdefinierten Reportdefinitionen aktualisiert wird. In der TAS-Struktur wird das Titelattribut des Reportelements angezeigt.

Name	Vorgang (Berechtigungen)
Projects	<p>Wenn kein Vorgang ausgewählt wurde, ist der Benutzerzugriff auf das Projekt nicht möglich.</p> <p>VIEW – Der Benutzer darf das Projekt nur anzeigen. Hierdurch erhält er auch Zugriff auf alle integrierten Reports und darf Abfragen anzeigen und ändern.</p> <p>CHANGE – Der Benutzer darf Projekte bearbeiten, löschen und erstellen und Jobs für ein Projekt starten und stoppen.</p> <p>FULLCONTROL – Der Benutzer darf alle verfügbaren Vorgänge ausführen und kann Vorgänge zulassen oder sperren.</p>
User-created Reports	<p>VIEW – Der Benutzer darf den Report anzeigen.</p> <p>FULLCONTROL – Der Benutzer darf alle verfügbaren Vorgänge ausführen und kann Vorgänge zulassen oder sperren.</p>
Analyzer	<p>EXECUTE – Der Benutzer darf Analyzer starten und stoppen.</p> <p>FULLCONTROL – Der Benutzer darf alle verfügbaren Vorgänge ausführen und kann Vorgänge zulassen oder sperren.</p>
User Data	<p>VIEW – Der Benutzer darf Benutzerdaten anzeigen.</p> <p>FULLCONTROL – Der Benutzer darf alle verfügbaren Vorgänge ausführen und kann Vorgänge zulassen oder sperren.</p>

Berechtigungen für Benutzerdaten

Einige Transaktionen und Ereignisinhalte, die von TransactionVision erfasst werden, können Daten enthalten, die innerhalb des Unternehmens vertraulich behandelt werden müssen. Hierzu können beispielsweise Daten wie Kreditkartennummern, Sozialversicherungsnummern usw. gehören. Falls sich herausstellt, dass TransactionVision sensible Daten erfasst, die nur bestimmten Personen zugänglich sein dürfen, können Sie den Zugriff auf diese Daten mit der Ressource **User Data** steuern. Indem Sie den Zugriff auf Benutzerdaten deaktivieren, werden die in TransactionVision angezeigten Informationen auf die allgemeinen Datenfelder begrenzt, die für alle Ereignis- und/oder Transaktionsdaten verfügbar sind.

Falls die Anzeigeberechtigung (VIEW) für **User Data** verweigert wird, hat dies folgende Auswirkungen:

- ▶ Im Fenster **Event Details** für ein Ereignis wird lediglich die folgende Meldung angezeigt: "You do not have permission to view user data."
- ▶ Im SQL-Statistikreport wird nicht die vollständige SQL-Anweisung angezeigt.
- ▶ Im Transaktionsverfolgungsreport werden keine benutzerdefinierten Spalten angezeigt.
- ▶ In **Detail for Business Transaction** werden keine benutzerdefinierten Daten im Abschnitt **Summary** angezeigt.
- ▶ Auf der Seite **Transaction Definition Editor Edit Match Condition** ist die Auswahl einer Transaktion nicht zulässig und der Wertebereich wird nicht mit Transaktionswerten aufgefüllt, falls für die als vertraulich eingestufteten Daten zugeordnete Transaktionsinformationen vorhanden sind.

Hinweis: Die Rolle **Business Availability Center Administrator** verfügt über das Recht, alle TransactionVision-Berechtigungen zu erteilen und zurückzuziehen; dem Business Availability Center-Administrator wird somit faktisch der vollständige Zugriff auf TransactionVision gewährt. Wenn es notwendig ist, den Zugriff auf **User Data** noch mehr einzuschränken, ein Benutzer jedoch weiter in der Lage sein muss, auf die TransactionVision-Registerkarte **Administration** zuzugreifen, können Sie diesem Benutzer die Rolle **TransactionVision Administrator** anstelle der Rolle **Business Availability Center Administrator** zuweisen.

Standardrollen

Für den Zugriff auf TransactionVision sind eine Reihe von Standardrollen definiert. Diese Rollen werden in der folgenden Tabelle beschrieben.

Rolle	Vorgänge	Zugelassene Registerkarten in TransactionVision
TransactionVision Administrator	FULLCONTROL für Projects, Analyzer und User-created Reports	Current Projects, Administration
TransactionVision Operator	VIEW für Projects und User-created Reports EXECUTE für Analyzer	Current Projects
TransactionVision User	VIEW für Projects und User-created Reports	Current Projects

Darüber hinaus wurden die vordefinierten Business Availability Center-Rollen **Administrator** und **System Viewer** aktualisiert, sodass sie nun dieselben Vorgänge einschließen, die für die Rollen **TransactionVision Administrator** und **TransactionVision User** definiert sind. **TransactionVision Administrator** und **Business Availability Center Administrator** sind die einzigen Rollen, die auf die Registerkarte **Administration** der TransactionVision-Applikation von Business Availability Center zugreifen können.

Um die Gefahr zu minimieren, dass der Zugriff auf die potenziell sensiblen Daten in **User Data** versehentlich gewährt wird, wurde nur dem Business Availability Center-Administrator der Zugriff auf **User Data** gewährt (Berechtigung FULLCONTROL).

TransactionVision-Authentifizierung und Lightweight Single Sign-On

Damit der sichere Zugriff auf TransactionVision sichergestellt ist, muss Lightweight Single Sign-on (LWSSO) aktiviert werden. Die von LWSSO verwendeten Sicherheitstoken werden zwischen TransactionVision und Business Availability Center übergeben und stellen sicher, dass nur ordnungsgemäß authentifizierte Benutzer auf die Seiten zugreifen, für die ihnen der Zugriff gewährt wurde. Wenn Sie beabsichtigen, LWSSO in Business Availability Center zu verwenden, ist keine weitere Konfiguration von TransactionVision erforderlich, da sämtliche Einstellungen bereits vordefiniert sind.

Damit LWSSO verwendet werden kann, muss der Zugriff auf die TransactionVision- und Business Availability Center-Server mit einem vollqualifizierten Domänennamen erfolgen. In einigen Bereitstellungen ist die Verwendung eines vollständig qualifizierten Hostnamens für den Zugriff auf Business Availability Center und/oder TransactionVision möglicherweise nicht gewünscht. In diesen Fällen ist LWSSO in Business Availability Center deaktiviert. Bedenken Sie, dass es im Hinblick auf die Sicherheit in TransactionVision notwendig ist, LWSSO und vollständig qualifizierte Domänennamen zu verwenden, um die ordnungsgemäße Benutzerauthentifizierung zu ermöglichen und die Benutzerautorisierung bestimmen zu können.

Wenn Sie LWSSO aus bestimmten Gründen deaktivieren möchten, obwohl TransactionVision weiterhin ausgeführt wird, werden die an TransactionVision UI/Job Server übergebenen Anmeldeinformationen mit einem nicht sicheren Verfahren übermittelt. Das Deaktivieren von LWSSO führt somit zu einer Schwachstelle, die es möglich machen kann, die Authentifizierung zu umgehen. Diese Vorgehensweise wird daher nicht empfohlen, wenn Sie sensible Daten verwenden oder sicherstellen müssen, dass der Zugriff auf TransactionVision-Daten und die TransactionVision-Verwaltung nur ordnungsgemäß autorisierten Benutzern möglich ist. Wenn Sie LWSSO in Business Availability Center deaktivieren, müssen Sie die Einstellung **bac_lwssso_enabled** in `<TVHOME>/config/ui/UI.properties` ändern und auf **false** festlegen. Solange **bac_lwssso_enabled** auf **false** festgelegt ist, unterstützt TransactionVision UI/Job Server eine weniger sichere Benutzerauthentifizierung.

Schützen der TransactionVision-Konfigurationsdateien

Alle TransactionVision-Konfigurationsdateien befinden sich im Verzeichnis `<TVISION_HOME>/config`. Wenn das Skript `<TVISION_HOME>/bin/TVisionSetupInfo.[sh|bat]` ausgeführt wird, wird der Pfad der Protokollverzeichnisse angegeben, in denen die TransactionVision-Protokolldateien gespeichert werden.

Der TransactionVision-Administrator, der als einziger über Schreibzugriff für diese Dateien verfügen sollte, sollte Besitzer aller TransactionVision-Konfigurationsdateien sein. Allgemeiner ausgedrückt: Allen relevanten TransactionVision-Gruppen sollte nur Leseberechtigung erteilt werden.

In der folgenden Tabelle sind die empfohlenen Berechtigungen für TransactionVision-Konfigurationsdateien aufgeführt:

Verzeichnis	Transaction Vision UI/ Job Server	Transaction Vision Analyzer	Beschreibung
<TVISION_HOME>/ config/codepage/*	Lesen	Lesen	Codepagekonfiguration
<TVISION_HOME>/ config/datamgr/*	Lesen	Lesen	Konfiguration der Datenbankverbindung
<TVISION_HOME>/ config/job/*	Lesen	Keine	Konfiguration des Job-Managers
<TVISION_HOME>/ config/license/*	Lesen	Lesen	TransactionVision-Lizenz
<TVISION_HOME>/ config/logging/*	Lesen	Lesen	Protokollierungsrichtlinie
<TVISION_HOME>/ config/rmi/*	Keine	Lesen	RMI-Richtlinie
<TVISION_HOME>/ config/services/*	Keine	Lesen	Analyzer-Konfiguration
<TVISION_HOME>/ config/setup/*	Lesen	Lesen	Konfiguration des TransactionVision-Setupprogramms
<TVISION_HOME>/ config/ technologyconst/*	Lesen	Keine	Verschiedene Konstantendefinitionen
<TVISION_HOME>/ config/ typeconversion/*	Lesen	Keine	Konfiguration der Typkonvertierung für Ereignisdetails
<TVISION_HOME>/ config/ui/*	Lesen	Keine	Konfiguration der Benutzeroberfläche, einschließlich Reporteinstellungen
<TVISION_HOME>/ config/usermgr/*	Lesen	Keine	Konfiguration von Benutzerdaten

Verzeichnis	Transaction Vision UI/ Job Server	Transaction Vision Analyzer	Beschreibung
<TVISION_HOME>/config/xdm/*	Lesen	Lesen	Verschiedene XDM-Definitionen
<TVISION_HOME>/config/xmlschema/*	Lesen	Lesen	Verschiedene XSL-Definitionen
<benutzerdef_verzeichnis>/logs/*	Schreiben	Schreiben	TransactionVision-Protokolle

Hinweis: <TVISION_HOME>/config/datamgr/Database.properties kann DB2-Benutzernamen und Kennwörter in Klartext enthalten. Beim Erteilen von Leseberechtigungen für diese Datei sollten Sie daher sehr sorgsam vorgehen. Um ein Maximum an Sicherheit zu erreichen, sollten Sie nur den Dateibesitzern und dem Systemadministrator die Berechtigung zum Lesen der Datei erteilen.

Führen Sie die folgenden Schritte aus, um unter Windows NT Dateizugriffsberechtigungen für eine Datei zu definieren:

- 1** Melden Sie sich als Systemadministrator an.
- 2** Öffnen Sie Windows-Explorer.
- 3** Klicken Sie mit der rechten Maustaste auf die Eigenschaften- oder Richtliniendatei und wählen Sie **Eigenschaften** aus. Das Dialogfeld mit den Dateieigenschaften wird geöffnet.
- 4** Klicken Sie auf die Registerkarte **Sicherheit** und weisen Sie den verschiedenen Benutzerkonten oder Sicherheitsgruppen Berechtigungen für die Datei zu.

Führen Sie die folgenden Schritte aus, um Konfigurationsdateiberechtigungen unter UNIX festzulegen:

- 1** Melden Sie sich als Systemadministrator an.

- 2 Führen Sie **chmod** aus, um den verschiedenen TransactionVision-Gruppen die geeigneten Zugriffsrechte zu erteilen. Empfohlene Einstellungen sind **rw--r--** für **/datamgr/database.properties** und **rw-r--r--** für alle anderen Konfigurationsdateien.
- 3 Verwenden Sie, falls notwendig, **chgrp**, um Dateigruppen zu ändern, und **chown**, um den Besitz für die Dateien zu ändern.

Schützen von TransactionVision Analyzer

TransactionVision Analyzer kommuniziert mit TransactionVision-Sensoren und -Agents und verarbeitet die erfassten Ereignisdaten, um aussagekräftige Analysen zu erstellen. Analyzer wird unter Windows als Windows-Dienst und unter UNIX als Daemon ausgeführt.

In TransactionVision wird RMI (Remote Method Invocation, Remote-Methodenaufruf) verwendet, um über einen angegebenen Port (standardmäßig 21100) mit Analyzer zu kommunizieren. TransactionVision UI/Job Server sowie die Befehlszeilenprogramme verwenden RMI für die Kommunikation mit Analyzer, um das Verhalten des Analyzer-Dienstes zu steuern, die Beendigung des Dienstes zu initiieren, das Starten und Beenden der Datenerfassung zuzulassen und Statusinformationen abzurufen. Informationen zu Dienstprogrammen, die für Analyzer ausgeführt werden, finden Sie unter "Administration Utilities" im Handbuch *Using TransactionVision*.

In TransactionVision werden RMI-Sicherheitsrichtlinien in der Datei **<TransactionVision_Installationsverzeichnis>/config/rmi/RMI.policy** definiert. Standardmäßig wird die Berechtigung **java.security.AllPermission** erteilt. Das Wichtigste an dieser Einstellung ist, dass hierdurch allen Benutzern, die Zugriff auf eine TransactionVision-Installation haben, der Remotezugriff auf Analyzer ermöglicht wird. Administratoren von TransactionVision sollten dies bedenken. Falls ein sichererer Zugriff auf Analyzer gewünscht ist, kann eine Kombination aus anderen Optionen notwendig sein, um den Zugriff zu steuern.

Am einfachsten lässt sich der Zugriff auf Analyzer auf dem Host, auf dem Analyzer ausgeführt wird, mithilfe der Dateiberechtigungen für die TransactionVision-Installationsdateien steuern. Diese Berechtigungen können nach Bedarf angepasst werden. Unter UNIX sollten die Berechtigungen zum Lesen, Schreiben und Ausführen beispielsweise nur dem Besitzer der Installationsdateien oder den Mitgliedern einer dedizierten Gruppe erteilt werden, die berechtigt ist, TransactionVision zu verwalten.

Zum Steuern des Remotezugriffs auf Analyzer kann die zuvor erwähnte RMI-Richtliniendatei verwendet werden. Durch das Zuweisen von **java.net.SocketPermission** können Sie steuern, von welchen Computern der Remotezugriff auf Analyzer zulässig ist. Mithilfe der nachfolgenden Vorlage können Sie dies erreichen. Als Mindestvoraussetzung muss die Datei **RMI.policy** dem Computer, auf dem UI/Job Server ausgeführt wird, Socketberechtigungen erteilen; andernfalls ist es nicht möglich, Analyzer über die Webschnittstelle zu steuern.

```
grant {
  permission java.net.SocketPermission "webserver:*", "accept, connect, listen, resolve";
  permission java.net.SocketPermission "remotepc:*", "accept, connect, listen, resolve";
  permission java.net.SocketPermission "localhost:*", "accept, connect, listen, resolve";
  permission java.io.FilePermission "<<ALL FILES>>", "read, write, delete, execute";
  permission java.net.NetPermission "**";
  permission java.awt.AWTPermission "**";
  permission java.util.PropertyPermission "**", "read, write";
  permission java.lang.reflect.ReflectPermission "**";
  permission java.lang.RuntimePermission "**";
  permission java.security.SecurityPermission "**";
  permission java.io.SerializablePermission "**";
};
```

In diesem Beispiel sind die Computer **webserver** und **remotepc** sowie der lokale Computer, auf dem Analyzer ausgeführt sind, die einzigen Computer, von denen Analyzer Befehle akzeptiert.

Die anderen Berechtigungen neben **java.net.SocketPermission** repräsentieren einen Mindestsatz an Berechtigungen, die von Analyzer selbst benötigt werden. Diese Berechtigungen sollten nicht geändert werden. Weitere Informationen zu den Besonderheiten dieser Berechtigungen finden Sie unter <http://java.sun.com/j2se/1.3/docs/guide/security/permissions.html>.

Schützen der TransactionVision-Datenbank

Mit dem Schützen der TransactionVision-Datenbank werden folgende Ziele verfolgt:

- ▶ Verhindern des nicht autorisierten Zugriffs auf vertrauliche Daten durch Personen, die diese Informationen nicht berufsbedingt benötigen.
- ▶ Verhindern, dass nicht autorisierte Benutzer Schaden anrichten, indem Daten böswillig gelöscht oder manipuliert werden.
- ▶ Überwachen des Benutzerzugriffs auf Daten mithilfe von Überwachungstechniken.

Der Datenbankzugriff wird normalerweise mittels Benutzerauthentifizierung und Benutzerautorisierung gesteuert. Im Rahmen der Authentifizierung wird die Identität eines Benutzers überprüft, wohingegen im Rahmen der Autorisierung bestimmt wird, ob ein Benutzer berechtigt ist, auf eine angeforderte Ressource zuzugreifen. Die Parameter zum Durchführen der Datenbankauthentifizierung werden in TransactionVision konfiguriert. Bei der Ausführung von **TVisionSetupInfo** werden Sie aufgefordert, den Datenbank-Benutzernamen und das zugehörige Kennwort einzugeben. Beide Angaben werden in der Konfigurationsdatei `<TVISION_HOME>/config/datamgr/Database.properties` gespeichert (das Kennwort wird in verschlüsselter Form gespeichert). Der Mechanismus, der der Benutzerauthentifizierung zugrunde liegt, kann je nach Datenbankprodukt variieren. In TransactionVision wird die betriebssystembasierte Authentifizierung für DB2, die Datenbankauthentifizierung für Oracle und die SQL Server-Authentifizierung für SQL Server unterstützt.

TransactionVision-Tabellen und -Schemas

In TransactionVision werden zwei verschiedene Arten von Datenbankschemas verwendet: das Systemschema mit der Bezeichnung TVISION und die benutzerdefinierten Projektschemas. Im Schema TVISION werden nicht ereignisbezogene Informationen gespeichert, während im Projektschema Projektereignisse gespeichert werden, die durch TransactionVision-Sensoren erfasst wurden.

Das Systemschema TVISION kann entweder automatisch bei der Ausführung von **TVisionSetupInfo** oder manuell durch Ausführen des Skripts `<TVISION_HOME>/bin/CreateSqlScript.[sh|bat] -c -e -system` erstellt werden. Anschließend können Sie mit der Arbeit mit TransactionVision beginnen und über die Webapplikation oder über **CreateSqlScript** TransactionVision-Projektschemas erstellen.

Datenbankberechtigungen

Damit TransactionVision erfolgreich in einer bestimmten Datenbankumgebung ausgeführt werden kann, müssen Sie sicherstellen, dass der für TransactionVision konfigurierte Datenbenutzer über die notwendigen Datenbankberechtigungen verfügt, um auf alle erforderlichen Datenbankobjekte zugreifen zu können. Diese Berechtigungen umfassen Folgendes:

- ▶ SELECT-Berechtigungen für die Systemkatalogtabellen der Datenbank.
- ▶ SELECT-, INSERT-, UPDATE- und DELETE-Berechtigungen für alle Tabellen im Schema TVISION.
- ▶ SELECT-, INSERT-, UPDATE- und DELETE-Berechtigungen für alle Tabellen in den Projektschemas.
- ▶ CREATE TABLE-Berechtigungen zum Erstellen der TVISION-Systemtabellen zur Installationszeit und der Projektschematabellen mithilfe des Projekterstellungs-Assistenten von TransactionVision.

Wenn für TransactionVision eine dedizierte Datenbank verwendet wird, genügt es häufig, einen Benutzer mit vordefinierten, hinreichenden Datenbankberechtigungen (z. B. ein Benutzer, dem die Rolle DBA zugewiesen ist) für den Zugriff auf die Datenbank auszuwählen. Wenn dies nicht möglich ist, müssen die oben aufgeführten Berechtigungen individuell erteilt werden. Die CREATE TABLE-Berechtigungen sind zum Ausführen von TransactionVision nicht obligatorisch, sobald die Tabellen erstellt wurden. Es ist daher möglich, das Erteilen dieser Berechtigungen zu vermeiden, indem der Datenbankadministrator angewiesen wird, die TVISION- und Projekttabellen manuell zu erstellen. Mit dem Dienstprogramm **CreateSqlScript** können die notwendigen SQL-Skripts zum Erstellen von TVISION- und Projekttabellen sowie Skripts erzeugt werden, mit denen dem konfigurierten Benutzer die erforderlichen Zugriffsberechtigungen für die TVISION- und Projekttabellen erteilt werden:

- CreateSqlScript -create -system
- CreateSqlScript -create -schema SCHEMA
- CreateSqlScript -grant -system
- CreateSqlScript -grant -schema SCHEMA

Wenn der konfigurierte Datenbankbenutzer nicht über die CREATE TABLE-Berechtigungen verfügt, ist es ihm nicht möglich, die Tabellen für ein neues Projekt über die TransactionVision-Applikation in Business Availability Center zu erstellen. Stattdessen müssen die Tabellen manuell erstellt werden, bevor der Assistent für neue Projekte ausgeführt wird.

Teil VI

Anhänge

A

Referenz der Dienstprogramme

Dieses Kapitel umfasst die folgenden Themen:

- CreateSqlScript auf Seite 360
- DB2RunStats auf Seite 363
- DB2Test auf Seite 365
- MigrateDB auf Seite 366
- nanny auf Seite 367
- OracleRunStats auf Seite 370
- OracleTest auf Seite 372
- PassGen auf Seite 374
- rebind_sensor auf Seite 375
- rebind_tux_sensor auf Seite 376
- runSupportSnapshot auf Seite 378
- ServicesManager auf Seite 381
- SetupModule auf Seite 385
- SQLServerTest auf Seite 386
- TVisionSetupInfo auf Seite 387

CreateSqlScript

Speicherort

TVISION_HOME/bin/CreateSqlScript.[sh|bat]

Zweck

Ermöglicht dem Benutzer, ein SQL-Skript zu erstellen und optional auszuführen, mit dem TransactionVision-Systemtabellen oder -Projekttabellen erstellt, gelöscht, importiert oder exportiert werden.

Hinweis: TransactionVision Analyzer und UI/Job Server müssen vor Import- oder Exportvorgängen in Datenbanken angehalten werden. Dadurch wird verhindert, dass Import- oder Exportvorgänge in Datenbanken aufgrund von Sperren von Analyzer oder UI/Job Server fehlschlagen. Wenn ein Schema gelöscht wird, stellen Sie sicher, dass keine aktiven Projekte oder angemeldeten Benutzer dieses Schema nutzen.

Syntax

```

CreateSqlScript
  {-create(-c) | -drop(-d) | -import(-i) | -export(-ex) | {-system(-sys) |
-schema(-s)SCHEMA | -table(-t) TABLE SCHEMA}
  [[-noscript(-n)] -execute(-e)]
    [-noprompt(-np)] [-noinsert(-ni)] (
    [-tablespace(-ts) TABLESPACE] [-dbMove(-m)]
    [-fileType(-f) IXF|DEL] [-lobPath(-lp) PATH]
    [-noLob(-nl)] [-dbproperties(-db) FILE]

```

Optionen

Option	Beschreibung
-drop (-d)	Löschen von Tabellen
-create (-c)	Erstellen von Tabellen
-execute (-e)	Ausführen des Skripts
-noscript (-n)	Keine Skripterstellung
-system (-sys)	Erstellen/Löschen von Systemtabellen (Schema TVISION)
-schema (-s) SCHEMA	Erstellen/Löschen von Projekttabellen im Schema SCHEMA
-table (-t) TABLE SCHEMA	Erstellen/Löschen der Tabelle TABLE im Schema SCHEMA
-tablespace (-ts) TBSPC	Verwenden von Tablespace TBSPC
-dbproperties (-db) FILE	Verwenden der Database.properties -Datei FILE
-import (-i)	Erzeugt ein Datenbankimportskript. Sie können Datenbankskripts mit dem Befehl db2 -n -t -f <Dateiname für das SQL-Skript> ausführen. Für eine DB2-Datenbank können Sie diese Option mit den Optionen -fileType und -lobPath kombinieren, um das Datenformat und den Speicherort von LOB anzupassen. Sie können diese Option NICHT mit den Optionen -noscript und -execute kombinieren.

Option	Beschreibung
-export (-ex)	Erzeugt ein Datenbankexportskript. Sie können Datenbankskripts mit dem Befehl db2 -n -t -f <Dateiname für das SQL-Skript> ausführen. Für eine DB2-Datenbank können Sie diese Option mit den Optionen -fileType und -lobPath kombinieren, um das Datenformat und den Speicherort von LOB anzupassen. Sie können diese Option NICHT mit den Optionen -noscript und -execute kombinieren.
-resetseq (-r)	Setzt die Sequenzstartnummer zurück, sodass sie den importierten Daten entspricht.
-dbMove (-m)	Zeigt den entsprechenden db2move -Befehl zum Importieren/Exportieren eines Schemas an, statt ein IMPORT/EXPORT-Skript zu erzeugen (nur in Verbindung mit -import/ -export).
-noprompt (-np)	Fordert beim Löschen von Datenbanktabellen keine Bestätigung an.
-noinsert (-ni)	Beim Erstellen von Datenbanktabellen werden keine ersten Tabellenzeilen eingefügt, die in der XDM-Datei definiert sind.
-fileType (-f) IXF DEL	Angabe des DB2-Dateiformats für die Datenausgabe, das beim Importieren/Exportieren von Daten verwendet wird. Der Standardtyp ist IXF. Beim IXF-Dateityp verwendet das Import-/Exportskript die Option LOBFILE für Zeilen mit mehr als 32K Daten. Beim DEL-Typ exportiert das Import-/Exportskript LOBFILES in eine Datei (erfordert Fixpack 8).

Option	Beschreibung
-lobPath (-lp) PATH	Gibt das Verzeichnis für LOBFILES von DB2 an. Der Standardwert ist das aktuelle Verzeichnis.
-noLob (-nl)	Erzeugt keinen DB2-Export/-Import von SQL mit der Option LOBINFILE. Mit dieser Option werden die LOB-Daten auf die ersten 32K Byte verkürzt.

Beispiele

- 1 Erstellen Sie Systemtabellen mit Schema als TVISION und führen Sie das Verfahren aus, ohne ein SQL-Skript zu erzeugen:

```
CreateSqlScript -e -n -c -sys
```

- 2 Erzeugen Sie ein SQL-Skript zum Erstellen von Projekttabellen mit Schema als PROJECT, ohne das Verfahren auszuführen:

```
CreateSqlScript -c -s PROJECT
```

- 3 Löschen Sie die Tabelle EVENT im Schema PROJECT und führen Sie das Verfahren aus, ohne ein SQL-Skript zu erzeugen:

```
CreateSqlScript -e -n -d -t EVENT PROJECT
```

DB2RunStats

Speicherort

TVISION_HOME/bin/DB2RunStats.[sh|bat]

Beschreibung

Aktualisiert Statistiken mit den physischen Merkmalen einer Tabelle und der zugeordneten Indizes. Zu diesen Merkmalen gehören die Anzahl der Datensätze, die Anzahl der Seiten und die durchschnittliche Datensatzlänge. Diese Statistiken werden vom Optimierer verwendet, um die Zugriffspfade für die Daten zu bestimmen.

Dieser Befehl wird aufgerufen, wenn an einer Tabelle viele Aktualisierungen vorgenommen wurden, beispielsweise wenn von TransactionVision Analyser fortlaufend Daten in DB2 erfasst werden. Dies kann zu beträchtlichen Leistungsverbesserungen bei Abfragen, die von TransactionVision-Ansichten und -Reports ausgeführt werden, sowie bei Abfragen führen, die intern durch TransactionVision Analyser zum Korrelieren von Ereignissen ausgeführt werden.

- ▶ Dieses Skript kann so eingerichtet werden, dass es als geplanter Batch-Job mit dem UNIX-Mechanismus **cron** oder dem Windows-Planer ausgeführt wird.
- ▶ Während dieses Skript ausgeführt wird, ist die TransactionVision Analyser-Verarbeitung verlangsamt.
- ▶ Dieses Skript sollte normalerweise täglich ausgeführt werden. Bei mehr Nachrichten kann die Frequenz auch höher sein.
- ▶ Dieses Skript muss basierend auf Ihrem System angepasst werden, um den richtigen DB2-Installations- und TVISION_HOME-Speicherort festzulegen
- ▶ Wenn der Benutzer zusätzliche Tabellen für das Projektschema definiert hat, müssen neue RUNSTATS-Anweisungen hinzugefügt werden, um die zusätzlichen Tabellen abzudecken.

Syntax

```
DB2RunStats username passwd database-name schema-name [-v7]
```

Optionen

Option	Beschreibung
-username	Das Benutzerkonto mit der Berechtigung zum Ausführen von RUNSTATS und zum Herstellen von Datenbankverbindungen
-passwd	Das dem Benutzernamen zugeordnete Kennwort
-database-name	Der Name der Datenbank, mit der eine Verbindung hergestellt werden soll
-schema-name	Der Name des Schemas, das vom Projekt gelesen und mit Daten gefüllt wird
-v7	Verwendung in einer DB2 7.x-Umgebung. Der Standardvorgang ist für DB2 8.1.

DB2Test**Speicherort**

com.bristol.tvision.admin.DB2Test

Beschreibung

Misst die Leistung von INSERT der DB2-Datenbank.

Mit dem Dienstprogramm werden Beispiele von Ereignisdaten in die Tabelle RAW_EVENT des angegebenen Schemas eingefügt. Erstellen Sie vor Ausführung des Tests ein neues Projektschema mit **CreateSqlScript** (das nach dem Test gelöscht werden kann). Details zum Einrichten der erforderlichen Testumgebung finden Sie im Handbuch *TransactionVision Planning Guide*.

Syntax

```
java com.bristol.tvision.admin.DB2Test databaseName user passwd schema
eventCount eventSize threadCount {-commit n}{-jdbcBatch}
```

Optionen

Option	Beschreibung
-databaseName	Der Name der DB2-Datenbank mit dem Schema, das für den Test verwendet werden soll
-user	DB2-Benutzername
-passwd	DB2-Kennwort
-schema	Das TransactionVision-Schema, in dem die Beispielergebnisdaten gespeichert werden
-eventCount	Anzahl der zu erzeugenden Ereignisse
-threadCount	Anzahl von Threads für die Erzeugung von Ereignissen
-commit n	Führt alle n Insert-Anweisungen als Teil eines Batch-Vorgangs aus. Standardmäßig wird jede Insert-Anweisung getrennt übergeben.
-jdbcBatch	Verwendung der standardmäßigen JDBC-Batch-Verarbeitung

MigrateDB

Speicherort

TVISION_HOME/bin/MigrateDB.[sh|bat]

Beschreibung

Migriert Projektdatenbankdateien von einer älteren Version von TransactionVision zur aktuellen Version. Dieses Skript muss nach einer Upgrade-Installation ausgeführt werden. Es muss in einer konfigurierten TransactionVision-Umgebung ausgeführt werden. **Database.properties** muss richtig für die Kommunikation mit der Datenbank festgelegt werden.

Syntax

```
MigrateDB
```

nanny**Speicherort**

```
TVISION_HOME/bin/nanny.[sh|bat]
```

Beschreibung

Verwaltet die über Nanny Manager gesteuerten Dienste.

TransactionVision-Komponenten wie SonicMQ Domain Manager, SonicMQ Broker, Analyzer und UI/Job Server sind Dienste, die über Nanny Manager verwaltet (gestartet, angehalten, überwacht) werden.

Nanny Manager ist ein Dienst (namens HP Business Availability Center) unter Windows und ein Prozess unter UNIX (nannyManager). Mit dem Dienstprogramm **nanny** kann festgestellt werden, welche Komponenten über Nanny Manager verwaltet werden.

Syntax

```
nanny
-s <Hostname> -p <Port> -disableService <Dienstname> -enableService
<Dienstname> -getServiceInfo <Dienstname> -isServiceRunning <Dienstname>
-listAllDisabledServicesNames -listAllServicesNames -listDeadServicesNames
-listLiveServicesNames -listServiceInfo -listStartingServicesNames -restartService
-<Dienstname> -retrieveNannyManagerHTMLAdapterPort -showStackTrace
<Dienstname> -showStackTrace <Dienstname> -startAllServices -startService
<Dienstname> -stopAllServices -stopService <Dienstname>
```

Optionen

Option	Beschreibung
-s <Hostname>	Hostname des Hosts, auf dem Analyzer installiert ist. Der Standardwert ist localhost .
-p <Port>	Port des Servers für das Dienstprogramm nanny . Der Standardwert ist 11020 .
-disableService <Dienstname>	Deaktiviert den mit <Dienstname> angegebenen Dienst.
-enableService <Dienstname>	Aktiviert den mit <Dienstname> angegebenen Dienst.
-getServiceInfo <Dienstname>	Zeigt Dienstinformationen zu <Dienstname> an.
-isServiceRunning <Dienstname>	Überprüft, ob der von <Dienstname> angegebene Dienst ausgeführt wird
-listAllDisabledServicesNames	Führt alle deaktivierten Dienstnamen auf
-listAllServicesNames	Führt alle Dienstnamen auf
-listDeadServicesNames	Führt alle Dienste auf, die nicht ausgeführt werden
-listLiveServicesNames	Führt alle Dienste auf, die ausgeführt werden
-listServiceInfo	Führt Attribute für alle Dienste auf
-listStartingServicesNames	Führt alle Dienste auf, die im Startstatus sind
-restartService <Dienstname>	Hält den mit <Dienstname> angegebenen Dienst an und startet ihn

Option	Beschreibung
-retrieveNannyManagerHTMLAdpaterPort	Zeigt den JMX-Port für das Dienstprogramm nanny an
-showStackTrace <Dienstname>	Zeigt die Stapelüberwachung für den mit <Dienstname> angegebenen Dienst an
-startAllServices	Startet alle Dienste
-startService <Dienstname>	Startet einen bestimmten im Argument angegebenen Dienst
-stopAllServices	Hält alle Dienste an
-stopService <Dienstname>	Hält einen bestimmten im Argument angegebenen Dienst an

Beispiele

Geben Sie Folgendes ein, um alle Dienste aufzuführen, die von Nanny Manager verwaltet werden:

```
nanny.bat listAllServiceNames
```

Mit dem Befehl wird die folgende Ausgabe erzeugt:

```
Executing: listAllServicesNames Service: on (localhost,11020)
Result = [tv_as, tv_message_broker, tv_domain_manager, tv_analyzer]
```

Mit dem Dienstprogramm **nanny** kann eine Komponente aktiviert oder deaktiviert werden. Sobald eine Komponente deaktiviert wurde, wird sie erst wieder von Nanny Manager verwaltet, wenn sie wieder aktiviert ist.

Führen Sie den folgenden Befehl aus, um Analyzer zu deaktivieren:

```
nanny.bat disableService tv_analyzer
```

Führen Sie den folgenden Befehl aus, um Analyzer zu aktivieren:

```
nanny.bat enableService tv_analyzer
```

Mit dem Dienstprogramm **nanny** kann eine Komponente auch gestartet werden, nachdem sie aktiviert wurde.

Führen Sie den folgenden Befehl aus, um die Analyzer-Komponente zu starten:

```
nanny.bat startService tv_analyzer.
```

OracleRunStats

Speicherort

TVISION_HOME/bin/OracleRunStats.[sh|bat]

Beschreibung

Hinweis: Zu Oracle 10g gehört ein integrierter Job-Scheduler, mit dem automatisch Statistiken erfasst werden. Sie müssen dieses Skript für Version 10g nicht ausführen.

Aktualisiert Statistiken mit den physischen Merkmalen einer Tabelle und der zugeordneten Indizes. Zu diesen Merkmalen gehören die Anzahl der Datensätze, die Anzahl der Seiten und die durchschnittliche Datensatzlänge. Diese Statistiken werden vom Optimierer verwendet, um die Zugriffspfade für die Daten zu bestimmen.

Dieser Befehl wird aufgerufen, wenn an einer Tabelle viele Aktualisierungen vorgenommen wurden, beispielsweise wenn von TransactionVision Analyzer fortlaufend Daten in Oracle erfasst werden. Dies kann zu beträchtlichen Leistungsverbesserungen bei Abfragen, die von TransactionVision-Ansichten und -Reports ausgeführt werden, sowie bei Abfragen führen, die intern durch TransactionVision Analyzer zum Korrelieren von Ereignissen ausgeführt werden.

Dieses Skript kann so eingerichtet werden, dass es als geplanter Batch-Job mit dem UNIX-Mechanismus **cron** oder dem Windows-Planer ausgeführt wird.

Hinweis: Während dieses Skript ausgeführt wird, ist die TransactionVision Analyzer-Verarbeitung verlangsamt.

Dieses Skript sollte normalerweise täglich ausgeführt werden. Bei mehr Nachrichten kann die Frequenz auch höher sein.

Führen Sie dieses Skript über ein Benutzerkonto aus, das über die Berechtigung zur Durchführung von Datenbankoperationen verfügt. Dieses Skript muss mit einem Oracle-Benutzerkonto ausgeführt werden.

Hinweis: Dieses Skript muss basierend auf Ihrem System angepasst werden, damit das richtige Skript für die Initialisierung der Umgebung aufgerufen wird (wie PROFILE oder BASHRC) und damit der richtige Oracle-Installationspeicherort und der richtige TVISION_HOME-Speicherort festgelegt werden. Darüber hinaus muss sich die Datei **OracleRunStats.sql** im aktuellen Verzeichnis befinden, wenn das Skript ausgeführt wird.

Syntax

```
OracleRunStats user_name passwd database_name schema_name
```

Optionen

Option	Beschreibung
-user_name	Das Oracle-Benutzerkonto für die Ausführung des Skripts
-passwd	Das dem Benutzernamen zugeordnete Kennwort
-database_name	Der Name der Datenbank, mit der eine Verbindung hergestellt werden soll
-schema_name	Der Name des Schemas, das vom Projekt gelesen und mit Daten gefüllt wird

OracleTest

Speicherort

com.bristol.tvision.admin.OracleTest

Beschreibung

Misst die Leistung von INSERT der Oracle-Datenbank.

Mit dem Dienstprogramm werden Beispiele von Ereignisdaten in die Tabelle RAW_EVENT des angegebenen Schemas eingefügt. Erstellen Sie vor Ausführung des Tests ein neues Projektschema mit **CreateSqlScript** (das nach dem Test gelöscht werden kann). Details zum Einrichten der erforderlichen Testumgebung finden Sie im Handbuch *TransactionVision Planning Guide*.

Syntax

```
java com.bristol.tvision.admin.OracleTest databaseName host port user passwd
schema eventCount eventSize threadCount [-VARCHAR | -BLOB |
-LONGRAW] {-commit n} {-jdbcBatch} {-OracleBatch} {-thin} {-parallel} {-url URL}
```

Optionen

Option	Beschreibung
-databaseName	Der Name der Oracle-Datenbank mit dem Schema, das für den Test verwendet werden soll
-host	Name des Hostsystems mit dem Oracle-Server
-user	Oracle-Benutzername
-passwd	Oracle-Kennwort
-schema	Das TransactionVision-Schema, in dem die Beispielergebnisdaten gespeichert werden
-eventCount	Anzahl der zu erzeugenden Ereignisse
-eventSize	Größe des Puffers für Ereignisbenutzerdaten (standardmäßig sind es 1024 Byte)
-threadCount	Anzahl von Threads für die Erzeugung von Ereignissen
-VARCHAR	Verwenden Sie diese Option, wenn die Tabelle RAW_EVENT mit einer Spaltendefinition VARCHAR erstellt wurde.
-BLOB	Verwenden Sie diese Option, wenn die Tabelle RAW_EVENT mit einer Spaltendefinition BLOB erstellt wurde.
-LONGRAW	Verwenden Sie diese Option, wenn die Tabelle RAW_EVENT mit einer Spaltendefinition LONGRAW erstellt wurde.
-commit n	Führt alle <i>n</i> Insert-Anweisungen als Teil eines Batch-Vorgangs aus. Standardmäßig wird jede Insert-Anweisung getrennt übergeben.
-jdbcBatch	Verwendung der standardmäßigen JDBC-Batch-Verarbeitung

Option	Beschreibung
-oracleBatch	Verwendung der Oracle-Batch-Verarbeitung für die Aktualisierung
-thin	Verwendung eines Thin Client-Treibers. Standardmäßig wird der OCI-Clienttreiber verwendet.
-parallel	Verwendung der Oracle-Option INSERT PARALLEL statt der Standardoption INSERT INTO
-url URL	Standardmäßig verwendet OracleTest einen geeigneten JDBC-URL für die Thin oder OCI-Clienttreiber. Die Standardeinstellung kann jedoch überschrieben werden, indem hier der JDBC-URL angegeben wird.

PassGen

Speicherort

TVISON_HOME/bin/PassGen

Beschreibung

Ein Kennwort kann mit dem Dienstprogramm **PassGen** verschleiert werden.

Syntax

```
PassGen /system <Kennwort>
```

Optionen

Option	Beschreibung
-password	Eine alphanumerische Zeichenfolge mit maximal 128 Zeichen

Beispiel

```
cd <Installationsverzeichnis>/bin
$ PassGen /system TheLazyFoxJumpedHigh
```

password:

OBF:3q6r3xxz3y3r3xjs3wx03yc63n0r3lbr3vc03wd745893wre44u0413j3kn93zw
y40vi432i44fr3m453m894493439040pc40303kjd419r44na3wx0451h3wir3v6m3
lfr3mwj3yi03wre3xpi3xxz3y3r3q23

rebind_sensor

Speicherort

`TVISION_HOME/bin/rebind_sensor.sh`

Beschreibung

Mit diesem Skript wird der TransactionVision WebSphere MQ-Sensor auf der AIX-Plattform erneut gebunden.

In WebSphere MQ-Supportpacks können sich interne Symbole, die vom TransactionVision WebSphere MQ-Sensor auf der AIX-Plattform exportiert wurden, ändern. Wenn ein internes Symbol, das aus der Sensorbibliothek exportiert wurde, in der WebSphere MQ-Bibliothek nicht mehr verfügbar ist, kann die Applikation nicht gestartet werden und es treten verschiedene Fehler im Zusammenhang mit der Symbolauflösung auf.

Daher muss das Skript **rebind_sensor** immer ausgeführt werden, wenn ein WebSphere MQ-Support- oder Fixpack installiert wurde.

Damit werden die TransactionVision-Sensorbibliotheken in **TVISION_HOME/lib** geändert.

In WebSphere MQ 6.0 und höher muss dieses Dienstprogramm zweimal wie folgt ausgeführt werden, um die 32-Bit- und die 64-Bit-Bibliotheken zu instrumentieren:

\$TVISION_HOME/bin/rebind_sensor.sh (bindet die 32-Bit-Bibliothek erneut)
\$TVISION_HOME/bin/rebind_sensor.sh -64 (bindet die 64-Bit-Bibliothek erneut)

Syntax

```
rebind_sensor.sh [-v|-s|-h][-64]
```

Optionen

Option	Beschreibung
-v	Schreibt Fehler in die Konsole. Standardmäßig werden die Fehler in die Datei TVISION_HOME/logs/mqsensorbind.log geschrieben
-s	Verwendet den unbeaufsichtigten Modus, bei dem vor der Ausführung keine Bestätigung erfolgt
-h	Zeigt eine Verwendungsmeldung an
-64	Bindet die 64-Bit-Sensorbibliothek erneut. Wenn diese Option nicht vorhanden ist, wird standardmäßig die 32-Bit-Bibliothek erneut gebunden

rebind_tux_sensor

Speicherort

TVISION_HOME/bin/rebind_tux_sensor.sh

Beschreibung

Mit diesem Skript wird der TransactionVision BEA Tuxedo-Sensor auf AIX-, Solaris- und HP-UX-Plattformen erneut gebunden.

Syntax

```
rebind_tux_sensor.sh [-v|-s|-h]
```

Optionen

Option	Beschreibung
-v	Schreibt Fehler in die Konsole. Standardmäßig werden die Fehler in die Datei TVISION_HOME/logs/tuxsensorbind.log geschrieben
-s	Verwendet den unbeaufsichtigten Modus, bei dem vor der Ausführung keine Bestätigung erfolgt
-h	Zeigt eine Verwendungsmeldung an

Hinweise zur Verwendung

Die BEA Tuxedo-Sensorbibliothek muss mit dem vollständigen Pfad mit der BEA Tuxedo-Bibliothek verknüpft werden. Geschieht dies nicht, können bei einer BEA Tuxedo-Applikation, die vom TransactionVision-Sensor überwacht wird, Fehler auftreten, die auf nicht aufgelöste Bibliotheksabhängigkeiten oder Symbole hinweisen.

Daher muss das Skript **rebind_tux_sensor.sh** nach der Installation des TransactionVision-Sensors ausgeführt werden. Zudem muss es immer ausgeführt werden, wenn BEA Tuxedo aktualisiert wird oder das Installationsverzeichnis für BEA Tuxedo geändert wird.

Die Umgebungsvariablen **TVISION_HOME** und **TUXDIR** müssen vor der Ausführung des Skripts festgelegt werden. **TUXDIR** sollte auf den vollständigen absoluten Pfad des Installationsverzeichnisses für den BEA Tuxedo-Sensor festgelegt werden. Mit dem Skript werden ggf. die TransactionVision-Sensorbibliotheken in **TVISION_HOME/lib** oder **TVISION_HOME/lib64** geändert.

runSupportSnapshot

Speicherort

<Installationsverzeichnis_Diagnostics_Probe>/contrib/JASUtilities/
snapins/runSupportSnapshot.[sh|cmd]

Hinweis: Das <Installationsverzeichnis_TransactionVision_Sensor> befindet sich in <Installationsverzeichnis_Diagnostics_Probe>.

Beschreibung

Mit dem Dienstprogramm **runSupportSnapshot** wird eine ZIP-Datei erstellt, die alle Dateien enthält, die für die Fehlerbehebung einer oder mehrerer Instanzen des Java-Agents in einer Diagnostics- oder TransactionVision-Bereitstellungsumgebung relevant sind. Diese ZIP-Datei ist für die Kundenunterstützung vorgesehen.

Die ZIP-Datei enthält Folgendes:

- ▶ Dateien aus dem Verzeichnis
<Installationsverzeichnis_Diagnostics_Probe>\etc
- ▶ Dateien aus dem Verzeichnis
<Installationsverzeichnis_Diagnostics_Probe>\log
- ▶ Dateien aus dem Verzeichnis
<Installationsverzeichnis_TransactionVision_Sensor>\config
- ▶ Dateien aus dem Verzeichnis
<Installationsverzeichnis_TransactionVision_Sensor>\logs
- ▶ Bericht **Property Scanner**, mit dem ein geändertes Agentverzeichnis mit der Freigabeversion verglichen wird. Es werden Unterschiede zwischen den Eigenschaften-, Verweis- und (nur für TransactionVision-Sensoren) XML-Dateien aufgeführt.
- ▶ Proben- oder Sensorinstanzinformationen, einschließlich der Eigenschafteneinstellungen. Für Agents, die in 1.5 JVMs ausgeführt werden, werden auch Umgebungsvariablen, Stapelabbilder und Informationen zu Klassenladeprogrammen aufgenommen.

Hinweis: Wenn eine Diagnostics-Probe für die Unterstützung von SSL konfiguriert ist, kann das Dienstprogramm die Instanzinformationen nicht erfassen.

Einfache Syntax

Für dieses Dienstprogramm werden Standardeinstellungen verwendet, die für die meisten Agentumgebungen gelten. Daher sieht eine typische Syntax einfach wie folgt aus:

```
runSupportSnapshot.[sh|cmd] -console
```

Vollständige Syntax

Mit diesem Dienstprogramm können Sie aber auch Optionen angeben, um die Standardeinstellungen zu überschreiben. So kann das Dienstprogramm in untypischen Agentumgebungen eingesetzt werden. Diese optionale Syntax sieht wie folgt aus:

```
runSupportSnapshot.[sh|cmd] -console [ -Zipfile ] [ -AddFiles ]  
[ -JAUser ] [ -JAPass ] [ -JAPort ] [ -JACount ]  
[ -FTPSite ] [ -User ] [ -Pass ] [ -RemoteFile ]  
[ -Source1 ] [ -Source2 ] [ -DiffOnly ] [ -SubDirs ] [ -Sort ]
```

Optionale Syntaxoptionen

Option	Beschreibung
-ZipFile	Der Name der ZIP-Datei, die erstellt werden soll. Standard: HPCustomerSupport.zip
-AddFiles	Weitere Verzeichnisse, die in die ZIP-Datei aufgenommen werden sollen. Damit alle Unterverzeichnisse eines Verzeichnisses aufgenommen werden, fügen Sie dem Verzeichnisnamen ein "+" an.
-JAUser	Der Benutzername, der für den Zugriff auf den Host verwendet werden soll, auf dem der Agent ausgeführt wird. Standard: admin
-JAPass	Das entsprechende Kennwort. Standard: admin
-JAPort	Der Port, bei dem die Abfrage für Agentinstanzen begonnen werden soll. Standard: 35000
-JACount	Die maximale Anzahl von Java-Agentinstanzen auf diesem Host, für die Daten in die ZIP-Datei aufgenommen werden. Standard: 20
-FTPSite	Der Name der FTP-Site.
-User	Der Benutzername für den Zugriff auf die FTP-Site.
-Pass	Das entsprechende Kennwort.
-RemoteFile	Der Remote-Dateiname. Stellen Sie bei Bedarf dem Dateinamen den Verzeichnispfad voran. Beispiel: Kunde/diag/AgentBaseline.zip .
-Source1	Das Verzeichnis etc der aktuellen Agentinstallation. Standard: <Installationsverzeichnis_Java_Probe>\ etc
-Source2	Eine archivierte Kopie des vorinstallierten Verzeichnisses etc in dem Verzeichnis. Standard: Ein internes Verzeichnis der Agent-Installation
-DiffOnly	Festlegung, ob nur Eigenschaften mit Unterschieden angezeigt werden sollen. Standard: Yes
-SubDirs	Angabe, ob auch Unterverzeichnisse verglichen werden sollen. Standard: No
-Sort	Angabe, ob die Ausgabe nach Eigenschaftennamen sortiert werden soll. Standard: No

Beispiel

```

> cd C:\MercuryDiagnostics\JavaAgent\DiagnosticsAgent\contrib\JASUtilities
\Snapins
> java -Dcom.hp.javaagent.diagnostics.home="..\..\.." -jar "..\..\lib\setupModule.jar"
-launchClass com.mercury.opal.javaprobe.setupModule.SetupModule -launchMethod
launchSetupModule -importJarList
probe.jar,org.mortbay.jetty-jdk1.2.jar,javax.servlet.jar,mail.jar,activation.jar
-importJarsFrom "..\..\lib,lib" -customerSnapshot -console
INFORMATION-> [Looking for Java Agent Instances]: Looking for Java Agent
Instances...please wait...
INFORMATION-> [Java Agent found at http://localhost:35000/inst/
customerSupportSnapshot]: Java Agent Discovery
INFORMATION-> [Creating Probe Information file]: Java Agent Information
INFORMATION-> [Creating Configuration Information file]: Java Agent Configuration
Information
INFORMATION-> [Zip File Created]: Zip file
C:\HPCode\diag_head\javaprobe\build\contrib\JASUtilities\Snapins\HPCustomerSup
portFile.zip Created with length 159215

```

Prüfen Sie bei Bedarf die ZIP-Datei auf vertrauliche Daten, bevor Sie sie an die Kundenunterstützung übermitteln.

ServicesManager**Speicherort**

TVISION_HOME/bin/ServicesManager.[sh|bat]

Zweck

Verwalten des TransactionVision Analyzer-Diensts. Analyzer verwendet eine eingebettete RMI-Registrierung, damit Analyzer-Instanzen über die TransactionVision-Webbenutzeroberfläche gesteuert werden können, die auf Remote-Hosts ausgeführt wird.

Wenn Analyzer mit der Option **-keepcollect** des Dienstprogramms **ServicesManager** angehalten oder beendet wird, können die Sensoren nur vollständig angehalten werden, indem Analyzer und das Projekt gestartet werden und dann das Projekt normal angehalten wird. Andernfalls erzeugen die Sensoren weiterhin Ereignisse, die in der Ereigniswarteschlange erfasst werden, bis die letzte von Analyzer gesendete Konfigurationsnachricht abgelaufen ist. Siehe "Configuration Message Expiry" in *Using TransactionVision*.

Syntax

```

ServicesManager
{(-start [-project (-proj) PROJECTNAME]) |
 (-stop [-quiesce][[-project (-proj) PROJECTNAME]
 [-keepcollect]) |
 (-exit [-quiesce][[-keepcollect]) |
 (-status [-project (-proj) PROJECTNAME]) |
 (-killserver) |
 (-reconfig {classification | logging | analyzer}) |
 (-versioninfo)
 ([-host HOST][[-rmiregp PORTNUMBER][[-debug])

```

Optionen

Option	Beschreibung
-start	Startet den Analyzer-Prozess, wenn er noch nicht ausgeführt wird
-stop	Analyzer beendet die Erfassung von Ereignisdaten
-exit	Analyzer beendet die Erfassung von Ereignisdaten, dann wird der Prozess beendet
-quiesce	Das Standardverhalten von Analyzer beim Anhalten oder Beenden ist die sofortige Beendigung der Erfassung. Ist dieses Kennzeichen festgelegt, löscht Analyzer ausstehende Ereignisse in der Ereigniswarteschlange vor dem Anhalten oder Beenden. Wenn es einen großen Rückstand an Ereignissen gibt, kann es eine Weile dauern, bis Analyzer angehalten oder beendet wurde.
-keepcollect	Es wird keine Nachricht zum Anhalten an den Sensor gesendet. Diese Option wird zusammen mit -stop oder -exit verwendet, damit der Sensor die Erfassung fortsetzt.
-status	Meldet den aktuellen Analyzer-Status

Option	Beschreibung
-reconfig	<p>Lädt die Analyzer-Konfigurationseinstellungen erneut (Klassifizierung/Protokollierung/alle Einstellungen), ohne die Ereigniserfassung zu beenden.</p> <ul style="list-style-type: none"> ▶ logging – Lädt die log4j-XML-Konfigurationsdateien erneut in das Verzeichnis <TVISION_HOME>/config/logging. ▶ analyzer – Initialisiert Analyzer erneut, indem die meisten Analyzer-Einstellungen in der Datei Analyzer.properties und alle in der Datei Beans.xml beschriebenen Beans erneut geladen werden, die Protokollierung und alle XML-basierten Regeldateien erneut initialisiert werden. <p>Einstellungen im Abschnitt Collection Properties der Datei Analyzer.properties, außer der Eigenschaft write_to_buffer_table, werden von Analyzer mit der Option -reconfig nicht erneut geladen.</p>
-versioninfo	Gibt Analyzer-Versionsinformationen zurück
-killserver	Führt Analyzer sofort herunter. Dieses Kennzeichen ist nicht empfehlenswert. Die bevorzugte Methode, um Analyzer reibungslos herunterzufahren, ist -exit .
-project (-proj) PROJECTNAME	<p>Der Name des Projekts für den angegebenen Befehl. Wenn der Projektname ein Leerzeichen enthalten muss, schließen Sie den Projektnamen in Anführungszeichen ein (z. B. -proj "Name des Projekts"). Das Projekt muss über eine Kommunikationsverbindung verfügen, damit Analyzer Ereignisse verarbeiten kann. Nur die Befehle -start, -stop und -status können für ein einzelnes Projekt ausgeführt werden. Diese Option kann nicht zusammen mit den Optionen -host oder -rmiregp verwendet werden. Wenn die Option -project verwendet wird, werden Host und Port von Analyzer in der Datenbank gesucht.</p>
-host HOST	Name des Hosts, auf dem Analyzer ausgeführt wird. Dies kann der Name oder die IP-Adresse sein. Falls kein Host angegeben ist, wird der lokale Host verwendet.

Option	Beschreibung
-rmiregp PORTNUMBER	Portnummer, die Analyzer auf RMI-Verbindungen abhört. Der Standardwert ist der Wert, der von der Eigenschaft analyzer_port in der Datei Analyzer.properties angegeben wird. Diese Option ist nur wirksam, wenn mit einer Analyzer-Instanz kommuniziert wird, die bereits ausgeführt wird. Beim Start verwendet Analyzer immer den Wert, der in Analyzer.properties angegeben ist.
-debug	Startet den Analyzer-Prozess im Debug-Modus.

Beispiele

- Starten Sie das Projekt PROJECT:


```
ServicesManager -start -proj PROJECT
```
- Halten Sie Analyzer auf dem Host HOST an:


```
ServicesManager -stop -host HOST
```

SetupModule

Speicherort

<Installationsverzeichnis_Java-Agent>\DiagnosticsAgent\lib

Hierbei ist <java_agent_install_verz> der Pfad des Java-Agent-Installationsverzeichnisses. Der Standardpfad ist **C:\MercuryDiagnostics\JavaAgent** unter Windows und **\opt\MercuryDiagnostics\JavaAgent** unter UNIX.

Beschreibung

Mit diesem Skript wird das Java-Agent-Setupmodul gestartet.

Syntax

```
SetupModule -recordFile name.rec -installFile name.rec -console
```

Optionen

Option	Beschreibung
-recordFile name.rec	Zeichnet auf
-installFile name.rec	Gibt die Aufzeichnung wieder oder automatisiert
-console	Startet im Konsolenmodus

SQLServerTest

Speicherort

com.bristol.tvision.admin.SQLServerTest

Beschreibung

Misst die Leistung von INSERT der SQL Server-Datenbank.

Mit dem Dienstprogramm werden Beispiele von Ereignisdaten in die Tabelle RAW_EVENT des angegebenen Schemas eingefügt. Erstellen Sie vor Ausführung des Tests ein neues Projektschema mit **CreateSqlScript** (das nach dem Test gelöscht werden kann).

Syntax

```
java com.bristol.tvision.admin.SQLServerTest databaseName user passwd schema
eventCount eventSize threadCount {-commit n}{-jdbcBatch}
```

Optionen

Option	Beschreibung
-databaseName	Der Name der SQL Server-Datenbank mit dem Schema, das für den Test verwendet werden soll
-user	SQL Server-Benutzername
-passwd	SQL Server-Kennwort
-schema	Das TransactionVision-Schema, in dem die Beispielergebnisdaten gespeichert werden
-eventCount	Anzahl der zu erzeugenden Ereignisse
-threadCount	Anzahl von Threads für die Erzeugung von Ereignissen
-commit n	Führt alle n Insert-Anweisungen als Teil eines Batch-Vorgangs aus. Standardmäßig wird jede Insert-Anweisung getrennt übergeben.
-jdbcBatch	Verwendung der standardmäßigen JDBC-Batch-Verarbeitung

TVisionSetupInfo**Speicherort**

TVISION_HOME/bin/TVisionSetupInfo.[sh|bat]

Zweck

Erfasst Informationen vom Administrator zu externen Werkzeugen, die von TransactionVision verwendet werden, sodass CLASSPATH und der Bibliothekspfad vor der Ausführung von TransactionVision richtig eingerichtet werden können. Mit diesem Dienstprogramm wird Folgendes ausgeführt:

- Speichern des Installationspfads für Softwarewerkzeuge in **TVISION_HOME/config/setup/DefaultInstallPath.xml**.
- Ändern von **TVISION_HOME/config/datamgr/Database.properties** basierend auf Benutzereingaben.
- Ändern von **TVISION_HOME/config/ldap/Ldap.properties** basierend auf Benutzereingaben.
- Erzeugen von **TVISION_HOME/bin/SetupEnv.[sh|bat]**. Damit wird der Mindestsatz an Umgebungsvariablen (JAVA_HOME, CLASSPATH, freigegebener Bibliothekspfad für unterschiedliche Plattformen usw.) festgelegt, die für TVision erforderlich sind.

Syntax

```
TVisionSetupInfo.[sh|bat] - cleanweb
```

Optionen

Option	Beschreibung
-cleanweb	Entfernen der TransactionVision-Webapplikation.

Hinweise

Mit **TVISION_HOME/config/setup/Setup.properties** wird die XML-Datei angegeben, in der der Standardinstallationspfad der Software und die Setupprotokolldatei gespeichert werden.

Der Benutzer kann später geänderte/erzeugte Dateien ändern.

Da mit diesem Dienstprogramm die Konfigurationsdateien von TransactionVision geändert werden, muss der Benutzer über die Berechtigung zum Ändern von Dateien verfügen, um es auszuführen (**root** unter Unix oder **Administrator** unter Windows).

Die Datei **TVISION_HOME/config/setup/DefaultInstallPath.xml** weist folgendes Format auf:

```
<?xml version="1.0"?>
<DefaultInstallPath>
  OS name="Windows">
    <DB2>C:\Programme\IBM\sqlib</DB2>
    <WebSphereMQ>C:\Programme\IBM\WebSphere MQ</WebSphereMQ>
    <WebSphereMQJava>C:\Programme\IBM\WebSphere MQ\Java</
WebSphereMQJava>
    <WebSphereAppServer>C:\Programme\Websphere\ AppServer</
WebSphereAppServer>
    <WebLogic>C:\bea\weblogic81</WebLogic>
  </OS>
  <OS name="SunOS">
    <DB2>/opt/IBMdb2/V7.1</DB2>
    <Oracle></Oracle>
    <WebSphereMQ>/opt/mqm</WebSphereMQ>
    <WebSphereMQJava>/opt/mqm/java</WebSphereMQJava>
    <WebSphereAppServer>/opt/WebSphere/AppServer </WebSphereAppServer>
  <WebLogic>/usr/local/bea/weblogic81</WebLogic>
  </OS>
  <OS name="AIX">
    <DB2>/usr/lpp/db2_07_01</DB2>
    <Oracle></Oracle>
    <WebSphereMQ>/usr/lpp/mqm</WebSphereMQ>
    <WebSphereMQJava>/usr/lpp/mqm/java</WebSphereMQJava>
    <WebSphereAppServer>/usr/WebSphere/AppServer </WebSphereAppServer>
  <WebLogic></WebLogic>
  </OS>
  <OS name="Linux">
    <DB2>/usr/IBMdb2/V7.1</DB2>
    <Oracle></Oracle>
    <WebSphereMQ>/opt/mqm</WebSphereMQ>
    <WebSphereMQJava>/opt/mqm/java</WebSphereMQJava>
    <WebSphereAppServer>/opt/WebSphere/AppServer </WebSphereAppServer>
    <WebLogic></WebLogic>
  </OS>
</DefaultInstallPath>
```


B

Konfigurationsdateien

Mit den TransactionVision-Setupdienstprogrammen werden Analyzer-Konfigurationsinformationen in den folgenden Konfigurationsdateien gespeichert. Sie können diese Dateien direkt ändern, wenn Sie Änderungen an der Konfiguration vornehmen müssen.

Dieses Kapitel umfasst die folgenden Themen:

- Analyzer.properties auf Seite 392
- CacheSize.properties auf Seite 398
- Database.properties auf Seite 399
- JobManager.properties auf Seite 403
- Sensor.properties auf Seite 403
- SensorConfiguration.xml auf Seite 403
- Setup.properties auf Seite 405
- StatisticsCache.properties auf Seite 406
- UI.properties auf Seite 407

Hinweis: Wenn Sie Eigenschaftendateien ändern, müssen Sie die entsprechende Applikation neu starten, damit die Änderungen wirksam werden. Weitere Informationen zum Neustart finden Sie unter "Verwalten von Analyzer" auf Seite 89.

Analyzer.properties

Die Datei <TVISION_HOME>/config/services/Analyzer.properties enthält Konfigurationsinformationen allgemeiner Art, zur Erfassung und zur Analyse für TransactionVision Analyzer.

Allgemeine Eigenschaften

- ▶ **rmi_client_timeout.** Mit diesem Eintrag wird die Zeitüberschreitung in Minuten für den RMI-Client angegeben. Der Standardwert ist **2**.
- ▶ **logging_xml.** Gibt den Namen der von Analyzer verwendeten Konfigurationsdatei für die Protokollierung an. Der Standardwert ist **Analyzer.Logging.xml**.
- ▶ **trace.** Mit dieser Eigenschaft wird angegeben, ob die Analyzer-Ablaufprotokollierung aktiviert ist. Der Standardwert ist **ON**.
- ▶ **debug.** Mit dieser Eigenschaft wird angegeben, ob die Analyzer-Debug-Protokollierung aktiviert ist. Der Standardwert ist **OFF**. Aktivieren Sie diese Einstellung nur, wenn die Kundenunterstützung Ihnen dies empfiehlt.
- ▶ **service_jvm_flags.** Gibt zusätzliche JFM-Kennzeichen an, mit denen Analyzer ausgeführt werden soll.
- ▶ **service_additional_classpath.** Gibt die CLASSPATH-Einstellungen für benutzerdefinierte Beans an, die mit Analyzer verwendet werden sollen.
- ▶ **jvm_dll**
 - service_classpath**
 - service_libpath.** Unter Windows werden diese Eigenschaften automatisch von **TVisionSetupInfo** erzeugt. Es sollte nicht erforderlich sein, sie zu ändern. Falls Sie sie ändern, beachten Sie, dass sie bei einer erneuten Ausführung von **TVisionSetupInfo** zurückgesetzt werden. Diese Eigenschaften haben keine Auswirkungen auf die TransactionVision-Umgebung. Sie bewirken lediglich, dass Analyzer als Windows-Dienst ausgeführt wird.

- **analyzer_port.** Mit diesem Eintrag wird die standardmäßige Portnummer für Analyzer angegeben. Der Standardwert ist **21100**. Beim Start verwendet Analyzer immer diese Portnummer. Mit dem Befehl **ServicesManager.[bat|sh] -rmiregp** kann die Portnummer für eine Analyzer-Instanz festgelegt werden, die bereits ausgeführt wird.
- **rmi_server_port.** Mit diesem Eintrag wird die standardmäßige Portnummer angegeben, die vom RMI-Serverobjekt abgehört wird.
Der Standardwert ist **0**. Dieser Wert bedeutet, dass die Java-Runtime bei jedem Start von Analyzer einen beliebigen, nach dem Zufallsprinzip ausgewählten Port verwendet. Wenn Sie Analyzer hinter einer Firewall ausführen und ein fester Port erforderlich ist, können Sie diese Eigenschaft auf einen bestimmten Portwert festlegen.
- **time_server.** Wird auf **on** festgelegt, damit der Zeitserver von Analyzer ausgeführt wird. Der Zeitserver ist für TIBCO EMS- und allgemeine JMS-Kommunikationsverbindungen erforderlich, um die Zeitabweichungsinformationen zu berechnen. Sie können den Zeitserver auf einem anderen Host ausführen, indem Sie das Dienstprogramm **TimeServer.[sh|bat]** verwenden.
- **time_server_port.** Gibt den für den Zeitserver zu verwendenden Listenerport an, falls für **time_server** der Wert **on** festgelegt ist. Der Standardport ist **9037**.

Erfassungseigenschaften

Die folgenden Eigenschaften werden für Vorgänge der Ereigniserfassung von Analyzer verwendet:

- **batch_commit_count.** Gibt an, wie viele Ereignisse in einen Batch aufgenommen werden, bevor es übergeben wird. Der Standardwert ist **50**.
- **commit_time_threshold.** Gibt die Zeitspanne in Sekunden an, nach der ein Commit erzwungen wird. Der Standardwert ist **1**.
- **commit_byte_threshold.** Gibt die Anzahl von Bytes an, nach der ein Commit erzwungen wird. Der Standardwert ist **1.000.000**.

- ▶ **write_to_buffer_table.** Gibt an, ob zusätzlich zur normalen Verarbeitung eine Kopie der Rohereignisse aus der Warteschlange in die Tabelle RAW_EVENT geschrieben wird. Der Standardwert ist **False**. Die Einstellung gilt für alle Projekte, anders als die entsprechende Einstellung auf der Seite **Communication Link** der Benutzeroberfläche, mit der das Schreiben von Rohereignissen für eine bestimmte Kommunikationsverbindung aktiviert werden kann. Diese Eigenschaft sollte nicht auf **True** festgelegt werden, wenn auch **read_from_buffer_table** auf **True** festgelegt ist. Mit dieser Eigenschaft können Sie benutzerdefinierte Beans testen oder Probleme beheben, sofern die Kundenunterstützung Sie dazu auffordert. In einer Produktionsumgebung sollte sie immer auf **False** festgelegt werden.
- ▶ **read_from_buffer_table.** Gibt an, ob Ereignisse nicht nur aus der Warteschlange gelesen, sondern auch aus der Tabelle RAW_EVENT abgerufen werden sollen. Der Standardwert ist **False**. Diese Eigenschaft sollte nicht auf **True** festgelegt werden, wenn auch **write_to_buffer_table** auf **True** festgelegt ist. Mit dieser Eigenschaft können Sie benutzerdefinierte Beans testen oder Probleme beheben, sofern die Kundenunterstützung Sie dazu auffordert. In einer Produktionsumgebung sollte sie immer auf **False** festgelegt werden.
- ▶ **keep_events.** Gibt an, ob Rohereignisse nach einer erfolgreichen Verarbeitung von Analyzer gelöscht werden sollen. Legen Sie die Eigenschaft auf **true** fest, damit die Rohereignisse in der Tabelle RAW_EVENT verbleiben. Deren Spalte **event_status** wird auf PROCESSED festgelegt, damit sie nicht noch einmal verarbeitet werden. Wenn Sie die Ereignisse noch einmal verarbeiten möchten (beispielsweise mit einer benutzerdefinierten Java-Bean für eine andere Form der Analyse), müssen Sie **event_status** erst auf NEW festlegen. Legen Sie die Eigenschaft auf **false** (den Standardwert) fest, damit Rohereignisse nach der Verarbeitung von Analyzer aus der Tabelle RAW_EVENT gelöscht werden.
- ▶ **write_to_jar.** Ähnelt **write_to_buffer_table**, jedoch werden die Rohereignisse von Analyzer in einer JAR-Datei und nicht in einer Tabelle in der Datenbank gespeichert. Der Speicherort der JAR-Datei ist **\$TVISION_HOME/logs** und der Dateiname lautet **SCHEMA_raw_events.jar**. Für jedes Projektschema wird eine separate JAR-Datei erstellt. Sie müssen das Projekt beenden, damit Sie die JAR-Datei richtig schließen können.

- ▶ **read_from_jar.** Ähnelt **read_from_buffer_table**, jedoch werden die Rohereignisse von Analyzer aus einer JAR-Datei und nicht aus einer Tabelle in der Datenbank gelesen. Bei einem Projekt mit dem Datenbankschema **SCHEMA** sucht Analyzer nach einer Datei **SCHEMA_raw_events.jar** im Verzeichnis **<TVISION_HOME>/logs**. Sie können die JAR-Datei kopieren, während Analyzer ausgeführt wird. Das Verzeichnis wird regelmäßig von Analyzer auf die entsprechenden Daten durchsucht. Sobald Analyzer mit der Verarbeitung der Ereignisse in der JAR-Datei beginnt, wird sie in **SCHEMA_raw_events.jar.processed.TIMESTAMP** umbenannt. Sie sollten diese Einstellung nicht auf **True** festlegen, wenn für **write_to_jar** die Einstellung **True** festgelegt ist.
- ▶ **fail_safe_collection_shutdown.** Bei **true** beendet Analyzer das Senden von Konfigurationsnachrichten, wenn bei der Ereigniserfassung und -verarbeitung ein schwerwiegender Fehler auftritt. Der Standardwert ist **true**.
- ▶ **jdbc_batching.** Der Standardwert **on** für **jdbc_batching** führt dazu, dass TransactionVision Datenbankankweisungen im Batch-Modus ausführt.
- ▶ **jdbc_batch_count.** Gibt die Anzahl von SQL-Operationen in Batches für den JDBC-Modus der Batch-Verarbeitung an. Diese Anzahl sollte der Eigenschaft **batch_commit_count** entsprechen oder ein Mehrfaches davon sein. Der Standardwert ist **50**.

Analyseeigenschaften

Die folgenden Eigenschaften werden für Vorgänge der Ereignisanalyse von Analyzer verwendet:

- ▶ **save_event_document.** Gibt an, ob Ereignisdokumente in der Datenbank gespeichert werden sollen. Der Standardwert ist **on**. Wird diese Eigenschaft auf **off** festgelegt, stehen keine Ereignisdetails zur Verfügung. Darüber hinaus kann Analyzer keine asynchron gelöschten Transaktions- und statischen Topologiedaten wiederherstellen. Dies bedeutet, dass nach einem Absturz oder einer anderen nicht normalen Programmbeendigung die Transaktionsdaten und die Statistikdaten für die statische Topologieansicht möglicherweise nicht mehr genau sind.
- ▶ **eventmatching_interval.** Gibt die Zeitspanne in Sekunden an, die zwischen einzelnen Aufrufen des Ereignisabgleichs gewartet wird. Der Standardwert ist **600**.

- **partial_event_lifetime.** Die maximale Dauer in Minuten, die ein unvollständiger Ereigniseintrag in der Tabelle **partial_event** vorhanden sein kann. Wenn diese Dauer erreicht ist, wird das unvollständige Ereignis gelöscht. Der Standardwert liegt bei 10 Minuten.
- **latency_resolution.** Die Auflösung, die für die Berechnung der Latenz zwischen Ereignissen verwendet wird. Mögliche Werte:

Wert	Beschreibung
1	Sekunden
10	1/10 Sekunden
100	1/100 Sekunden (Standard)
1000	Millisekunden

- **generate_api_only_txns.** Standardmäßig führt Analyzer keine Transaktionsanalyse für Ereignisse durch, die mit einem DataCollection-Filter **API only** erfasst wurden. Legen Sie diese Eigenschaft auf **true** fest, um dieses Verhalten zu ändern.
- **correlation_limit.** Mit dieser Einstellung wird die Anzahl von Ereignisbeziehungen begrenzt, die von Analyzer für einen bestimmten Korrelationschlüssel erstellt werden. Sie kann als Schutz vor einem fehlerhaften Korrelationsregelentwurf bei der benutzerdefinierten Korrelation dienen, der zu Leistungsproblemen führt. Der Standardwert ist **-1** (unbegrenzt).
- **local_txn_limit.** Mit dieser Einstellung wird die Anzahl von lokalen Transaktionen begrenzt, die von Analyzer einer Geschäftstransaktion zugewiesen werden. Sie kann als Schutz vor einem fehlerhaften lokalen Transaktionsregelentwurf bei der benutzerdefinierten lokalen Transaktionsanalyse dienen, der zu Leistungsproblemen führt. Der Standardwert ist **-1** (unbegrenzt).
- **separate_child_thread_txns.** Wenn ein Servlet einen Thread ausgliedert, um JMS-Aufrufe abzusetzen, übergibt das Servlet standardmäßig Verfolgungsinformationen an den untergeordneten Thread. Dies führt dazu, dass sowohl Servlet- als auch JMS-Ereignisse zu derselben Geschäftstransaktion gehören. Mitunter kann es jedoch gewünscht sein, diese Ereignisse in unterschiedlichen Transaktionen zu trennen. So könnte

beispielsweise aus einem Servlet ein lang ausgeführter Thread ausgegliedert werden, der jedoch nicht zur selben Transaktion wie das Servlet gehören soll. Wenn von einem Servlet ausgegliederte Threads in dieselbe Geschäftstransaktion wie das Servlet aufgenommen werden sollen, geben Sie den Servlet-Programmnamen als Wert dieser Eigenschaft an. Mehrere Programmnamen werden durch ein Komma getrennt, wie im folgenden Beispiel veranschaulicht:

```
separate_child_thread_txns=programm1, programm2
```

- ▶ **enable_dbcaching.** Aktiviert den Standardmodus zusätzlich zum oder anstelle des Fehlermodus. Wenn in allen Projekten nur der Fehlermodus verwendet wird, kann der Standardmodus deaktiviert werden, um Systemressourcen zu sparen.
- ▶ **enable_failure_mode.** Aktiviert den Fehlermodus, sodass von Analyzer nur Ereignisdaten für fehlgeschlagene Geschäftstransaktionen (oder Transaktionen, die einen SLA verletzen) gespeichert werden. Für erfolgreiche Geschäftstransaktionen werden nur die entsprechenden Geschäftstransaktionszeilen gespeichert. Da für diesen Modus ein aussagekräftiger Wert für das Attribut **result** der Geschäftstransaktion sehr wichtig ist, muss dafür eine Transaktionsklassifikation mit geeigneten Regeln für die Spalte **result** implementiert sein. Der Fehlermodus kann pro Projekt aktiviert werden. So können bestimmte Projekte im Fehlermodus ausgeführt werden, während andere im herkömmlichen Standardmodus ausgeführt werden. Der Standardwert ist **off**. Weitere Informationen zum Fehlermodus finden Sie unter "Failure Mode" auf Seite 74 im Handbuch *Using TransactionVision*.
- ▶ **dbcache_thread_count.** Gibt die Anzahl der Threads an, die zum Schreiben der Analysedaten im Cache in die Datenbank verwendet werden. Der optimale Wert hängt von der Hardware ab, mit der Analyzer ausgeführt wird. Er kann nur durch Leistungsmessungen mit anderen Werten bestimmt werden. Ein guter Ausgangspunkt wäre, die Anzahl auf die Hälfte der Erfassungsthreads festzulegen, die für ein bestimmtes Schema verwendet werden. Der Standardwert ist **2**.
- ▶ **failure_mode_thread_count.** Definiert die Anzahl von Löschthreads, die für den Fehlermodus verwendet werden. Diese Anzahl sollte so hoch festgelegt werden, sodass die Daten im Cache schnell genug gelöscht werden. Gleichzeitig sollte sie nicht zu hoch festgelegt werden, damit nicht zu viele Systemressourcen verschwendet werden. Der Standardwert ist **2**.

- ▶ **failure_mode_process_delay.** Definiert die Zeitspanne in Millisekunden, die dem SLA-Wert zur Bestimmung der Zeitüberschreitung beim Veralten hinzugefügt wird. Dies ist ein entscheidender Wert für den Fehlermodus. Ist dieser Wert zu niedrig festgelegt, veralten viele Daten erfolgreicher Transaktionen unnötigerweise und werden in die Datenbank geschrieben. Dies führt zu einer Leistungsverschlechterung und zur Verschwendung von Speicherplatz für die Datenbank. Ist dieser Wert zu hoch festgelegt, werden zu viele Daten in Caches des Speichers abgelegt. Reichen die Cache-Größen für die Systemlast nicht aus, kommt es häufig zu Cache-Überläufen und somit zu einer sehr schlechten Leistung. Der Standardwert liegt bei 5 Sekunden.
- ▶ **failure_mode_discard_overflow.** Bei einem Cache-Überlauf werden alle Daten in den Caches normalerweise in die Datenbank geschrieben. Wurde der Überlauf jedoch durch eine länger anhaltende Verlangsamung des Systems verursacht, kann das ständige Löschen des Caches die Situation noch weiter verschlechtern und die Leistung von Analyzer weiter beeinträchtigen. Wenn in solchen Situationen vorrangig versucht wird, das Auffüllen der Ereigniswarteschlange zu vermeiden, können Sie den oben angegebenen Parameter auf **true** festlegen und das Verwerfen aller Daten im Cache bei einem Überlauf erzwingen. Der Standardwert ist **false**.

CacheSize.properties

Mit der Datei <TVISION_HOME>/config/services/CacheSize.properties wird die Größe der Caches angegeben, die für die Ereignisanalyse verwendet werden. Größere Caches minimieren den Datenbankzugriff und steigern die Leistung. Dadurch wird aber auch die von TransactionVision genutzte Speichermenge erhöht. Mit der Datei werden die folgenden Eigenschaften festgelegt:

- ▶ **system_model_objects.** Gibt die Anzahl von Systemmodellobjekten, außer PII, im Cache an.
- ▶ **pii_objects.** Gibt die Anzahl von PII-Systemmodellobjekten im Cache an.
- ▶ **event_based.** Gibt die Anzahl von Ereignissen im Cache an.
- ▶ **transaction_based.** Gibt die Anzahl von Transaktionen im Cache an.

Database.properties

Mit der Datei <TVISION_HOME>/config/datamgr/Database.properties wird die Datenbank angegeben, die von Analyzer und TransactionVision UI/Job Server gelesen wird und in die diese Komponenten schreiben.

Erforderliche Einträge

Die folgenden Pflichteinträge sind erforderlich:

- ▶ **jdbc_driver=COM.ibm.db2.jdbc.app.DB2Driver.** Dieser Eintrag ist der Klassenname des verwendeten JDBC-Treibers. Der Standardwert für den IBM DB2-JDBC-Treiber ist oben angegeben. Für Oracle ist der Standardwert **oracle.jdbc.driver.OracleDriver** und für SQL Server ist es **com.microsoft.jdbc.sqlserver.SQLServerDriver**.
- ▶ **database_connection_name.** Für DB2 ist dieser Eintrag der Name der zu verwendenden Datenbankverbindung (normalerweise der Datenbankalias). Für Oracle ist dieser Eintrag der Datenbankname für eine OCI-Clientverbindung oder der SID (System Identifier, Systembezeichner) für eine Thin Client-Verbindung. Für SQL Server wird dieser Eintrag nicht verwendet.
- ▶ **database_name.** Für DB2 ist dieser Eintrag der Name der Datenbank auf dem Server, mit dem eine Verbindung hergestellt werden soll. Dieser Name kann von dem Namen für **database_connection_name** abweichen, wenn eine Client-Datenbankverbindung verwendet wird. Für Oracle ist dieser Eintrag der SID. Für SQL Server ist dies der Name der Datenbank.
- ▶ **database_host.** Dieser Eintrag ist der Host, auf dem der Datenbankserver ausgeführt wird.
- ▶ **database_port.** Für Oracle gibt dieser Eintrag die Portnummer des Oracle-Listeners für den Zielhost an. Der Standardwert ist **1521**. Dieses Attribut wird nur verwendet, wenn Sie den Thin Client von Oracle nutzen. Weitere Informationen zur Verwendung von Oracle mit TransactionVision finden Sie in Kapitel 6, "Konfigurieren von Datenbanken.". Für SQL Server wird mit diesem Eintrag der Port für die Verbindung mit dem Server angegeben. Der Standardwert ist **1433**.

- ▶ **oracle_client_type.** Gibt an, ob der Treiber für den Thin Client von Oracle oder der JDBC-Treiber verwendet wird. Legen Sie diese Einstellung für Oracle 9 auf **thin** oder **oci** bzw. für Oracle 8.1.7 auf **oci8** fest. Der Standardwert ist **thin**. Dieses Attribut wird nur verwendet, wenn Sie eine Oracle-Datenbank nutzen. Weitere Informationen zur Verwendung von Oracle mit TransactionVision finden Sie in Kapitel 6, "Konfigurieren von Datenbanken".
- ▶ **oracle_user_password.** Mit diesem Eintrag wird das Benutzerkennwort angegeben, das für den Oracle-Benutzer festgelegt wird, der für ein neues Projektschema erstellt wird. Der Standardwert ist **ABC99DEF**. Sie können diesen Wert ändern, wenn Sie bestimmte Kennwortrichtlinien einhalten müssen.
- ▶ **db2_instance_env.** Mit diesem Eintrag wird der Wert der DB2-Umgebungsvariablen DB2INSTANCE angegeben.

Optionale Einträge, die die automatische Erkennung überschreiben

Die folgenden optionalen Einträge können in dieser Datei eingerichtet werden. Diese Einträge überschreiben die automatische Erkennung.

- ▶ **connection_type.** Dieser Eintrag definiert, wie JDBC-Verbindungen abgerufen werden. Verwenden Sie einen der folgenden Werte:
 - ▶ **JDBC:** Verwendet Verbindungen, die vom JDBC-Treiber abgerufen werden, ohne Treiberverbindungs pooling (Standard).
 - ▶ **DB2DataSource:** Verwendet DB2-Verbindungs pooling.
 - ▶ **OracleDataSource:** Verwendet Oracle-Verbindungs pooling.
 - ▶ **JNDI:** Verwendet die in JNDI registrierte Datenquelle. Damit Sie eine JNDI-Verbindung verwenden können, müssen Sie auch die Eigenschaft **jni_url** festlegen.
- ▶ **database_url.** Mit diesem Eintrag wird ein benutzerdefinierter URL definiert, der für den JDBC-Treiber-Manager verwendet wird.
- ▶ **jni_url.** Mit diesem Eintrag wird der JNDI-Name für die Datenbank (definiert in WebSphere) definiert, wenn JNDI-Verbindungen verwendet werden.

- **unicode_db.** Ist diese Eigenschaft festgelegt, werden alle zeichenbasierten XDM-Spalten, für die das Attribut **unicode=true** festgelegt ist, mit der doppelten Bytegröße erzeugt, um das Speichern der angegebene Anzahl an Zeichen in der Datenbank zu ermöglichen.
- **unicode_bytes_per_character.** Ist diese Eigenschaft festgelegt, werden alle zeichenbasierten XDM-Spalten mit dem Attribut **unicode_true** mit einer Größe erzeugt, für die dieser Wert als Multiplikator des Basiswerts verwendet wird. Der Standardwert **2** entspricht dem Verhalten der Einstellung **unicode_db=true**, ohne dass dieser Wert festgelegt wird. Werte über 3 können zu Problemen bei der Datenbankerstellung führen. Diese Eigenschaft ist nur wirksam, wenn auch **unicode_db=true** festgelegt wurde.
- **jdbc_url.** Mit diesem Eintrag wird der zu verwendende Datenbanktreiber angegeben, wenn dies nicht der DB2 Universal JDBC-Treiber (Typ 4) oder der Thin Client-Treiber für Oracle ist.

Für den DB2 Universal Driver (Typ 2) hat dieser Eintrag das folgende Format:

```
jdbc:db2:<Name_der_Datenbank>
```

Dabei ist <Name_der_Datenbank> die Datenbank, mit der Analyzer eine Verbindung herstellt.

Für den Oracle OCI-Treiber hat dieser Eintrag das folgende Format:

```
jdbc:oracle:oci:<Benutzer>/<Kennwort>@<Name_der_Datenbank>
```

Dabei ist <Name_der_Datenbank> die Datenbank, mit der Analyzer eine Verbindung herstellt, und <Benutzer> und <Kennwort> sind der Benutzername und das Kennwort für die Oracle-Verbindung.

Optionale Einträge mit Standardwerten

Die folgenden optionalen Einträge können in dieser Datei eingerichtet werden. Werden Sie nicht angegeben, werden die Standardwerte verwendet.

- **user.** Dieser optionale Eintrag ist der Benutzername, der für die Herstellung der Datenbankverbindung verwendet werden soll. Wenn dieses Feld leer ist, wird die Datenbankverbindung für den momentan angemeldeten Benutzer hergestellt. Stellen Sie sicher, dass der angegebene oder momentan angemeldete Benutzer berechtigt ist, auf die Datenbank zuzugreifen.

- **passwd.** Dieser optionale Eintrag ist das Kennwort, das für die Herstellung der Datenbankverbindung verwendet werden soll. Wenn dieses Feld leer ist, wird das Kennwort des momentan angemeldeten Benutzers verwendet, um die Datenbankverbindung herzustellen.
- **jce_provider.** Dieser optionale Eintrag gibt den Namen des JCE-Pakets (Java Cryptographic Extension) für die Verschlüsselung des Datenbankkennworts an. In der folgenden Tabellen werden die gültigen Paketnamen aufgeführt. Ist dieser Eintrag leer, speichert TransactionVision das Kennwort als reinen Text.

Provider	Paketname
Sun	com.sun.crypto.provider.SunJCE
IBM	com.ibm.crypto.provider.IBMJCE

Das Dienstprogramm **TVisionSetupInfo** sucht automatisch nach einem installierten JCE-Provider, der den DES-Verschlüsselungsalgorithmus unterstützt. Wird ein solcher Provider gefunden, legt **TVisionSetupInfo** den Wert für **jce_provider** auf den Klassennamen des JCE-Providers fest. Wird kein JCE-Provider gefunden, wird von **TVisionSetupInfo** die folgende Meldung angezeigt:

Java Cryptography Extension (JCE) is not present in your current JDK.
Password encryption feature will be disabled and stored in plain text.

- **reconnect_interval.** Mit diesem Eintrag wird die Frequenz in Sekunden angegeben, mit der von TransactionVision versucht werden soll, erneut eine Verbindung mit der Datenbank herzustellen. Der Standardwert ist **10**.
- **reconnect_timeout.** Mit diesem Eintrag wird die Zeitdauer in Sekunden angegeben, in der von TransactionVision versucht werden soll, erneut eine Verbindung mit der Datenbank herzustellen (mit der Frequenz von **reconnect_interval**). Der Standardwert ist **600**.
- **query_timeout.** Mit diesem Feld wird für Abfragen, die in der Benutzeroberfläche ausgeführt werden, die Zeitüberschreitung in Sekunden angegeben. Diese Einstellung ist standardmäßig deaktiviert.

JobManager.properties

Mit der Datei <TVISION_HOME>/config/jobs/JobManager.properties werden Konfigurationsinformationen für TransactionVision-Jobs angegeben. In dieser Datei werden die folgenden Eigenschaften angegeben:

- ▶ **BeansXMLFile.** Gibt den Speicherort der Konfigurationsdatei mit der Liste der Jobvorlagen an, die beim Erstellen eines neuen Projekts zur Verfügung stehen.
- ▶ **logSize.** Gibt die Anzahl der Einträge an, die in die Protokolldatei aufgenommen werden sollen.

Sensor.properties

Die Datei <TVISION_HOME>/config/sensor/Sensor.properties gibt Informationen über die Servlet- und JMS-Sensoren an. Die Datei weist die folgenden Einträge auf:

- ▶ **logging_xml.** Gibt den Namen der Konfigurationsdatei für die Protokollierung an.
- ▶ **configuration_file.** Gibt den Pfadnamen der XML-Konfigurationsdatei für Sensoren an.

SensorConfiguration.xml

Mit der Datei **SensorConfiguration.xml** wird die Interaktion zwischen dem Sensor und der Konfigurationswarteschlange definiert. Es werden die folgenden Attribute definiert, die beim Erstellen oder Bearbeiten einer Kommunikationsverbindung angegeben werden.

Attribut	Beschreibung
ConfigurationQM	Der Name des Konfigurationswarteschlangen-Managers
ConfigurationQ	Der Name der Konfigurationswarteschlange

Attribut	Beschreibung
ConfigurationQMHost	Der Hostname des Konfigurationswarteschlangen-Managers
ConfigurationQMPort	Die Nummer des Listenerports des Konfigurationswarteschlangen-Managers
ConfigurationQMChannel	Der Kanalname des Konfigurationswarteschlangen-Managers
ConnectionRetryDelay	Die Verzögerung in Millisekunden zwischen Verbindungsversuchen, wenn die Verbindung mit dem Konfigurationswarteschlangen-Manager unterbrochen wurde. Der Standardwert ist 1000 .
ConnectionRetryTimeout	Die Zeitspanne in Millisekunden, die versucht wird, erneut eine Verbindung mit dem Konfigurationswarteschlangen-Manager herzustellen, nachdem sie unterbrochen wurde. Mit dem Wert -1 (dem Standardwert) wird es immer wieder versucht.
ConfigurationRetrieveInterval	Das Zeitintervall in Millisekunden, in dem die Konfigurationswarteschlange auf neue Konfigurationsnachrichten überprüft wird. Der Standardwert ist 10000 .
SensorClientTimeSkewInterval	Das Zeitintervall in Millisekunden, in dem die Zeitabweichung zwischen dem Host mit dem Sensor und dem Host mit dem Konfigurationswarteschlangen-Manager überprüft wird. Der Standardwert ist 300000 (5 Minuten).

Attribut	Beschreibung
EventPackageFlushTimeout	Die Zeitdauer in Millisekunden, die ein Ereignispaket im Leerlauf bleibt (keine Ereignisse werden hinzugefügt), bevor das Schreiben in die Ereigniswarteschlange erzwungen wird. Der der Mindestwert ist 10000 , Standardwert ist 300000 (5 Minuten).
RepeatLogInterval	Die Zeitdauer in Millisekunden, die ein Wiederholungsfehler, der normalerweise unterdrückt werden würde, wiederholt wird. Der Standardwert ist 600000 .

Setup.properties

Mit der Datei <TVISION_HOME>/config/setup/Setup.properties werden die folgenden Eigenschaften angegeben:

- **default_tool_install_path.** Der Verzeichnispfad der Datei **DefaultInstallPath.xml**, in der die Speicherorte der Softwarekomponenten aufgeführt sind, die für TransactionVision erforderlich sind.
- **logs_dir.** Der Name des Verzeichnisses, in dem die Protokolldateien gespeichert werden sollen.
- **logging_xml.** Der Name der Konfigurationsdatei für die Protokollierung.
- **minimum_java_version.** Die Java-Version, die für TransactionVision mindestens erforderlich ist.
- **minimum_java_version_sun.** Die Java-Version, die für TransactionVision auf der Solaris-Plattform mindestens erforderlich ist.
- **maximum_java_version.** Die höchste Java-Version, die von TransactionVision unterstützt wird.

StatisticsCache.properties

Mit der Datei <TVISION_HOME>/config/services/StatisticsCache.properties werden die Einstellungen definiert, die im statischen Modus der Komponententopologieanalyse verwendet werden. Die Datei enthält die folgenden Einträge:

- ▶ **flush_interval.** Gibt an, wie häufig der Cache für Statistiken in die Datenbank geschrieben wird. Die Daten werden erst im Diagramm dargestellt, nachdem sie in die Datenbank geschrieben wurden. Abhängig vom Umfang eingehender Anforderungen kann es sinnvoll sein, die Leerung häufiger oder weniger häufig vorzunehmen. Bei einer größeren Zeitspanne zwischen Leerungen wird seltener in die Datenbank geschrieben, aber die Größe des Caches nimmt zu.
- ▶ **timeslice_interval.** Gibt in Minuten die Dauer der Zeitscheibe an, für die Ereignisstatistiken berechnet werden sollen. Das Konzept der Zeitscheibe wird verwendet, um Daten in verschiedene Teile zu partitionieren, sodass die Daten, je nach gewünschtem Zeitrahmen, schnell abgerufen werden können. Eine Zeitscheibe steht für einen Zeitraum, für den alle Ereignisse, die zum gleichen Programm- und MQ-Objekt gehören, in einer Statistik zusammengefasst werden. Wenn z. B. ein Programm Nachrichten in eine bestimmte Warteschlange eingestellt hat, werden alle Statistikdaten im Zusammenhang mit dem Programm und der Warteschlange (Latenz, Erfolgsanzahl usw.) während einer bestimmten Zeitscheibe in derselben Datenbankzeile gespeichert. Indem Sie die Werte für das Zeitscheibenintervall ändern, können Sie steuern, wie umfangreich und wie effizient der Zugriff auf die Statistiktabelle ist. Je größer die Zeitscheibe, desto effizienter die Speicherung der Ergebnisse. Bedenken Sie dabei jedoch, dass das Zeitscheibenintervall den kleinsten Zeitabschnitt darstellt, für den Sie Ihre Daten anzeigen können. Wenn Sie beispielsweise als Zeitscheibe einen Tag festlegen, können Sie diese Statistiken nicht auf Stundenbasis anzeigen.

UI.properties

Mit der Datei <TVISION_HOME>/config/ui/UI.properties werden die Eigenschaften angegeben, die für die TransactionVision-Webbenutzeroberfläche verwendet werden. Die Datei enthält die folgenden Einträge:

- ▶ **ui_config_dir.** Gibt den Speicherort der Konfigurationsdateien für die Benutzeroberfläche an.
- ▶ **view_config_filename.** Gibt die Datei mit den Definitionen der TransactionVision-Ansichten an.
- ▶ **reports_xml_filename.** Gibt den Speicherort der Datei mit der Liste der verfügbaren Reports an.
- ▶ **category_xml_filename.** Gibt den Speicherort der Konfigurationsdatei für Abfragen an.
- ▶ **category_xsl_filename.** Gibt den Speicherort des Stylesheets für Abfragen an.
- ▶ **filter_config_xml.** Gibt den Speicherort der Konfigurationsdatei für Datenerfassungsfiler an.
- ▶ **filter_config_xsl.** Gibt den Speicherort des Stylesheets für Datenerfassungsfiler an.
- ▶ **detail_modifier_xsl.** Gibt den Speicherort des Stylesheets für die Ansicht **Event Detail** an.
- ▶ **logging_xml.** Gibt den Speicherort der Protokolldatei für die Benutzeroberfläche an.
- ▶ **support_url.** Der URL für Produktunterstützung.
- ▶ **autologin.** Aktiviert das Kontrollkästchen **Remember Login** auf der TransactionVision-Anmeldeseite. Wenn dieses Kontrollkästchen aktiviert ist, werden der Benutzername und das Kennwort als Cookie gespeichert. Die Benutzer müssen beim nächsten Zugriff auf TransactionVision den Benutzernamen und das Kennwort also nicht angeben. Das Kennwort wird mit dem angegebenen Java JCE-Provider verschlüsselt.
- ▶ **jceProvider.** Gibt den Namen des JCE-Pakets an, das zum Verschlüsseln des Kennworts verwendet wird, wenn **autologin** aktiviert ist. Wenn Sie nicht über ein JCE-Paket verfügen, können Sie es von **java.sun.com** herunterladen.

- **trace.** Aktiviert die Ablaufprotokollierung für die TransactionVision-Webbenutzeroberfläche. Weitere Informationen finden Sie unter "Verwenden von Windows- und UNIX-Systemprotokollen" auf Seite 108.
- **logout_Redirect.** Diese Eigenschaft gibt den URL an, der angezeigt werden soll, wenn sich ein Benutzer von TransactionVision abmeldet, beispielsweise eine SSO-Abmeldeseite.
- **dashboard_xml_filename.** Der Name der Konfigurationsdatei für Dashboard-Reports.
- **hasProxySensor.** Wird auf **True** festgelegt, damit in der dynamischen Ansicht **Component Topology Analysis** proxy-bezogene Links angezeigt werden, wenn Sie den Proxy-Sensor verwenden.
- **bac_lwssso_enabled.** Wird auf **False** festgelegt, wenn LW-SSO in Business Availability Center deaktiviert ist. So können Sie auf TransactionVision zugreifen, ohne dass LW-SSO erforderlich ist.

Weitere Informationen zu LW-SSO finden Sie im Handbuch *HP Business Availability Center Hardening Guide* (PDF)

C

Datenbankmigration

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Dauer und Speicherplatzanforderungen auf Seite 409
- ▶ Deaktivieren nicht verwendeter Integrationsspalten auf Seite 410
- ▶ Migration benutzerdefinierter Datenbankschemas auf Seite 410
- ▶ Datenbankmigration – Technische Details auf Seite 410
- ▶ Optimieren von TransactionVision in Umgebungen ohne Integration auf Seite 413

Dauer und Speicherplatzanforderungen

TransactionVision, Version 7.50, umfasst wesentliche Änderungen am Datenbankschema, die beim Migrationsprozess Auswirkungen auf dessen Dauer und die erforderlichen Ressourcen haben. Die meisten Tabellendaten eines Projekts vor 7.50 müssen während der Migration kopiert werden. Aufgrund der ausfallsicheren Implementierung des Migrationsprozesses ist dafür zusätzlicher freier Speicherplatz erforderlich, der dem Speicherplatz für das größte vorhandene Projekt in der Datenbank entspricht. Wenn z. B. für das größte TransactionVision-Projekt 2 GB Speicherplatz erforderlich sind, sind für eine erfolgreiche Migration weitere 2 GB notwendig. Anhängig von der Datenmenge in der Datenbank kann die Migration darüber hinaus eine beträchtliche Zeit dauern.

Deaktivieren nicht verwendeter Integrationsspalten

Das neue Datenbankschema enthält einige neue Tabellenspalten, die für die Integration von TransactionVision in andere Produkte der Business Availability Center-Suite verwendet werden. Wenn Sie TransactionVision in einer leistungskritischen Umgebung einsetzen und die Integrationsmöglichkeiten nicht nutzen möchten, können Sie die Erstellung dieser neuen Spalten im Migrationprozess vermeiden. Führen Sie das Skript **MigrateDB** (siehe Anhang A, "Referenz der Dienstprogramme.") mit der folgenden Befehlszeilenoption aus:

```
MigrateDB.sh|.bat -disableMigration,
```

Migration benutzerdefinierter Datenbankschemas

Wenn Sie in der vorherigen TransactionVision-Installation benutzerdefinierte XDM-Definitionsdateien hinzugefügt haben, bleiben diese in der neuen Konfiguration erhalten. Sie müssen jedoch für benutzerdefinierte Ereignistabellen das Schlüsselfeld **proginst_id** von INTEGER in BIGINT und für benutzerdefinierte Transaktionstabellen das Schlüsselfeld **business_trans_id** von INTEGER in BIGINT ändern. Wenn Sie Spalten in den standardmäßigen XDM-Dateien hinzugefügt oder geändert haben, bleiben diese Änderungen in der Neuinstallation nicht erhalten. Speichern Sie die alten Konfigurationsdateien beim Setup und nehmen Sie die erforderlichen Aktualisierungen nach der Installation vor.

Datenbankmigration – Technische Details

Die folgende Liste enthält alle Änderungen am Datenbankschema seit TransactionVision, Version 5.00:

TVISION-Systemschema

- ▶ Tabelle SCHEMA_VERSION: Spalte **schema** wurde umbenannt -> **schema_name**.
- ▶ Tabelle ID_TABLE: Spalte **key** wurde umbenannt -> **key_name**.
- ▶ Tabelle SCHEMA wurde umbenannt -> SCHEMA_TABLE.

- Tabelle CLASSIFICATION wurde hinzugefügt.
- Tabelle CLASSIFICATION_REL wurde hinzugefügt.
- Tabelle TXN_CLASS_ATTRIBUTE wurde hinzugefügt.
- Tabelle TXN_CLASS_ATTR_VALUE wurde hinzugefügt.
- Tabelle PROPERTIES wurde hinzugefügt.

Projektschema

- Tabelle ID_TABLE: Spalte **key** wurde umbenannt -> **key_name**.
- Tabelle SCRATCH: Spalte **key** wurde umbenannt -> **key_name**.
- Für die folgenden Tabellen wurde der Spaltentyp von **proginst_id** von INTEGER in BIGINT geändert: PARTIAL_EVENT, EVENT, EVENT_OVERFLOW, USER_DATA, USER_DATA_OVERFLOW, RELATION_LOOKUP.
- In allen Tabellen, die über XDM-Dateien in \$TVISION_HOME/config/xdm mit **documentType="/Event"** definiert wurden, wurde der Spaltentyp von **proginst_id** von INTEGER in BIGINT geändert.
- In Tabelle LOCAL_TRANSACTION wurde der Spaltentyp von **local_trans_id** und **business_trans_id** von INTEGER in BIGINT geändert.
- In Tabelle TRACKING_OVERFLOW wurde der Spaltentyp von **local_trans_id** von INTEGER in BIGINT geändert.
- In Tabelle BUSINESS_TRANSACTION wurde der Spaltentyp von **business_trans_id** von INTEGER in BIGINT geändert.
- In Tabelle EVENT_RELATION wurde der Spaltentyp von **proginst_id** und **proginst_id2** von INTEGER in BIGINT geändert.
- In Tabelle SYS_MDL_OBJECT wurde der Spaltentyp von **object_id** von INTEGER in BIGINT geändert.
- In Tabelle SYS_MDL_OBJECT_RELATION wurde der Spaltentyp von **object_id** und **object_id2** von INTEGER in BIGINT geändert.
- Tabelle LOCAL_TRANSACTION: Spalte **key** wurde umbenannt -> **key_name**.
- Tabelle BUSINESS_TRANSACTION: Spalte **sequential_id** wurde umbenannt -> **update_id** und der Spaltentyp wurde von INTEGER in BIGINT geändert.

- Tabelle EVENT_LOOKUP: Spalte **sequential_id** wurde umbenannt -> **seq_id** und der Spaltentyp wurde von INTEGER in BIGINT geändert.
- Tabelle TRANSACTION_STATS: Spalten **begin** wurde umbenannt -> **begin_time** und **end** -> **end_time**.
- Tabelle JMS_LOOKUP: Spalte **exception_class** (INTEGER) wurde hinzugefügt.
- Tabelle BUSINESS_TRANSACTION: Spalte **exception** (INTEGER) wurde hinzugefügt.
- Die folgenden Spalten wurden zu Integrationszwecken SERVLET_LOOKUP hinzugefügt: **probe_id, probe_group_id, uri, bpm_profile_id, bpm_location_id, bpm_txn_id, client_ip, session_id, url** (Datentypen sind in **\$TVISION_HOME/config/xdm/Servlet.xdm** aufgeführt).
- Die folgenden Spalten wurden zu Integrationszwecken BUSINESS_TRANSACTION hinzugefügt: **trans_id, is_bpievent** (Datentypen sind in **\$TVISION_HOME/config/xdm/Transaction.xdm** aufgeführt).
- Der eindeutige Index (**tracking_id, tracking_seq**) für Tabelle TRACKING_OVERFLOW wurde in einen nicht eindeutigen Index geändert.
- Tabelle JDBC_LOOKUP wurde hinzugefügt.
- Tabelle JDBC_STATS wurde hinzugefügt.
- Tabelle SYS_MDL_OBJECT_ATTR wurde hinzugefügt.

Optimieren von TransactionVision in Umgebungen ohne Integration

TransactionVision 7.50 bietet viele neue Projekttabellenspalten für die Integration in andere Produkte der Business Availability Center-Produktfamilie. Wenn Sie die Integrationsmöglichkeiten nicht nutzen, werden während der Analyzer-Verarbeitung NULL-Werte in diese Spalte geschrieben. Wenn Ihr Datenbankprodukt Indexdateien aktualisiert, kann es auch bei NULL-Werten möglich sein, dass die Leistung von Analyzer im geringen Maße beeinträchtigt wird. Wenn Sie also TransactionVision Analyzer in einer Umgebung mit hoher Leistung ausführen und eine Integration in andere Produkte nicht beabsichtigen, sollten Sie die Integrationsspalten wie folgt deaktivieren:

- ▶ Führen Sie **TVisionSetupInfo.bat|.sh** über **\$TVISION_HOME/bin** mit der Befehlszeilenoption **-integrations** aus.
- ▶ Beantworten Sie die folgenden Fragen mit **n**:

```
-----
Integration Settings
-----
Retrieving current integration settings...
Notice:
  If you disable integrations, projects created afterwards may not work properly
  as the database columns won't exist. It's safer to enable them now and just
  disable them in the UI later.
IntegrationEnableBAC (y/n) [n]:
IntegrationEnableDiag (y/n) [n]:
```

So werden alle Integrationsspalten in den XDM-Dateien deaktiviert (derzeit **Servlet.xdm** und **Transaction.xdm**). Bei neuen Projektschemas werden die Tabellen ohne diese Spalten erstellt. Bei vorhandenen Projektschemas verbleiben die Spalten in den Tabellen, Analyzer greift aber nicht mehr darauf zu. Sie können diese Einstellung jederzeit ändern, indem Sie **TVisionSetupInfo** wieder mit der entsprechenden Befehlszeilenoption ausführen.

D

Zusätzliche z/OS-Einstellungen

Dieses Kapitel umfasst die folgenden Themen:

- RACF-Autorisierungen auf Seite 415
- Firewall-Einstellungen auf Seite 418
- MIPS-Anforderungen auf Seite 418

RACF-Autorisierungen

RACF-Autorisierungen werden in großem Umfang auf die Umgebung des Benutzers zugeschnitten, deshalb gibt es keine speziellen Anforderungen für die TransactionVision-Sensoren. Verwenden Sie die in den folgenden Tabellen aufgeführten Programme und Transaktionen, um die erforderlichen RACF-Autorisierungen zu bestimmen.

CICS-Sensor von TransactionVision

Transaktion	Muss ausgeführt werden	Benötigt Zugriff auf WebSphere MQ	Muss als globaler Benutzerexit von CICS ausgeführt werden
SLDS	Ja	Nein	Nein
SLDM	Ja	Nein	Nein
SLDP	Ja	Nein	Nein
SLDC	Ja	Nein	Nein
SLDD	Ja	Nein	Nein
SLDI	Ja	Nein	Nein

Transaktionsprogramm	Muss ausgeführt werden	Benötigt Zugriff auf WebSphere MQ	Muss als globaler Benutzerexit von CICS ausgeführt werden
SLDPCSX	Ja	Nein	Nein
SLDPCMXX	Ja	Nein	Nein
SLDPCCX	Ja	Nein	Nein
SLDPDSX	Ja	Nein	Nein
SLDPUXI	Ja	Nein	Nein
TVISION	Ja	Nein	Nein
TVISIONC	Ja	Ja	Nein

Exitprogramm	Muss ausgeführt werden	Benötigt Zugriff auf WebSphere MQ	Muss als globaler Benutzerexit von CICS ausgeführt werden
SLDPTCX	Ja	Nein	Ja
SLDPPSX	Ja	Nein	Ja

Exitprogramm	Muss ausgeführt werden	Benötigt Zugriff auf WebSphere MQ	Muss als globaler Benutzerexit von CICS ausgeführt werden
SLDPICX	Ja	Nein	Ja
SLDPTDX	Ja	Nein	Ja
SLDPPCX	Ja	Nein	Ja
SLDPFCX	Ja	Nein	Ja
SLDPTSX	Ja	Ja	Ja

CICS WebSphere MQ-Sensor (WMQ) von TransactionVision

Transaktion	Muss ausgeführt werden	Benötigt Zugriff auf WebSphere MQ	Muss den CICS-Steuerübergabeexit aktivieren/deaktivieren
SLMC	Ja	Nein	Ja

Transaktionsprogramm	Muss ausgeführt werden	Benötigt Zugriff auf WebSphere MQ	Muss den CICS-Steuerübergabeexit aktivieren/deaktivieren	Zugriff des CICS-Steuerübergabeexits ist erforderlich
CSQCAPX	Ja	Ja	Nein	Ja
SLMC	Ja	Ja	Ja	Nein
SLMBCNFG	Ja	Ja	Nein	Nein

IMS WebSphere MQ-Sensor (WMQ) von TransactionVision

Der IMS WMQ-Sensor von TransactionVision wird über die IMS WMQ-Applikation ausgeführt, die überwacht wird. Dazu wird über eine Verbindung ein TransactionVision-Stub in Abhängigkeit von der IMS WMQ-Applikation bearbeitet. Aus diesem Grund müssen möglicherweise einige RACF-Autorisierungen vorgenommen werden.

Die folgenden Bibliotheken werden verwendet und müssen möglicherweise auch für RACF autorisiert werden, dies hängt vom lokalen RACF-Autorisierungsschema ab.

Bibliotheken	APF-autorisiert	In DFHRPL
SSLDLOAD	Nein	Ja
SSLDAUTH	Ja	Ja

Firewalleinstellungen

Da die TransactionVision-Sensoren WebSphere MQ für die Kommunikation mit TransactionVision Analyzer verwenden, sind keine Firewalleinstellungen erforderlich. Die gesamte Kommunikation wird über WebSphere MQ konfiguriert.

MIPS-Anforderungen

Es gibt keine speziellen MIPS-Anforderungen für den TransactionVision-Sensor. Sensoren sind so konzipiert, dass sie nur bei Bedarf aktiv sind, und auch dann benötigen sie nur in sehr geringem Umfang Ressourcen, um die gewünschten Funktionen zu erreichen.

E

Agents für die uCMDB-Discovery

In diesem Kapitel wird beschrieben, wie der Agent für die uCMDB-Discovery unter IBM z/OS installiert, konfiguriert und verwendet wird.

Dieses Kapitel umfasst die folgenden Themen:

- ▶ Installieren und Konfigurieren des Agents für die uCMDB-Discovery auf Seite 420
- ▶ Komponenten und Betrieb des Agents für die uCMDB-Discovery auf Seite 435
- ▶ Sicherheitsanforderungen des Agents für die uCMDB-Discovery auf Seite 437
- ▶ Zusammenfassung der uCMDB-Befehle auf Seite 441
- ▶ Konsolenmeldungen des Agents für uCMDB-Mainframe-Dienste auf Seite 443
- ▶ Fehlermeldungen bei der uCMDB-Discovery unter z/OS auf Seite 451

Wichtig: Lesen Sie die Abschnitte Komponenten und Betrieb des Agents für die uCMDB-Discovery auf Seite 435 und Sicherheitsanforderungen des Agents für die uCMDB-Discovery auf Seite 437, bevor Sie den Agent installieren und konfigurieren. So erhalten Sie nützliche Hintergrund- und Kontextinformationen.

Installieren und Konfigurieren des Agents für die uCMDB-Discovery

Die folgenden Aufgaben müssen ausgeführt werden, um den Agent für die uCMDB-Discovery und die ebenfalls erforderlichen Produkte zu installieren und zu konfigurieren:

Aufgabe 1: Überprüfen der Systemanforderungen

Aufgabe 2: Installieren des Agents für die uCMDB-Discovery

Aufgabe 3: Installieren von IBM HTTP Server

Aufgabe 4: Aktivieren von IBM HTTP Server für die uCMDB-Discovery

Aufgabe 5: Anpassen des Agents für den uCMDB-Discovery-Dienst

Aufgabe 1: Überprüfen der Systemanforderungen

Überprüfen Sie, ob der für die Installation vorgesehene Host diese Systemanforderungen erfüllt:

- ▶ Betriebssystem IBM z/OS V1.7 oder höher.
- ▶ IBM HTTP Server für z/OS, Version 5.3 (5694-A01), installiert und konfiguriert wie in den Aufgaben 3 und 4 beschrieben.
- ▶ IBM SDSF (System Display and Search Facility, Systemanzeige- und Suchfunktionen) für z/OS. Dieses Produkt ist erforderlich, um Merkmale von IMS-Untersystemen über eine Discovery zu ermitteln.

IBM HTTP Server für z/OS, Version 5.3, wird im Folgenden als IBM HTTP Server bezeichnet.

Aufgabe 2: Installieren des Agents für die uCMDB-Discovery

Das allgemeine Verfahren für die Installation des Agents für die uCMDB-Discovery sieht wie folgt aus:

1. Übertragen Sie die Installationsdateien von den Installationsmedien auf die zSeries-Plattform. Hierfür gibt es zwei Möglichkeiten:
 - ▶ Verwenden Sie das auf den Installationsmedien bereitgestellte Skript **tvinstall_800_zos_zseries.bat**, um die Dateien zu übertragen (empfohlen).
 - ▶ Übertragen Sie die Installationsdateien manuell mit FTP.
2. Führen Sie den TSO-Befehl RECEIVE für die übertragenen Dateien aus, um Produktdatasets zu erstellen. Hierfür gibt es zwei Möglichkeiten:
 - ▶ Verwenden Sie das in Schritt 1 durch das Skript **tvinstall_800_zos_zseries.bat** erstellte REXX-Skript (TSORECV), um die TSO-RECEIVE-Befehle auszuführen (empfohlen).
 - ▶ Führen Sie die erforderlichen TSO-RECEIVE-Befehle manuell aus.
3. Bestimmen Sie den besten Installationsansatz für Ihre Umgebung. Hierfür gibt es zwei Möglichkeiten:
 - ▶ SMP/E (empfohlen). Sie können entweder eine neue globale SMP/E-Zone erstellen (empfohlen) oder eine bestehende globale SMP/E-Zone verwenden.
 - ▶ Nicht-SMP/E
4. Passen Sie die Installationsjobs an und führen Sie sie aus. Der Umfang der erforderlichen Anpassung hängt von der Installationsmethode ab. Hierfür gibt es zwei Möglichkeiten:
 - ▶ SMP/E-Installationsjobs
 - ▶ Nicht-SMP/E-Installationsjobs
5. Überprüfen Sie die Installationsbibliotheken.

Dateinamenkonventionen

Die Dateinamen auf den Installationsmedien entsprechen dem folgenden Muster:

dsslr<dateiqual>_<ver>_zos_zseries.xmit

Die Dateinamen der FTP-Ausgabedatasets sollten dem folgenden Muster entsprechen:

<&hlq>.<prodcode>|<ver>.<dateiqual>

Hierbei gilt Folgendes:

<dateiqual> = f1 | f2 | f3 | f4 | mcs

<prodcode> = SLR

<ver> = 800

<&hlq> = ein kundendefiniertes übergeordnetes Qualifikationsmerkmal für die Ausgabe des FTP-Unterbefehls PUT auf dem z/OS-Zielsystem.

Ein Dateiname auf den Installationsmedien lautet beispielsweise folgendermaßen:

dsslrf1_800_zos_zseries.xmit

Ein Dateiname für ein FTP-Ausgabedataset lautet:

DSCVRY.SLR800.F1

Schritt 1: Übertragen Sie die Dateien von den Installationsmedien auf den zSeries-Host

Hierfür gibt es zwei Möglichkeiten:

- ▶ Führen Sie das Skript **tvisntall_800_zos_zseries.bat** aus.
- ▶ Übertragen Sie die Installationsdateien manuell mit FTP.

Möglichkeit 1: Führen Sie das Skript "tvisntall_800_zos_zseries.bat" aus

- 1** Melden Sie sich an einer Arbeitsstation unter Microsoft Windows an, von der mit FTP eine TCP/IP-Verbindung zum z/OS-Zielsystem hergestellt werden kann.
- 2** Starten Sie unter Windows eine Eingabeaufforderungssitzung (**Start > Alle Programme > Zubehör > Eingabeaufforderung**). Passen Sie die Fenstereigenschaften an, indem Sie Größe und Puffer nach Bedarf verändern.
- 3** Navigieren Sie mithilfe des Befehls **change directory (cd)** zu dem Ordner mit den Installationsmedien.
- 4** Überprüfen Sie die Informationen, die vom Skript erfasst werden indem Sie Folgendes eingeben:

```
tvinstall_800_zos_zseries.bat /h | more
```

- 5 Rufen Sie **tvinstall_800_zos_zseries.bat** ohne Parameter auf, und reagieren Sie auf die Eingabeaufforderungen.

Hinweis: Wenn Sie mit den eingegebenen Werten nicht zufrieden sind, können Sie **tvinstall_800_zos_zseries.bat** beenden, ohne eine Dateiübertragung zu initiieren. Darüber hinaus können Sie die Skriptausführung jederzeit beenden, indem Sie wiederholt STRG+C drücken.

- 6 Prüfen Sie sorgfältig die Ergebnisse der FTP-Befehlsausgabe, die von **tvinstall_800_zos_zseries.bat** generiert werden.
 - ▶ Es sollten keine Fehler aufgeführt werden.
 - ▶ Prüfen Sie, ob Installations-RELFILes mit dem angegebenen übergeordneten Qualifikationsmerkmal auf dem z/OS-Zielsystem vorhanden sind.
 - ▶ Prüfen Sie, ob das REXX-Skript (TSORECV) in der angegebenen PDS-Bibliothek vorhanden ist.
 - ▶ Wenn Datasets fehlen oder FTP-Fehler festgestellt wurde, sollten Sie diese Fehler untersuchen und korrigieren und **tvinstall_800_zos_zseries.bat** erneut ausführen.

Möglichkeit 2: Übertragen Sie die Installationsdateien manuell mit FTP

Führen Sie diesen Schritt nicht aus, wenn Sie die Dateien mit dem Skript **tvinstall_800_zos_zseries.bat** installiert haben.

- 1 Melden Sie sich an einer Arbeitsstation unter Microsoft Windows an, von der mit FTP eine TCP/IP-Verbindung zum z/OS-Zielsystem hergestellt werden kann.
- 2 Starten Sie unter Windows eine Eingabeaufforderungssitzung (**Start > Alle Programme > Zubehör > Eingabeaufforderung**). Passen Sie die Fenstereigenschaften an, indem Sie Größe und Puffer nach Bedarf verändern.

- 3 Navigieren Sie mithilfe des Befehls **change directory (cd)** zu dem Ordner mit den Installationsmedien.
- 4 Verwenden Sie ein z/OS-Konto, das über ausreichende Berechtigungen zum Erstellen von Datasets mit dem gewünschten übergeordneten Qualifikationsmerkmal verfügt, und starten Sie eine FTP-Sitzung. Beispiel:
 > ftp hostname
- 5 Setzen Sie die folgenden FTP-Unterbefehle ab:

a ftp> **quote site fixrecfm 80 lrecl=80 recfm=fb blksize=3120 vol=&custvol u=&custunit pri=30 sec=5 tr**

Hierbei ist **&custvol** eine gültige Datenträgerfolgennummer und **&custunit** ein gültiger Datenträger-Einheitentyp oder ein privater Einheitenname

b ftp> **bin**

c ftp> **put dsslr<dateiqual>.<ver>_zos_zseries.xmit '<&hlq>.slr<|ver>.<dateiqual>'**

Hierbei gilt Folgendes:

<&hlq> = kundendefiniertes übergeordnetes Qualifikationsmerkmal

<dateiqual> = f1 | f2 | f3 | f4 | mcs

<ver> = 800

Beispiel:

PUT dscvry_agent_slr_800_zos_zseries.xmit '&hlq.slr|ver.dateiqual'

- d** Setzen Sie FTP-Befehle des Typs **put** für alle verbleibenden Produktinstallationsdateien mit dem Dateinamensuffix **fn** oder **mcs** ab.
- e** Es ist wichtig, die FTP-Befehlsausgabe sorgfältig im Hinblick auf Fehler zu untersuchen. Falls Fehler gefunden werden, kann es notwendig sein, das Zieldataset auf dem z/OS-System zu löschen und den FTP-Befehl **put** erneut auszuführen.
- f** ftp> **quit**

Schritt 2: Führen Sie den TSO-Befehl RECEIVE für die übertragenen Dateien aus, um Produktdatasets zu erstellen

Hierfür gibt es zwei Möglichkeiten:

- Führen Sie das in "Schritt 1: Übertragen Sie die Dateien von den Installationsmedien auf den zSeries-Host" auf Seite 422 durch das Skript `tvinstall_800_zos_zseries.bat` erstellte REXX-Skript (TSORECV) aus.
- Führen Sie die erforderlichen TSO-RECEIVE-Befehle manuell aus

Möglichkeit 1: Führen Sie das REXX-Skript (TSORECV) aus

Führen Sie an einer TSO- oder TSO/ISPF-Befehlszeile das in dem in Schritt 1 oben identifizierten PDS enthaltene REXX-Skript (TSORECV) aus, ggf. mit dem Präfix TSO.

```
exec 'PDS.identified.instep1(TSORECV)'
```

Damit werden die TSO-Befehle RECEIVE für alle z/OS-Zieldatasets ausgeführt, die in Schritt 1, 6 oben angegeben sind. Die mit RECEIVE empfangenen Datasets werden mithilfe desselben übergeordneten Qualifikationsmerkmals zugeordnet, wobei jedoch der Buchstabe "A" als Präfix zum Produktcode hinzugefügt wird.

Wenn das Eingabemuster beispielsweise **DSCVRY.SLR800.dateiqual** entspricht, würde das Ausgabemuster **DSCVRY.ASLR800.dateiqual** sein.

Möglichkeit 2: Führen Sie die erforderlichen TSO-RECEIVE-Befehle manuell aus

- 1 Führen Sie an einer TSO- oder TSO/ISPF-Befehlszeile RECEIVE-Befehle, falls notwendig mit dem Präfix TSO, wie in der folgenden Beispielstabelle gezeigt aus. Geben Sie als Antwort auf die Aufforderung "INMR906A Enter restore parameters or 'DELETE' or 'END'" DSN(&hlq.ASLR800.dateiqual) ein.

Befehl	Dateiname
RECEIVE INDSNAME('&hlq.SLR8000.F1')	DSN('&hlq.ASLR800.F1')
RECEIVE INDSNAME('&hlq.SLR8000.F2')	DSN('&hlq.ASLR800.F2')
RECEIVE INDSNAME('&hlq.SLR800.F3')	DSN('&hlq.ASLR800.F3')
RECEIVE INDSNAME('&hlq.SLR800.F4')	DSN('&hlq.ASLR800.F4')
RECEIVE INDSNAME('&hlq.SLR800.MCS')	DSN('&hlq.ASLR800.SMPMCS')

Schritt 3: Bestimmen Sie den besten Installationsansatz

Hierfür gibt es zwei Möglichkeiten:

- SMP/E (empfohlen). Sie können entweder eine neue globale SMP/E-Zone (empfohlen) erstellen oder eine bestehende globale SMP/E-Zone verwenden.
- Nicht-SMP/E

Schritt 4: Passen Sie die Installationsjobs an und führen Sie sie aus

Hierfür gibt es zwei Möglichkeiten: SMP/E oder Nicht-SMP/E.

Möglichkeit 1: SMP/E-Installationsjobs

Passen Sie die Jobs an und führen Sie sie in der angegebenen Reihenfolge aus. Jobs befinden sich im Dataset **&hlq.ASLR800.F3**. Lesen Sie die Kommentare am Anfang jedes Jobs und passen Sie die Jobs entsprechend an.

Prüfen Sie im Anschluss an die Ausführung sorgfältig alle Ausgaben, um sicherzustellen, dass alle Schritte erfolgreich waren. Fahren Sie erst dann mit dem nächsten Job fort. Falls die Installation in einer bestehenden globalen SMP/E-Zone erfolgt, sollten SMP/E-Schritte, Parameter und Eingaben sorgfältig geprüft werden.

SMP/E-Installationsjobs:

Reihenfolge	Job	Beschreibung
1.	SLRALLOC	Ordnet Ziel- und Verteilungsbibliotheken zu
2.	SLRGZON	Definiert die globale SMP/E-Zone (nicht erforderlich, wenn eine bestehende globale Zone verwendet wird)
3.	SLRDZON	Definiert die SMP/E-Verteilerzone
4.	SLRTZON	Definiert die SMP/E-Zielzone
5.	SLRDDDEF	Definiert SMP/E-DDDEF-Einträge
6.	SLRRECV	SMP/E RECEIVE
7.	SLRAPPLY	SMP/E APPLY

Reihenfolge	Job	Beschreibung
8.	SLRIVP	Installationsüberprüfung
9.	SLRACCTP	SMP/E ACCEPT Wichtig! Führen Sie diesen Job erst aus, wenn die Installation konfiguriert und vollständig getestet wurde. Sobald die SMP/E-Funktion ACCEPT ausgeführt wurde, ist es nicht möglich, das Produkt aus Ziel- oder Verteilungsbibliotheken zu entfernen.

Möglichkeit 2: Nicht-SMP/E-Installationsjobs

Jobs befinden sich im Dataset **&hlq.ASLR800.F3**. Lesen Sie die Kommentare am Anfang jedes Jobs und passen Sie die Jobs entsprechend an. Prüfen Sie sämtliche Ausgaben nach der Ausführung und stellen Sie sicher, dass alle Schritte erfolgreich abgeschlossen wurden.

Nicht-SMP/E-Installationsjobs:

Reihenfolge	Job	Beschreibung
1.	SLRINSTL	
2.	SLRIVP	Installationsüberprüfung

Schritt 5: Überprüfen Sie die Installationsbibliotheken

DS-Nr.	Datasetname	Beschreibung
1	DSCVRY.SSLRCARD	Steuerkarten, Konfigurationsdaten
2	DSCVRY.SSLREXEC	REXX-CGI-Programmquelle
3	DSCVRY.SSLRINST	Installations-JCL
4	DSCVRY.SSLRLOAD	Dienst-Agent-Ladebibliothek (APF-Autorisierung)
5	DSCVRY.SSLRPROC	Dienst-Agent-Prozedurbibliothek
6	DSCVRY.SSLRSAMP	Beispiel-JCL, Befehle usw.
7	DSCVRY.SLR800.F1	Installations-RELFILE - JCLIN
8	DSCVRY.SLR800.F2	Installations-RELFILE - Lademodule
9	DSCVRY.SLR800.F3	Installations-RELFILE - Installations-JCL
10	DSCVRY.SLR800.F4	Installations-RELFILE - REXX-Programme
11	DSCVRY.SLR800.SMPMCS	Installations-RELFILE - SMP/E MCS-Anweisungen

- ▶ Die Tabelle umfasst nicht die SMP/E-Verteilungsbibliotheken (Distribution Libraries, DLIBS) der Installation. Sie werden erst mit der Ausführung von SLRACCPT gefüllt.
- ▶ DSCVRY ist das standardmäßige übergeordnete Qualifikationsmerkmal; das von Ihnen verwendete Qualifikationsmerkmal kann davon abweichen.

Aufgabe 3: Installieren von IBM HTTP Server

IBM HTTP Server für z/OS, Version 5.3, ist eine z/OS-Basisinstallationskomponente und steht IBM z/OS-Kunden kostenlos zur Verfügung. Erforderliche Informationen zum Produktservice und zu Korrekturen finden Sie im z/OS-Programmverzeichnis für dieses Produkt, in der PSP-Dokumentation und über [IBMLink](#).

Aktuelle Informationen zu diesem Produkt sind im IBM Publications Center unter folgendem URL verfügbar:

<http://www.elink.ibm.link.ibm.com/public/applications/publications/cgi-bin/pbi.cgi>

Die Hinweise in diesem Abschnitt beziehen sich auf die folgende IBM-Veröffentlichung: Handbuch *HTTP Server Planning, Installing, and Using* (SC34-4826-09).

Sie sollten vor der Installation des Produkts Kapitel 1 dieses Handbuchs lesen.

Folgen Sie diesen Schritten bei der Installation von IBM HTTP Server.

1 Beachten Sie die folgenden Punkte im Zusammenhang mit der Installation:

- ▶ Komponenten für die uCMDB-Discovery verwenden keine benutzerdefinierten DLLs.
- ▶ Der Fast Response Cache Accelerator wird zum jetzigen Zeitpunkt nicht eingesetzt.
- ▶ Im Hinblick auf Leistung und Optimierung erfordern Aktivitäten der uCMDB-Discovery wenige oder keine parallelen Benutzerverbindungen. Daher können die Einstellungen für **MaxActive Thread**, **MAXSOCKETS**, **MAXFILEPROC** und ähnliche Leistungseinstellungen relativ niedrig angesetzt werden, wenn davon ausgegangen wird, dass die Aktivitäten der uCMDB-Discovery die primäre Applikationsarbeitslast der primären IBM HTTP Server-Instanz sind.
- ▶ Anpassungen, Befehle und Beispiele zu ESM (External Security Manager, externer Sicherheits-Manager) in diesem Handbuch beziehen sich auf den Sicherheitsserver von IBM (RACF). Für die Implementierung der uCMDB-Discovery ist RACF nicht erforderlich. Der Einsatz anderer ESM-Produkte wie ACF2, TopSecret usw. macht es jedoch erforderlich, dass entsprechende Befehle und Funktionen in der ESM-Umgebung ohne RACF des Kunden erkannt und ausgeführt werden.
- ▶ Das ESM-Konto, mit dem IBM HTTP Server ausgeführt wird, muss über die folgenden Berechtigungen verfügen:
 - ▶ Ausreichende Berechtigungen zum Lesen der Ladebibliothek des Agents für den Discovery-Dienst (DS 4).
 - ▶ Ausreichende Berechtigungen zum Ausführen der REXX-Skripts im Verzeichnis **../cgi-gin** von IBM HTTPS Server.

- 2** Folgen Sie den Anweisungen im Kapitel 3 des Handbuchs *HTTP Server Planning, Installing, and Using*, um IBM HTTP Server zu installieren.

Wenn Sie die einfache Installation abgeschlossen haben, prüfen Sie die Schritte 1 bis 10 von Kapitel 3 sorgfältig und führen Sie sie aus. Verschiedene dieser Schritte umfassen Sicherheitsanpassungen wie SSL, private Schlüsselsammlungszertifikate, Erstellung von Webserverkonten mit nicht leeren Benutzer-IDs usw. Zwar sind diese Anpassungen für den Webserverbetrieb und die Verarbeitung der uCMDB-Discovery nicht unbedingt erforderlich, Hewlett-Packard Software rät Kunden der uCMDB-Discovery im Hinblick auf die plattformübergreifenden Merkmale der Discovery-Verarbeitung aber nachdrücklich, die Funktionen zur Optimierung des Gesamtsicherheitsprofils der Umgebung voll auszuschöpfen.

Die folgenden Beispieljobs sollen Ihnen bei der Konfiguration der Sicherheitsaspekte des Servers helfen. Einige der Beispiele gelten nicht für jede Umgebung. Sie sollten angepasst und in Verbindung mit den Anweisungen aus Kapitel 3 des Handbuchs *HTTP Server Planning, Installing, and Using* ausgeführt werden.

Entsprechende Anweisungen finden Sie in den Schritten 1 bis 5 von Kapitel 3. Der Einfachheit halber sind sie hier aufgeführt:

- **SLRSEC01.** Definitionen von IMWEB-Gruppen und WEBADM-Benutzern
- **SLRSEC02.** WEBSRV-Benutzerdefinition
- **SLRSEC03.** Definieren eines Benutzers mit einer nicht leeren USS-Benutzer-ID
- **SLRSEC04.** Ersetzen von Klassenprofildefinitionen
- **SLRSEC05.** Aktivieren der Programmsteuerung für MVS-Datasets
- **SLRSEC06.** Aktivieren der Programmsteuerung für SSL in z/OS
- **SLRSEC07.** Zulassen von WEBSRV-Zugriff auf eine Schlüsselsammlung
- **SLRSEC08.** Zulassen von WEBSRV-Zugriff auf kryptografische Hardwaredienste
- **SLRSEC09.** Gewähren von WEBSRV-Zugriff für Profile in kryptografischen Diensten

- **SLRSEC10.** Zulassen von WEBSRV-Zugriff auf die z/OS-Arbeitslastverwaltung
 - **SLRSEC11.** Zulassen von WEBSRV-Zugriff auf SMF-Funktionen
 - **SLRSEC12.** Hinzufügen der IBM HTTP Server-PROC zur Tabelle gestarteter Tasks
- 3** Testen Sie die Grundfunktionen von IBM HTTP Server, indem Sie den Verfahren in den Kapiteln 4, 5 und 6 des Handbuchs *HTTP Server Planning, Installing, and Using* folgen.

Es scheint kein IVP für IBM HTTP Server selbst zu geben. Das Material in diesen Kapiteln hilft Ihnen aber beim Starten, Beenden und Aufrufen der Standardtitelseite des Webservers, die eine alternative Konfigurationsoberfläche aufweist. Sie sollten die grundlegenden Serverfunktionen überprüfen, bevor Sie weitere Anpassungen vornehmen.

Aufgabe 4: Aktivieren von IBM HTTP Server für die uCMDB-Discovery

Nachdem die Grundfunktionen von IBM HTTP Server überprüft wurden, nehmen Sie diese Anpassungen hinsichtlich des Betriebs der Komponenten für die uCMDB-Discovery vor.

- 1** Sie sollten unbedingt die SSL-Kommunikation und grundlegende Authentifizierungsfunktionen von IBM HTTP Server aktivieren. Weitere Informationen finden Sie in Kapitel 8 des Handbuchs *HTTP Server Planning, Installing, and Using*.

Das folgende Element (aus DS 6 - Beispiel-JCL) kann als Beispiel der RACF-Befehle und ähnlicher Tasks, die für die Aktivierung der grundlegenden SSL-Kommunikation erforderlich sind, nützlich sein.

```
SLRSSLEX  SSL-Setup - Beispielschritte
```

Verwenden Sie dieses Material zusammen mit dem in Kapitel 8 des oben genannten Handbuchs.

- 2** Aktualisieren Sie die STEPLIB-Verkettung aus der gestarteten Task-JCL für IBM HTTP Server. Der standardmäßige Prozedurname gestarteter Tasks für IBM HTTP Server ist IMWEBSRV. Zwei Bibliotheken müssen der STEPLIB-Verkettung hinzugefügt werden:

- a DS 4, Ladebibliothek des Dienst-Agents, sollte dem Ende der Verkettung hinzugefügt werden.
 - b Falls DB2 installiert ist, fügen Sie auch die DB2-Bibliothek SDSNLOAD gefolgt von der DB2-Bibliothek RUNLIB.LOAD hinzu.
- 3 Aktualisieren Sie die primäre Konfigurationsdatei, normalerweise **httpd.conf**, die dieser Instanz von IBM HTTP Server zugeordnet ist, sodass die Exec-Zuordnungsregeln für REXX-CGI-Skripts eingeschlossen werden. Standardspeicherort dieser Datei:

/etc/httpd.conf

Dieser Speicherort kann jedoch geändert werden. Überprüfen Sie den Speicherort und den Namen der primären Konfigurationsdatei, indem Sie die folgende Zeichenfolge in den Ausgabenachrichten von IBM HTTP Server suchen und dann eine Bearbeitungssitzung für die Datei initiieren.

"Using configuration file"

Fügen Sie die folgende Exec-Zuordnungsregel für die Ausführung von CGI-Programmen und -Skripts hinzu. Erstellen Sie bei Bedarf das Zielverzeichnis in den angegebenen Pfaden.

Exec	/uCMDB/*.cgi	/etc/HP/Discovery/cgi-bin/*.cgi
Exec	/uCMDB/*.sh	/etc/HP/Discovery/cgi-bin/*.sh

Hinweis: Sie können den tatsächlichen Speicherort der CGI-Programme und -Skripts ändern. Dies sollte kein Problem darstellen, sofern auch die Sicherheitsattribute geändert werden.

- 4 Die in Schritt 1 von Kapitel 3 im Handbuch *HTTP Server Planning, Installing, and Using* identifizierte Benutzer-ID muss über die Berechtigungen zum Lesen und Ausführen der REXX-CGI-Skripts im oben angegebenen Verzeichnis verfügen. Passen Sie die Sicherheitsattribute entsprechend an.

Aufgabe 5: Anpassen des Agents für den uCMDB-Discovery-Dienst

Nachdem die grundlegende Funktionalität von IBM HTTP Server für z/OS überprüft wurde, müssen Anpassungen für den Betrieb der uCMDB-Discovery vorgenommen werden.

- 1 Nachdem Sie die entsprechenden Ersetzungen vorgenommen haben, führen Sie den folgenden Befehl der TSO/ISPF-Befehls-Shell aus (Option 6). Damit werden die REXX-CGI-Skripts in das USS-Dateisystem (.../**dscvry/cgi-bin**), das in Schritt 3 auf Seite 432 bestimmt wurde, kopiert.

```
OPUTX 'sourcePDS' 'targetUSSfilesys' TEXT CONVERT(YES)
```

Beispiel:

```
OPUTX 'DSCVRY.SSLREXEC' '/etc/HP/Discovery/cgi-bin' TEXT  
CONVERT(YES)
```

- 2 Der Agent für den uCMDB-Discovery-Dienst führt eine SAF-Standardauthentifizierung der Benutzer-ID und des Kennworts der Discovery-Anforderung durch. Diese Benutzer-ID und das Kennwort müssen in ESM bekannt sein, damit die Standardauthentifizierung erfolgreich abgeschlossen werden kann.
- 3 Passen Sie das PDS-Element SLRSRVIN aus DS 1 (Steuerkarten und Konfigurationsdaten) an.
- 4 Überprüfen Sie die ACL-Datei (Authorized Command List, Liste autorisierter Befehle) bzw. das ACL-Element, auf das in Aufgabe 5, Schritt 3, weiter oben verwiesen wird, und nehmen Sie ggf. Änderungen vor. Das standardmäßige ACL-Element SLRACL befindet sich in DS 1 (Steuerkarten und Konfigurationsdaten).
- 5 Richten Sie die APF-Autorisierung der uCMDB-Discovery-Bibliothek **&hlq.SSLRLOAD** (DS 4) ein. Hilfe finden Sie im Handbuch **MVS Initialization and Tuning Reference** (SA22-7592).
- 6 Passen Sie die JCL-Prozeduren gestarteter Task nach den Anweisungen oben in jedem Element an. Kopieren Sie sie dann in die entsprechende JCL-Prozedurbibliothek für Ihre Website.

Beispiel-JCL für Prozeduren gestarteter Task finden Sie in DS 5 - Dienst-Agent-Prozedurbibliothek.

Hinweis: Falls die unten aufgeführten Prozedurnamen nicht mit den Websitestandards oder -einstellungen konsistent sind, können Sie die Namen ändern. Bei SLRTSREX ist es dann jedoch auch erforderlich, dass die Änderung auch in der SLRSERV-Konfigurationsdatei wiedergegeben wird.

- SLRSERV
 - SLRTSREX
- 7** SLRSERV und SLRTSREX werden als gestartete MVS-Tasks ausgeführt. Die standardmäßige Benutzer-ID, die diesen Tasks zugeordnet ist, muss über die folgenden Berechtigungen verfügen:
- Erstellen von Dateien im USS-Dateisystem **/tmp/Discovery**.
 - Lesen aller Bibliotheken, die in der STEPLIB DD-Anweisung angegeben sind
 - Lesen der Bibliotheken, die in der SYSEXEC DD-Anweisung angegeben sind
 - Lesen der Konfigurationsdatei und der Liste autorisierter Befehle
 - Erstellen und Verwenden einer MVS MCSE-Konsole
 - Ausführen aller Befehle in der Liste autorisierter Befehle (Authorized Command List, ACL)
- 8** Lesen Sie "Komponenten und Betrieb des Agents für die uCMDB-Discovery" auf Seite 435 um die gesamte Architektur, die Komponenten und den einfachen Betrieb besser zu verstehen.
- 9** Nachdem Sie Erfahrungen mit dem Produkt gesammelt haben, kehren Sie zu Schritt 3 von "Aufgabe 2: Installieren des Agents für die uCMDB-Discovery" auf Seite 421 zurück und führen Sie den Job **SLRACCT (SMP/E Accept)** aus.

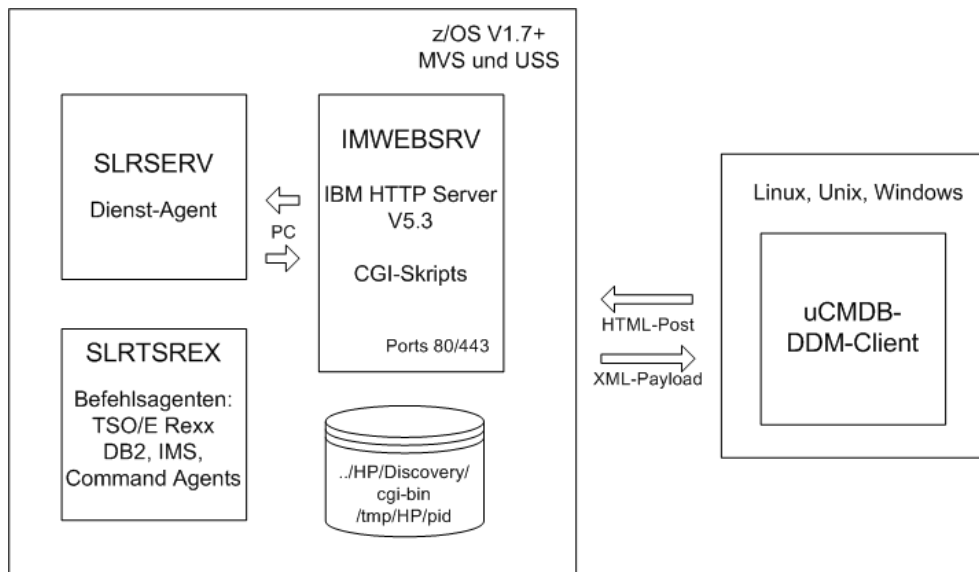
Komponenten und Betrieb des Agents für die uCMDB-Discovery

Dieser Abschnitt gibt eine kurze Übersicht über die Architektur des Agents für die uCMDB-Discovery sowie Anweisungen zum Starten und Anhalten von dessen z/OS-basierten Komponenten.

In der folgenden Tabelle sind die wichtigsten Komponenten des Agents für die uCMDB-Discovery aufgeführt:

Prozedur	Beschreibung
IMWEBSRV	IBM HTTP Server V5.3 für z/OS (standardmäßig verwendeter Prozedurname)
SLRSERV	Dienst-Agent für die MVS-Konsole (gestarteter Task)
SLRTSREX	TSO/E REXX-Befehlsprozessor (gestarteter Task)
DDM Client	HP-Client für uCMDB-DDM (Remote-Client für Discovery and Dependency Mapping (DDM) auf einer Linux-, Unix- oder Windows-Plattform)

Die uCMDB-Architektur sieht wie folgt aus:



IBM HTTP Server muss installiert sein und ausgeführt werden, damit SLRSERV ordnungsgemäß funktioniert. Weitere Informationen finden Sie in den Kapiteln 4 und 6 des Handbuchs *HTTP Server Planning, Installing, and Using*.

Starten von SLRSERV:

Verwenden Sie zum Starten von SLRSERV den Startbefehl einer MVS-Konsole:

```
S SLRSERV
```

Hinweis: Die Prozedur SLRSERV startet SLRTSREX automatisch, indem ein interner Startbefehl an das MVS-Betriebssystem ausgegeben wird. Versuchen Sie nicht, SLRTSREX manuell zu starten.

Nach dem Start können SLRSERV und der begleitende Task SLRTSREX bis zum nächsten IPL ausgeführt werden.

Anhalten von SLRSERV:

Bei Bedarf können Sie SLRSERV mit einem MVS-Anhaltebefehl beenden:

```
P SLRSERV
```

Hinweis: Wenn SLRSERV beendet wird, wird auch SLRTSREX automatisch beendet. Versuchen Sie nicht, SLRTSREX manuell anzuhalten.

Aktualisieren der ACL:

Dynamische Konfigurationsdaten, darunter eine Liste autorisierter Befehle (Authorized Command List, ACL), werden mit dem Parameter PARM= der Anweisung EXEC an SLRSERV übergeben. Verwenden Sie den MVS-Änderungsbefehl REFRESH, um SLRSERV anzuweisen, die ACL erneut zu lesen. So werden die letzten Änderungen ergänzt. Beispiel:

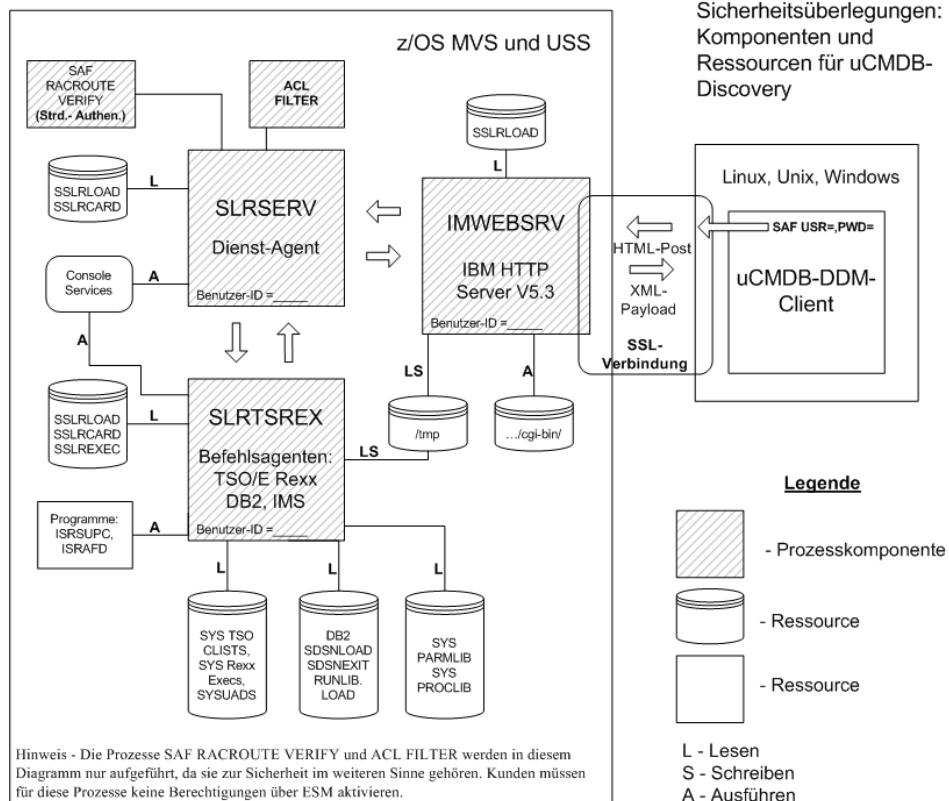
```
F SLRSERV,REFRESH
```

Sicherheitsanforderungen des Agents für die uCMDB-Discovery

Die folgende Liste soll zusammen mit einem z/OS ESM (External Security Manager, externer Sicherheits-Manager) verwendet werden, um ESM-spezifische Befehlszeichenfolgen zu entwickeln, die erforderlich sind, um Berechtigungen zu erteilen und die Agent-Komponenten für die uCMDB-Discovery zu autorisieren.

Alle Elemente der Liste sind auch in verschiedenen Abschnitten zur Installation und Konfiguration in diesem Dokument enthalten. Sie sind hier zusammengefasst, um Ihnen einen einfachen Überblick zu geben. Die Liste ist nach Discovery-Agent-Komponenten aufgegliedert. Einer Benutzer-ID, die den einzelnen Discovery-Agent-Komponenten zugeordnet ist (primär, sekundär oder Gruppe), müssen ausreichende Berechtigungen gewährt werden, um die beschriebenen Aufgaben auszuführen.

Im folgenden Diagramm sind die verschiedenen beteiligten Ressourcen und die Discovery-Agent-Komponenten aufgeführt, für die Zugriff erforderlich ist.



IBM HTTP Server

- Berechtigung zum Ausführen für **cgi-bin**-Skripts und -Programme in dem Ordner, der von der Datei **httpd.conf** des HTTP-Servers für REXX-Zuordnungsregeln angegeben ist. Der Speicherort dieses Ordners kann angepasst werden. Der vorgeschlagene Pfad ist **/etc/HP/Discovery/cgi-bin/**
- Lese- und Schreibzugriff auf Dateien und Unterverzeichnisse im Dateisystem **/tmp**

- ▶ Lesezugriff auf DS 4 – DSCVRY.SSLRLOAD (in der Tabelle in Aufgabe 2, Schritt 5) (diese Bibliothek muss auch für MVS als APF-autorisiert definiert sein)
- ▶ Es ist zwar für den Agent für den uCMDB-Discovery-Dienst nicht unbedingt erforderlich, Hewlett-Packard Software empfiehlt jedoch, dass der Kunde alle geeigneten Vorsichtsmaßnahmen ergreift, um die IBM HTTP Server-Umgebung zu sichern. Dazu gehören auch die Anpassungen für SSL und die Verwaltung, die im IBM-Handbuch **HTTP Server Planning, Installing, and Using** (SC34-4826) beschrieben sind. Beispiel-Jobstreams, die für die RACF-Sicherheitsumgebung gelten, wurden aufgenommen. Dies kann nützlich für die Aktivierung von SSL und das Sichern des HTTP-Servers sein (siehe SLRSSLEX, SLRSEC01-SLRSEC12 in DS 6 – DSCVRY.SSLRSAMP, Beispiel-JCL, Befehle usw.)

Gestarteter SLRSERV-Task

- ▶ MVS-autorisierte Dienstaufrufe, die mit der Erstellung und Verwendung der MCSE-Konsolendienste verbunden sind: MCSOPER (ACTIVATE,DEACTIVATE), MGCRE, MCOPMSG
- ▶ Ausführung verschiedener MVS-Befehle (üblicherweise Anzeigebefehle), TSO-Befehle und verschiedener REXX Exec- und TSO CLIST-Befehle. Einige Befehle zielen über eine "Erkennungszeichenfolge" auf bestimmte Untersysteme ab. Daher müssen die speziellen Berechtigungen auch die Verwendung solcher Zeichenfolgen einschließen. Eine vollständige Liste aller Befehle, die aufgerufen werden können, finden Sie im Abschnitt mit der Befehlsreferenz in diesem Dokument. Der Kunde kann die Ausführung eines Befehls über die ACL (Authorized Command List, Liste autorisierter Befehle) steuern, die weiter unten erläutert wird.
- ▶ Lesezugriff auf DS 1 – DSCVRY.SSLRCARD (in der Tabelle in Aufgabe 2, Schritt 5) oder eine entsprechende QSAM-Datei des Kunden oder PDS mit der SLRSERV-Konfigurationsdatei und ACL.
- ▶ Lesezugriff auf DS 4 – DSCVRY.SSLRLOAD (in der Tabelle in Aufgabe 2, Schritt 5) (diese Bibliothek muss auch für MVS als APF-autorisiert definiert sein)

- Neben einem granularen Operator können auch Befehlsberechtigungen, die der Kunde über den ESM und die ACL definiert hat, verwendet werden, um zu steuern, welche Befehle, Exec-Befehle und CLIST-Befehle vom Agent für den Discovery-Dienst ausgeführt werden. Der Speicherort der ACL wird in der SLRSERV-Konfigurationsdatei definiert und über die Anweisung PARM= in der SLRSERV-Prozedur gelesen. Wird keine ACL angegeben, verwendet SLRSERV eine intern definierte Standardentsprechung im Inhalt der Standard-ACL (SLRACL) aus DS 1 – Steuerkarten und Konfigurationsdaten für DSCVRY.SSLRCARD. Bitte prüfen Sie den Inhalt der verwendeten ACL-Datei und nehmen Sie ggf. Anpassungen vor. Beim Fehlen der Sicherheit auf Befehlsebene für den ESM-Operator oder in Verbindung damit gibt die ACL dem Kunden die Möglichkeit zu steuern, welche Befehle oder Klassen von Befehlen (falls Platzhalter verwendet werden) vom Agent für den Discovery-Dienst ausgeführt werden können.

Gestarteter SLRTSREX-Task

- Lesezugriff auf DS 4 – DSCVRY.SSLRLOAD (in der Tabelle in Aufgabe 2, Schritt 5) (diese Bibliothek muss auch für MVS als APF-autorisiert definiert sein)
- Lesezugriff auf alle DB2-Bibliotheken SDSNLOAD, SDSNEXIT und RUNLIB.LOAD, falls DB2-Komponenten über eine Discovery ermittelt werden sollen.
- Lesezugriff auf lokale SYSUADS-Datasets
- Lesezugriff auf lokale TSO CLIST- und REXX Exec-Bibliotheken
- Lesezugriff auf DS 2 – DSCVRY.SSLREXEC (in der Tabelle in Aufgabe 2, Schritt 5)
- Lesezugriff auf DS 1 – DSCVRY.SSLRCARD (in der Tabelle in Aufgabe 2, Schritt 5)
- Lese- und Schreibzugriff auf Dateien und Unterverzeichnisse im Dateisystem **/tmp**
- Lesezugriff auf Systemparameter und Prozedurbibliotheken (wie SYS1.PARMLIB, USER.PROCLIB usw.)
- Berechtigung zum Ausführen des IBM-Programms ISRSUPC zur Suche von Zeichenfolgen

- ▶ Berechtigung zum Ausführen des IBM SDSF-Programms ISRAFD (für die Discovery von IMS-Ressourcen)
- ▶ Berechtigung zum Ausführen der DB2-Pläne DSNREXX und DSNTEP2
- ▶ MVS-autorisierte Dienstaufrufe, die mit der Erstellung und Verwendung der MCSE-Konsolendienste verbunden sind: MCSOPER (ACTIVATE,DEACTIVATE), MGCRE, MCOPMSG
- ▶ Ausführung verschiedener MVS-Befehle (üblicherweise Anzeigebefehle), TSO-Befehle und verschiedener REXX Exec- und TSO CLIST-Befehle. Einige Befehle zielen über eine "Erkennungszeichenfolge" auf bestimmte Untersysteme ab. Daher müssen die speziellen Berechtigungen auch die Verwendung solcher Zeichenfolgen einschließen.

Sonstiges

- ▶ Der uCMDB-DDM-Client weist eigene Sicherheitsanforderungen auf.
- ▶ Allen Anforderungen des uCMDB-DDM-Clients sind eine Benutzer-ID und ein Kennwort zugeordnet, die vom gestarteten SLRSERV-Task für die SAF-Standardauthentifizierung verwendet werden. Die Benutzer-ID muss im z/OS ESM definiert sein.

Zusammenfassung der uCMDB-Befehle

Die folgenden Befehle werden vom Agent für den uCMDB-Discovery-Dienst ausgeführt. In den verschiedenen Befehlstypen sind sie nach Befehlsuntertypen unterteilt. Die folgenden Befehlstypen werden vom Dienst-Agent erkannt und haben in der ACL eine Bedeutung: MVS, TSO und SAF. Einige der Befehle unter MVS sind tatsächlich DB2-, CICS- und MQSeries-Befehle. In künftigen Versionen werden voraussichtlich weitere Beschreibungen von Befehlstypen ergänzt.

MVS

D ASM
D M=CPU
D OPDATA
D NET,MAJNODES
D PROD,REGISTERED
D PROD,STATE
D R
D SSI
D SYMBOLS
D TCPIP,,NETSTAT,CONN
D TCPIP,,NETSTAT,DEV
D TCPIP,,NETSTAT,HOME
D TCPIP,,NETSTAT,ROUTE
D VIRTSTOR,HVSHARE
D XCF,GRP
D XCF,GRP,groupname,ALL

DISPLAY CHANNEL	MQSeries
DISPLAY CHINIT	MQSeries
DISPLAY SYSTEM	MQSeries

CEMT I TAS	CICS
------------	------

DISPLAY DATABASE	DB2
DISPLAY GROUP	DB2
DISPLAY LOCATION	DB2

TSO

TSO ALTLIB DISPLAY
TSO LISTA SYSNAMES
TSO LISTALC SYSNAMES
TSO NETSTAT
TSO STATUS
TSO TIME

SLRRDFLE REXX Exec zum Lesen einer Datei
SLRREXSC REXX Exec für die dynamische SQL-Ausführung
SLRSRPDS REXX Exec zum Suchen nach einem PDS-Element
SLRDTEP2 REXX Exec zum Aufrufen von DSNTEP2 für die dynamische
SQL-Ausführung
SLRGETSS
SLRIMSDS REXX Exec für die Ausführung des Befehls IMS DISPLAY
ACTIVE
SLREXPRD TSO CLIST zum Erzeugen eines Inventar-Reports der
Systemsoftware

SAF

SAF USR=,PWD=Interne Anforderung zur SAF-Standardauthentifizierung

Konsolenmeldungen des Agents für uCMDB-Mainframe-Dienste

Die folgenden Meldungen werden in der Operatorkonsole angezeigt. Diese Meldungen werden zusammen mit verschiedenen anderen Meldungen auch in das MSGOUT-Dataset geschrieben.

SLR-0001-E: Not APF authorized

Für den Agent der uCMDB-Dienste wurde die APF-Autorisierung nicht eingerichtet. Diese Meldung kann darauf hinweisen, dass für die Ladebibliothek des Agents der uCMDB-Dienste die APF-Autorisierung nicht eingerichtet wurde, dass das Lademodul SLRAGENT einen leeren Autorisierungscode aufweist, dass STEPLIB der Ausführungs-JCL mindestens eine nicht autorisierte Bibliothek enthält oder dass SLRAGENT von nicht autorisierter Software aufgerufen wurde.

SLR-0002-I: TRANSWAP completed successfully

Diese Meldung weist darauf hin, dass der Adressraum des Agents der uCMDB-Dienste jetzt nicht mehr austauschbar ist. Diese Meldung wird nicht angezeigt, wenn der Adressraum des Agents der uCMDB-Dienste bereits zu dem Zeitpunkt nicht mehr austauschbar war, als SLRAGENT die Steuerung zugewiesen wurde.

SLR-0003-I: uCMDB services agent (SLRvvvvv) main task starting

Diese Meldung wird angezeigt, wenn der Agent der uCMDB-Dienste die Start- und Initialisierungssequenz startet. Der Kunde kann diese Meldung zur Automatisierung nutzen. Die Meldung umfasst (wie durch vvvvvv gekennzeichnet) einen Versions-/Freigabecode, der bis zu fünf Zeichen lang ist.

SLR-0011-E: Operating system is too old, no longer supported

Das Hostbetriebssystem bietet nicht alle Funktionen und Möglichkeiten, die für den Agent der uCMDB-Dienste erforderlich sind.

SLR-0012-E: Error in authorized command list

Mit der Klausel CMDLIST= in der Konfigurationsdatei wird der Name einer ACL-Datei (Authorized Command List, Liste autorisierter Befehle) angegeben. Auf die angegebene ACL-Datei kann jedoch nicht zugegriffen werden oder sie enthält Fehler. Weitere Details finden Sie in MSGOUT.

SLR-0013-E: Error in configuration file

Die EXEC-Anweisung der Ausführungs-JCL enthält eine nicht leere PARM=-Klausel, in der der Name einer Konfigurationsdatei angegeben ist. Auf die angegebene Konfigurationsdatei kann jedoch nicht zugegriffen werden oder sie enthält Fehler. Weitere Details finden Sie in MSGOUT.

SLR-0014-E: Invalid PARM= clause in execution JCL

Die EXEC-Anweisung der Ausführungs-JCL enthält eine nicht leere PARM=-Klausel. Der von der PARM=-Klausel angegebene Text ist fehlerhaft. Beispielsweise kann die Länge des PARM-Texts mehr als 100 Byte umfassen.

SLR-0015-E: ALESERV rc=X'hhhh'

Die Makroanweisung ALESERV EXTRACTH hat einen nicht leeren Rückgabecode zurückgegeben. Die Meldung enthält den Rückgabecode, der vom Makro ALESERV zurückgegeben wurde. ALESERV-Rückgabecodes sind in *MVS Programming: Authorized Assembler Services Reference, Volume 1 (ALESERV-DYNALLOC)* dokumentiert.

SLR-0016-E: Services agent already active

Eine andere Instanz des Agents der uCMDB-Dienste ist im aktuellen z/OS-Image bereits aktiv.

SLR-0017-E: Unable to load xxxxxxxx, retc=X'hhhhhhh', reas=X'hhhh'

Das angegebene Modul (gekennzeichnet durch xxxxxxxx) konnte nicht geladen werden. Diese Meldung kann auf zu wenig Speicher oder darauf hinweisen, dass ein bestimmtes Modul in STEPLIB nicht gefunden wurde.

SLR-0025-I: Attach of subtask xxxxxxxx failed, rc=X'hhhh'

Die Makroanweisung ATTACH hat einen nicht leeren Rückgabecode zurückgegeben. Dies deutet normalerweise auf zu wenig Speicher hin.

SLR-0026-I: Subtask xxxxxxxx terminated, ecb=X'40ssuuu'

Der angegebene Subtask wurde beendet. Die Meldung zeigt den Hauptprogrammnamen des beendeten Subtasks und den 32-Bit-Inhalt für den Beendigungs-ECB des Subtasks. Im Allgemeinen steht sss für einen Systemabsturzcode und uuu für einen Benutzerabsturzcode.

SLR-0028-I: uCMDB Services Agent main task is now active

Diese Meldung wird angezeigt, wenn der Agent der uCMDB-Dienste die Start- und Initialisierungssequenz beendet hat. Der Kunde kann diese Meldung zur Automatisierung nutzen.

SLR-0029-I: RQE successfully reclaimed

Mit dieser Meldung wird angegeben, dass ein Clientprogramm eine Anforderung ausgegeben hat, die Ergebnisse dieser Anforderung aber nicht rechtzeitig abrufen konnte. Das Anforderungswarteschlangenelement (Request Queue Element, RQE), das der Anforderung zugeordnet ist, wurde freigegeben und kann erneut verwendet werden.

SLR-0030-E: Severe error in time management routine

Wenden Sie sich an den technischen Support, sollte diese Meldung angezeigt werden!

SLR-0040-W: Unrecognized console command ignored

Der Konsolenoperator hat einen fehlerhaften Konsolenbefehl an den Adressraum des Agents der uCMDB-Dienste ausgegeben. Nach dem Start erkennt der Agent der uCMDB-Dienste nur die Befehle MODIFY und STOP des Konsolenoperators. Alle anderen Konsolenbefehle vom Konsolenoperator werden ignoriert.

SLR-0041-I: Modify command data: xxxxxxxx

Ein Befehl vom Typ MODIFY wurde vom Konsolenoperator empfangen. Die Daten zu diesem MODIFY-Befehls werden in der Meldung angegeben.

SLR-0042-I: STOP command accepted

Ein Befehl vom Typ STOP wurde vom Konsolenoperator empfangen.

SLR-0043-E: Unable to free a CIB

Die Makroanweisung QEDIT hat einen nicht leeren Rückgabecode beim Versuch zurückzugeben, einen Befehlseingabepuffer (Command Input Buffer, CIB) freizugeben.

SLR-0052-E: Unable to obtain memory for an ECB table, rc=X'hhhh'

Die Makroanweisung STORAGE OBTAIN hat einen nicht leeren Rückgabecode bei dem Versuch zurückgegeben, Speicher für eine ECB-Tabelle abzurufen.

SLR-0065-E: IEANTRT failed, rc=X'hhhh'

Die z/OS-Dienstroutine für das Abrufen von Namen/Token (IEANTRT) hat einen nicht leeren Rückgabecode zurückgegeben, der nicht auf eine ergebnislose Suche hinweist. Diese Meldung zeigt (als Hex-Wert) den Rückgabecode (**hhhh**), der von IEANTRT zurückgegeben wurde. Rückgabecodes für Namen/Token sind in *MVS Programming: Assembler Services Reference, Volume 2 (IARR2V-XCTLX)* dokumentiert.

SLR-0069-E: IEANTRT failed, rc=X'hhhh'

Die z/OS-Dienstroutine für die Erstellung von Namen/Token (IEANTCR) hat einen nicht leeren Rückgabecode zurückgegeben. Diese Meldung zeigt (als Hex-Wert) den Rückgabecode (**hhhh**), der von IEANTRT zurückgegeben wurde. Rückgabecodes für Namen/Token sind in *MVS Programming: Assembler Services Reference, Volume 2 (IARR2V-XCTLX)* dokumentiert.

SLR-0072-E: IEANTDL failed, rc=X'hhhh', name= tokename

Die z/OS-Dienstroutine zum Löschen von Namen/Token (IEANTDL) hat einen nicht leeren Rückgabecode zurückgegeben, als ein angegebenes Token gelöscht werden sollte. Mit dieser Meldung wird der Rückgabecode (**hhhh**), der von IEANTDL zurückgegeben wurde, sowie der Name des Token (**tokename**) angegeben, der von IEANTDL gelöscht werden sollte. Rückgabecodes für Namen/Token sind in *MVS Programming: Assembler Services Reference, Volume 2 (IARR2V-XCTLX)* dokumentiert.

SLR-0212-E: Not APF authorized

Für den TSO/E-REXX-Adressraum ist keine APF-Autorisierung eingerichtet. Diese Meldung kann auf ein oder mehrere der folgenden Probleme in der JCL-Prozedur hinweisen, die vom Parameter TSOPROC= in der Konfigurationsdatei angegeben ist: (1) Für die TSO/E-REXX-Ladebibliothek ist keine APF-Autorisierung eingerichtet. (2) Der Autorisierungscode des SLRTSREX-Lademoduls ist nicht leer. (3) STEPLIB der Ausführungs-JCL enthält mindestens eine nicht autorisierte Bibliothek. (4) Das Programm SLRTSREX wurde von nicht autorisierter Software aufgerufen.

SLR-0213-E: Unable to retrieve named token, name= tokename

Vom TSO/E-REXX-Adressraum konnte das Token nicht abgerufen werden, dessen Name von **tokename** angegeben ist. Weitere Informationen finden Sie in MSGOUT, beispielsweise den Rückgabecode, der von IEANTRT zurückgegeben wurde.

SLR-0214-E: Terminating due to ALESERV error

Von der Makroanweisung ALESERV EXTRACTH oder ALESERV ADD wurde ein nicht leerer Rückgabecode zurückgegeben. Weitere Informationen finden Sie in MSGOUT, beispielsweise den Wert des ALESERV-Rückgabecodes.

SLR-0215-E: Terminating due to ASEXT error

Von der Makroanweisung ASEXT wurde ein nicht leerer Rückgabecode zurückgegeben. Weitere Informationen finden Sie in MSGOUT, beispielsweise den Wert des ASEXT-Rückgabecodes.

SLR-0216-E: Unable to open TASKLIB

Die JCL-Prozedur, die vom Parameter TSOPROC= in der Konfigurationsdatei angegeben ist, enthält keine gültige Anweisung TASKLIB DD.

SLR-0331-E: Unable to allocate ACL file

Mit dem Parameter CMDLIST= in der Konfigurationsdatei wird der Name einer ACL-Datei (Authorized Command List, Liste autorisierter Befehle) angegeben. Die angegebene ACL-Datei kann jedoch nicht zugeordnet werden. Möglicherweise ist die ACL-Datei nicht vorhanden oder der Agent der uCMDB-Dienste verfügt nicht über die Berechtigung, die ACL-Datei zu lesen. Weitere Informationen finden Sie in MSGOUT, einschließlich des dynamischen Zuordnungsrückgabecodes, des Fehlerursachencodes und des Fehlerinformationscodes.

SLR-0332-E: Unable to de-allocate ACL file

Die dynamische Aufhebung der Zuordnung der ACL-Datei ist fehlgeschlagen. Weitere Informationen finden Sie in MSGOUT, einschließlich des dynamischen Zuordnungsrückgabecodes, des Fehlerursachencodes und des Fehlerinformationscodes.

SLR-0333-E: ddname not allocated

Diese Meldung gibt an, dass der zuvor zugeordnete DD-Name in TIOT nicht mehr vorhanden ist. Wenden Sie sich an den technischen Support, sollte diese Meldung angezeigt werden!

SLR-0334-E: Unable to open ACL file

Die Makroanweisung OPEN hat einen nicht leeren Rückgabecode beim Versuch zurückgegeben, eine zuvor zugeordnete ACL-Datei zu öffnen. Die wahrscheinliche Ursache ist, dass das Format der ACL-Datei vom Agent der uCMDB-Dienste nicht unterstützt wird. Die ACL-Datei muss eine "unstrukturierte" MVS-Datei, ein Element eines partitionierten Datasets (PDS) oder eine EBCDIC-Textdatei in einem hierarchischen Unix Systems Services-Dateisystem sein. Andere Dateiformate wie VSAM werden nicht unterstützt. Weitere Informationen finden Sie in MSGOUT.

SLR-0335-E: Unable to open ACL file

Die Makroanweisung CLOSE hat einen nicht leeren Rückgabecode beim Versuch zurückzugeben, eine zuvor geöffnete ACL-Datei zu schließen. Weitere Informationen finden Sie in MSGOUT.

SLR-0356-E: Stmt nnnnn invalid key xxxxxxxx

Diese Meldung wird angezeigt, wenn eine Anweisung in der Konfigurationsdatei ein unbekanntes Schlüsselwort angibt. In der Meldung werden die Zeilennummer (**nnnnn**) der falschen Anweisung und das fehlerhafte Schlüsselwort (**xxxxxxx**) in der Anweisung angegeben.

SLR-0358-E: Stmt nnnnn has no equal sign

Diese Meldung wird angezeigt, wenn eine Anweisung in der Konfigurationsdatei kein Gleichheitszeichen aufweist. Jede Anweisung in der Konfigurationsdatei, die kein Kommentar ist, muss aus einem Schlüsselwort, gefolgt von einem Gleichheitszeichen, gefolgt von einem Zeichenfolgenwert, der dem angegebenen Schlüsselwort zugewiesen werden soll, bestehen. Eine Anweisung ist ohne das Gleichheitszeichen ungültig. Die Meldung gibt die Zeilennummer (**nnnnn**) der fehlerhaften Anweisung an.

SLR-0359-E: Configuration file exceeds max record count

Die Gesamtanzahl der Datensätze (einschließlich der Kommentare) in der Konfigurationsdatei liegt über 999.

SLR-0362-E: Unable to allocate configuration file

In der Klausel PARM= der EXEC-Anweisung der Ausführungs-JCL wurde der Name einer Konfigurationsdatei angegeben. Die angegebene Datei konnte jedoch nicht zugeordnet werden. Möglicherweise ist die Datei nicht vorhanden oder der Agent der uCMDB-Dienste verfügt nicht über Lesezugriff auf die Datei. Weitere Informationen finden Sie in MSGOUT, einschließlich des dynamischen Zuordnungsrückgabecodes, des Fehlerursachencodes und des Fehlerinformationscodes.

SLR-0363-E: Unable to de-allocate configuration file

Beim Versuch, die Zuordnung einer zuvor zugeordneten Konfigurationsdatei aufzuheben, ist ein Fehler eingetreten. Weitere Informationen finden Sie in MSGOUT, einschließlich des dynamischen Zuordnungsrückgabecodes, des Fehlerursachencodes und des Fehlerinformationscodes.

SLR-0364-E: Unable to find ddname in TIOT

Diese Meldung gibt an, dass der zuvor zugeordnete DD-Name in TIOT nicht mehr vorhanden ist. Wenden Sie sich an den technischen Support, sollte diese Meldung angezeigt werden!

SLR-0365-E: Unable to open configuration file

Die Makroanweisung OPEN hat einen nicht leeren Rückgabecode beim Versuch zurückgegeben, eine zuvor zugeordnete Konfigurationsdatei zu öffnen. Die wahrscheinliche Ursache ist, dass das Format der Konfigurationsdatei vom Agent der uCMDB-Dienste nicht unterstützt wird. Die Konfigurationsdatei muss eine "unstrukturierte" MVS-Datei, ein Element eines partitionierten Datasets (PDS) oder eine EBCDIC-Textdatei in einem hierarchischen Unix Systems Services-Dateisystem sein. Andere Dateiformate wie VSAM werden nicht unterstützt.

SLR-0366-E: Unable to close configuration file

Die Makroanweisung CLOSE hat einen nicht leeren Rückgabecode beim Versuch zurückgegeben, eine zuvor geöffnete Konfigurationsdatei zu schließen.

SLR-0368-W: No configuration file name specified

In der Klausel PARM= der EXEC-Anweisung der Ausführungs-JCL wird kein Name einer Konfigurationsdatei angegeben. Die Klausel PARM= ist leer oder wird ausgelassen. Die standardmäßigen Konfigurationsparameter werden verwendet.

Fehlermeldungen bei der uCMDB-Discovery unter z/OS

<Rückgabe-code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
00010	get_process_vars\getcwd {working_directory}\{errno}\{errnojr}	IBM HTTP Server wurde nicht richtig eingerichtet	<ul style="list-style-type: none"> - Überprüfen Sie {errno} und {errnojr}, sofern vorhanden, auf die Ursache. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen.
00010	Get_process_vars\getpid\ {errno}\{errnojr}	IBM HTTP Server wurde nicht richtig eingerichtet	<ul style="list-style-type: none"> - Überprüfen Sie {errno} und {errnojr}, sofern vorhanden, auf die Ursache. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen.
00020	Invalid HTTP request method<Methode>{meth}	Fehler bei der Einrichtung der uCMDB-Software	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die SSL-Einrichtung.
00100	UserID Required	Der uCMDB-Client wurde nicht für die Übergabe von Benutzer-ID/Kennwort eingerichtet, für den uCMDB-Agent sind diese Angaben jedoch erforderlich	Richten Sie den uCMDB-Client mit Benutzer-ID/Kennwort ein.
00110	Password Required	Der uCMDB-Client hat das Kennwort nicht übergeben	Richten Sie das Kennwort für den uCMDB-Client ein.

Anhang E • Agents für die uCMDB-Discovery

<Rückgabe -code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
00120	Authentication Failed - service {service error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {service error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Wenden Sie sich an den Administrator für Zugriff/ Sicherheit.
00130	SAF {service error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {service error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Wenden Sie sich an den Administrator für Zugriff/ Sicherheit.
00140 *	Authentication Failed - bpxwunix retval={retval}	Sicherheit	Wenden Sie sich an den Administrator für Zugriff/ Sicherheit.
00150 *	Authentication Failed - no response	Sicherheit	Wenden Sie sich an den Administrator für Zugriff/ Sicherheit.
00160 *	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Wenden Sie sich an den Administrator für Zugriff/ Sicherheit.
00500	Invalid Command = {cmd}	Fehler bei der Einrichtung der uCMDB-Software. Fehler bei der SSL-Einrichtung.	- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen.

<Rückgabe-code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
11000	{error message}	Ein Sicherheits- oder Ressourcenproblem	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen.
11010	Für CONNLIST wurde eine leere Meldung zurückgegeben		Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen.
13000	{error message}	Verursacht durch Sicherheits- oder DB2-Zugriffsprobleme	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX. - Überprüfen Sie den DB2-Zugriff für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.
13010	Für DSNTEP2 wurde eine leere Meldung zurückgegeben	Sicherheit oder DB2-Zugriff	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX. - Überprüfen Sie den DB2-Zugriff für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.
14000	{error message}		<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen.

Anhang E • Agents für die uCMDB-Discovery

<Rückgabe -code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
14010	Für GETSUBSYS wurde eine leere Meldung zurückgegeben		- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen.
15000	{error message}		- Überprüfen Sie {error message} .
15010	Für GETMSTJCL wurde eine leere Meldung zurückgegeben		
16000	{error message}		- Überprüfen Sie {error message} .
16010	Für HOMELIST wurde eine leere Meldung zurückgegeben		
17000	{error message}		- Überprüfen Sie {error message} .
17010	Für IMSCMD wurde eine leere Meldung zurückgegeben		
18000	{error message}		
18010	Für LINKLIST wurde eine leere Meldung zurückgegeben		
20000	{error message}	- Sicherheit - Zeitüberschreitung bei Ressourcen	- Überprüfen Sie {error message} . - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.

<Rückgabe -code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
20010	Für LISTPRD2 wurde eine leere Meldung zurückgegeben	<ul style="list-style-type: none"> - Sicherheit - Zeitüberschreitung bei Ressourcen 	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.
21000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
21010	Für LISTPROD wurde eine leere Meldung zurückgegeben	Sicherheit	Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
22000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
22010	Für LISTPROD wurde eine leere Meldung zurückgegeben	Sicherheit	- Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
23000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
23010	Für MAJNODES wurde eine leere Meldung zurückgegeben	Sicherheit	- Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.

<Rückgabe -code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
24000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
24010	Für MVSCMD wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
25000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
25010	Für PAGELIST wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
26000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
26010	Für ROUTE wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.

<Rückgabe -code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
27000	{error message}	Sicherheit oder DB2-Zugriff	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX. - Überprüfen Sie den DB2-Zugriff für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.
27010	Für DSNREXX wurde eine leere Meldung zurückgegeben	Sicherheit oder DB2-Zugriff	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX. - Überprüfen Sie den DB2-Zugriff für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.
28000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.

Anhang E • Agents für die uCMDB-Discovery

<Rückgabe -code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
28010	Für SEARCHPDS wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTERSREX.
29000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRTERSREX.
29010	Für SSILIST wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRTERSREX.
30000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRTERSREX.
30010	Für SYMLIST wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRTERSREX.
31000	{error message}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTERSREX.

<Rückgabe-code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
31010	Für TSOCMD wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie TSO-Sicherheitsinformationen für die Benutzer-ID des uCMDB TSO-Diensts SLRTSREX.
32000	{error message}	Sicherheit Zeitüberschreitung bei Ressourcen	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die Unix System Services-Autorisierung für die IMWEBSRV-Benutzer-ID.
32010	Für UNIXCMD wurde eine leere Meldung zurückgegeben	Sicherheit Zeitüberschreitung bei Ressourcen	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die Unix System Services-Autorisierung für die IMWEBSRV-Benutzer-ID.
33000	{error message}	Sicherheit Zeitüberschreitung bei Ressourcen	<ul style="list-style-type: none"> - Überprüfen Sie {error message}. - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die Unix System Services-Autorisierung für die IMWEBSRV-Benutzer-ID.
33010	Für UPLOAD wurde eine leere Meldung zurückgegeben	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die Unix System Services-Autorisierung für die IMWEBSRV-Benutzer-ID.

<Rückgabe -code>	<Meldung>	Wahrscheinliche Ursache	Maßnahmen
34000	{error message}	Sicherheit Zeitüberschreitung bei Ressourcen	- Überprüfen Sie {error message} . - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
34010	Für XCFLIST wurde eine leere Meldung zurückgegeben	Sicherheit Zeitüberschreitung bei Ressourcen	- Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
35000	{error message}	Sicherheit Zeitüberschreitung bei Ressourcen	- Überprüfen Sie {error message} . - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
35010	Für XCFLIST wurde eine leere Meldung zurückgegeben	Sicherheit Zeitüberschreitung bei Ressourcen	- Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
36000	{error message}	Sicherheit Zeitüberschreitung bei Ressourcen	- Überprüfen Sie {error message} . - Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.
36010	Für XCFLIST2 wurde eine leere Meldung zurückgegeben	Sicherheit Zeitüberschreitung bei Ressourcen	- Überprüfen Sie die Autorisierung der Benutzer-ID des uCMDB-Diensts SLRSERV.

Im Folgenden finden Sie weitere Informationen zu **{error message}**:

{error message}	Wahrscheinliche Ursache	Maßnahmen
do_console_cmd\address tso {cmd}\{errno}\{errnojr}	Sicherheit	- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die TSO-Sicherheitsinformationen für die Benutzer-ID von IMWEBSRV.
do_dsnrexx\address tso {cmd}\{errno}\{errnojr}	- Sicherheit - DB2-Zugriff	- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die TSO-Sicherheitsinformationen für die Benutzer-ID von IMWEBSRV. - Überprüfen Sie die DB2-Autorisierung für die Benutzer-ID von IMWEBSRV.
do_dsntep2\address tso {cmd}\{errno}\{errnojr}	- Sicherheit - DB2-Zugriff	- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die TSO-Sicherheitsinformationen für die Benutzer-ID von IMWEBSRV. - Überprüfen Sie die DB2-Autorisierung für die Benutzer-ID von IMWEBSRV.
do_get_parmmembs\address tso {cmd}\{errno}\{errnojr}	Sicherheit	- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die TSO-Sicherheitsinformationen für die Benutzer-ID von IMWEBSRV.
do_listprod_clist\address tso {cmd}\{errno}\{errnojr}	- Sicherheit - Zeitüberschreitung bei Ressourcen	- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die TSO-Sicherheitsinformationen für die Benutzer-ID von IMWEBSRV.
do_search_pds\address tso {cmd}\{errno}\{errnojr}	- Sicherheit - Zeitüberschreitung bei Ressourcen	- Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die TSO-Sicherheitsinformationen für die Benutzer-ID von IMWEBSRV.

{error message}	Wahrscheinliche Ursache	Maßnahmen
do_tso_cmd \address tso {cmd} \{errno} \{errnojr}	<ul style="list-style-type: none"> - Sicherheit - Zeitüberschreitung bei Ressourcen 	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die TSO-Sicherheitsinformationen für die Benutzer-ID von IMWEBSRV.
do_unix_cmd \bpxwunix {cmd} \{errno} \{errnojr}	<ul style="list-style-type: none"> - Sicherheit - Zeitüberschreitung bei Ressourcen 	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die Unix System Services-Autorisierung für die Benutzer-ID von IMWEBSRV.
{filename} file not opened, error codes {errno} - {errnojr}	Sicherheit	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die Unix System Services-Autorisierung für die Benutzer-ID von IMWEBSRV.
INVALID uCMDB_mode - {uCMDB_mode} lib - {uCMDB_lib}	Fehler bei der uCMDB-Einrichtung	<ul style="list-style-type: none"> - Überprüfen Sie IMWEBSRV SYSOUT auf Fehlermeldungen. - Überprüfen Sie die uCMDB-Einrichtung, einschließlich SSL.
SLRCMD {command} rc={rc}, *** Buffer Overflow : Buffer Size={buffer size} Returned={length}	Kritischer uCMDB-Fehler	
SLRCMD {command} rc={rc}, Console server is not active	Gestarteter SLRSERV-Task wird nicht ausgeführt	<ul style="list-style-type: none"> - Überprüfen Sie, ob der gestartete SLRSERV-Task ausgeführt wird
SLRCMD {command} rc={rc}, Named token not found	Zeitüberschreitung bei Ressourcen	<ul style="list-style-type: none"> - Überprüfen Sie SLRSERV SYSOUT auf Fehlermeldungen.
SLRCMD {command} rc={rc}, Named token services not available	Zeitüberschreitung bei Ressourcen	<ul style="list-style-type: none"> - Überprüfen Sie SLRSERV SYSOUT auf Fehlermeldungen.

{error message}	Wahrscheinliche Ursache	Maßnahmen
SLRCMD {command} rc={rc}, Unable to pass command to console server	Zeitüberschreitung bei Ressourcen	- Überprüfen Sie SLRSERV SYSOUT auf Fehlermeldungen.
SLRCMD {command} rc={rc}, No reply from server within timeout interval	Zeitüberschreitung bei Ressourcen	- Überprüfen Sie SLRSERV SYSOUT auf Fehlermeldungen.
SLRCMD {command} rc={rc}, Non-zero completion code posted in ECB	Zeitüberschreitung bei Ressourcen	- Überprüfen Sie SLRSERV SYSOUT auf Fehlermeldungen.
SLRCMD {command} rc={rc}, Cannot retrieve results from console server	Zeitüberschreitung bei Ressourcen	- Überprüfen Sie SLRSERV SYSOUT auf Fehlermeldungen.
SLRCMD {command} rc={rc}, Unrecognized return code {return code}	Zeitüberschreitung bei Ressourcen	- Überprüfen Sie SLRSERV SYSOUT auf Fehlermeldungen.

Index

A

Ablaufprotokollierung 141, 332
Abwärtskompatibilität von
 TransactionVision-Komponenten 28
Agent für die uCMDB-Discovery
 Betrieb 435
 Komponenten 435
Agent für uCMDB-Mainframe-Dienste
 Konsolenmeldungen 443
Agents
 Abwärtskompatibilität 28
 Übersicht 24
Agents für die uCMDB-Discovery
 Installieren 420
 Sicherheitsanforderungen 437
Agents. Siehe auch Sensoren
Aktualisierungen, Dokumentation 17
Analyzer
 Abwärtskompatibilität 28
 Beans 93
 Deinstallieren 54, 61
 Dienst
 Verwalten 367
 Fehlermodus 95
 Fehlerprotokollierung 91
 in der Bereitstellungsumgebung 25
 Informationen 47
 Korrelieren von Multithread-Servlet-/
 JMS-Ereignissen 93
 Tread-Anzahl 91
 Übersicht 24
 unter Windows installieren 51
 Unterstützte Plattformen 33
 Upgrade 28
 Upgrade-Installation 60
analyzer.log 91, 105
Analyzer.properties 392

analyzer_startup.log 91, 105
APP CTL HEAP SZ 66
APPLHEAPSZ 66
ASP.NET-Applikationen 148
auto_detect.points (Datei) 197

B

Batch MQ
 Applikationen, die überwacht werden
 können 148
BEA Tuxedo-Sensoren
 Applikationen, die überwacht werden
 können 148
 Deinstallieren 253
 Erneutes Binden 253
 Installieren 251
 Übersicht 153
BEA Tuxedo-Server 148
Beans.xml 102, 318
Benutzerdefinierte Benutzerereignisse
 Konfigurieren 197

C

CacheSize.properties 398
CICS-Sensoren
 Applikationen, die überwacht werden
 können 148
 Konfigurieren 233
 Übersicht 153
 Zusätzliche Einstellungen 416
Clientapplikationsüberwachung 307
COLDMON 282
CreateSqlScript 360

D

- Database.properties (Datei) 73, 399
- Datenbank
 - Auflösung für Typ-2-Treiber für DB2 99
 - Auflösung für Typ-2-Treiber für Oracle 100
 - Auflösung für Typ-4-Treiber für DB2 100
 - Auflösung für Typ-4-Treiber für Oracle 101
 - in der Bereitstellungsumgebung 26
 - Konfiguration 63, 399
 - Konfiguration und Optimierung 68
 - Leistung 68
 - Migration 409
 - Reduzieren der Größe 102
 - Speichern von Unicode-Daten 73
 - Übersicht 25
 - Unterstützte Plattformen 34
 - Upgrade 29
 - Verwaltung 360, 363
- Datenbankverwaltung 14
- DB2INSTANCE (Umgebungsvariable) 78, 80
- db2move-Befehl 362
- DB2-Pufferpool 67
- DB2RunStats 363
- DB2Test 70
- DB2Test (Dienstprogramm) 365
- DB2-Variableneinstellungen 66
- DBWriteEventCompressedBean 103
- DBWriteEventDefaultBean 103
- Deaktivieren des strikten lokalen Transaktionsabgleichs 94
- Deinstallieren
 - WebSphere MQ- und User Event-Sensoren unter UNIX 253
- Dokumentation
 - Online 14
- Dokumentationsaktualisierungen 17

E

- EJB-Beans 148
- EJB-Sensoren
 - Applikationen, die überwacht werden

- können 148
 - Übersicht 152
 - Unterstützte Plattformen 39
- Ereignisappender
 - UNIX 109, 143
 - Windows 108, 142
- Ereigniskomprimierungsbean 102
- Ereignisprotokoll 332
- exit_sensor.deny 299
- Exit-Sensoren für WebSphere MQ-APIs auf verteilten Plattformen
 - konfigurieren 299
- Übersicht 150
- unter i5/OS konfigurieren 299
- unter Windows konfigurieren 303
- Verwerfen von WMQ-Ereignissen in TransactionVision-Warteschlange n 305

F

- FASTPATH_BINDING 306
- Fehlerbehebung und Wissensdatenbank 16
- Fehlerprotokollierung 91
- Flash Player-Unterstützung 43

G

- Gestarteter SLRSERV-Task 439
- Gestarteter SLRTSREX-Task 440

H

- Heapeinstellungen für SonicMQ Broker 90
- HP Software-Website 17

I

- IBM HTTP Server 438
 - für die uCMDB-Discovery aktivieren 431
- IMSBridgeObject.xml 318
- IMS-MQ-Sensoren
 - Applikationen, die überwacht werden können 148
- Internationalisierungsunterstützung 43

J

- Java-Agents
 - Ausführen des JRE-Instrumentierers 184
 - Informationen 156
 - Installationsdateien 157
 - Protokollierung 329
 - Starten des Installationsprogramms unter Windows 158
 - Unbeaufsichtigte Installation 183
 - unter UNIX installieren 172
 - unter Windows installieren 157
- Java-Agent-Setupmodul 160
- Java-Servlet-Sensoren
 - Applikationen, die überwacht werden können 148
- Java-Unterstützung 43
- JDBC-Ereignisse 95
- JDBC-Sensoren
 - Datenbankauflösung 95
 - Übersicht 152
 - Unterstützte Plattformen 41
- JDBCSystemModelDefinition.xml 96
- JDBC-URL-Zuordnung 98
- JMSPubSubRelationBean 94
- JMS-Sensoren
 - Korrelieren von Multithread-Ereignissen 93
 - Übersicht 152
 - Unterstützte Plattformen 40
- Jobkonfiguration 403
- JobManager.properties 403
- JRE-Instrumentierer
 - unter UNIX ausführen 191
 - unter Windows ausführen 186
 - Verarbeitungsschritte 185

K

- Komponenten 205
- Konfigurationsdateien 391
- Konfigurationswarteschlangen-Prüfintervall 297
- Konfigurieren des Nachrichtenablaufs 91
- Konventionen, typografische 17

L

- LD_LIBRARY_PATH (Umgebungsvariable) 287
- LIBPATH (Umgebungsvariable) 287
- Lineare Protokollierung 107, 141
- LOCKLIST 67
- Lokale Transaktionen
 - Deaktivieren des strikten Abgleichs 94
- Lokalisierung 43

M

- MAXAPPL 66
- Message Oriented Middleware 148
- MigrateDButility 366
- Migrationsanforderungen 409
- MQ IMS Bridge-Sensoren
 - Applikationen, die überwacht werden können 148
- MQ_CONNECT_TYPE 306

N

- Nachrichtensystemprovider
 - Konfigurieren von SonicMQ 196
 - Konfigurieren von TIBCO EMS 196
 - Konfigurieren von WebLogic JMS 196
 - Konfigurieren von WebSphere MQ 297
- Name der Konfigurationswarteschlange 292
- nanny (Dienstprogramm)
 - Syntax und Beschreibung 367
- .NET Remoting 149
- .NET-Agents
 - Bestimmen der Version 276
 - Deinstallieren 276
 - Installieren 260
 - Konfigurieren 267
 - Übersicht 153
 - Unterstützte Plattformen 42
- NonStop TMF-Sensoren
 - Applikationen, die überwacht werden können 149
 - Beenden 282
 - Informationen 279
 - Installieren 280

Index

- Konfigurieren 283
- Starten 282
- Übersicht 154
- Unterstützte Plattformen 42

NT_EVENT_LOG 108, 142

O

- Onlinedokumentation 14
- Onlineressourcen 16
- Operatorkonsolenprotokoll 332
- OracleRunStats (Dienstprogramm) 370
- OracleTest 70
- OracleTest (Dienstprogramm) 372
- Oracle-Variableneinstellungen 68

P

- PassGen (Dienstprogramm) 374
- PATH (Umgebungsvariable) 287
- Protokolldateien
 - Ablaufprotokollierung 141
 - Konfigurieren 139
 - NT_EVENT_LOG 142
 - SYSLOG 143
 - ui.log 139
- Protokollierung
 - Ablaufverfolgung 332
 - Getrennte Protokolldateien für mehrere Sensoren 333
 - Java-Agents 329
 - Linear 107, 141
 - Mehrere Protokolldateien 333
 - SMTP 109
 - SNMP 111
 - Umlaufprotokollierung 106, 139, 330
 - UNIX-Ereignisappender 336
 - WebSphere MQ-Sensoren 330
 - Windows-Ereignisappender 335
- Proxy-Sensor
 - Unterelemente 326
- ProxySensorDef.xml 325
- Proxy-Sensoren
 - Aktivieren 324
 - Applikationsanforderungen 324
 - Konfigurieren 323

- Konfigurieren der Benutzeroberfläche 328
- Konfigurieren der Definitionsdatei 325
- Optionale Attribute 328
- Übersicht 151

Pufferpool 67

R

- RACF-Autorisierungen
 - Zusätzliche Einstellungen 415
- rebind_sensor (Dienstprogramm) 375
- rebind_tux_sensor (Dienstprogramm) 376
- RUNSTATS
 - DB2 363
 - Oracle 370
- runSupportSnapshot (Dienstprogramm) 378

S

- Sensor.properties 403
- Sensoren
 - Ablaufprotokollierung 332
 - Abwärtskompatibilität 28
 - Clientapplikationsüberwachung 307
 - Laden von WebSphere MQ 291
 - Mehrere Protokolldateien 333
 - Protokollierung 329
 - Typen 149
 - Übersicht 24
 - unter IBM i5/OS installieren 213
- Sensoren. Siehe auch CICS-Sensoren, Java-Agents, .NET-Agents, WebSphere MQ-Sensoren, Servlet-Sensoren, BEA Tuxedo-Sensoren, NonStop TMF-Sensoren
- ServicesManager (Dienstprogramm) 381
- Servlet-Sensoren
 - Korrelieren von Multithread-Ereignissen 93
 - Übersicht 151
- Setup.properties (Datei) 405
- SHLIB_PATH (Umgebungsvariable) 287
- SLRSERV 436

- SMTP-Protokollierung 109
- SNMP-Protokollierung 111
- SonicMQ
 - Konfigurieren 196
- SonicMQ Broker
 - Dienst
 - Verwalten 367
- SonicMQ Domain Manager
 - Dienst
 - Verwalten 367
- SQLServerTest (Dienstprogramm) 386
- StatisticsCache.properties 406
- STOPMON 283
- Strikter lokaler Transaktionsabgleich 94
- STRTMON 283
- SYSLOG 109, 143
- Systemanforderungen 31
- Systemprotokoll 332

- T**
- TIBCO EMS
 - Konfigurieren 196
- Transaction Constructor (Algorithmus) 21
- TransactionVision 253
 - Architektur 23
 - Bereitstellungsumgebung 25
- Tuxedo-Server 148
- TVISION_CONFIG_CHECK_INTERVAL
 - (Umgebungsvariable) 297
- TVISION_CONFIGURATION_QUEUE
 - (Umgebungsvariable) 292
- TVISION_HOME 130
- TVISION_SYSLOG (Umgebungsvariable) 332
- tvisionapiexit 298
- TVisionSetupInfo
 - Erforderliche Informationen 76, 130
 - Geänderte Dateien 76
 - Syntax 131
 - Syntax und Verwendung 387
 - Übersicht 130
- Typografische Konventionen 17

- U**
- uCMDB-Architektur 435
- uCMDB-Discovery-Dienst
 - Zusammenfassung der Befehle 441
- uCMDB-Discovery-Dienste 433
- UI.properties 407
- UI/Job Server
 - Abwärtskompatibilität 28
 - Dienst
 - Verwalten 367
 - in der Bereitstellungsumgebung 25
 - Konfigurieren von Protokolldateien 139
 - Übersicht 23
 - Upgrade 29
- Umgebungsvariablen
 - LD_LIBRARY_PATH 287
 - LIBPATH 287
 - SHLIB_PATH 287
- Umlaufprotokollierung 106, 139, 330
- Unicode-Daten 73
- UNIX
 - Ereignisappender 109
- Unterelemente des Proxy-Sensors 326
- Upgrade 201
- User Event-Sensor
 - unter Windows bearbeiten 203
- User Event-Sensoren
 - unter UNIX deinstallieren 210
 - unter UNIX installieren 207
 - unter Windows aktualisieren 201
 - unter Windows deinstallieren 205

- W**
- WBI
 - Installation benutzerdefinierter Brokerknoten 321
 - Integration 320
- Webbrowser
 - Unterstützte Konfigurationen 42
- WebLogic JMS
 - als Nachrichtensystemprovider konfigurieren 196
- Website der HP Software-Unterstützung 16
- WebSphere Business Integration-Sensoren
 - Übersicht 151
- WebSphere Message Broker

Index

- Unterstützte Plattformen 37
- WebSphere MQ
 - Konfigurieren des Nachrichtensystemproviders 297
 - Konfigurieren von Nachrichtensystem Providern
 - Konfigurieren von WebSphere MQ 195
- WebSphere MQ IMS Bridge-Sensor
 - Übersicht 151
- WebSphere MQ-Sensor
 - Protokollierung 330
- WebSphere MQ-Sensorbibliothek
 - Übersicht 150
- WebSphere MQ-Sensoren
 - Applikationen, die überwacht werden können 148
 - Übersicht 150
 - unter AIX erneut binden 210
 - unter UNIX deinstallieren 210
 - User Event-Sensoren
 - unter UNIX deinstallieren 210
 - unter UNIX installieren 207
 - unter Windows aktualisieren 201
 - unter Windows bearbeiten 203
 - unter Windows deinstallieren 205
 - Unterstützte Plattformen 35
- Windows
 - Ereignisappender 108
- Wissensdatenbank 16
- WMQ Batch-Sensoren
 - Konfigurieren und starten 233
- WMQ IMS Bridge-Sensoren
 - Verwenden 313
- wmqsensor 288

Z

- z/OS
 - Zusätzliche Einstellungen 415
- z/OS WebSphere MQ-Sensoren
 - Übersicht 150