



# webMethods Process Performance Manager OPERATION GUIDE

Version 9.10

April 2016

Dieses Dokument gilt für PPM ab Version 9.10. Hierin enthaltene Beschreibungen unterliegen Änderungen und Ergänzungen, die in nachfolgenden Release Notes oder Neuauflagen bekanntgegeben werden.

Urheberrechtlich geschützt © 2000 - 2016 Software AG, Darmstadt, Deutschland und/oder Software AG USA Inc., Reston VA, USA und/oder ihre Tochtergesellschaften und/oder ihre Lizenzgeber.

Der Name Software AG und die Namen der Software AG Produkte sind Marken der Software AG und/oder Software AG USA Inc., einer ihrer Tochtergesellschaften oder ihrer Lizenzgeber. Namen anderer Gesellschaften oder Produkte können Marken ihrer jeweiligen Schutzrechtsinhaber sein. Genaue Informationen über die geschützten Marken und Patente der Software AG und ihrer Tochtergesellschaften sind veröffentlicht unter <http://softwareag.com/licenses>.

Die Nutzung dieser Software unterliegt den Lizenzbedingungen der Software AG. Diese Bedingungen sind Bestandteil der Produktdokumentation und befinden sich unter <http://softwareag.com/licenses> und/oder im Wurzelverzeichnis des lizenzierten Produkts.

Diese Software kann Teile von Software-Produkten Dritter enthalten. Urheberrechtshinweise, Lizenzbestimmungen sowie zusätzliche Rechte und Einschränkungen dieser Drittprodukte können dem Abschnitt „License Texts, Copyright Notices and Disclaimers of Third Party Products“ entnommen werden. Diese Dokumente enthalten den von den betreffenden Lizenzgebern oder den Lizenzen wörtlich vorgegebenen Wortlaut und werden daher in der jeweiligen Ursprungssprache wiedergegeben. Für einzelne, spezifische Lizenzbeschränkungen von Drittprodukten siehe PART E der Legal Notices, abrufbar unter dem Abschnitt „License Terms and Conditions for Use of Software AG Products / Copyrights and Trademark Notices of Software AG Products“. Diese Dokumente sind Teil der Produktdokumentation, die unter <http://softwareag.com/licenses> oder im Verzeichnis der lizenzierten Produkte zu finden ist.

## Inhalt

1	Textkonventionen .....	1
2	Allgemeines .....	2
3	IT Service und Applikationsmanagement .....	3
3.1	Frühwarnsystem bei Speicherknappheit .....	3
3.1.1	Speicherüberwachung .....	3
3.1.2	Systemmeldungen .....	4
3.1.3	Konfiguration .....	5
3.2	Workload Monitoring .....	6
3.3	Maintenance-Konzept .....	7
3.3.1	Systemaktualisierung .....	7
3.3.2	Archivierung .....	7
3.3.3	Rücksichern .....	8
3.4	Hochverfügbarkeitssysteme .....	8
3.5	Support-Desk-Management-Konzept .....	9
4	Third-Party-Software .....	12
5	Performance Aspekte .....	13
6	Administration .....	15
6.1	Cloud Agent .....	15
6.2	Systemstatus .....	15
6.3	Konfiguration .....	16
6.4	Systemmeldungen .....	19
7	PPM Systemmeldungen .....	20
7.1	Überblick .....	20
7.2	Verzeichnisse der log-Dateien .....	22
7.3	Systemmeldungen (log-Ausgaben) .....	23
7.4	Audit-Meldungen (Audit-logging) .....	28
7.5	Kommandozeilenprogramme .....	33
8	Globalisierung .....	36
9	Auftragsautomatisierung .....	37
10	Barrierefreiheit .....	38
11	SSO-Integration .....	39
11.1	Parameter konfigurieren .....	39
11.1.1	SAML2-Anbindung .....	39
11.1.2	Kerberos .....	40
12	Sicherheitsaspekte .....	41
12.1	Sicherheit der Kommunikationskanäle .....	41
12.2	Load Balancer .....	41
12.2.1	SSL-Zertifikat einbinden .....	41
12.2.2	Weitere Absicherung des Load Balancer .....	42
12.3	PPM Server .....	43
12.3.1	SSL-Keystore erzeugen und signieren .....	43
12.3.2	Absicherung des Server Systems .....	44

13	Kommandozeilenprogramme .....	45
13.1	Allgemeiner Aufbau und funktionsspezifische Struktur.....	45
13.2	PPM-Kommandozeilenprogramme .....	46
13.3	Gemeinsame Argumente .....	46
13.4	runinitdb .....	48
13.5	runppmanalytics.....	49
13.5.1	Argumente runppmanalytics .....	50
13.6	runppmconfig.....	51
13.6.1	Argumente runppmconfig .....	51
13.6.2	Export der Systemkonfiguration .....	55
13.6.3	Import und Export von Benutzerdaten.....	55
13.6.4	Import und Export von Favoriten .....	57
13.7	runppmcpic.....	59
13.7.1	Argumente runppmcpic .....	59
13.8	runppmimport.....	61
13.8.1	Argumente runppmimport .....	61
13.8.2	Neuberechnung.....	65
13.9	runprocessimport .....	65
13.9.1	Argumente runprocessimport.....	66
13.9.2	Importeinstellungen .....	67
13.9.3	Import-Logging.....	68
13.9.4	Import mehrere Datenquellen .....	68
13.10	runppmcompress.....	69
13.10.1	Argumente runppmcompress .....	70
13.11	runppmdelete .....	72
13.11.1	Argumente runppmdelete .....	73
13.12	runppmadmin .....	75
13.12.1	Argumente runppmadmin.....	75
13.13	runppsendmail .....	77
13.13.1	Argumente runppsendmail.....	77
13.14	runtableimport.....	78
14	Häufig gestellte Fragen .....	79
14.1	Systemintegration .....	79
14.1.1	Datenbankanbindung .....	83
14.1.2	Prozessextraktor SAP-2-PPM .....	85
14.2	Systemadministration .....	87
14.2.1	PPM-Kommandozeilenprogramme: Protokollausgaben .....	92
14.2.2	Inkonsistenz der Datenbestände.....	94

## 1 Textkonventionen

Im Text werden Menüelemente, Dateinamen usw. folgendermaßen kenntlich gemacht:

- Menüelemente, Tastenkombinationen, Dialoge, Dateinamen, Eingaben usw. werden **fett** dargestellt.
- Eingaben, über deren Inhalt Sie entscheiden, werden **<fett und in spitzen Klammern>** dargestellt.
- Einzeilige Beispieltex te werden am Zeilenende durch das Zeichen ↵ getrennt, z. B. ein langer Verzeichnispfad, der aus Platzgründen mehrere Zeilen umfasst.
- Dateiauszüge werden in folgendem Schriftformat dargestellt:

Dieser Absatz enthält einen Dateiauszug.

## 2 Allgemeines

Dieses Dokument versteht sich als Leitfaden zur Systemintegration der PPM-Produktkomponenten **webMethods Process Performance Manager**, **PPM Customizing Toolkit** und **PPM Prozessextraktoren** in eine komplexe Systemlandschaft. Es beschreibt bestimmte Aspekte, die Sie bei der Einführung in eine existierende Systemlandschaft unterstützen sollen.

Des Weiteren enthält das Dokument Verweise zu weiterführenden, komponentenspezifischen Dokumenten.

## 3 IT Service und Applikationsmanagement

webMethods Process Performance Manager ist ein Werkzeug zur Analyse von real existierenden Prozessen. Die Daten der Arbeitsschritte dieser Prozesse werden aus Anwendungssystemen ausgelesen und zu Prozessinstanzen zusammengefügt. Anschließend werden Kennzahlen für diese Prozessinstanzen berechnet. PPM ermöglicht zeitnah einen umfassenden Überblick über die Prozesse eines Unternehmens und unterstützt Sie beim Erkennen von Schwachstellen.

Als Repository verwendet PPM ein SQL-RDBMS, das alle Konfigurationen und Daten speichert. PPM ist als Client-Server-Applikation in Java entwickelt.

### HOCHVERFÜGBARKEIT

PPM ist eine analytische Applikation, die Daten importiert, berechnet und in einem Datenbankschema speichert. Die Integrität der Daten muss vom verwendeten Datenbanksystem gewährleistet werden. Durch erneutes Einlesen der Nutzdaten lässt sich ein bestimmter Systemzustand jederzeit wiederherstellen. PPM selbst unterstützt direkt weder Hochverfügbarkeitskriterien noch Adaptive Computing-Konzepte.

### 3.1 Frühwarnsystem bei Speicherknappheit

Der PPM-Analyseserver kann beim Starten maximal die in den Mandanteneigenschaften angegebene Menge an Hauptspeicher belegen. Wird diese Grenze erreicht, kommt es zu Speicherfehlern (Out of Memory). Diese Speicherfehler können zu Instabilitäten des Analyseservers führen, was besonders beim Datenimport kritisch ist. Weiterhin können die Speicherfehler bewirken, dass der Datenbestand des Analyse- und Mandantenservers voneinander abweichen, wodurch eine Reinitialisierung des Analyseservers erforderlich wird. Die Reinitialisierung des Analyseservers sollte vermieden werden, da bei großen Datenbeständen die Reinitialisierung des Analyseservers bis zu mehreren Stunden dauern kann. Andererseits verschlechtert sich die Leistung des Analyseservers bereits bei hoher Speicherauslastung durch häufiges Ausführen interner Verwaltungsaufgaben des Arbeitsspeicher (Garbage Collection).

#### 3.1.1 Speicherüberwachung

Um der Leistungsverschlechterung und den Speicherfehlern mit der verbundenen Instabilität bei Speicherknappheit vorzubeugen, wird der Speicherverbrauch des Analyseservers überwacht. Es wird rechtzeitig gemeldet, wenn die Gefahr besteht, dass sich die Leistung des Servers durch Speicherknappheit verschlechtern könnte und Speicherfehler drohen. Die Überprüfung des Speicherverbrauchs des Analyseservers benötigt bestimmte Systemressourcen. Um die Auswirkung auf die Gesamtleistung des Systems gering zu halten, findet die Überprüfung der Speicherauslastung zyklisch und bei bestimmten Systemaktivitäten statt:

- Datenimport
  - Bei Beginn des mit den Kommandozeilenprogrammen **runppmimport** oder **runprocessimport** gestarteten Imports neuer Daten (oder spezifische Neuberechnung mit Paramset-Angabe)
  - Bevor beim Import neuer Daten berechnete EPKs in die Datenbank geschrieben werden.

Dadurch wird erreicht, dass die EPKs, deren berechnete Daten nicht mehr im Analyseserver gespeichert werden können, auch nicht in die Datenbank geschrieben werden und Analyseserver und Wiederherstellungsdateien konsistent bleiben.

- Am Ende des Datenimports

- Zu Beginn und beim Beenden des Bearbeitungsmodus der manuellen Dateneingabe prozessinstanzunabhängiger Kennzahlen
- Zu Beginn und beim Beenden des Einlesen prozessinstanzunabhängiger Kennzahlen mittels Kommandozeilenprogramm **runpikidata** und bevor eingelesene PIKI-Datenblöcke in die Datenbank geschrieben werden.

### 3.1.2 Systemmeldungen

Die Überprüfung der Speicherauslastung des Systems kann zu folgenden Ergebnissen führen:

- normal (Regelfall)  
Es steht noch genügend Speicherplatz zur Verfügung, um weiterhin uneingeschränkt Daten zu importieren.
- ungünstig  
Der noch frei zur Verfügung stehende Speicher ist so gering, dass mit Leistungsver schlechterung zu rechnen ist. Auch ist absehbar, dass bei weiterem Datenimport bald die kritische Schwelle erreicht wird und keine Daten mehr importiert werden können.
- kritisch  
Der freie Speicherplatz ist so gering, dass keine Daten mehr importiert werden können. In diesem Fall ist kein Datenimport mehr möglich, ein aktiver Datenimport wird abgebrochen. Ab jetzt ist das System für einen weiteren Datenimport gesperrt.

Ungünstige oder kritische Systemzustände werden bei der Überprüfung als Warnung in den Log-Ausgaben des Analyseservers ausgegeben.

Wenn das System durch kritische Speicherauslastung für einen weiteren Datenimport gesperrt wurde, können Sie diese Sperre durch folgende Maßnahmen aufheben:

- Stoppen des Mandanten- und Analyseservers und Speicherbelegung des Analyseserver erhöhen. Anschließend Analyse- und Mandantenserver wieder starten.
- Verdichten oder Löschen einer ausreichenden Menge von Daten

Wenn während des Datenimport das System durch kritische Speicherauslastung gesperrt wurde, können Sie diesen Zustand nicht direkt durch Verdichten oder Löschen von Daten aufheben. Starten Sie stattdessen die PPM-Server mit ausreichend hoher Speicherbelegung des Analyseservers neu und führen Sie den abgebrochenen Datenimport durch erneutes Ausführen des Kommandozeilenprogramms **runppmimport** bzw. **runprocessimport** zu Ende. Anschließend können Sie dann verdichten oder löschen. Abschließend können Sie die Speicherbelegung wieder auf den ursprünglichen Wert ändern.

#### Tipp

Durch Ausführen des Kommandozeilenprogramms **runppmadmin** mit der Option **-memoryinfo** können Sie jeder Zeit eine manuelle Überprüfung der Speicherauslastung durchführen.



### 3.1.3 Konfiguration

Die Konfiguration der Überwachung der Systemspeicherauslastung (auch MemoryLoadGuard genannt) wird durch bestimmte Schlüsselwerte in der Konfigurationsdatei

**Analyseserver\_settings.properties** festgelegt. Wenn einer dieser Schlüssel nicht angegeben ist oder ungültige Werte enthält, wird in den Log-Ausgaben des Analyseservers eine Warnung ausgegeben und für diesen Schlüssel ein Vorgabewert gesetzt.

Der Schlüssel **MemoryLoadGuard.Enabled** bestimmt, ob die Überwachung der Systemspeicherauslastung aktiv ist oder nicht. Mögliche Werte sind **true** und **false**. Vorgabewert ist **true**.

Sie können die Konfiguration während des Betriebs des Analyseservers ändern. Bei der nächsten Überprüfung der Speicherauslastung des Systems werden die geänderten Werte berücksichtigt.

#### SCHWELLWERTE

Die beiden Schwellwerte, die die Grenze zu ungünstiger und kritischer Speicherbelegung festlegen, werden als Prozentsatz des maximal zur Verfügung stehenden Speichers des Analyseservers angegeben. Ausschlaggebend ist die Speicherauslastung der Old Generation (Speicherbereich, in dem Java Objekte persistiert sind. Die Java-Dokumentation spricht von Committed Size, dem Verhältnis von verwendeter und maximaler Größe.)

Werte kleiner 50% werden ignoriert, da sie nicht praktikabel sind.

Der Schlüssel **MemoryLoadGuard.Warn.PercentageOfMemoryUsed** gibt die Grenze an, ab der die ungünstige Speicherbelegung beginnt. Vorgabewert ist 90%. Dieser Wert ist so gewählt, dass er effizient mit der internen Komponente **Garbage Collection** der Java-Laufzeitumgebung zusammenarbeitet. Es kann frühzeitig auf bevorstehende Speicherknappheit des Systems reagiert werden.

Der Schlüssel **MemoryLoadGuard.PreventImport.PercentageOfMemoryUsed** gibt die Grenze an, ab der die kritische Speicherbelegung beginnt. Der Vorgabewert 95% ist so gewählt, dass eine kritische Speicherknappheit frühzeitig erkannt wird und auch größere Datenvolumen noch sicher importiert werden können.

#### TAGESZEITEN

Die zyklische Überprüfung der Speicherauslastung kann auf bestimmte Tageszeiten beschränkt werden, um die gesamte Leistung des Analyseservers für die aktuelle Sitzung zur Verfügung zu haben. Start- und Endzeitpunkt, zwischen denen die zyklische Überprüfung der Speicherauslastung stattfindet, werden in den Schlüsseln

**MemoryLoadGuard.BackgroundCheck.TimeWindow.Start** und

**MemoryLoadGuard.BackgroundCheck.TimeWindow.End** angegeben. Die Zeitpunkte werden im 24-Stundenformat **hh:mm** angegeben (ISO 8601). Dabei gilt die Systemzeit des Analyseservers. Sie können die Werte im laufenden Betrieb des Analyseservers ändern.

- Eine begonnene Überprüfung der Speicherauslastung wird auf jeden Fall zu Ende geführt, auch wenn dies länger dauert als der angegebene Endzeitpunkt. Die durch bestimmte Systemaktivitäten ausgelöste Überprüfung der Speicherauslastung wird unabhängig von der zyklischen Überprüfung durchgeführt.

- Wir empfehlen die Überprüfung in einen Zeitraum zu legen, in dem möglichst wenige Anfragen an den Server kommen.

### Beispiel

Der folgende Dateiauszug zeigt die Standardkonfiguration des Frühwarnsystems bei Speicherknappheit. Das System ist aktiv, die Grenze für ungünstige Speicherauslastung ist 90%, die Grenze für kritische Speicherauslastung ist 95%. Die zyklische Überprüfung der Speicherauslastung ist aktiv zwischen 8 Uhr abends und 6 Uhr morgens des folgenden Tages.

```
#### Memory Load Guard ####
MemoryLoadGuard.Enabled=true
MemoryLoadGuard.Warn.PercentageOfMemoryUsed=90
MemoryLoadGuard.PreventImport.PercentageOfMemoryUsed=95
# Time window where background checks may occur.
MemoryLoadGuard.BackgroundCheck.TimeWindow.Start=20:00
# End minute is not included. However, if both Start and End are set to the same value
checks are possible all day.
MemoryLoadGuard.BackgroundCheck.TimeWindow.End=06:00
```

## 3.2 Workload Monitoring

Zur Überwachung der PPM-Serverprozesse können Sie die Überwachungsprogramme des Betriebssystems verwenden. Für Windows-Systeme gibt es die Standardprogramme **Taskmanager**, **Perfomancemonitor** und ab Windows Vista/Server 2008 zusätzlich das Programm **Resourcenmonitor**. Welche Überwachungsprogramme für Linux verfügbar sind, hängt von der Distribution und der Installationsvariante ab.

Die Belegung von Systemressourcen durch die PPM-Server-Prozesse lässt sich sehr gut mit dem plattformunabhängigen Java-Programm **Visual VM** überwachen.

### JAVA VISUAL VM

Dieses Programm visualisiert in verschiedenen Ansichten die Belegung bestimmter Systemressourcen wie z. B. CPU- und Speicherauslastung. Hierbei wird nur die vom Java-Prozess (z. B. PPM-Server) innerhalb der Java-VM belegte Systemlast analysiert. Das Programm **Visual VM** ist Bestandteil des Java-SDK 1.7. Zum Starten unter Windows führen Sie die Datei **jvisualvm.exe** im Unterverzeichnis **bin** der SDK-Installation aus. Zum Starten unter Linux führen Sie die Datei **jvisualvm** im Unterverzeichnis **bin** der SDK-Installation aus. Visual VM verwendet die Standard-JMX-Funktionalität der Java VM. PPM selbst unterstützt JMX nicht.

Visual VM kann nur Java-Prozesse analysieren, die mit der selben Benutzerkennung gestartet wurden, mit der auch das Programm **jvisualvm** gestartet wurde. Ein einfacher Weg, um auch Java-Prozesse zu überwachen, die als Windows Dienst gestartet wurden, ist, über die Systemsteuerung von Windows den entsprechenden Dienst so zu konfigurieren, dass dieser mit der Kennung des Benutzers ausgeführt wird, der das Programm **jvisualvm** startet.

Mit Visual VM können Sie auch Java Prozesse überwachen, die auf einem entfernten Rechner ausgeführt werden. Starten Sie hierfür auf dem Rechner, auf dem die zu überwachenden Java-Prozesse laufen, das Programm **jstatd** im Verzeichnis **bin** der SDK-Installation.

Anschließend können auf entfernten Rechnern gestartete Visual VM-Programme eine Verbindung mit **jstatd** aufbauen und lokale Java-Prozesse überwachen.

## 3.3 Maintenance-Konzept

### 3.3.1 Systemaktualisierung

Sie können nach der Installation der initialen Release Version von PPM und MashZone mit Hilfe des Software AG Installer (siehe Dokumentation **Installing webMethods and Intelligent Business Operations Products**) das Produkt nur mit Hilfe des Software AG Update Managers aktualisieren. Produktaktualisierungen werden im Allgemeinen als **Official Fix** bezeichnet, die meist wichtige Fehlerbehebungen für das Produkt enthalten. Wir empfehlen dringend, nach erfolgter Installation der Release Version auch bereits verfügbare Official Fixes mit Hilfe des Software AG Update Managers einzuspielen.

Eine detaillierte Beschreibung der Vorgehensweise zur Aktualisierung erhalten Sie im Handbuch **PPM Installation**.

### 3.3.2 Archivierung

Alle auf Basis der importierten Quellsystemdaten erzeugten Prozessinstanzen, prozessinstanzunabhängige Kennzahlen, mittels Konfigurationsprogramm **runppmconfig** eingelesenen Konfigurationen (Customizing) sowie vom Benutzer über das Frontend bearbeitete bzw. erweiterte Konfigurationen (z. B. PPM-Benutzer, Benutzerdefinierte Kennzahlen) werden in der Datenbank des PPM-Mandanten gespeichert. Sie sind automatisch mit einem Backup des Datenbankschemas gesichert. Führen Sie daher regelmäßig Sicherungen der Datenbank aus.

Standardmäßig werden die Konfigurationsdateien eines PPM-Mandanten mandantenspezifisch in den Verzeichnissen **config** und **custom** unter

`<Installationsverzeichnis>\ppmmashzone\server\bin\work\data_ppm\` gespeichert. Sichern Sie diese Verzeichnisse, wenn Sie Änderungen an der Konfiguration vorgenommen haben.

Sichern Sie das Verzeichnis **patch** im Verzeichnis

`<Installationsverzeichnis>\ppmmashzone\server\bin\work\data_ppm`. In diesem Verzeichnis werden Laufzeitkomponenten gespeichert, die die PPM-Funktionalität global anpassen. Die Mandantenspezifische Laufzeitkomponenten zur Anpassung der PPM-Funktionalität werden im Verzeichnis

`<Installationsverzeichnis>\ppmmashzone\server\bin\work\data_ppm\custom\<Mandant>\patch` gespeichert. Durch Sichern des Verzeichnisses **custom** werden diese automatisch mit archiviert.

Das PPM-Verzeichnis des Web-Servers und alle anwenderspezifischen Reportdefinitionen müssen manuell gesichert werden. Den Speicherort des Web-Serververzeichnisses, den Sie zum Installationszeitpunkt angegeben haben, erfahren Sie aus dem Schlüssel

**ManagementViews.BaseDirectory** in der mandantenspezifischen Konfigurationsdatei **Templates\_settings.properties**.

Wenn Performance Dashboard zur http-basierten Publikation Ihrer Analyseergebnisse (Dashboards und Management Views) installiert haben, können Sie die Konfigurationen aller Mandanten durch Sichern des Verzeichnisses

`<Installationsverzeichnis>\ppmmashzone\server\bin\work\data_ppm\webapp\WEB-INF\config` sichern.

Wenn Sie alle globalen und mandantenspezifischen Konfigurationen und Daten sichern möchten, dann erstellen Sie eine Sicherungskopie des Verzeichnisses **data\_ppm** inklusive aller Unterverzeichnisse.

### 3.3.3 Rücksichern

Stellen Sie in einem Crash-Recovery-Fall die PPM-Installation durch Neuinstallation mit den gleichen Optionen, die Sie bei der ursprünglichen Installation verwendet haben, wieder her. Beachten Sie bitte, dass benötigte 3rd-Party-Software (ggf. JRE für PPM bzw. Tomcat für Performance Dashboard) bereits installiert sein muss. Anschließend müssen Sie mittels CTK alle Mandanten mit den ursprünglichen Einstellungen wieder anlegen. Stellen Sie ggf. die Datenbanken aller PPM-Mandanten wieder her. Abschließend können Sie in umgekehrter Reihenfolge wie im Kapitel **Archivierung** (Seite 7) beschrieben die archivierten Dateien an den jeweiligen Speicherort zurück kopieren und das PPM-System starten.

## 3.4 Hochverfügbarkeitssysteme

### ÜBERBLICK

Hochverfügbarkeitssysteme werden eingesetzt, um das Risiko eines Gesamtsystemausfalls, bedingt durch Ausfall einer einzelnen Komponente (SPOF, Single-Point-Of-Failure), zu minimieren. Solche Systeme beruhen i.d.R. auf redundanten Einzelsystemen, d. h. mehrere gleichartige Systeme mit identischem Datenbestand werden parallel betrieben. Eines dieser Einzelsysteme ist das Hauptsystem, auf das im Normalfall von den Anwendern zugegriffen wird. Ein weiteres System ist das Reservesystem. Fällt das Hauptsystem aus, wird automatisch auf das Reservesystem umgeschaltet. Dabei spielt es keine Rolle, ob der Ausfall durch eine plötzliche, nicht vorhersehbare Ursache (z. B. Hardware-Defekt) oder durch geplante Wartungsaktivitäten verursacht wurde.

Zur Feststellung eines Fehlers kann eine bestimmte Meldung des System-Monitoring verwendet werden (z. B. das S.M.A.R.T.-Protokoll zum Erkennen von Festplattenfehlern). Ergänzt wird dieses Verfahren durch regelmäßige Systemanfragen, die korrekt beantwortet werden müssen. Diese Art der Anfragen nennt man auch Heartbeats. Heartbeats können auf verschiedenen Systemebenen aktiv sein, bspw. bestimmte Applikationsanfragen (z. B. application pings) oder Systemressourcenüberwachung, wie z. B. Hardware-Monitoring. Der Fehlerfall tritt beim Auftreten einer Systemfehlermeldung oder beim Ausbleiben der Antwort auf ein Heartbeat ein. Automatisch wird in diesem Fall auf das Reservesystem umgeschaltet und eine Nachricht an den Systemadministrator verschickt.

Nach Umschalten auf das Ersatzsystem kann der Fehler des Hauptsystems analysiert und behoben werden. Wenn der Fehler behoben ist, kann nach Synchronisation des Datenbestandes wieder auf das Hauptsystem umgeschaltet werden. Hochverfügbarkeitssysteme unterscheiden die beiden Strategien Hot-Standby und Cold-Standby.

Hot-Standby bedeutet, dass ein System im Fehlerfall nach außen immer ohne Unterbrechung auch während einer aktiven Sitzung verfügbar bleibt. Wenn das Hauptsystem ausfällt und ein Ersatzsystem aktiviert wird, merkt der Anwender nichts von diesem Vorgang. Die Umschaltung

auf das Ersatzsystem erfolgt verzögerungsfrei und mit Beibehaltung aktiver Anwendersitzungen. Diese Strategie wird überwiegend für sogenannte mission critical Systeme eingesetzt, bei denen z. B. Sicherheit der Allgemeinheit und Gesundheit von Menschen bedroht sind.

Im Gegensatz zu Hot-Standby erlaubt Cold-Standby bei Ausfall des Hauptsystems eine bestimmte Zeitspanne, währenddessen das Reservesystem aktiviert wird und die Umschaltung auf das Reservesystem erfolgt. Während der Umschaltungsphase ist das System nicht verfügbar. Das bedeutet auch, dass nicht garantiert ist, dass aktive Anwendersitzungen wieder aufgenommen werden können, wenn das System nach Umschalten auf das Ersatzsystem wieder zur Verfügung steht.

### PPM HOCHVERFÜGBARKEITSSYSTEM

PPM ist eine analytische Applikation, die Daten importiert, berechnet und in einem Datenbankschema speichert. Die Integrität der Daten muss vom verwendeten Datenbanksystem gewährleistet werden. Durch erneutes Einlesen der Nutzdaten lässt sich ein bestimmter Systemzustand jederzeit wiederherstellen. PPM selbst unterstützt direkt weder Hochverfügbarkeitskriterien noch Adaptive Computing-Konzepte, wie die Wiederherstellung einer unterbrochenen Sitzung zwischen PPM-Server und -Client. Indirekt lassen sich aber Szenarien realisieren, in denen zwischen dem Hauptsystem und einem vorhandenen Reservesystem umgeschaltet werden kann. Dabei sind aber jedoch bestimmte Ausfallzeiten in Kauf zu nehmen, in denen das PPM-System nicht verfügbar ist (Cold-Standby).

Zur Einrichtung eines solchen PPM-Hochverfügbarkeitssystems empfiehlt sich die 3-Ebenen-Systemkonzeption. Das heißt, der von PPM verwendete Datenbankserver wird in ein eigenständiges System ausgelagert, das bereits Hochverfügbarkeitskriterien entspricht. I.d.R. bieten die Hersteller von Datenbanken für ihre Produkte auch Hochverfügbarkeitsversionen an. Die übrigen Komponenten des PPM-Systems werden auf einem weiteren System installiert und betrieben.

Archivieren Sie in regelmäßigen Abständen wie im Kapitel **Archivierung** (Seite 7) beschrieben das PPM-System. Bei einem Ausfall des PPM-Systems verfahren Sie für ein neues System mit vergleichbaren Hardware-Eigenschaften wie im Kapitel **Rücksichern** (Seite 8) beschrieben.

## 3.5 Support-Desk-Management-Konzept

Treten im laufenden Betrieb unerwartetes Verhalten oder Fehler auf, sollten Sie zunächst die log-Ausgaben des Systems prüfen. Meist findet sich dort ein Hinweis, der das unerwartete Verhalten erklärt oder die Ursache des Fehlers erkennen lässt. Das PPM-System verwendet die Schnittstelle **log4J** zur Ausgabe von Systemmeldungen. Dadurch ist es leicht möglich, während der Laufzeit des PPM-Servers die Konfiguration der Ausgabe von Systemmeldungen durch Anpassen der log-Level zu ändern (verfeinern), um detailliertere Informationen über die Fehlerursache zu erhalten. Die Konfiguration der Ausgabe von Systemmeldungen ist ausführlich im Kapitel **PPM Systemmeldungen** (Seite 20) beschrieben. Die Anbindung an externe Support-Desk-Systeme, z. B. SAP Solution Manager, wird von PPM nicht unterstützt.

Treten beim Importieren von Daten oder Konfigurationen Fehler auf, sind die log-Ausgaben des PPM-Systems für die Analyse sehr wichtig. Zunächst sollten Sie die Meldungen des

Importprogramms auf der Konsole bzw. in der angegebenen log-Datei analysieren. In vielen Fällen kann die Ursache auf diese Weise bereits ermittelt werden. Die Handhabung der Ausgabe von Systemmeldungen der Kommandozeilenprogramme ist detailliert im Kapitel **Kommandozeilenprogramme** (Seite 33) beschrieben. Die Systemmeldungen des PPM-Mandanten- und Analyseservers können die Fehleranalyse unterstützen. PPM-Mandanten- und Analyseserver schreiben ihre Systemmeldungen jeweils in eine Datei. Der Speicherort dieser Dateien ist über die entsprechenden Appender in der Datei **Server\_Log\_settings.properties** bzw. **Analysisserver\_Log\_settings.properties** angegeben. Voreingestellt ist das Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\log\<Mandant>**, die log-Dateien haben das Namensmuster **<Mandant>\_<Typ>.log**, log-Dateien des Analyseserver haben das Suffix **\_as** nach dem Mandantennamen. Beispielsweise schreibt der Analyseserver des Mandanten **umg\_en** Fehlermeldungen standardmäßig in die Datei **umg\_en\_as\_error.log** im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\custom\umg\_en\log\**.

Die Systemmeldungen der PPM-Server können auf verschiedene Weise weiter verarbeitet werden, z. B. durch Verwendung eigener Appender. Die Konfiguration der Ausgabe von Systemmeldungen ist detailliert im Kapitel **Systemmeldungen (log-Ausgaben)** (Seite 23) beschrieben.

Bei der Analyse mittels PPM-Frontend auftretende Fehler werden in der Sitzung des Frontend in einem eigenen Dialogfenster angezeigt. Die Anzeige der Systemmeldung lässt sich zusätzlich um die Ausgabe von Laufzeitmeldungen des Programms (Stacktraces) erweitern. Klicken Sie hierfür auf den Button **Details** des Fehlerdialogs.

Wenn Sie einen Fehler an den Support des Herstellers melden, sollte ihre Anfrage folgende Informationen enthalten:

- Eine möglichst genaue Beschreibung, bei welchen Anwender- und Systemaktivitäten der Fehler auftritt.
- Wie ist der Fehler reproduzierbar? Tritt der Fehler nur sporadisch auf? (Diese Informationen sind sehr wichtig, um den Fehler nachstellen zu können.)
- Mit welchen projektspezifischen Laufzeitbibliotheken ist die PPM-Installation gegebenenfalls erweitert?
- Falls vorhanden, log-Ausgaben, die während des Auftretens des Fehlers ausgegeben werden.
- Welche Software-Versionen sind im Einsatz (PPM, Java, Datenbank)?

Die Versionen der installierten Software können Sie wie folgt ermitteln:

- Java

Alle Applikationen verwenden immer die gleiche JAVA-Version.

Zum Ermitteln der jeweiligen Version führen Sie in einer Eingabeaufforderung die Kommandozeile **<Installationsverzeichnis>\ppmmashzone\server\jre\bin\java -version** aus.

#### **Beispiel**

```
C:\> SoftwareAG\ppmmashzone\server\jre\bin\java -version
```

- PPM

Zum Ermitteln der installierten PPM-Version führen Sie das Kommandozeilenprogramm **runppmadmin** mit der Option **-version** aus.

**Beispiel**

```
C:\> cd
SoftwareAG\ppmmashzone\server\bin\agentLocalRepo\.unpacked\ppm-client-run-pr
od-95.1.0-RC8-trunk-20130404.122823-4-runnable.zip\ppm\bin & runppmadmin
-version
S: 21.04.11 08:59:16: [SRV] Applikation: 95.1.0 (10055)
S: 21.04.11 08:59:16: [SRV] Datenbankschema: 95
```

- Datenbank (und JDBC-Treiber)

Beim Starten gibt der PPM-Server Meldungen aus, welche Datenbank- und JDBC-Treiberversion verwendet werden. Diese Informationen können auch den log-Dateien des Servers entnommen werden.

**Beispiele**

- Oracle

```
I: ...: [SRV] Database version used: Oracle9i Enterprise Edition Release 9.2.0.7.0
- Production
I: ...: [SRV] JDBC driver used: Oracle JDBC driver (10.2.0.4.0).
```

- IBM DB2

```
I: ...: [SRV] Verwendete Datenbankversion: SQL09013.
I: ...: [SRV] Verwendeter JDBC-Treiber: IBM DB2 JDBC Universal Driver Architecture
(3.4.65).
```

- MS SQL Server

```
I: ...: [SRV] Verwendete Datenbankversion: 8.00.2039.
I: ...: [SRV] Verwendeter JDBC-Treiber: Microsoft SQL Server 2005 JDBC Driver
(1.1.1501.101).
```

## 4 Third-Party-Software

Zum Betrieb benötigt PPM bestimmte Software von Fremdherstellern. PPM wird in der Programmiersprache Java als Client-Server-Architektur entwickelt. Als Kommunikationsprotokolle werden RMI und Corba verwendet.

### JAVA LAUFZEITUMGEBUNG

Um PPM ausführen zu können, wird eine Java-Laufzeitumgebung benötigt, die bei der Installation des Produktes bereits mitgeliefert wird. Eine separate Installation ist nicht notwendig. Um zusätzliche, Performance steigernde Optionen des PPM-Servers in der Laufzeitumgebung aktivieren zu können, konsultieren Sie bitte das PPM-Installationshandbuch. Der PPM-Client wird zum einen im Java-Plugin des Web-Browsers als Applet ausgeführt und setzt eine entsprechende Installation des Java-Plugins für den verwendeten Browser voraus. Außerdem können Sie den PPM-Client auch als Java Applikation verwenden, indem Sie im Software AG-Installationsprogramm die Komponente PPM Analysis GUI wählen.

### RELATIONALES DATENBANKSYSTEM

PPM speichert eingelesene und berechnete Daten, sowie Konfigurationen eines Mandanten in dem Datenbankschema eines einzigen Datenbankbenutzers. Getestet und freigegeben ist PPM für die Verwendung der Datenbanksysteme **Oracle**, **IBM DB2** und **MS SQL-Server**.

Weiterhin verwendet PPM intern kommerzielle und Open Source-lizenzierte (z. B. Apache-Lizenz) Laufzeitbibliotheken.



## 5 Performance Aspekte

Das PPM-System ist für den Betrieb im LAN ausgelegt. Das zwischen PPM-Server und -Client zu übertragende Datenaufkommen hängt entscheidend von der am Client ausgeführten Aktion ab. Analyseanfragen an den PPM-Server können je nach Anzahl der angefragten Kennzahlen und Dimensionen umfangreiche Ergebnismengen zur Folge haben, die zum Client übertragen werden müssen. Aus diesem Grund überträgt das PPM-System die Daten standardmäßig komprimiert, da der Performance-Gewinn den zusätzlich zur Komprimierung erforderlichen Rechenaufwand bei weitem übertrifft.

Die Systemleistung lässt sich weiterhin steigern, wenn anstelle der direkten PPM-Client-Server-Kommunikation bestimmte vorberechnete Analyse-Ergebnisse über einen Web-Server publiziert werden.

Die Import Zeiten des Systems sind im Wesentlichen von den Hardware-Voraussetzungen abhängig: Hauptspeicher, Rechenleistung (ausgedrückt durch die Anzahl der CPUs) und frei verfügbarer Speicherplatz auf der Festplatte. Zur besseren Hardware-Abschätzung für ein individuelles System bietet das Setup drei vordefinierte Szenarien, die hier im Anschluss kurz beschrieben werden sollen. Sie dienen jedoch lediglich als grobe Richtwerte, da der tatsächliche Speicherbedarf noch von vielen weiteren Faktoren abhängig ist. Eine individuelle Hardware-Abschätzung ist deshalb nur nach Rücksprache mit dem Software AG-Support möglich.

Beim Einsatz von PPM auf Basis eines virtualisierten Betriebssystems (VMWare) ist beim Einrichten der VM darauf zu achten, dass der Hauptspeicher für diese VM exklusiv für diese VM bereitgestellt wird. Werden physikalische Ressourcen des VMWare Servers auf mehrere VMs verteilt, kann es zu Problemen in der VM kommen, auf der PPM installiert ist. Durch die Verwendung der In Memory-Technologie ist es zu empfehlen, den zur Verfügung stehenden Hauptspeicher exklusiv für die VM mit PPM zu reservieren.

Das Setup unterscheidet drei Szenarien, die in der folgenden Tabelle aufgelistet sind.

### Hardware-Szenarien für PPM und MashZone

PPM und MashZone Szenarien/Besonderheiten (4)		Demo scenario (1) Small setup scenario (64bit)	Medium scenario (2) (Proof of concept) Medium setup scenario (64bit)	Large scenario (3) (Production) Large setup scenario (64bit)
PPM 9.10	YPI	bis zu 200 000 PI/DR 1 PPM Client bis zu 2 Benutzer	bis zu 5 Millionen PI/DR 1 PPM Client bis zu 10 Benutzer	bis zu 25 Millionen PI/DR 1 PPM Client bis zu 100 Benutzer
MashZone 9.10	YCZ	2 CPU Kerne 4 GB RAM 10 GB freier HD-Platz	4 CPU Kerne 16 GB RAM 20 GB freier HD-Platz	8 CPU Kerne 64 GB RAM 50 GB freier HD-Platz

Legende	
PI	Prozessinstanzen
DR	Datenzeilen (verwendet in Data Analytics)
1	Unterstützt nur 64 Bit OS, Datenbank auf PPM-Servermaschine (Tabellenplatz: 1 GB)
2	Unterstützt nur 64 Bit OS, Datenbank auf PPM-Servermaschine oder auf einer eigenen Maschine (Tabellenplatz: 50 GB)
3	Unterstützt nur 64 Bit OS, Datenbank auf eigener Maschine (Tabellenplatz: 150 GB)
4	Die angegebenen Werte für jedes Szenario sind nur für die definierte Anzahl an PPM Clients gültig. Die Hardware-Anforderungen wachsen mit der Anzahl an PPM Clients und Benutzern.

## 6 Administration

Eine Produktinstallation umfasst die einzelnen PPM spezifischen Produktkomponenten und als deren gemeinsame Basis verschiedene Infrastrukturkomponenten zur Benutzer- und Lizenzverwaltung. Eine detaillierte Auflistung der Komponenten und deren Bedeutung finden Sie im PPM Installationshandbuch. In den nachfolgenden Kapiteln werden die wichtigsten administrativen Aufgaben und die verwendeten Werkzeuge beschrieben.

### 6.1 Cloud Agent

Das installierte PPM Systems wird durch den Cloud Agent mit Hilfe des Cloud Controller gesteuert. Der Cloud Controller ist bei der Installation standardmäßig so konfiguriert, dass er den auf dem lokalen Rechner (localhost) installierten Cloud Agent steuern kann und eine Anmeldung des Benutzers nicht erforderlich ist. Cloud Agent, Benutzername und Kennwort sind fest vorgegeben und werden als Parameter dem Cloud Controller beim Start mitgegeben.

Sie können den Cloud Controller in der Windows-Programmgruppe Start > Alle Programme > Software AG > Administration starten.

Es ist aber auch möglich den Cloud Controller direkt über die Kommandozeile zu starten. Wechseln Sie dazu in das Verzeichnis

**<PPM-Installationsverzeichnis>\ppmmashzone\server\acc**

und geben folgenden Befehl ein:

```
acc.bat -c "<PPM-Installationsverzeichnis>\ppmmashzone\server\generated.apptypes.cfg" -h localhost -p 17005 -u Clous -pwd g3h31m.
```

Starten Sie den Cloud Controller und geben Sie in der Kommandozeile das Kommando **help** ein, um die verfügbaren Kommandos des Cloud Controller und die entsprechenden Beschreibungen anzuzeigen.

### 6.2 Systemstatus

Der Status der einzelnen installierten Systemkomponenten kann im Cloud Agent mit dem Kommando **list** abgefragt werden. Es werden alle installierten Komponenten mit Name, Status und Version ausgegeben. Im Folgenden werden die wichtigsten Kommandos zur Steuerung der einzelnen Komponenten aufgelistet.

#### KOMMANDOS

Cloud Controller-Befehl	Beschreibung
list	Listet alle installierten Komponenten auf dem ausgewählten Cloud Agent Node auf. Standard ist localhost
start <instance id>	Startet die mit <instance id> angegebene Komponente
stop <instance id>	Stoppt die mit <instance id> angegebene Komponente

Cloud Controller-Befehl	Beschreibung
startall	Startet alle Komponenten in einer bestimmt Reihenfolge. Diese kann im Cloud Agent eingestellt werden.
stopall	Stoppt alle Komponenten in der umgekehrten Startreihenfolge
killall	Schließt alle Komponenten ungeachtet des aktuellen Zustandes oder der bestehenden Verbindungen. Mit kill <instance id> kann eine bestimmt Komponente geschlossen werden. Dieser Befehl sollte nur dann angewendet werden, wenn Komponenten nicht mehr auf andere Befehle reagieren.

## 6.3 Konfiguration

Mit Hilfe des Cloud Controller ist es auch möglich Konfigurationen einzelner Systemkomponenten zu ändern. So können zum Beispiel neue Datenbanktreiber für PPM hinzugefügt oder neue Benutzer in der Benutzerdatenbank angelegt werden. Im Folgenden werden die wichtigsten Kommandos zur Konfiguration der einzelnen Komponenten aufgelistet.

### KOMMANDOS

Cloud Controller-Befehl	Beschreibung
show config	Zeigt die aktuelle Konfiguration des Cloud Agent und Cloud Controller
show instance <instance id>	Zeigt die aktuelle Konfiguration einer Komponente.  <b>Beispiel</b> show instance ppm_core Gibt alle Konfigurationsparameter der Komponente <b>ppm_core</b> aus.

Cloud Controller-Befehl	Beschreibung
enhance [Treiber]	<p>Mit Hilfe des Befehls <b>enhance</b> können benötigte Treiber nachinstalliert werden.</p> <p><b>Installation von Datenbanktreibern</b>  enhance &lt;Mandant&gt; with dbDriver local file &lt;path to DB driver&gt;</p> <p><b>Installation von SAP JCO-Treibern</b>  enhance &lt;Mandant&gt; with dbDriver local file &lt;path to SAP JCO driver&gt;</p> <p>Bitte beachten Sie, dass die Pfadangaben im Java-Standard angegeben werden müssen. Also auch unter Windows Betriebssystemen geben Sie bitte „/" anstelle von „\" an.</p> <p><b>Beispiel</b></p> <p>Unter C:\Temp liegt der Oracle Datenbank Treiber ojdbc6.jar. geben Sie folgenden Befehl ein, um diesen Treiber für den PPM-Mandanten <b>umg_en</b> zu installieren.</p> enhance umg_en with dbDriver local file C:/Temp/ojdbc6.jar
enhance [PPM Komponenten]	<p>Um nicht installierte PPM-Komponenten wie Content Packages oder auch Extraktoren nach zu installieren kann folgender Befehl verwendet werden:</p> <p><b>CSV2PPM Extractor</b>  enhance &lt;Mandant&gt; with extractor com.aris.ppm.enhancements.csv2ppm &lt;version&gt; type zip</p> <p><b>ppm4mm Content Package</b>  enhance &lt;Mandant&gt; with contentPackage com.aris.ppm.enhancements.ppm4mm &lt;version&gt; type zip</p> <p>Bitte beachten Sie, dass die gewählte Komponente im Repository Ihrer Installation bereits vorhanden sein muss. Das lokale Repository Ihrer Installation für die PPM-Erweiterungen finden sie hier:  &lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\agentLocalRepo\com\aris\ppm\enhancements. Die Version entnehmen Sie bitte dem Namen des Unterverzeichnisses der jeweiligen Komponente. Ist z. B. unter  &lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\agentLocalRepo\com\aris\ppm\enhancements\jdbs2ppm das Verzeichnis <b>95.1.0</b> angegeben, lautet die Version <b>95.1.0</b>.</p>

Cloud Controller-Befehl	Beschreibung
enhance [Benutzer]	<p>Der Befehl <b>enhance</b> kann auch eingesetzt werden, um in der zentralen Benutzerverwaltung einen neuen Benutzer anzulegen.</p> <pre>enhance &lt;instance id&gt; with createUser trigger only options tenant.name="default" tenant.user.name=system tenant.user.pwd=manager affected.user=user1 affected.pwd=user1 affected.first.name="John" affected.last.name="Doe" affected.email="test@test.de" affected.description="description"</pre> <p>Legt einen neuen Benutzer <b>user1</b> mit dem Passwort <b>user1</b> an. Für die Instanz-ID nutzen Sie bitte den Namen der Komponente <b>Benutzerverwaltung</b> (default ist umcadmin).</p>
reconfigure <instance id> <Parameter>	<p>Der Befehl wird verwendet, um Komponenten neu zu konfigurieren. Dabei können alle Parameter geändert werden, die in der Konfigurationsdatei unter</p> <pre>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\work_&lt;instance id&gt;\runtimeinfo.properties</pre> <p>gespeichert sind. Die Änderung der Werte in dieser Datei hat keinen Einfluss auf die laufende Instanz. Parameter können nur mittels Cloud Controller geändert werden.</p> <p>Bitte beachten Sie, dass die Pfadangaben im Java-Standard angegeben werden müssen. Also auch unter Windows Betriebssystemen geben Sie bitte „/" anstelle von „\" an. Ersetzen Sie im Kennwort einen verwendeten Rückwärtsschrägstrich \ durch einen doppelten Rückwärtsschrägstrich \\. </p> <p><b>Beispiel</b></p> <p>Die RMI-Kommunikation zwischen PPM-Mandant und -Server soll auf SSL-Verschlüsselung umgestellt werden. Dazu sind folgende Befehle im Cloud Controller auszuführen.</p> <ul style="list-style-type: none"> <li>▪ stop ppm_core</li> <li>▪ reconfigure ppm_core ssl.enabled="true" ssl.keystorefile="&lt;path to keystore file&gt;" ssl.keystorepassword="&lt;keystorepassword&gt;"</li> <li>▪ start ppm_core</li> </ul>

## 6.4 Systemmeldungen

Im Falle eines Fehlers in der Infrastruktur können die Systemmeldungen (log-Ausgaben) der einzelnen Komponenten zur Analyse des Problems herangezogen werden. In der folgenden Tabelle sind die einzelnen log-Dateien der Infrastrukturkomponenten aufgelistet. Die PPM Systemmeldungen (Seite 20) werden im nachfolgenden Kapitel detailliert behandelt.

Infrastrukturkomponente	Pfad zur log-Datei
Cloud Agent	< Installationsverzeichnis>\ppmmashzone\server\logs\CloudAgent.log < Installationsverzeichnis>\ppmmashzone\server\bin\log\agent.log
Cloud Controller	< Installationsverzeichnis>\ppmmashzone\server\acc\log\acc.log
Cloudsearch	< Installationsverzeichnis>\ppmmashzone\server\bin\work\work_cloudsearch_<memory model>\defaultLog\log.txt
Elastic Search	< Installationsverzeichnis>\server\bin\work\work_elastic_<memory model>\elasticsearch\logs\elasticsearch.log
Load Balancer	< Installationsverzeichnis>\ppmmashzone\server\bin\work\work_loadbalancer_<memory model>\httpd\logs\*.log
Zentrale Benutzerverwaltung	< Installationsverzeichnis>\ppmmashzone\server\bin\work\work_umcadmin_<memory model>\base\logs\*.log
PostgreSQL DB	< Installationsverzeichnis>\ppmmashzone\server\bin\work\work_postgres_<memory model>\pgworkdata\pg_log\*.log

Um alle relevanten log-Dateien in einem Schritt zusammen zu tragen und zu verpacken, wird ein eigenes Werkzeug zur Verfügung gestellt. Dies kann z. B. bei einer Support Anfrage notwendig sein, wenn zur Analyse eines Problems alle log-Dateien angefordert werden. Zum Sammeln der log-Dateien wechseln Sie in den folgenden Order Ihrer PPM-Installation.

```
< Installationsverzeichnis>\ppmmashzone\server\support
```

Führen dort den Befehl **collectlogfiles.bat** (unter Windows) und **collectlogfiles.sh** (unter Linux) aus.

Sie können alle log-Dateien mit Hilfe der Batchdatei **deletelogfiles.bat** unter Windows und dem Shell-Skript **deletelogfiles.sh** unter Linux löschen.

## 7 PPM Systemmeldungen

### 7.1 Überblick

Der PPM- Server verwendet zur Ausgabe von Systemmeldungen (log-Ausgaben) die Java-Standardschnittstelle **log4J**. log4J ist ein flexibles Framework zur Steuerung der Ausgabe von Java-Anwendungsmeldungen über verschiedene Medien (Ausgabekanäle). Die Konfiguration der log4J-Schnittstelle erfolgt entweder in einer XML-Datei oder einer properties-Datei. Die Schnittstelle log4J setzt sich aus drei Komponenten zusammen: Logger (Sammler), Layout (Formatierung) und Appender (Ausgabe).

PPM liest die log4J-Konfiguration aus einer Property-Datei.

#### LOGGER

Die Java-Anwendung übergibt ihre Meldungen an einen bestimmten Logger (Sammler), der diese je nach Art der Meldung weiterverarbeitet. Meldungen können vom Typ (Loglevel) **ALL**, **TRACE**, **INFO**, **WARN**, **ERROR**, **FATAL**, **OFF** sein.

Loglevel	Beschreibung
ALL	Alle Meldungen werden ungefiltert ausgegeben.
TRACE	Sehr ausführliche Laufzeitmeldungen, häufig mit Ausgaben kompletter Fehlerzustände der Applikation
DEBUG	Ausführliche Laufzeitmeldungen, die oft den internen Zustand der Applikation beschreiben
INFO	Allgemeine Informationen oder Warnungen
WARN	Hinweise auf nicht laufzeitkritische Applikationszustände, z. B. fehlende oder falsche Konfigurationen
ERROR	Fehler, die oft zum Abbruch der aktiven Komponente führen
FATAL	Fehler, die zum Abbruch der Anwendung führen, z. B. Ressourcenmangel
OFF	Keine log-Ausgaben

In der log4J-Konfigurationsdatei wird ein Logger durch die folgende Zeile definiert:

```
<Logger-Name> = <Loglevel>, <Name Appender 1>, ... , <Name Appender n>
```

Die Schreibweise der Logger-Namen beachtet die Groß- und Kleinschreibung. Die Namen bilden eine hierarchische Struktur. Das Wurzelement ist der root-Logger. Die Hierarchieebenen werden durch das Zeichen **Punkt** (.) getrennt. Die höhere Hierarchieebene vererbt ihre Konfiguration an die niedrigere, die dann von spezifischen Einstellungen überschrieben wird.

Die folgende Tabelle veranschaulicht die Vererbungshierarchie:



Logger Name	zugewiesener Loglevel	Ererbter Loglevel
log4j.rootLogger	INFO	keiner
log4j.logger.LOG	DEBUG	INFO
log4j.logger.LOG.SRV	ERROR	INFO

## APPENDER

Der Logger gibt die Meldungen an die Appender (Empfänger) weiter, die die Meldungen dann in einer bestimmten Form (Layout) ausgeben. Die Appender unterscheiden sich in der Art der Ausgabe, beispielsweise:

PPM Beispiel Appender	Beschreibung
console	Meldungen werden an der Konsole ausgegeben
logFile	Meldungen werden in eine Datei geschrieben
errorFile	Fehlermeldungen werden in eine Datei geschrieben

In der log4J-Konfigurationsdatei wird ein Logger durch die folgende Zeile definiert:

```
log4j.appender.<Name>=<Klassenname>
```

Für einen Logger können mehrere Appender angegeben werden, alle Appender eines Loggers geben die Meldungen entsprechend ihrer Konfiguration aus.

Es gibt auch Appender, die die Meldungen in das log-System des Betriebssystems schreiben, als Mail versenden oder die Meldungen über das Netzwerk an einen bestimmten log-Server schicken.

## LAYOUT

Das Format der Ausgabe lässt sich durch bestimmte Formatvorlagen festlegen. Diese Formatvorlagen werden für den gewünschten Appender angegeben. Die wichtigsten Platzhalter für eine Formatvorlage sind:

Platzhalter	Beschreibung
%c	Kategorie / Logger (entspricht PPM-Modul)
%C	Klassenname, vollqualifiziert
%d	Datum, z. B. %d{HH:mm:ss,SSS}
%F	Dateiname
%L	Zeilennummer
%m	Die Meldung selbst

Platzhalter	Beschreibung
%M	Name der Methode
%n	Zeilenumbruch
%p	Priorität, Level (INFO, WARN, ERROR, etc.)
%t	Name des Thread
%throwable{0}	Ausgabe von Stacktraces wird unterdrückt.

### Beispiel

Die Formatvorlage **[%t] %-5p %C - %m%n** ergibt folgende Ausgabe:

```
[main] INFO org.apache.log4j.Action - Der Prozess wurde gestartet
```

## 7.2 Verzeichnisse der log-Dateien

Die log-Dateien der verschiedenen PPM-Komponenten sind in folgenden Verzeichnissen abgelegt.

- Cloud Controller:
  - < Installationsverzeichnis > \ppmmashzone\server\acc\log
- Runnable-Instanzen:
  - < Installationsverzeichnis > \ppmmashzone\server\bin\work\work\_<instance>\ und deren Unterverzeichnisse
- log-Ausgaben des Cloud Agent:
  - < Installationsverzeichnis > \ppmmashzone\server\bin\log
- Mandantenspezifischen Kommandozeilenprogramme:
  - < Installationsverzeichnis > \ppmmashzone\server\bin\work\data\_ppm\custom\  
<mandant>\log
  - < Installationsverzeichnis > \ppmmashzone\server\bin\work\data\_ppm\config\  
<mandant>\log
- log-Dateien des PPM- und des Analyseservers:
  - < Installationsverzeichnis > \ppmmashzone\server\bin\work\data\_ppm\log\  
<mandant> \*)
- log-Ausgaben des Client-Setup:
  - < Installationsverzeichnis > \ppmmashzone\server\bin\work\data\_ppm\system\log
- Konsole-Ausgaben der CORBA- und der RMI-Registry:
  - < Installationsverzeichnis > \ppmmashzone\server\bin\work\work\_ppm\_core\defaultLog
- Konsole-Ausgaben des PPM-Servers:
  - < Installationsverzeichnis > \ppmmashzone\server\bin\work\work\_<mandant>\_cs\  
defaultLog
- Konsole-Ausgaben des Analyse-Servers:

<Installationsverzeichnis>\ppmmashzone\server\bin\work\work\_<mandant>\_as\  
defaultLog

\*) Standard-Einstellung: Die Verzeichnisse und die Namen der log-Dateien werden in den mandantenspezifischen Konfigurationsdateien **AnalysisServer\_Log\_settings.properties** und **Server\_Log\_settings.properties** eingestellt.

### 7.3 Systemmeldungen (log-Ausgaben)

Der PPM-Server verwendet die standardisierte Logging-Schnittstelle **log4J** zum Ausgeben von Systemmeldungen. Hierfür verwendet jede PPM-Komponente (z. B. Datenimport, Merger, Kennzahlenberechner) eigene Logger. Die Module (Komponenten) des PPM-Systems verfügen über folgende übergeordnete Logger: **log**, **trace** und **audit**.

Standardmäßig verfügt das PPM-System über folgende Appender zur formatierten Ausgabe der Meldungen.

Appender	Beschreibung
console	Meldungen werden an der Konsole ausgegeben. Nur noch für Kommandozeilenprogramme relevant.
logFile	Meldungen werden in eine Datei geschrieben. Der Name der Datei wird durch den Konfigurationsschlüssel <b>log4j.appender.logFile.File</b> bestimmt.
errorFile	Fehlermeldungen und Exceptions (log-Ausgaben, die mit E: oder F: beginnen) werden in eine Datei geschrieben. Der Name der Datei wird durch den Konfigurationsschlüssel <b>log4j.appender.errorFile.File</b> bestimmt. Die Ausgabe von Exceptions und Stack-Traces erfolgt ausschließlich in diese Datei.
traceFile	Meldungen werden ausschließlich in eine Datei geschrieben. Der Name der Datei wird durch den Konfigurationsschlüssel <b>log4j.appender.traceFile.File</b> bestimmt. Dieser Appender wird verwendet, um bestimmte Meldungen zu sammeln, die die Fehlersuche erleichtern sollen.
auditFile	Schreibt Audit-log-Meldungen in eine Datei. Der Name der Datei wird durch den Konfigurationsschlüssel <b>log4j.appender.auditFile.File</b> bestimmt.

Für den Mandantenserver wird die Ausgabe von Meldungen in der Datei **Server\_Log\_settings.properties** konfiguriert, für den Analyseserver in der Datei

**AnalysisServer\_Log\_settings.properties.** Diese beiden Konfigurationsdateien werden während der Installation automatisch mit Standardeinträgen erzeugt.

Die Konfigurationsdateien **Server\_Log\_settings.properties** und **AnalysisServer\_Log\_settings.properties** werden alle 60 Sekunden (fest vorgegeben) auf Veränderungen geprüft. Hierdurch kann die Konfiguration der Log-Ausgaben ohne Neustart der PPM-Server geändert werden.

## MODULBASIERTES LOGGING

Zur Ausgabe von Systemmeldungen des PPM-Mandanten- und Analyseservers werden die bekannten PPM-Modulnamen verwendet. Die entsprechende Konfiguration erfolgt in den jeweiligen Dateien **Server\_Log\_settings.properties** (Mandantenserver) bzw.

**AnalysisServer\_Log\_settings.properties** (Analyseserver).

Folgende Logger-Module werden von PPM unterstützt:

Modulname	Beschreibung
LOG.XML	XML-Quellsystemdatenimport
LOG.EIM	Internes Einlesen der importierten Fragmentinstanzen
LOG.MGR	Zusammenführung von Prozessfragmenten (Merge)
LOG.TYP	Typisieren von Prozessinstanzen
LOG.KIC	Berechnung und Ermittlung von Kennzahlen und Dimensionen
LOG.PSC	Anfragen des Dashboard
LOG.PLV	Planwertberechnung
LOG.CFG	Internes Konfigurationsmanagement
LOG.STD	PPM-Server im Modus <b>STANDARD</b>
LOG.SRV	Standardmodul, das für Protokollausgaben des PPM-Servers verwendet wird
LOG.CNV	Datenbankkonverter
LOG.IMP	Im- und Export von Konfigurationen (z. B. mittels <b>runppmconfig</b> )
LOG.ADM	Administration des Mandanten, (z. B. mittels <b>runppmadmin</b> )
LOG.KG	Schlüsselerzeugung für importierte Prozessfragmente (Keygenerator)
LOG.PRF	Profiler (SA)

Modulname	Beschreibung
LOG.MV	Management Views (Applet basiert)
LOG.OLA	Abfrageschnittstelle
LOG.REP	Erstellung von Reporten
LOG.RAU	Reportautomatisierung
LOG.MST	PPM-Server im Modus <b>MASTER</b>
LOG.SUB	PPM-Server im Modus <b>SUBSERVER</b>
LOG.LOG	Allgemeine log-Ausgaben
LOG.EAL	Frühwarnsystem
LOG.ALY	Unregelmäßigkeitsanalyse ( <b>runppmanalytics</b> )
LOG.MTAB	Minitab
LOG.CCS	Attributberechner im Modus <b>SILENT</b>
LOG.CCD	Attributberechner im Modus <b>DEFAULT</b>
LOG.CCV	Attributberechner im Modus <b>VERBOSE</b>
LOG.PFM	sysmon-Mandant
LOG.CRC	Corba-Verbindungen
LOG.MON	Online-Monitoring
LOG.PFS	Statistiken benutzerspezifischer Favoriten
LOG.SFS	Statistiken gemeinsamer Favoriten
LOG.CPI	Verbesserungen, Nachrichtenverwaltung
LOG.ASRV	Standardmodul, das für Protokollausgaben des Analyseservers verwendet wird
LOG.MEM	Frühwarnsystem bei Speicherknappheit
LOG.PPI	Prozessimport
LOG.EIP	Prozessimport - Eventimport
LOG.CLC	Prozessimport - Kennzahlberechnung
LOG.UPD	Prozessimport - Aktualisierung
LOG.DAI	Data-Analytics-Import

Bei der Ausgabe von Meldungen werden die Modulkürzel (entspricht dem Modulnamen ohne Präfix **LOG.**) nicht lokalisiert ausgegeben, d. h. für Ausgaben von Meldungen für den Kennzahlenberechner wird das Modulkürzel **KIC** verwendet.

Für jedes Logger-Modul muss ein Loglevel angegeben werden. Folgende Loglevel werden unterstützt: **INFO**, **WARN**, **ERROR**, **FATAL**, **OFF**. Die Affinität des gewählten Loglevel nimmt von rechts nach links zu, d. h. wenn Loglevel **WARN** eingestellt ist, werden auch Meldungen der Level **ERROR**, **FATAL** ausgegeben.

### Beispiel

Für den XML-Datenimport sollen Informationen, Warnungen, Fehler und kritische Fehler an der Konsole ausgegeben werden.

```
log4j.logger.LOG.XML=INFO,console
```

- Die Loglevel werden ohne Berücksichtigung der Schreibweise angegeben, d. h. **log4j.logger.LOG.XML=error** entspricht **log4j.logger.LOG.XML=ERROR**.
- Falsche Angaben des Loglevel, z. B. **log4j.logger.LOG.XML=eror** werden ignoriert. In diesem Fall wird der Loglevel des übergeordneten Logger verwendet, hier der Loglevel **log4j.logger.LOG=INFO**.
- Wenn für ein Logger-Modul kein Loglevel angegeben ist, wird der Loglevel des übergeordneten Logger verwendet, hier der Loglevel **log4j.logger.LOG=INFO**.

### AUSGABE VON TRACE-MELDUNGEN

Im Gegensatz zu den Logger-Modulen gibt es für die Trace-Module keine Loglevel, die Ausgaben von Meldungen können lediglich ein- bzw. abgeschaltet werden. Diese Ausgaben sind gedacht, um im konkreten Fehlerfall weiterführende Informationen zu erhalten. Es gibt folgende Trace-Module: **ASRV**, **SRV**, **CONFIG**, **ABT**, **CONPOOL**, **FRQ**, **RET**, **RETDS**, **QPL**, **SELITY**, **DIM**, **IMP**, **RMI**, **MEM**, **PPI**, **EIP**, **CLC**, **UPD**, **DST**, **QUE**

Anstelle des Loglevel können für die Trace-Module folgende Einstellungen vorgenommen werden: **OFF**, **TRACE**.

### Beispiel

Für den Mandantenserver sollen Trace-Meldungen, für Analyseserver sollen keine Trace-Meldungen ausgegeben werden.

```
log4j.logger.TRACE.SRV=TRACE
```

```
log4j.logger.TRACE.ASRV=OFF
```

### PRÄFIXE VON LOG-AUSGABEN

Jede Log-Ausgabe beginnt mit einem Präfix, der den entsprechenden Log-Level angibt. Dieser Präfix ist für alle Sprachen gleich.

Log-Level	Präfix alle Sprachen
Statistik	S:
FATAL	F:
ERROR	E:
WARN	W:
INFO	I:

Log-Level	Präfix alle Sprachen
TRACE	T:

Statistikmeldungen werden immer ausgegeben und können nur durch den Log-Level **OFF** unterdrückt werden.

## LOG-AUSGABEN FORMATIEREN

Das Format der Log-Ausgaben wird in den beiden Konfigurationsdateien **Server\_Log\_settings.properties** und **AnalysisServer\_Log\_settings.properties** angegeben. Das PPM-Standardformat des Datums wird in diesen Dateien durch **%X{ppmdate}** angegeben.

Das verwendete Datumsformat ist abhängig von der Sprache, mit der Mandanten- und Analyseserver gestartet wurden. Für die von PPM unterstützten Sprachen werden folgende Datumsformate verwendet:

Sprache	Locale	Datumsformat
englisch	EN	dd/MM/yy HH:mm:ss
englisch USA	US	MM/dd/yy HH:mm:ss
deutsch	DE	dd.MM.yy HH:mm:ss
französisch	FR	dd.MM.yy HH:mm:ss
ungarisch	HU	dd.MM.yy HH:mm:ss
japanisch	JA	dd/MM/yy HH:mm:ss
niederländisch	NL	dd.MM.yy HH:mm:ss
russisch	RU	dd.MM.yy HH:mm:ss
chinesisch	ZH	dd/MM/yy HH:mm:ss

### Beispiel

```
log4j.appender.error.File.layout.ConversionPattern=%p %X{ppmdate} [%c{1}] %m%n
```

## SKALIERTE SYSTEME

Die Konfiguration der Ausgabe von Log-Meldungen wird für ein skaliertes PPM-System grundsätzlich gehandhabt wie für ein Standardsystem mit nur einem PPM-Server. Folgende Besonderheiten sollten beachtet werden:

- Alle auftretenden Exception- und Trace-Meldungen werden auf den entsprechenden Sub-Server ausgegeben. Exceptions werden an den Master-Server weitergeleitet und in das Errorlog des Master-Servers geschrieben.
- Da der Master-Server nicht mit einem eigenen Analyseserver verbunden ist, können auf dem Master-Server nur Trace-Meldungen des Mandantenservers ausgegeben werden.

Trace-Meldungen der Analyseserver werden in die Log-Ausgaben der jeweiligen Sub-Server geschrieben.

## LOG-AUSGABEN VERTEILEN

Die Systemmeldungen der PPM-Server können Sie durch geeignete Appender zu beliebigen log-Server senden, die diese Meldungen dann weiter verarbeiten.

## WINDOWS EREIGNISANZEIGE

Der folgende Dateiauszug zeigt die erweiterte Konfiguration eines Appender, um die Systemmeldungen der PPM-Server an den lokalen log-Server von Windows zu schicken.

```
...
log4j.appender.Logserver=org.apache.log4j.nt.NTEventLogAppender
log4j.appender.Logserver.layout=org.apache.log4j.EnhancedPatternLayout
log4j.appender.Logserver.layout.ConversionPattern=%d | %-5p | %-20c{1} | %m%n
...
```

Die Systemmeldungen werden im Windows Systemprogramm Ereignisanzeige (Computerverwaltung -> System -> Ereignisanzeige) angezeigt.

## ENTFERNTE LOG-SERVER

Um die Systemmeldungen der PPM-Server an einen entfernten Logging-Server (z. B. Lumbermill oder Apache Chainsaw) in Ihrem Netzwerk zu schicken, können Sie im folgenden Dateiauszug gezeigte erweiterte Konfiguration eines Appender verwenden.

```
...
log4j.appender.AuditLogserver=org.apache.log4j.net.SocketAppender
log4j.appender.AuditLogserver.Port=4445
log4j.appender.AuditLogserver.RemoteHost=pcirgendwo
log4j.appender.AuditLogserver.layout=org.apache.log4j.EnhancedPatternLayout
log4j.appender.AuditLogserver.layout.ConversionPattern=%X{ppmdate} [%c{1}] %m%n
...
```

## 7.4 Audit-Meldungen (Audit-logging)

Audit-Meldungen (Audit-Logs) dokumentieren wichtige Vorgänge in einem laufenden System und werden von bestimmten Personen (Auditoren) gelesen. Eine Audit-Meldung enthält verschiedene Informationen, so dass der Auditor den Vorgang detailliert nachvollziehen kann.

Information	Beschreibung
Zeitstempel	Zeitpunkt, zu dem die vom Vorgang ausgelösten Änderungen gespeichert wurden
Vorgang	Identifizierer und Beschreibung des Vorgangs
Komponente	Komponentenkürzel
Benutzer	Login-ID des Anwenders, der den Vorgang ausgeführt hat
IP-Adresse	Rechner, von dem aus der Vorgang ausgelöst wurde



Information	Beschreibung
Anwendung	ID der Anwendung, die zum Auslösen des Vorgangs geführt hat

Es werden nur Vorgänge protokolliert, bei deren Ausführung tatsächlich Daten gespeichert (persistiert) wurden. Vom Anwender zurück genommene Änderungen werden nicht protokolliert.

### Beispiel

Die folgende Meldung zeigt an, dass Benutzer **SYSTEM** zur angegebenen Zeit über die Benutzeroberfläche (GUI) am lokalen Rechner des Serversystems (127.0.0.1) durch Ausführen des Vorgangs mit der ID 800 die Benutzerverwaltung (Komponente USR) verändert hat. Der folgende Text gibt die genaue Aktion des Vorgangs an.

27.09.2010 16:36:25.527, USR, [800] Die Benutzerkonfiguration wurde geändert, SYSTEM, 127.0.0.1, GUI

### KOMPONENTEN-ID

Jede Komponente ist durch einen eindeutigen, sprachunabhängigen Identifizierer gekennzeichnet. Die folgende Tabelle zeigt die möglichen Werte und Beschreibungen.

Komponenten-ID	Beschreibung
KIC	Kennzahlenkonfiguration
OKI	Benutzerdefinierte Kennzahlen
RKI	Ranking Kennzahlen
ABC	Benutzerdefinierte Dimensionen
TFD	Top-Flop-Dimensionen
KID	Kennzahlabhängige Dimensionen
PRT	Prozessbaumkonfiguration
USR	Benutzerverwaltung
REP	Reportdefinition
RAU	Reportautomatisierung
FAV	Gemeinsame Favoriten
MGR	Merger
EPK	Prozessinstanzen
PIK	Prozessinstanzunabhängige Kennzahlen
PLV	Planwerte
ORG	Organisationseinheiten
FTC	Werkskalender

Komponenten-ID	Beschreibung
SRV	Server

## ANWENDUNGS-ID

Mit Hilfe der Anwendungs-ID kann der Auditor erkennen, welche Anwendung die Änderung bewirkt hat. Die folgende Tabelle zeigt die möglichen Werte und Beschreibungen.

Anwendungs-ID	Beschreibung
GUI	PPM-Benutzeroberfläche
EXP	Export
CIM	Konfigurationsimport
IMP	Datenimport (Prozessinstanzunabhängigen Kennzahlen, Dimensionsdaten, PPM)
MMV	Management View
CHK	Überprüfungen, z. B. Planwertabweichungen
CMP	Verdichten und Löschen von Instanzen
CVT	Datenbankkonverter
INT	interne Anwendung
SHR	Session-Verwaltung
ADT	Adapter
ADM	Admin-Tool
PFC	Performance Dashboard
MST	Verbindung Master-Sub-Server
OTH	andere Anwendung

## KONFIGURATION

Zur Ausgabe von Audit-Meldungen verwendet PPM den log4J-Logger **AUDIT** der im vorangegangenen Kapitel beschriebenen logging-Schnittstelle. Das Audit-logging wird in der mandantenspezifischen Konfigurationsdatei **Server\_Log\_settings.properties** auf Basis der log4J-Schnittstelle konfiguriert (siehe auch Kapitel Überblick (Seite 20)).

Nur der PPM-Mandantenserver protokolliert Audit-Meldungen, nicht jedoch der Analyseserver.

## PROTOKOLLMELDUNGEN

Zusätzlich zu den Basisinformationen (Zeitpunkt, Komponenten- und Vorgang, usw.) geben die verschiedenen PPM-Komponenten jeweils bestimmte Arten von Meldungen aus, die im Folgenden aufgeführt sind.

Es werden ausschließlich Vorgänge protokolliert, die Daten im PPM-System dauerhaft verändern. Exportieren von Daten wird nicht protokolliert.

### KENNZAHLENKONFIGURATION

Mittels Kommandozeilenprogramm **runppmconfig ... -keyindicator** durchgeführte Änderungen der Kennzahlenkonfiguration werden zusätzlich zu den Basisinformationen folgende statistische Angaben ausgegeben: Anzahl der bearbeiteten Attributberechnerfunktionen, Anzahl der bearbeiteten Attribute, Anzahl der bearbeiteten Kennzahlen, Dimensionen und Relationen, Anzahl der bearbeiteten prozessunabhängigen Kennzahlreihen.

### PROZESSBAUMKONFIGURATION

Es gibt unterschiedliche Meldungen, je nachdem, ob die Prozessbaumkonfiguration mittels Kommandozeilenprogramm **runppmconfig ... -processtree** verändert wurde oder automatisch durch Import neuer Prozessinstanzen mit neuen Prozesstypen erweitert wurde. Im letzteren Fall werden die Namen der neuen Prozesstypgruppen und Prozesstypen als Zusatzinformation ausgegeben.

### ERWEITERTE KENNZAHLENKONFIGURATION

Es werden bestimmte Meldungen ausgegeben, unabhängig davon, ob die erweiterten Kennzahlenkonfigurationen (Benutzerdefinierte Kennzahlen, Ranking-Kennzahlen, ABC-Dimensionen, Benutzerdefinierte Dimensionen, Top-Flop-Dimensionen) mittels Kommandozeilenprogramm **runppmconfig** oder über die Benutzeroberfläche verändert wurden.

### BENUTZERVERWALTUNG

Diese Komponente meldet Änderungen an Benutzern und Benutzergruppen, Änderung der Gruppenzugehörigkeit und Datenzugriffsrechten, Ändern von Kennwörtern, Funktions- und Zugriffsrechten sowie An- und Abmelden am System. An- und Abmeldevorgänge von Benutzern, die sich über Performance Dashboard, Abfrageschnittstelle oder Management Views anmelden, werden nicht ausgegeben.

Wird die Benutzerkonfiguration mittels Kommandozeilenprogramm **runppmconfig ... -users** verändert, werden zusätzlich statistische Ausgaben des Importvorgangs ausgegeben.

Auf Grund der Art, wie das Performance Dashboard mit PPM verbunden ist, können beim Ändern des Kennworts über das Performance Dashboard die Adresse des Rechners und die Login-ID des Benutzers nicht eindeutig ermittelt werden.

#### **Anbindung an die zentrale Benutzerverwaltung**

Die von PPM angestoßenen Änderungen, die sich auf die zentrale Benutzerverwaltung auswirken, werden nicht protokolliert. Es werden ausschließlich Änderungen, die in PPM stattfinden, ausgegeben. Beim Aktivieren oder Deaktivieren von Benutzern oder Benutzergruppen in PPM wird ein entsprechender Text samt Zusatzinformationen ausgegeben.

Beim Import mittels Kommandozeilenprogramm **runppmconfig** können neue Benutzer in der zentralen Benutzerverwaltung und in PPM oder nur in einem von beiden Systemen angelegt werden. Wird mindestens ein neuer Benutzer in PPM angelegt, wird ein entsprechender Text ausgegeben. Änderungen in der zentralen Benutzerverwaltung werden nicht protokolliert.

Die Benutzeranzahl wird entsprechend Ihrer Lizenz durch die zentrale Benutzerverwaltung geprüft. Sollte es zu einer Überschreitung der im Lizenzschlüssel angegebenen Benutzeranzahl kommen, wird ein entsprechender Text ausgegeben.

### ORGANISATIONSEINHEITEN

Anlegen, Bearbeiten und Löschen von Organisationseinheiten wird entsprechend protokolliert, unabhängig davon, ob der Vorgang an der Oberfläche oder mittels Kommandozeilenprogramm ausgelöst wurde.

### WERKSKALENDER

Änderungen der Bezugszeiträume, Arbeitstage und Umrechnungsfaktoren eines Werkskalenders werden protokolliert.

### REPORTE

Das Speichern von Reportdefinitionen und das Ändern von Zugriffsrechten einer Reportdefinition werden protokolliert. Das Ausführen einer Reportdefinition wird nicht protokolliert.

### REPORTAUTOMATISIERUNG

Anlegen, Bearbeiten und Löschen von Reportautomatisierungen werden entsprechend protokolliert, unabhängig davon, ob der Vorgang an der Oberfläche oder mittels Kommandozeilenprogramm ausgelöst wurde. Der Benutzer **SYSTEM** kann durch Importieren im Modus **replace** einer für alle Benutzer gültigen Gesamtkonfiguration die Reportautomatisierungen aller Benutzer löschen. Dieser Vorgang wird durch den Text **Reportautomatisierung aller Benutzer gelöscht** protokolliert.

### FAVORITEN

Da Gemeinsame Favoriten in der Kennzahlkonfiguration verwendet werden können, werden Änderungen der Gemeinsamen Favoriten (Anlegen, Bearbeiten, Änderung der Zugriffsrechte, Umbenennen, Verschieben und Löschen) protokolliert, unabhängig davon, ob der Vorgang an der Oberfläche oder mittels Kommandozeilenprogramm ausgelöst wurde.

Änderungen der privaten Favoriten und Änderungen am Favoriten-Cache werden nicht protokolliert.

### PLANWERTE

Anlegen, Bearbeiten und Löschen von Planwerten wird entsprechend protokolliert, unabhängig davon, ob der Vorgang an der Oberfläche oder mittels Kommandozeilenprogramm ausgelöst wurde.

### MANAGEMENT VIEWS

Das Erstellen, Bearbeiten und Ausführen von Management Views wird nicht protokolliert.

### KOMMANDOZEILENPROGRAMME

Bestimmte Kommandozeilenprogramme melden Systemveränderungen.

Das Verdichten und Löschen von Instanzen mittels **runppmcompress** und **runppmdelete** wird nur protokolliert, wenn das Programm im Modus **execute** ausgeführt wird.

Es gibt unterschiedliche Meldungen, je nach dem, ob die Datenreihen prozessinstanzunabhängiger Kennzahlen mittels Kommandozeilenprogramm **runpikidata** oder über die Benutzeroberfläche verändert wurden. Auch der Import von Dimensionsdaten mittels Kommandozeilenprogramm **rundimdata** wird entsprechend protokolliert.

### PPM-DATENIMPORT

Mittels Kommandozeilenprogramm **runppmimport** werden die im PPM-System vorhandenen Daten verwaltet (Prozessfragmente zu Prozessinstanzen zusammenführen, Bearbeiter zu Organisationseinheiten anonymisieren, Prozessinstanzen typisieren, Kennzahlen neu berechnen, Shared Fragments und Prozessschlüssel löschen, Datenbankindizes erzeugen, Analyseserver reinitialisieren). Das Ausführen dieser Vorgänge wird mit einer entsprechenden Meldung ausgegeben.

Das Neuberechnen von Datenbankstatistiken wird nicht protokolliert.

### STARTEN UND STOPPEN DES SERVERS

Starten und Stoppen des PPM-Mandantenservers wird entsprechend protokolliert.

Wird der Server unkontrolliert mittels Taskmanager oder Schließen der Eingabeaufforderung, in der der Server ausgeführt wird, beendet, wird dieser Vorgang nicht protokolliert.

### SKALIERTE SYSTEME

Bei Einsatz eines skalierten Systems werden vom Master-Server und den Sub-Servern dieselben Audit-Informationen protokolliert wie bei einem Server im Standalone-Betrieb. Zu beachten ist, dass bestimmte Vorgänge nur vom Master-Server bzw. einem einzelnen oder mehreren Sub-Servern ausgeführt und protokolliert werden.

## 7.5 Kommandozeilenprogramme

Die Kommandozeilenprogramme schreiben ihre Meldungen in die Konsole oder in die beim Aufruf angegebene log-Datei (Argument **-logfile**, siehe Kapitel **Gemeinsame Argumente** (Seite 46)). Eine Unterscheidung zwischen Error- und Trace-Datei gibt es nicht. Exceptions werden mit anderen log-Ausgaben in die angegebene Datei oder auf der Konsole ausgegeben. Ob der Stack-Trace einer Exception mit ausgegeben wird, bestimmt der Wert des Schlüssels **PRINT\_STACKTRACE\_ON\_EXCEPTION** in der Konfigurationsdatei

**Server\_settings.properties**. Mögliche Werte sind **TRUE** und **FALSE**, Standardwert ist **FALSE**.

Die angegebenen Protokolloptionen (Argumente **-information**, **-warning**, **-error**) setzen den Loglevel modulspezifisch auf den angegebenen Wert **INFO**, **WARN** oder **ERROR**. Dabei wird der in der Datei **Server\_Log\_settings.properties** für das Modul eingestellte Loglevel berücksichtigt. Aufgrund der hierarchischen Loglevel-Struktur ergeben sich einige Besonderheiten bei der Auswertung der angegebenen Protokolloptionen. Der verwendete Loglevel wird wie folgt bestimmt:

1. In der Reihenfolge der angegebenen Argumente **-information**, **-warning**, **-error** wird der letzte gefundene Parameter **no** ermittelt.
2. In der Reihenfolge der angegebenen Argumente **-information**, **-warning**, **-error** wird der erste gefundene Parameter **yes** ermittelt.
3. Aus dem im ersten Schritt als letzten gefundenen no-Wert und dem modulspezifischen Standard-Loglevel wird der restriktivere Loglevel ermittelt. Ist kein no-Wert angegeben, wird der Standard-Loglevel genommen.
4. Aus dem im zweiten Schritt als ersten gefundenen yes-Wert und dem im dritten Schritt ermittelten Loglevel wird der am wenigsten restriktive Loglevel ermittelt.

### Beispiel 1

Standard-Loglevel für das Modul ist INFO. Beim Ausführen eines Kommandozeilenprogramms mit den Parametern **-information yes -warning no -error yes** wird der Loglevel wie folgt bestimmt.

- Letzter no-Wert ist bei **-warning**, LogLevel ist ERROR
- Erster yes-Wert ist bei **-info**, LogLevel ist INFO
- $\text{MAX}(\text{INFO}, \text{ERROR})$  ist ERROR
- $\text{LogLevel} = \text{MIN}(\text{ERROR}, \text{INFO})$  ist INFO

Der resultierende Loglevel ist INFO.

### Beispiel 2

Standard-Loglevel für das Modul ist WARN. Beim Ausführen eines Kommandozeilenprogramms mit dem Parameter **-information yes** wird der Loglevel wie folgt bestimmt.

- Letzter no-Wert ist nicht gesetzt.
- Erster yes-Wert ist bei **-info**, LogLevel ist INFO
- $\text{MAX}(\text{WARN})$  ist WARN
- $\text{LogLevel} = \text{MIN}(\text{WARN}, \text{INFO})$  ist INFO

Der resultierende Loglevel ist INFO.

### Beispiel 3

Standard-Loglevel für das Modul ist INFO. Beim Ausführen eines Kommandozeilenprogramms mit dem Parameter **-warning no** wird der Loglevel wie folgt bestimmt.

- Letzter no-Wert ist bei **-warning**, LogLevel ist ERROR
- Erster yes-Wert ist nicht gesetzt.
- $\text{MAX}(\text{ERROR}, \text{INFO})$  ist ERROR
- $\text{LogLevel} = \text{MIN}(\text{ERROR})$  ist ERROR

Der resultierende Loglevel ist ERROR.

## ZUSAMMENFASSUNG

- Ausführen eines Kommandozeilenprogramms mit dem Parameter **-information yes**

Es werden Informationen, Warnungen, Fehler und fatale Fehler ausgegeben, egal wie die restlichen Log-Argumente oder das modulspezifische Loglevel stehen.

- Ausführen eines Kommandozeilenprogramms mit den Parametern **-information no -warning yes**

Es werden Warnungen, Fehler und fatale Fehler ausgegeben, egal wie die restlichen Log-Argumente stehen.

Außerdem werden Informationen für Module ausgegeben, deren modulspezifisches Loglevel auf INFO steht.

- Ausführen eines Kommandozeilenprogramms mit den Parametern **-information no -warning no -error yes**

Es werden Fehler und fatale Fehler ausgegeben. Außerdem werden Informationen für Module ausgegeben, deren modulspezifisches Loglevel auf INFO steht und Warnungen für Module, deren modulspezifisches Loglevel auf INFO oder WARN steht.

- Ausführen eines Kommandozeilenprogramms mit den Parametern **-information no -warning no -error no**

Es werden nur fatale Fehlermeldungen ausgegeben. Für Module deren modulspezifisches Loglevel auf OFF steht, werden keinerlei Meldungen ausgegeben.

## LOG-AUSGABEN FORMATIEREN

Das Format der Log-Ausgaben der Kommandozeilenprogramme wird für Ausgaben an der Konsole vom eingestellten Format des Appender **console** bestimmt. Für Log-Ausgaben, die in eine Datei geschrieben werden, wird das Format durch den Appender **logFile** bestimmt.

Für die Kommandozeilenprogramme werden aus den Konfigurationsdateien

**Server\_Log\_settings.properties** und **Analysisserver\_Log\_settings.properties** keine weiteren Konfigurationen ausgewertet, ausgenommen die durch die entsprechenden Appender angegebene Formatvorlage. Zusätzliche Appender können für die Log-Ausgaben der Kommandozeilenprogramme daher nicht angegeben werden.

## LOG-AUSGABEN DER SERVER

Die Kommandozeilenprogramme **runppmconfig**, **runppmreport** und **runppmadmin** geben bestimmte Log-Meldungen des Mandanten- und Analyseservers aus. Diese Meldungen erscheinen sowohl in den Log-Ausgaben der Server als auch in denen des Kommandozeilenprogramms, gekennzeichnet durch den zusätzlichen Modulnamen.

### Beispiel

```
...
I: 05.10.10 14:41:53 [IMP] Importiere Konfiguration der Komponente "users"...
I: 05.10.10 14:41:53 [IMP] [SRV] Importiere Benutzer...
I: 05.10.10 14:41:53 [IMP] [SRV] Importiere Gruppen...
I: 05.10.10 14:41:53 [IMP] [SRV] Folgende 0 Benutzer wurden hinzugefügt: []
I: 05.10.10 14:41:53 [IMP] Die Konfiguration der Komponente "users" wurde aus der
Datei
...
```

## 8 Globalisierung

### LOKALISIERUNG

#### APPLIKATION

Die Applikation webMethods Process Performance Manager ist unter anderem für folgende Sprachen verfügbar:

- deutsch
- englisch
- französisch
- ungarisch
- japanisch
- chinesisch

Die Sprachauswahl wird während des Anmeldevorgangs angegeben und bezieht sich auf die PPM-Benutzeroberfläche und log-Ausgaben. Unabhängig von der Oberflächensprache werden eingelesene Daten in der Sprache des Quellsystems angezeigt, aus dem sie extrahiert wurden.

#### DOKUMENTATION

Die mit PPM ausgelieferte Online-Hilfe und ergänzende Dokumentationen im PDF-Format sind in den Sprachen **Deutsch** und **Englisch** verfügbar.

#### KALENDER

PPM unterstützt derzeit ausschließlich den gregorianischen Kalender.

#### BIDIREKTIONALE TEXTDARSTELLUNG

Die bidirektionale Darstellung RTL-basierter Oberflächentexte wird nur eingeschränkt unterstützt, Oberflächentexte (z. B. Favoriten-, Kennzahlen, Dimensionsnamen) werden nicht einheitlich korrekt dargestellt. Die Anordnung der Steuerelemente ist unabhängig von der gewählten Oberflächensprache immer einheitlich.



## 9 Auftragsautomatisierung

Auftragsautomatisierung ermöglichen Ihnen ausgewählte Programme zu bestimmten Zeitpunkten automatisiert ausführen zu lassen. In einer Auftragsautomatisierung geben Sie die Programme und die Ausführungsregel an, nach der die Programme regelmäßig ausgeführt werden sollen.

Sie können Auftragsautomatisierungen für einen einzelnen Mandanten definieren. Dazu stehen Ihnen die Programme des jeweiligen Mandanten zur Verfügung.

Nach der Ausführung der Auftragsautomatisierung wird automatisch eine Nachricht erzeugt und an festgelegte Empfänger versandt.

Mit Hilfe eines Assistenten können Sie in PPM Customizing Toolkit eigene Auftragsautomatisierungen anlegen.

Weitere detaillierte Informationen zu diesem Thema erhalten Sie in dem Handbuch PPM Customizing Toolkit und in der integrierten Hilfe des entsprechenden Assistenten oder Dialogs.

## 10 Barrierefreiheit

PPM unterstützt in der vorliegenden Version nur eingeschränkt die Barrierefreiheit, wie sie von **Section 508** (<http://www.section508.gov>) und **WCAG** (Web Content Accessibility Guidelines, <http://www.w3.org/TR/WAI-WEBCONTENT>) empfohlen wird.

## 11 SSO-Integration

Mit Hilfe der Anbindung von PPM an SAML2 und Kerberos können Sie PPM in ein Single-Sign-On-Szenario einbinden.

SAML (Security Assertion Markup Language) und Kerberos ermöglichen eine Einmalanmeldung eines Benutzers (Single Sign-on kurz SSO) an mehreren Anwendungen gleichzeitig. Das heißt, der Benutzer ist nach der Anmeldung an einer Anwendung automatisch auch zur Benutzung von weiteren Anwendungen berechtigt.

PPM unterstützt ab der Version 9.0 eine Benutzeranmeldung mittels SAML 2 und Kerberos.

### 11.1 Parameter konfigurieren

Um PPM in ein SSO-Szenario per SAML2 oder Kerberos einzubinden, müssen Sie die erforderlichen Parameter mit Hilfe der zentralen Benutzerverwaltung konfigurieren. Die Parameter können Sie auf der Seite Konfiguration in der zentralen Benutzerverwaltung konfigurieren.

Detaillierte Informationen zur Bedeutung und Konfiguration der Parameter erhalten Sie in der Online-Hilfe der zentralen Benutzerverwaltung (<https://<host>/umc/help/de/handling/index.htm#52345.htm>).

#### 11.1.1 SAML2-Anbindung

Für eine SAML2-Anbindung muss der HTTP-Anfrage beim Aufruf von PPM eine sogenannte SAML-Signed-Assertion mitgegeben werden, die von PPM gegenüber einer Authentifizierungsstelle geprüft wird. Die SAML-Signed-Assertion wird in den meisten Fällen auch von der rufenden Anwendung bereitgestellt. Die Assertion enthält unter anderem die Benutzerdaten des Benutzers, der sich an PPM anmelden möchte. Ist die Assertion gültig und der betreffende Benutzer in PPM bekannt und auch aktiv, so ist die Anmeldung erfolgreich und er bekommt Zugang zu PPM.

#### SAML 2-PARAMETER

Mit Hilfe folgender Einträge in der zentralen Benutzerverwaltung können Sie den SAML 2-Zugang konfigurieren.

- `com.aris.umc.saml.active = true`
- `com.aris.umc.saml.assertion.timeoffset = 60` // max. erlaubte Zeitdifferenz in Sekunden zwischen dem Aussteller der Signed-Assertion und dem Verifizierer der Signed-Assertion.
- `com.aris.umc.saml.assertion.ttl = 120` // max. Gültigkeitsdauer der Signed Assertion
- `com.aris.umc.saml.keystore.alias = Alias des Keystore-Users`
- `com.aris.umc.saml.keystore.location = Ort des Keystores` (Wird bei Upload des Keystore in die UMC automatisch eingetragen)

`com.aris.umc.saml.keystore.password = Kennwort des Keystore-Benutzers`

- `com.aris.umc.saml.keystore.type = Verschlüsselungstyp des Keystores, z.B. ‚JKS‘`

- `com.aris.umc.saml.truststore.alias` = Alias des Truststore-Benutzers.
- `com.aris.umc.saml.truststore.location` = Ort des Truststore (Wird bei Upload des Truststore in die UMC automatisch eingetragen)
- `com.aris.umc.saml.truststore.password` = Kennwort des Truststore-Benutzers
- `com.aris.umc.saml.truststore.type` = Verschlüsselungstyp des Truststore, z.B. ‚JKS‘

Der Truststore wird verwendet um die Signatur in der vom Fremdsystem erhaltenen Signed Assertion zu prüfen, während der Keystore verwendet wird um die eigenen Signed Assertions zu signieren. Somit muss jeweils das Zertifikat im Keystore des Ausstellers im Truststore des Empfängers vorhanden sein, damit die Verifizierung erfolgen kann. Idealerweise sind die Zertifikate auf beiden Seiten (Truststore / Keystore im Sender / Empfänger) identisch.

### 11.1.2 Kerberos

Kerberos ist ein Authentifizierungsprotokoll, das eine gegenseitige Authentifizierung von Anwendungen in einem Netzwerk mit Hilfe einer Schlüssel- Kryptografie ermöglicht. PPM kann die Benutzeranmeldedaten des Betriebssystems (z. B. MS Windows) verwenden, um den Benutzer automatisch an PPM im Web-Browser anzumelden.

#### KERBEROS-PARAMETER

Mit Hilfe folgender Einträge in der zentralen Benutzerverwaltung können Sie den Kerberos-Zugang konfigurieren.

- `com.aris.umc.kerberos.active`
- `com.aris.umc.kerberos.config`
- `com.aris.umc.kerberos.debug`
- `com.aris.umc.kerberos.kdc`
- `com.aris.umc.kerberos.keyTab`
- `com.aris.umc.kerberos.realm`
- `com.aris.umc.kerberos.servicePrincipalName`

## 12 Sicherheitsaspekte

Dieses Kapitel beschreibt alle sicherheitsrelevanten Aspekte für die Installation eines PPM-Systems mit allen benötigten Komponenten.

### 12.1 Sicherheit der Kommunikationskanäle

Die Kommunikation des PPM Clients erfolgt zum einen direkt mit dem Load Balancer und zum anderen mit dem PPM Server über ein TCP/IP Netzwerk. Im Folgenden werden die einzelnen Schritte zur Absicherung der Kommunikation mit dem Load Balancer und dem PPM Server näher erläutert.

### 12.2 Load Balancer

Der Load Balancer ermöglicht standardmäßig den unverschlüsselten Zugriff mittels HTTP (Port 4080) und den verschlüsselten Zugriff mittels HTTPS/SSL (Port 4443). Eine zusätzliche Konfiguration zur Aktivierung der SSL-Verschlüsselung ist hier nicht notwendig. Standardmäßig wird ein selbst erzeugtes Zertifikat des Load Balancer für die HTTPS-Unterstützung verwendet. Dies wird in den meisten Browsern jedoch abgelehnt, da es kein gültiges Root-Zertifikat einer zugelassenen Zertifizierungsstelle enthält. Wie Sie ein eigenes gültiges Zertifikat für die HTTPS-Unterstützung des Load Balancer erstellen können, ist im nachfolgenden Kapitel beschrieben.

#### 12.2.1 SSL-Zertifikat einbinden

In diesem Abschnitt wird erläutert, wie Sie ein eigenes gültiges Zertifikat für die HTTPS-Unterstützung des Load Balancer einrichten können. Dafür muss das zu erzeugende SSL-Zertifikat an den Load Balancer-Host-Namen angepasst werden, um Warnmeldungen seitens des Client insbesondere des verwendeten Web-Browser zu vermeiden und eine korrekte Funktionsweise zu gewährleisten.

Um SSL verwenden zu können, benötigen Sie für den Server, auf dem der Load Balancer läuft, ein gültiges Zertifikat einer Zertifizierungsstelle. Stellen Sie sicher, dass das ausgestellte Zertifikat kompatibel zur verwendeten Java-Version des Client ist.

- Sie können ein Zertifikat bei einer offiziellen Zertifizierungsstelle kaufen. Die meisten Clients insbesondere Web-Browser akzeptieren ein solches Zertifikat.
- Falls Ihr Unternehmen eine eigene Zertifizierungsstelle verwendet, binden Sie diese Zertifizierungsstelle in den Trust store jedes Clients ein.

Ihr Zertifikat sollte aus zwei Teilen bestehen, dem privaten Schlüssel (Dateiendung **.key**) zur Entschlüsselung der Information, die zurück an den Client gesendet wird, und dem Serverzertifikat (Dateiendung **.crt**). Ein Beispiel zur Erzeugung dieser Dateien ist in Kapitel PPM Server (Seite 43) beschrieben.

#### Vorgehen

1. Packen Sie beide Dateien in ein ZIP-Datei.

2. Kopieren Sie die ZIP-Datei an eine Stelle, um mit dem Cloud Controller darauf zugreifen zu können.
3. Starten Sie den Cloud Controller.
4. Stoppen Sie im Cloud Controller den Load Balancer.
5. Geben Sie im Cloud Controller den Befehl **enhance <instanceID of the load balancer component> with sslCertificate local file "<Pfad zur ZIP-Datei>"**.

Wenn die Instanz-ID des Load Balancer **loadbalancer\_m** ist und Ihre ZIP-Datei liegt unter **c:\temp\lbcert.zip**, geben Sie den Befehl **enhance loadbalancer\_m with sslCertificate local file "c:\\temp\\lbcert.zip"** ein.

Beachten Sie die Verwendung von doppelten Backslashes oder alternativ einfache Schrägstriche, z. B. **"c:/temp/lbcert.zip"**.

6. Starten Sie wieder den Load Balancer.

Das SSL-Zertifikat steht jetzt zur Verfügung.

## 12.2.2 Weitere Absicherung des Load Balancer

Zentrale Systeme wie der Load Balancer, der für die Kommunikation zwischen Client und Server verantwortlich ist, werden oft Ziel von Angriffen aus dem Internet. Um aktuelle und bekannte Sicherheitslücken zu schließen, können einige Maßnahmen getroffen werden. Diese werden folgend kurz beschrieben und diskutiert.

- Halten Sie Ihr Betriebssystem immer auf dem aktuellsten Stand. Führen Sie regelmäßig notwendige Sicherheitsaktualisierungen durch.
- Halten Sie Ihre Produktinstallation immer auf dem aktuellsten Stand. Führen Sie in regelmäßigen Abständen Prüfungen durch, ob neue Fixes für Ihre Installation verfügbar sind und installieren Sie diese.
- Verwenden Sie kein selbst signiertes Zertifikat für den Load Balancer, sondern lassen Sie Ihren Keystore von einer autorisierten Stelle signieren.
- Um die sogenannte „Logjam“ Sicherheitslücke zu schließen, ist es notwendig eine sogenannte „Diffie-Hellman“ Gruppe für den Load Balancer zu erzeugen. Dies kann mit den folgenden Schritten erreicht werden:
  - Wechseln Sie in das Verzeichnis `<Installationsverzeichnis>\server\bin\agentLocalRepo\unpacked\httpd-run-prod-<version>-runnable.zip\httpd\bin`. Ersetzen Sie dabei `<Installationsverzeichnis>` mit dem Pfad zu Ihrer PPM-Installation und `<version>` mit der bei Ihnen installierten Version.
  - Öffnen Sie eine Kommandozeile und geben folgenden Befehl ein:  
`openssl dhparam -out dhparams.pem 2048`  
Dies wird eine benutzerdefinierte DH-Gruppe in einer Datei **dhparams.pem** im gleichen Verzeichnis erzeugen.
  - Nun müssen Sie den Inhalt dieser Datei an das SSL-Zertifikat des Servers anhängen, das in folgendem Verzeichnis liegt.

<Installationsverzeichnis>\server\bin\work\work\_<loadbalancerInstanceld>\httpd\conf

- Starten Sie den Load-Balancer neu, damit die Änderungen wirksam werden.

## 12.3 PPM Server

Mit PPM Server ist hier der PPM Mandanten-Server gemeint, der die Anfragen an einen bestimmten Mandanten von einem beliebigen Client entgegennimmt. Die Kommunikation zwischen Client und Server ist nach der Installation nicht abgesichert. Im folgenden Abschnitt wird kurz beschrieben, wie die SSL Verschlüsselung zwischen Client und Server eingerichtet werden kann.

### 12.3.1 SSL-Keystore erzeugen und signieren

Um die RMI Kommunikation in PPM durch die Verwendung von SSL abzusichern, müssen Sie zunächst einen eigenen Keystore erstellen. Sie können mit Hilfe des Java-Keytool einen eigenen Keystore einrichten und administrieren. Detaillierte Informationen zu diesem Thema erhalten Sie in der Dokumentation **Keytool - Key and Certificate Management Tool** (<http://docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html>).

#### Warnung

Im Folgenden ist das Erzeugen des Keystores und des Certificate Signing Requests an einfachen Beispielen mit Demo Daten erläutert. Die verwendeten Werte der einzelnen Parameter müssen an Ihre lokale Umgebung angepasst werden (besonders die Informationen zu dem Parameter „-dname“). Eine detaillierte Beschreibung der Bedeutung der einzelnen Parameter finden Sie in der o.a. Dokumentation **Keytool - Key and Certificate Management Tool**.

#### Vorgehen

1. Erzeugen Sie mit dem Keytool einen eigenen Keystore mit den folgenden Parametern.

keytool

```
-keystore ppm_keystore
-alias ppm
-keypass changeit
-storepass changeit
-genkey
-v
-keyalg RSA
-dname "CN=PPM Admin , OU=PPM , O=SAG , L= Saarbruecken , S=SAL , C=DE"
```

2. Erzeugen Sie mittels Keytool den Certificate Signing Request mit den folgenden Parametern.

keytool

```
-certreq
-alias ppm
```

```
-file ppm_keystore.csr  
-keystore ppm_keystore
```

Das Programm erzeugt zwei Dateien: einen Schlüssel (Endung **.key**) und den eigentlichen Zertifikats Request (Endung **.csr**).

3. Reichen Sie den Inhalt der Datei mit der Endung **.csr** bei einer Zertifizierungsstelle ein.  
Diese signiert das Zertifikat mit einem gültigen Root Zertifikat, das in der Regel von allen Browsern und den standardmäßig ausgelieferten Java Laufzeitumgebungen anerkannt wird.
4. Importieren Sie mittels Keytool dieses signierte Zertifikat in Ihren keystore.

```
keytool  
-import  
-trustcacerts  
-alias ppm  
-file <DateiName>  
-keystore ppm_keystore
```

Die Datei **ppm\_keystore** kann nun zur SSL-Verschlüsselung in PPM verwendet werden.

### 12.3.1.1 PPM für SSL konfigurieren

Mit dem neu erzeugten Keystore kann nun die RMI-Kommunikation zwischen PPM-Client und PPM-Server auf SSL-Verschlüsselung umgestellt werden. Dazu müssen Sie folgende Befehle im Cloud Controller ausführen.

- stop ppm\_core
- reconfigure ppm\_core ssl.enabled="true" ssl.keystorefile="<Pfad zur keystore Datei>"  
ssl.keystorepassword="<keystorepassword>"
- start ppm\_core

### 12.3.2 Absicherung des Server Systems

Um Ihr System vor unbefugten Zugriffen zu schützen, sollten Sie nur einem eingeschränkten Nutzerkreis direkten Zugriff auf das System (z. B. remote per RDP oder direkt über eine Management-Konsole) gewähren.

Zudem sollten nur die notwendigen Dienste auf dem Server (wie Datenbank, PPM-System, Load Balancer) in der Firewall freigeschaltet sein.



## 13 Kommandozeilenprogramme

Dieses Kapitel gibt einen Überblick über die wichtigsten Kommandozeilenprogramme, mit deren Hilfe Anwendungsdaten im PPM-System verwaltet werden.

### 13.1 Allgemeiner Aufbau und funktionspezifische Struktur

Die Kommandozeilenprogramme (BAT-Dateien) zur Administration und Konfiguration des PPM-Systems im Verzeichnis **bin** der PPM-Installation sind vierstufig aufgebaut:

1. Ermitteln und Überprüfen der PPM-Umgebungsvariablen
2. Zusammenstellen einer Argumentzeichenkette aus allen übergebenen Kommandozeilenargumenten
3. Erstellen eines lokalen Klassenpfades
4. Starten der Java Virtual Machine mit Angabe des Klassenpfades und der Argumentzeichenkette sowie Aufruf einer Kommandozeilenprogramm-abhängigen Klasse in der VM

#### PPM-spezifische Batch-Dateien

- Windows: Die Windows-Batchdateien (\*.bat) befinden sich im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-<version>-runnable.zip\ppm\bin**.
- Unix: Im entsprechenden Verzeichnis befinden sich die Verknüpfungen zu dem Shell-Skript **.runppmcommand.sh**, das die gewünschte Funktionalität ausführt.

#### CTK-spezifische Batchdateien

- Windows: Die Windows-Batchdateien (\*.bat) befinden sich im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-<version>-runnable.zip\ppm\ctk\bin**.
- Unix: Im entsprechenden Verzeichnis liegen die Verknüpfungen zu dem Shell-Skript **.runctkcommand.sh**, das die gewünschte Funktionalität ausführt.

#### Mandantenspezifische Batch-Dateien

- Windows: Die Windows-Batchdateien (\*.bat) befinden sich im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\custom\<mandant>\bat**.
- Unix: Die Shell-Skripte befinden sich im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\custom\<mandant>\bin**.

Diese Batch-Dateien (sowohl Windows als auch Unix) werden auch im CTK im Modul **Mandant -> Programme** referenziert.

Komplexe Funktionalitäten (ACC-Start, Start/Stop Infrastruktur, Start/Stop Mandanten, CTK, PPM GUI usw.) werden durch Batch-Dateien in folgenden Verzeichnissen ausgeführt.

- **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\bin**
- **<Installationsverzeichnis>\ppmmashzone\server**

Unter Windows sind im Startmenü Verknüpfungen zu diesen Batch-Dateien angelegt.

## 13.2 PPM-Kommandozeilenprogramme

Folgende Kommandozeilenprogramme stehen Ihnen zur Verfügung.

- `runinitdb` (Seite 48)  
Initialisiert das Datenbankschema eines PPM-Mandanten
- `runppmanalytics` (Seite 49)  
Untersucht im Rahmen von Easy Mining Prozessinstanzen auf Unregelmäßigkeiten
- `runppmconfig` (Seite 51)  
Liest Konfigurationen in das Datenbankschema eines PPM-Mandanten ein
- `runppmcpi` (Seite 59)  
Löscht im Rahmen von Easy Mining automatisch mehrere Nachrichten gleichzeitig
- `runppmimport` (Seite 61)  
Steuert die Weiterverarbeitung der mittels **runxmlimport** eingelesenen Prozessinstanzfragmente (Zusammenführen, Typisieren, Kennzahlen berechnen)
- `runprocessimport` (Seite 65)  
Importiert prozessabhängige Daten nach PPM
- `runppmcompress` (Seite 69)  
Verdichtet importierte Prozessinstanzen
- `runppmdelete` (Seite 72)  
Löscht Prozessinstanzen testweise oder tatsächlich
- `runppmadmin` (Seite 75)  
Führt verschiedene administrative Funktionen aus
- `runppmsendmail` (Seite 77)  
Versendet einfache Textnachrichten
- `runtableimport` (Seite 78)  
Liest Daten aus einer Datei im Format **XML** oder **ZIP** (XML-Event-Format)

Alle Kommandozeilenprogramme werden im Verzeichnis

**<Installationsverzeichnis>\ppmmashzone\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-<version>-runnable.zip\ppm\bin** ausgeführt.

Für allgemein gültige Angaben (z. B. PPM-Benutzerkonto und Mandant) werden in allen Kommandozeilenprogrammen dieselben Argumente verwendet. Spezielle Argumente dienen der Steuerung der komponentenspezifischen Funktionalitäten.

Der Aufruf eines der folgend beschriebenen Programme ohne Parameter oder mit **-h** oder **-?** gibt auf der Konsole die Online-Hilfe aus, in der alle verfügbaren Optionen kurz beschrieben sind.

## 13.3 Gemeinsame Argumente

**-VERSION**

Auf der Konsole werden die Version der PPM-Software und des Datenbankschemas ausgegeben. Andere Argumente werden ignoriert.

**-USER <BENUTZERNAME> -PASSWORD <KENNWORT>**

Hier geben Sie den Benutzernamen und das Kennwort des PPM-Benutzers an. Der angegebene Benutzer muss für die gewünschte Aktion eine entsprechende Berechtigung besitzen.

Funktionsrecht	Kommandozeilen-programm	Beschreibung
<b>Datenimport</b>	runxmlimport, runprocessimport und andere Quellsystemadapter	Einlesen von Prozessinstanzfragmenten
<b>Datenberechnung</b>	runppmimport und runprocessimport	Berechnung von Prozessinstanzen
<b>Konfigurations- import</b>	runppmconfig runinitdb	Einlesen von XML-Konfigurationsdateien
<b>Prozessinstanz- verdichtung</b>	runppmcompress	Verdichten von Prozessinstanzen

Die Funktionsrechte werden in der Administrationskomponente **Benutzer** des Frontend verwaltet. Der Systembenutzer **system** hat alle Funktionsrechte.

Wenn Sie das Kennwort verdeckt eingeben möchten, geben Sie in der Kommandozeile nur das Argument **-user** mit dem gewünschten Benutzernamen an. Das Kennwort wird anschließend interaktiv abgefragt.

**-CLIENT <NAME>**

Hier geben Sie den PPM-Mandanten an, dessen Datenbankschema vom jeweiligen Kommandozeilenprogramm verwendet wird. Wenn Sie diese Angabe nicht vornehmen, wird der Standardmandant **default** verwendet.

**-LANGUAGE <ISO-CODE>**

Durch diese Option bestimmen Sie die Sprache, in der die Protokollausgaben erfolgen. Mögliche Werte sind **de** für Deutsch, **en** für Englisch und **fr** für Französisch.

**-PROTOCOLFILE <FILENAME>**

Diese Option schreibt die Protokollausgaben in die angegebene Datei. In der Konsole werden nur noch kritische Fehlermeldungen ausgegeben, die zum Programmabbruch führen.

## **-INFORMATION {YES|NO|DEFAULT}, -WARNING {YES|NO|DEFAULT}, -ERROR {YES|NO|DEFAULT}**

Diese Parameter steuern die Ausgabe von Meldungen. Die Kommandozeilenoptionen sind ausführlich im Kapitel **Kommandozeilenprogramme** (Seite 33) beschrieben.

### 13.4 runinitdb

Das Kommandozeilenprogramm **runinitdb** initialisiert das Datenbankschema des angegebenen PPM-Mandanten. Hierfür muss die Option **-init** mit angegeben werden. Der Vorgang läuft zweistufig ab.

Das Datenbankschema wird neu angelegt und mit Standardwerten für jede der in der Datei **Locales.xml** referenzierten Sprachen gefüllt. Der Dateiname der Datei **Locales.xml** kann beliebig gewählt werden und wird in der Datei **InitDB\_settings.properties** angegeben.

Um ein existierendes Datenbankschema zu initialisieren, müssen Sie die Parameter **-user system -password <Kennwort>** angeben. Nach erfolgreicher Initialisierung ist das Kennwort des Benutzers **system** auf den Standardwert **manager** zurückgesetzt.

Anschließend wird der PPM-Mandantenserver gestartet und die in der Datei **InitSystem\_settings.properties** angegebenen Konfigurationsdateien werden eingelesen.

Die Datei **InitSystem\_settings.properties** besteht aus verschiedenen Abschnitten. Jeder Abschnitt setzt sich aus drei Zeilen zusammen. Jede Zeile beginnt mit einem Schlüsselwort, dem durch ein Gleichheitszeichen ein Wert zugewiesen wird:

Schlüsselwort	Beschreibung
INIT_MODULE_X	Abschnittseinleitung
INIT_MODULE_X_NAME	Name der Konfigurationskomponente. Angabe entspricht dem Argument <b>command</b> des Programms <b>runppmconfig</b> .
INIT_MODULE_X_FILE	Dateiname mit Pfadangabe der XML-Konfigurationsdatei

**X** ist Platzhalter für die Nummer eines Abschnitts in der Datei. Alle drei Zeilen eines Abschnittes müssen angegeben werden. Die Nummerierung der Abschnitte muss mit **1** beginnen und zusammenhängend sein. Die Reihenfolge der Abschnitte in der Datei ist jedoch beliebig.

#### **Warnung**

Beachten Sie beim Bearbeiten der Datei **InitSystem\_settings.properties** unbedingt die Reihenfolge der Konfigurationskomponenten: Datentypen, Attribute, Prozesstypen, Kennzahlen, Prozessbaum, sonstige.

#### **Beispiel**

Auszug aus der Standarddatei **InitSystem\_Settings.properties**:

```
#
```

```
# Initialisierung aller runppmconfig-Komponenten entsprechend
# der angegebene Reihenfolge
#
# INIT_MODULE_XXX_NAME : Name des Moduls entsprechend der
#                       Kommandozeilenoption von runppmconfig
# INIT_MODULE_XXX_FILE : Name der XML-Datei mit den entspr. Konfig-Daten
#
```

```
INIT_MODULE_1 =
INIT_MODULE_1_NAME = datatypes
INIT_MODULE_1_FILE = D:/ppm/xml/DefaultDataTypes.xml
```

```
INIT_MODULE_2 =
INIT_MODULE_2_NAME = attributetypes
INIT_MODULE_2_FILE = D:/ppm/xml/DefaultAttributeTypes.xml
```

```
INIT_MODULE_3 =
INIT_MODULE_3_NAME = attributenames
INIT_MODULE_3_FILE = D:/ppm/xml/DefaultAttributeNames.xml
...
```

Um während der Initialisierung die Konfiguration von Organisationseinheiten einzulesen, editieren Sie die Datei **InitSystem\_settings.properties** im Konfigurationsverzeichnis des gewünschten Mandanten und fügen Sie am Ende der Datei folgende Zeilen ein:

```
...
INIT_MODULE_X =
INIT_MODULE_X_NAME = orgunits
INIT_MODULE_X_FILE = <Organisationseinheiten.xml>
```

Für **X** setzen Sie die um **1** erhöhte Nummer des vorangegangenen Abschnittes.

In der Praxis hat es sich bewährt, die Konfigurationsdateien mandantenabhängig in Verzeichnissen der Form

**<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\custom\<Mandant>\xml** abzulegen.

## 13.5 runppmanalytics

Im Rahmen des **Easy Mining** können Sie Prozessinstanzen auf Unregelmäßigkeiten untersuchen. Dabei werden mehrere analytische Funktionalitäten, wie Frühwarnsystem, Alarmwertabweichung, Prozess-Mining, Ausreißeranalyse und Planwertüberschreitung ausgeführt.

Neben der Bündelung der verschiedenen Analysen in einem PPM-Werkzeug werden für alle auftretenden Unregelmäßigkeiten entsprechende Nachrichten erstellt und den Benutzern von PPM mitgeteilt.

Mit Hilfe des Kommandozeilenprogramms **runppmanalytics** können die verschiedenen Easy-Mining-Analysen einzeln oder alle nacheinander gestartet werden.

Wird eine Analyse über alle Analysearten gemacht, so findet dies immer in einer festgelegten Reihenfolge statt.

1. Early-Alert-Analyse
2. Planwertanalyse

3. Alarmwertanalyse
4. Ausreißeranalyse
5. Process-Mining-Analyse

Das Programm liegt im Verzeichnis

**<Installationsverzeichnis>\ppmmashzone\server\bin\agentLocalRepo\unpacked\ppm-client-run-<version>-runnable.zip\ppm\bin.**

Zum Ausführen von **runppmanalytics** ist das Funktionsrecht **PPM-Anmeldung** erforderlich.

#### **Tipp**

Weitere Informationen zu **Easy Mining** erhalten Sie in der Online-Hilfe **PPM-Hilfethemen**.

### **13.5.1 Argumente runppmanalytics**

Sie können runppmanalytics mit folgenden spezifischen Argumenten ausführen.

Optional kann zu jedem Argument der Pfad zu einem Favoritenverzeichnis **<favoritepath>** angegeben werden.

z. B. **runppmanalytics -earlyalert <favoritepath>**

Ist ein Favoritenverzeichnis angegeben, so wird die entsprechende Analyse für alle darin befindlichen Favoriten ausgeführt.

Ohne Pfadangabe werden die Standardanalysen durchgeführt.

#### **-EARLYALERT**

Es wird eine **Frühwarnanalyse** durchgeführt.

#### **-PLANNEDVALUE**

Es wird eine **Planwertabweichungsanalyse** durchgeführt.

#### **-ALARMVALUE**

Es wird eine **Alarmwertabweichungsanalyse** durchgeführt.

#### **-PROCESSMINING**

Es wird eine **Auffälligkeitsanalyse** durchgeführt.

#### **-OUTLIER**

Es wird eine **Ausreißeranalyse** durchgeführt.

#### **-ALL**

Es werden nacheinander **alle Analysen** in einer fest definierten Reihenfolge durchgeführt (Siehe runppmanalytics (Seite 49)).

Der Aufruf des Programms mit dem Parameter **-h** gibt auf der Konsole die Online-Hilfe aus, in der alle verfügbaren Optionen kurz beschrieben sind.

**-REGISTRY <HOST:PORT>**

Durch Angabe dieses Parameter können Sie auf ein entferntes PPM-System zugreifen, dessen RMI-Registry unter dem angegebenen Rechnernamen und Portnummer zu erreichen ist.

**-USESSL <TRUE|FALSE|DEFAULT>**

Wenn Sie den Parameter **-registry** angegeben haben, können Sie mittels Parameter **-usessl** bestimmen, ob das entfernte System SSL-verschlüsselte Datenkommunikation verwenden soll oder nicht.

Parameter	Beschreibung
true	Das entfernte PPM-System verwendet verschlüsselte Datenübertragung. Stellen Sie sicher, dass das lokale System den gleichen Keystore wie das entfernte System verwendet.
false	Das entfernte System verwendet unverschlüsselte Datenübertragung.
default (Standardwert)	Es werden die Einstellungen des lokalen PPM-Systems verwendet.

## 13.6 runppmconfig

Das Kommandozeilenprogramm **runppmconfig** dient zum Einlesen von XML-Konfigurationsdateien in das PPM-System und zum Speichern der Konfiguration eines PPM-Mandanten in XML-Dateien.

### 13.6.1 Argumente runppmconfig

**-MODE <MODUS>**

Hier bestimmen Sie, ob die Konfiguration der mit dem Argument **<command>** angegebenen PPM-Komponente in die angegebene Datei gespeichert wird oder ob die in der Datei gespeicherte Konfiguration in das PPM-System eingelesen wird. Mögliche Werte sind **export** und **import**.

**-<COMMAND> <DATEINAME>**

Hier bestimmen Sie die PPM-Komponente, deren Konfiguration in der angegebenen Datei gespeichert bzw. aus dieser eingelesen werden soll.

command	PPM-Komponente
abcanalysis	Konfiguration der ABC-Analyse

command	PPM-Komponente
attributenames	Namen der PPM-Attribute. Zum Exportieren bestimmen Sie durch das Argument <b>-language &lt;ISO-code&gt;</b> die gewünschte Sprache der PPM-Attributnamen. Fehlt dieses Argument, werden die Attributnamen in der Standardsprache des PPM-Mandanten (XML-Element <b>defaultlanguage</b> der Datei <b>Locales.xml</b> ) exportiert. Beim Import wird das in der XML-Datei angegebene Sprachkürzel verwendet.
attributetypes	Definition der im PPM-System bekannten Attributtypen
copyattributerules	Regeln zum Kopieren der Objektattribute an die Prozessinstanz
cpi	Nachrichten des CPI-Moduls
datatypes	Erweiterte Datentypdefinition
dimgroup	Dimensionengruppen
epctemplate	Vorlagen zur EPK-Darstellung
factorycalendar	Werkskalender
favorites [-favoriteuser <user>]	Im- bzw. Export von Favoriten. Der Vorgang kann mittels Parameter <b>favoriteuser</b> auf die Favoriten eines Benutzers beschränkt werden. Geben Sie optional <b>-favoriteuser sharedfavorites</b> an, werden lediglich alle gemeinsamen Favoriten im- bzw. exportiert.
images	EPK-Darstellung mit Hintergrundbild und anderen Symbolen für EPK-Objekte versehen
keyindicator	Prozessinstanzabhängige und ~unabhängige Kennzahlen sowie Dimensionen, Attributberechnungen und Beziehungen
keyrules	Regeln zur Schlüsselerzeugung
kidepend	Benutzerdefinierte Dimensionen
kigroup	Kennzahlengruppen
locales	Oberflächensprachen von PPM
merger	Merge-Konfiguration
mysaptransactions	SAP-Transaktionen



command	PPM-Komponente
onlinekis	Benutzerdefinierte Kennzahlen
orgunits	Organisationseinheiten
processtree [-add]	Definition des Prozessbaumes Wenn Sie die Option <b>-add</b> angeben, werden dem Prozessbaum neue Elemente aus der angegebenen Datei hinzugefügt. Bereits vorhandene Elemente des Prozessbaums werden nicht ersetzt.
ranking	Konfiguration der Ranking-Analyse
realm	Name des Analysebereichs, der importiert werden soll.
reportautomations	Gesamtkonfiguration der Reportautomatisierungen
serverproperties	Export allgemeiner Servereinstellungen
system	Systemkonfiguration. Wird in XML-Dateien im angegebenen Verzeichnis gespeichert. (Kapitel <b>Export der Systemkonfiguration</b> (Seite 55))
targetvalues	Planwertdefinitionen
topflop	Top-Flop-Dimensionen
transformationfactors	Umrechnungsfaktoren
users	Benutzerrechte

Jeden möglichen Parameterwert können Sie auch in der Datei **InitSystem\_settings.properties** angeben, sodass die Konfiguration sofort nach Initialisieren der PPM-Datenbank aktiv ist. Beachten Sie dabei bitte die Abhängigkeit verschiedener Konfigurationen voneinander.

### **-OVERWRITE**

Dieser Schalter erzwingt, dass beim Einlesen einer Konfiguration existierende Konfigurationen ersetzt werden. Er ist nur für folgende Konfigurationskomponenten anwendbar:

-targetvalues, -abcanalysis, -datatypes, -factorycalendar, -keyindicator,  
-ranking, -onlineki, -favorites, -reportautomations

### **Warnung**

Das Ersetzen bestehender Konfigurationen durch Angabe der Option **-overwrite** kann zu unbeabsichtigten Effekten führen, wenn Sie z. B. die Definition bereits verwendeter Datentypen ändern.

**-REPLACE**

Dieser Schalter löscht vor dem Beginn des Imports eine existierende Konfiguration. Er ist nur für folgende Konfigurationskomponenten anwendbar: -targetvalues, -abcanalysis, -ranking -favorites, -reportautomations

**Beispiel**

Die Konfiguration der benutzerdefinierten Kennzahlen des PPM-Mandanten **umg\_de** soll in der XML-Datei **OnlineKI.xml** auf Laufwerk **C** gespeichert werden.

```
runppmconfig -client umg_de -user system -password manager -mode export
-onlinekis C:\OnlineKI.xml
```

**Warnung**

Ändern Sie in einer bereits importierten Kennzahlenkonfiguration nicht nachträglich die Auflösung (**precision**) von Zeitdimensionen (**timedim**) und Uhrzeitdimensionen (**hourdim**), z. B. von **HOURL** auf **SECOND**, da der Re-Import mit **runppmconfig -overwrite** nicht die dafür erforderlichen Änderungen in den Datenbanktabellen erzeugen kann.

Wenn Sie die Auflösung von Zeit- und Uhrzeitdimensionen einer bereits importierten Konfiguration ändern möchten, müssen Sie mit der geänderten Konfiguration eine erneute Datenbankinitialisierung durchführen.

**-REGISTRY <HOST:PORT>**

Durch Angabe dieses Parameter können Sie auf ein entferntes PPM-System zugreifen, dessen RMI-Registry unter dem angegebenen Rechnernamen und Portnummer zu erreichen ist.

**-USESSL <TRUE|FALSE|DEFAULT>**

Wenn Sie den Parameter **-registry** angegeben haben, können Sie mittels Parameter **-usessl** bestimmen, ob das entfernte System SSL-verschlüsselte Datenkommunikation verwenden soll oder nicht.

Parameter	Beschreibung
true	Das entfernte PPM-System verwendet verschlüsselte Datenübertragung. Stellen Sie sicher, dass das lokale System den gleichen Keystore wie das entfernte System verwendet.
false	Das entfernte System verwendet unverschlüsselte Datenübertragung.
default (Standardwert)	Es werden die Einstellungen des lokalen PPM-Systems verwendet.

## 13.6.2 Export der Systemkonfiguration

Mit dem Parameter **-system** des Kommandozeilenprogramms **runppmconfig** lässt sich fast die gesamte Konfiguration eines Mandanten mit Ausnahme der Reportautomatisierung auslesen und in XML-Dateien speichern. Dabei wird die Datei **InitSystem\_settings.properties** erzeugt, die die Namen der Konfigurationsdateien mit Pfadangabe enthält. Zum Initialisieren einer PPM-Datenbank kopieren Sie die Datei **InitSystem\_settings.properties** in das Konfigurationsverzeichnis des gewünschten Mandanten und führen anschließend das Kommando **runinitdb** aus.

### Beispiel

Das Ausführen der Kommandozeile

```
runppmconfig -user system -password manager -mode export -system  
<Installationsverzeichnis>\ppmmashzone\server\bin\work\data_ppm\config\default\xml
```

liest die Konfiguration des Mandanten **default** aus und speichert diese im Verzeichnis

```
<Installationsverzeichnis>\ppmmashzone\server\bin\work\data_ppm\config\default\xml
```

in folgenden XML-Dateien:

```
default_abcanalysis.xml  
default_attributenames_de.xml  
default_attributenames_en.xml  
default_attributenames_fr.xml  
default_attributetypes.xml  
default_copyattributerules.xml  
default_datatypes.xml  
default_dimgroup.xml  
default_epctemplate.xml  
default_factorycalendar.xml  
default_favorites.xml  
default_keyindicator.xml  
default_keyrules.xml  
default_kidepend.xml  
default_kigroup.xml  
default_merger.xml  
default_mysaptransactions.xml  
default_onlinekis.xml  
default_orgunits.xml  
default_processtree.xml  
default_ranking.xml  
default_realm_REALMNAME.xml (optional eine Datei für jeden vorhandenen Analyseraum)  
default_targetvalues.xml  
default_topflop.xml  
default_transformationfactors.xml  
default_users.xml  
InitSystem_settings.properties
```

## 13.6.3 Import und Export von Benutzerdaten

Mit Hilfe des Kommandozeilenprogramms **runppmconfig** können Sie PPM-Benutzerdaten importieren und exportieren.

Das Format für den Import und Export von Benutzerdaten wird in der Datei **userconfig.dtd** definiert.

Neue Benutzer und Benutzergruppen können Sie nur in der zentralen Benutzerverwaltung anlegen. In PPM selbst können Sie nur spezifische PPM-Rechte verwalten, z. B. Daten- oder Prozesszugriffsrechte vergeben. Detaillierte Informationen hierzu erhalten Sie in der Online-Hilfe PPM-Hilfethemen.

## IMPORT VON BENUTZERDATEN

Der Programmaufruf sieht wie folgt aus.

```
runppmconfig -user system -password manager -client <name> -mode import -users <filename>
```

Beim Import werden die bereits in PPM vorhandenen Benutzer oder Benutzergruppen nicht neu angelegt. Die für das Kommandozeilenprogramm **runppmconfig** verfügbaren Optionen – **replace**, **-overwrite** und **-add** stehen für den Import von Benutzer nicht zur Verfügung. Es wird eine entsprechende Fehlermeldung ausgegeben.

Benutzer und Benutzergruppen, die in PPM importiert werden aber nicht in der zentralen Benutzerverwaltung vorhanden sind, werden als zu löschend markiert.

Beim Ausführen von runppmconfig werden nur spezifische PPM Benutzerdaten in PPM importiert, wie z. B. Daten- und Prozesszugriffsrechte, sofern sie in PPM noch nicht bekannt sind.

Allgemeine Benutzerdaten, wie z. B. Gruppenzuordnung, Vorname, Nachname, E-Mail-Adresse, werden nicht in PPM gespeichert. Diese Daten können nur in der zentralen Benutzerverwaltung verwaltet werden.

Die Liste der Benutzerattribute umfasst folgende Elemente und Standardeinstellungen:

```
<!ATTLIST user
  login CDATA #REQUIRED
  password CDATA #IMPLIED
  name CDATA #IMPLIED
  lastname CDATA #IMPLIED
  email CDATA #IMPLIED
  isldapuser (yes | no) "no"
  isumcadmin (yes | no) "no"
>
```

Es ist nicht möglich, in der XML-Importdatei zu spezifizieren, ob die Daten nur in PPM oder nur in die zentrale Benutzerverwaltung importiert werden sollen.

Nur Benutzer, die das Funktionsrecht **Konfigurationsimport** und das Recht **Rechteverwaltung** haben, können Benutzerdaten importieren.

## EXPORT VON BENUTZERDATEN

Der Programmaufruf sieht wie folgt aus.

```
runppmconfig -user system -password manager -client <name> -mode export -users <filename>
```

Beim Export werden nur spezifische PPM Benutzerdaten exportiert, wie z. B. Daten- und Prozesszugriffsrechte. Es werden die Daten von aktiven und zu löschenden Benutzern exportiert.

## IM- UND EXPORT VON KENNWÖRTERN

Kennwörter können nur in der zentralen Benutzerverwaltung verwaltet werden.

## 13.6.4 Import und Export von Favoriten

Mit Hilfe des Kommandozeilenprogramms **runppmconfig** können Sie private und gemeinsame Favoriten nach PPM importieren.

### IMPORT VON FAVORITEN

Der Import der Favoriten per Kommandozeile unterstützt die Optionen **Ersetzen** (Parameter **-replace**) und **Überschreiben** (Parameter **-overwrite**). Ohne Angabe von Parametern erfolgt der Import standardmäßig rein additiv. Es werden nur neue Favoriten und Ordner angelegt. Bereits existierende Favoriten und Ordner bleiben unverändert bestehen.

### Voraussetzung

Sie haben das Funktionsrecht **Favoritenadministration**.

### IMPORT UND EXPORT VON GEMEINSAMEN FAVORITEN

Durch Angabe der Option **sharedfavorites** des Parameters **-favoriteuser** wird nur der gemeinsame Favoritenbaum im- bzw. exportiert. Wird der Parameter **-favoriteuser** nicht angegeben, wird der gemeinsame Favoritenbaum wie ein privater Favoritenbaum mit dem Loginnamen zusammen mit den privaten Bäumen exportiert.

Beim Export gemeinsamer Favoriten werden die Zugriffsrechte auf die jeweiligen gemeinsamen Favoritenordnern mit exportiert. Geerbte oder implizite Zugriffsrechte werden nicht exportiert. Zum Export von Favoriten sind keine speziellen Berechtigungen erforderlich.

### IMPORT UND EXPORT VORLAGEFAVORITEN

Der Import und Export von Vorlagefavoriten folgt dem gleichen Mechanismus. Sie können aber Vorlagefavoriten und Favoriten getrennt voneinander (Parameter **-favoritetemplates**) oder zusammen (Parameter **-favoritesall**) im- bzw. exportieren.

```
runppmconfig -user <username> -password <password>
    -mode {import|export}
    -favorites <filename>
    [-client <clientname>]
    [-favoriteuser <ppmuser>]
    [-language <ISO-code>]
    [-overwrite]
    [-replace]
    [protocoloptions]
    [-version]
    [-favoritetemplates]
    [-favoritesall]
```

Liste der spezifischen Parameter des Favoriten-Import und -Export

Parameter	Beschreibung
-favorites <filename>	Dateiname mit Pfadangabe der XML-Datei, in die Favoriten exportiert bzw. aus der Favoriten importiert werden. Angabe obligatorisch
[–favoriteuser <ppmuser>]	Gibt an, von bzw. zu welchem PPM-Benutzer Favoriten im- bzw. exportiert werden. Wird diese Option nicht angegeben, so erfolgt der Im- bzw- Export für alle PPM-Benutzer. Angabe optional
[-overwrite]	Fügt neue Favoriten und Ordner hinzu und bereits existierende Favoriten und Ordner werden geändert, d. h. Inhalte der Favoriten und Zugriffsrechte der gemeinsamen Favoritenordner. Angabe optional
[-replace]	Löscht zuerst alle Ordner und Favoriten und importiert dann die neuen Favoriten und Ordner samt Zugriffsrechte der gemeinsamen Favoritenordner. Angabe optional
[-favoritetemplates]	Exportiert oder importiert nur die Vorlagefavoriten aller Benutzer; Angabe optional
[-favoritesall]	Exportiert oder importiert die Favoriten und Vorlagefavoriten aller Benutzer; Angabe optional

### Beispiele

- Export der Favoriten aller Benutzer  
runppmconfig -user <username> -password <password> -mode export –favorites <filename>
- Export der Favoriten eines bestimmten Benutzers  
runppmconfig -user <username> -password <password> -mode export –favorites <filename> -favoriteuser <user>
- Import von Favoriten für alle Benutzer  
runppmconfig -user <username> -password <password> -mode import –favorites <filename>
- Import von Favoriten für einen bestimmten Benutzer  
runppmconfig -user <username> -password <password> -mode import –favorites <filename> -favoriteuser <user>
- Export aller Vorlagefavoriten des Benutzers system in die Datei TemplatesSystem.xml

```
runppmconfig -user <username> -password <password> -mode export --favoritetemplates  
TemplatesSystem.xml -favoriteuser system
```

- Export aller Vorlagefavoriten aller PPM-Benutzer in die Datei Templates.xml

```
runppmconfig -user <username> -password <password> -mode export --favoritetemplates  
Templates.xml
```

- Export aller Favoriten und Vorlagefavoriten des Benutzers system in die Datei FavoritesAllSystem.xml

```
runppmconfig -user <username> -password <password> -mode export --favoritesall  
FavoritesAllSystem.xml --favoriteuser system
```

## 13.7 runppmcpu

Im Rahmen des **Easy Mining** können sehr viele Nachrichten mit Hinweisen und Warnungen in PPM erzeugt werden. Um bei einer großen Anzahl angefallener Nachrichten (ab ca. 800-1000) eine deutliche Verschlechterung der Systemleistung zu verhindern, können vorhandene Nachrichten mit Hilfe der Benutzeroberfläche oder automatisiert über Kommandozeile gelöscht werden.

Mit Hilfe des Kommandozeilenprogramm **runppmcpu** können automatisch mehrere Nachrichten gleichzeitig gelöscht werden. Die Nachrichten können endgültig gelöscht werden oder erhalten den Status **Zum Löschen vorgesehen**.

Zum Ausführen von **runppmcpu** ist das Funktionsrecht **CPI-Administration** erforderlich.

### Tipp

Weitere Informationen zu Easy Mining erhalten Sie in der Online-Hilfe **PPM-Hilfethemen**.

### 13.7.1 Argumente runppmcpu

#### **-DELETECPIMESSAGES**

Parameter zum Starten des Löschvorgangs von CPI-Nachrichten. Als alleiniger Parameter ohne Auswirkung. Es muss mindestens einer der Parameter **-excluding**, **-messagetype** und **-owner** als Filter übergeben werden.

#### **-MODE <DELETE|MARK\_DELETED>**

Spezifiziert, ob Nachrichten gelöscht werden (Modus **delete**) oder in den Status zum Löschen vorgesehen (Modus **mark\_deleted**) gesetzt werden.

Im Modus **mark\_deleted** werden alle CPI-Nachrichten zum Löschen vorgesehen, ohne sie endgültig zu löschen. Diese Nachrichten können wieder hergestellt werden. Standardwert für **-mode** (ohne Parameter **-mode**) ist **mark\_deleted**.

Zum Starten des Löschvorgangs ist der Parameter **-deletecpimessages** zwingend erforderlich. Die Parameter **-excluding**, **-messagetype** und **-owner** werden UND-Verknüpft und schränken die Menge der zu löschenden Nachrichten ein.

**Beispiel 1****runppmcpic ... -deletecpimessages --excluding 5w --mode mark\_deleted**

Es werden alle CPI-Nachrichten in den Status **to\_delete** gesetzt, die älter als 5 Wochen sind.

**Beispiel 2****runppmcpic ... -deletecpimessages --excluding 5w --mode delete**

Es werden alle CPI-Nachrichten gelöscht, die älter als 5 Wochen sind.

**-EXCLUDING <N>[Y|Q|M|W|D]**

Ausschließender Filter mit aktuellem Bezugsdatum: y=Jahr, q=Quartal, m=Monat, w=Woche, d=Tag. "n" muss ganzzahlig sein.

Alle CPI-Nachrichten, die älter sind als der ausschließende Filter, werden endgültig gelöscht/in den Status **to\_delete** gesetzt.

**Beispiel****runppmcpic ... -deletecpimessages --excluding 5w --mode delete**

Es werden alle CPI-Nachrichten gelöscht, die älter als 5 Wochen sind.

**-MESSAGE TYPE <SYSTEM\_MESSAGE, ACTIVITY, COMMENT, OUTLIER\_MESSAGE, EARLYALERT\_MESSAGE, ALARM\_MESSAGE, TARGET\_MESSAGE, PROCESSMINING\_MESSAGE>**

Der Parameter filtert nach dem Typ der CPI-Nachricht. Alle CPI-Nachrichten des übergebenen **messagetype** werden gelöscht. Es können mehrere **messagetypes** übergeben werden.

**Beispiel 1****runppmcpic ... -deletecpimessages -messagetype comment --mode delete**

Es werden alle Kommentare gelöscht.

**Beispiel 2****runppmcpic ... -deletecpimessages -messagetype comment earlyalert\_message**

Es werden alle Kommentare UND Early Alert-Messages in den Status **to\_delete** gesetzt.

**-OWNER <OWNER>**

Die CPI-Nachrichten des durch **owner** angegebenen Benutzer werden gelöscht.

**Beispiel****runppmcpic ... -deletecpimessages --owner mueller -messagetype comment --mode delete**

Es werden alle Kommentare mit owner=mueller gelöscht.

**-REGISTRY <HOST:PORT>**

Durch Angabe dieses Parameter können Sie auf ein entferntes PPM-System zugreifen, dessen RMI-Registry unter dem angegebenen Rechnernamen und Portnummer zu erreichen ist.



**-USESSL <TRUE|FALSE|DEFAULT>**

Wenn Sie den Parameter **-registry** angegeben haben, können Sie mittels Parameter **-usessl** bestimmen, ob das entfernte System SSL-verschlüsselte Datenkommunikation verwenden soll oder nicht.

Parameter	Beschreibung
true	Das entfernte PPM-System verwendet verschlüsselte Datenübertragung. Stellen Sie sicher, dass das lokale System den gleichen Keystore wie das entfernte System verwendet.
false	Das entfernte System verwendet unverschlüsselte Datenübertragung.
default (Standardwert)	Es werden die Einstellungen des lokalen PPM-Systems verwendet.

- Nach dem Aufrufen von `runppmcp` wird die Anzahl der gelöschten CPI-Nachrichten ausgegeben.
- Nach dem Aufrufen von `runppmcp` wird der Eigentümer der CPI-Nachricht (`CURRENT_OWNER`) auf den im Argument **-user** angegebenen Benutzer gesetzt.
- Der Parameter **-excluding** kann max. taggenau angewendet werden (z. B. **-excluding 1d**). Eine Angabe von Stunden (z. B. **-excluding 1h**) ist nicht möglich.

**MASTER-SUB-SERVER SYSTEM**

CPI-Nachrichten können in einem skalierten System auf dem Master- und den Subserver erstellt werden. Beim Löschen von Nachrichten mittels `runppmcp` werden Master- und Subserver wie Standalone-Server betrachtet. Es findet kein Löschen statt.

**13.8 runppmimport**

Das Kommandozeilenprogramm `runppmimport` startet die Weiterverarbeitung der vom XML-Import eingelesenen Prozessinstanzfragmente in das PPM-System. Für jede Prozessinstanz werden nacheinander EPK-Import, Prozesszusammenführung, Typisierung und Kennzahlenberechnung ausgeführt.

**13.8.1 Argumente runppmimport****-MERGER ANONYMIZING**

Alle Prozessinstanzen der PPM-Datenbank, die noch nicht anonymisiert wurden, werden anonymisiert.

Normalerweise werden beim Import von Prozessinstanzfragmenten nur neu hinzukommende Prozessinstanzfragmente anonymisiert. Bereits vorhandene Instanzen, die vom Merger nicht bearbeitet werden, werden nicht anonymisiert.

Es ist nicht möglich, die Anonymisierung bereits anonymisierter Prozessinstanzen zu ändern, da die Informationen über die realen Bearbeiter beim Anonymisieren überschrieben werden.

### **-TYPIFIER NEW**

Alle Prozessinstanzen der PPM-Datenbank (neu hinzukommende und bereits typisierte) werden mit der angegebenen Prozesstypdefinition neu typisiert.

Die erneute Typisierung der Prozessinstanzen bedingt auch das erneute Berechnen der Kennzahlen und Dimensionen, da diese prozesstypabhängig sind.

### **-KEYINDICATOR NEW**

Die Kennzahlen und Dimensionen aller Prozessinstanzen werden neu berechnet. Verwenden Sie diese Option, wenn Sie eine neue Konfiguration des Kennzahlenberechners importiert haben (siehe auch Kapitel **Neuberechnung** (Seite 65)).

### **-REINITANALYSISSEVER {YES|NO|AUTO}**

Mit **-reinitanalysisserver yes** geben Sie an, dass der Analyseserver nach erfolgreicher Beendigung des PPM-Imports neu initialisiert werden soll. Wenn Sie das Argument **auto** verwenden, wird der Analyseserver nur falls notwendig neu initialisiert. Mit der Standardeinstellung **no** (default) verhindern Sie generell die Neuinitialisierung. Die Option gilt für Prozessinstanzdaten, prozessinstanzunabhängige Daten und Data-Analytics-Daten.

Wenn bei der Reinitialisierung des Analyseservers auch die Wiederherstellungsdateien neu erzeugt werden (`runppmimport --reinitanalysisserver [yes|auto] --recoveryfile yes`), werden alle Wiederherstellungsdateien erzeugt.

Details zur Reinitialisierung von Data-Analytics-Analyseräumen des Analyseservers erhalten Sie in der technischen Dokumentation **PPM Data Analytics**.

### **-PS <PARAMSET>**

Gibt die XML-Datei an, die das Paramset zur Einschränkung der Menge der Prozessinstanzen enthält, auf die eine der Optionen **-typifier new** oder **-keyindicator new** angewendet werden soll (siehe auch Kapitel **Neuberechnung** (Seite 65)). Geben Sie ggf. den kompletten Pfad zur Datei an.

Auch bei Angabe eines Paramsets wird die Einbeziehung der Fehlerinstanzen über den Parameter **-errorinstances** gesteuert und muss gegebenenfalls explizit ausgeschaltet werden.

### **-GENSTATS {BEFORE|AFTER|KEYINDICATOR|MERGER|DELETE}**

Aktualisiert bzw. löscht die Datenbankstatistiken vor Überprüfung der Planwerte, **before** führt die Funktion am Beginn, **after** am Ende, **merger** vor jedem Aufruf des Merger's und **keyindicator** vor jeder Kennzahlberechnung aus. Mit **delete** löschen Sie Datenbankstatistiken beim Start des PPM-Imports. Die Parameterwerte können kombiniert werden. Nur Oracle unterstützt das Erzeugen von Statistiken als Standard-Datenbankbenutzer. Für alle anderen

DB-Systeme ist das Berechnen von Statistiken eine administrative Aufgabe, die mit DB-Administratorrechten ausgeführt werden muss.

### -ERRORINSTANCES {YES|NO}

Mit **-errorinstances yes** geben Sie an, dass alle Fehlerinstanzen im Rahmen des PPM-Imports erneut berechnet werden. Vorgabewert: **yes**.

#### Beispiel

In der Analysekomponente wurde für den Prozesstyp Barverkauf der Prozesstypgruppe Auftragsabwicklung die Kennzahl Prozessanzahl über die Zeit mit Filtern auf den Dimensionen Material/MASC (Maschinen) und Zeit (2. Quartal 2002) dargestellt. Das entsprechende Paramset wurde in der XML-Datei paramset.xml gespeichert.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE paramset SYSTEM "paramset.dtd">
<paramset>
<analysewizparam selectedrow="0"/>
  <chartparam>
    <chart-chartposition>0.16388889,0.20876624
    </chart-chartposition>
    <chart-legend-visible>>false</chart-legend-visible>
    <chart-zoom>75</chart-zoom>
    <chart-scale>1.0,1.0,1.0</chart-scale>
    <chart-colinfo/>
  </chartparam>
  <cpiparam/>
  <epkparam/>
  <favoritparam usecache="yes"/>
  <kiquery usecache="yes" showzero="auto">
    <keyindicator>
      <criterion name="PNUM">Prozessanzahl</criterion>
    </keyindicator>
    <iteration>
      <criterion name="PROCESSTYPE">Prozesstyp</criterion>
      <refinement name="BY_LEVEL2">Fein</refinement>
    </iteration>
    <iteration>
      <criterion name="TIME">Zeit</criterion>
      <refinement name="BY_MONTH">Monatsweise</refinement>
    </iteration>
    <iteration>
      <criterion name="MATERIAL">Material</criterion>
      <refinement name="BY_LEVEL2">Fein</refinement>
    </iteration>
    <filter>
      <criterion name="PROCESSTYPE">Prozesstyp</criterion>
      <filteritem operator="or">
        <datalist usewildcard="no">
          Auftragsabwicklung
          <dataitem>
            Auftragsabwicklung\Barverkauf
            <datatype name="TEXTPAIR">Textpaar</datatype>
            <scale name="LEVEL1SCALE">Grob</scale>
          </dataitem>
        </datalist>
      </filteritem>
    </filter>
  </filter>
</filter>
```

```

<criteria name="TIME">Zeit</criteria>
<filteritem operator="or">
  <datarange operator="range">
    2. Quartal 02
    <dataitem value="2.QUARTER.2002">
      2. Quartal 02
      <datatype name="TIME">Zeit</datatype>
      <scale name="QUARTERSCALE" factor="7776000.0">
        Quartal(e)
      </scale>
    </dataitem>
    <dataitem value="2.QUARTER.2002">
      2. Quartal 02
      <datatype name="TIME">Zeit</datatype>
      <scale name="QUARTERSCALE" factor="7776000.0">
        Quartal(e)
      </scale>
    </dataitem>
  </datarange>
</filteritem>
</filter>
<filter>
  <criteria name="MATERIAL">Material</criteria>
  <filteritem operator="or">
    <datalist usewildcard="no">
      Maschinen (MASC)
      <dataitem>
        MASC{Maschinen}
        <datatype name="TEXTPAIR">Textpaar</datatype>
        <scale name="LEVEL2SCALE" factor="2.0">
          Fein
        </scale>
      </dataitem>
    </datalist>
  </filteritem>
</filter>
</kiquery>
<navparam>
  <nav-view>chart</nav-view>
</navparam>
<tableparam view="listtable">
  <tbl-cross totalresult="yes"/>
  <tbl-instance split="horizontal"/>
</tableparam>
</paramset>

```

Durch Ausführen der folgenden Kommandozeile werden ausschließlich die Prozessinstanzen neu typisiert und berechnet, die den Filterangaben im oben stehenden Paramset entsprechen:

```
runppmimport -user system -password manager -typifier new -ps paramset.xml
```

### **-RECOVERYFILE {YES|NO}**

Mit **-recoveryfile yes** (Standardeinstellung) geben Sie an, dass nach erfolgreichem Abschluss des PPM-Imports die für die Prozessanalyse relevanten Wiederherstellungsdateien des Analyseservers unter

<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\recovery\<client>  
erstellt werden sollen.

## 13.8.2 Neuberechnung

### Globale Neuberechnung

Wenn Sie Änderungen an der Kennzahlenkonfiguration oder Typisierungsregeln vornehmen und den vorhandenen Datenbestand aktualisieren möchten, müssen Sie folgende Schritte durchführen:

1. Retypisierung (Argument **-typifier new**) oder eine Neuberechnung (Argument **-keyindicator new**) durchführen.
2. Reinitialisierung des Analyseservers (Argument **-reinitanalysisserver auto**), bei der die Speicherstrukturen des Analyseserver komplett neu aufgebaut werden.

Sie können die beiden Schritte in einem gemeinsamen Aufruf **runppmimport** kombinieren oder unabhängig voneinander ausführen. Die neu berechneten Daten werden in der Analyse jedoch erst dann angezeigt, wenn beide Schritte erfolgreich abgeschlossen wurden.

Eine erneute Typisierung impliziert auch eine Neuberechnung der Kennzahlen, da durch die Änderung der Zugehörigkeit einer Prozessinstanz zu bestimmten Prozessstypen und Prozessgruppen i.d.R. auch neue prozesstypspezifische Kennzahlen berechnet werden müssen.

### Partielle Neuberechnung

Sie können die neu zu berechnende Menge der Prozessinstanzen einschränken, in dem Sie mittels Argument **-ps** ein Paramset angeben, das die neu zu berechnenden Prozessinstanzen bestimmt. In diesem Fall wird der Analyseserver automatisch aktualisiert und die Reinitialisierung ist nicht erforderlich.

## 13.9 runprocessimport

Mit Hilfe des Kommandozeilenprogramms **runprocessimport** können Sie den Prozessimport starten. Der Prozessimport ermöglicht eine zeitnahe Analyse der importierten Prozesse in PPM.

Das Programm liegt im Verzeichnis

**<PPM-Installationsverzeichnis>\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-<version>-runnable.zip\ppm\bin.**

Führen Sie das Programm mit den folgenden Parametern in der Kommandozeile aus. Die Parameter in den eckigen Klammern **[]** sind optional.

```
runprocessimport -user <username> -password <password>
    (-datasource <file>) | (-datasourcelist <file>) | (-recoverimport)
    [-client <name>]
    [-language <ISO-code>]
    [-recoveryfile {yes|no}]
    [-version]
    [protocoloptions]
```

Der Prozessimport ist eine alternative Importmethode zum standardmäßigen XML-Import (runxmlimport) und PPM-Import (runppmimport (Seite 61)) von prozessabhängigen Daten nach

PPM. Der Prozessimport ermöglicht den Import fast aller prozessbezogenen Datenquellen außer Daten im Format **GRAPH**. Es können Daten in den Formaten **EVENT**, **JDBC**, **CSV**, **SAP** und **Universal Messaging** importiert werden. Der Prozessimport umfasst die meisten Funktionen des PPM-Imports und XML-Imports, z. B. Event-Import, Schlüsselerzeugung, Merge, Attribut-, Kennzahl- und Dimensionsberechnung bezüglich Prozesshierarchien usw.

Folgende Funktionen des PPM-Imports und XML-Imports werden vom Prozessimport nicht unterstützt.

- Graph-Import
- Import vollständiger EPKs
- Shared-fragments-Import
- Anonymisierung (wie der Modus **anonymizing** des PPM-Imports)
- Neutypifizierung (wie der Modus **typifier new** des PPM-Imports)
- Neuberechnung (wie der Modus **keyindicator new** des PPM-Imports)
- Neuinitialisierung des Analyseservers (wie der Modus **reinitanalysisserver** des PPM-Imports)
- Autoattribute von Laufzeitdaten (Neue Attribute müssen zur Konfigurationszeit erstellt werden.)

Das Kommandozeilenprogramm startet mit eigenen VM-Einstellungen, die in der Datei **\_vmparam.bat** spezifiziert sind. Die Datei liegt im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\config\<Mandant>\bin**. Die Werte können mit Hilfe von CTK konfiguriert werden.

Standardwerte sind:

- LOCAL\_PPM5\_VM\_MIN\_MEM\_RT\_PROCESSIMPORT=1024
- LOCAL\_PPM5\_VM\_MAX\_MEM\_RT\_PROCESSIMPORT=4096

### Beispiel

```
runprocessimport -client e2ppm -user system -password manager -datasource  
C:\SoftwareAG\ppmmashzone\server\bin\work\data_ppm\custom\e2ppm\xml\datasource\data  
source.xml
```

## 13.9.1 Argumente runprocessimport

**-DATASOURCE <FILE> | -RECOVERIMPORT | -DATASOURCELIST <FILE>**

Die Parameter **datasource**, **datasourcelist** und **recoverimport** sind alternative Importmodi. Der Parameter **datasource** spezifiziert die Konfigurationsdatei der Datenquelle für den Prozessimport. Mit dem Parameter **datasource** wird ein neuer Prozessimport gestartet. Mit dem Parameter **recoverimport** wird der Prozessimport im Wiederherstellungsmodus gestartet und vervollständigt einen zuvor abgebrochenen Importvorgang.

Mit Hilfe des Arguments **-datasourcelist** können mehrere Datenquellen auf einmal importiert werden. Der Import entspricht dem mehrfachen, nacheinander Ausführen des Imports mittels des Arguments **-datasource**.

Siehe dazu Kapitel Gleichzeitiger Import mehrere Datenquellen (Seite 68)

### **-RECOVERYFILE {YES|NO}**

Mit **-recoveryfile yes** (Standardeinstellung) geben Sie an, dass nach erfolgreichem Abschluss des PPM-Imports die für die Prozessanalyse erforderlichen Wiederherstellungsdateien des Analyseservers unter

<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\recovery\**<Mandant>** erstellt werden sollen.

## **13.9.2 Importeinstellungen**

Mit Hilfe verschiedener Einstellungen können Sie das Verhalten des Prozessimports beeinflussen. Die Einstellungen nehmen Sie in der Datei **ProcessImport\_settings.properties** im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\config\**<Mandant>**** vor.

Die Datei enthält Einstellungen zur Konfiguration von PPM-Server, Analyseserver und dem Datenspeicher (Arbeitsspeicher und Festplatte), der zum Speichern von Datenobjekten (EPKs, Schlüssel, IDs usw.) während des Importvorgangs verwendet wird.

Die verschiedenen Komponenten des Prozessimports verwenden zum Datenaustausch mehrere Datenspeicher. Das sind Cache-Instanzen mit andauernder Lebenszeit der Datenobjekte und der Fähigkeit Elemente auf die Festplatte auszulagern, wenn nicht mehr genug Hauptspeicher vorhanden ist. Ein Benutzer kann die Größe des verfügbaren Arbeitsspeichers (**MAX\_BYTES\_LOCAL\_HEAP**) und Festplattenspeichers (**MAX\_BYTES\_LOCAL\_DISK**) spezifizieren. Er kann auch den Ort (**DISK\_STORE\_PATH**) angeben, wo die temporären Daten gespeichert werden sollen.

In der folgenden Tabelle sind die einzelnen Variablen beschrieben.

Key	Beispiel	Beschreibung
MAX_BYTES_LOCAL_HEAP	3072	Maximale Größe des verfügbaren Arbeitsspeichers in MB
MAX_BYTES_LOCAL_DISK	32768	Maximale Größe des verfügbaren Festplattenspeichers in MB
DISK_STORE_PATH	<Installationsverzeichnis>\ppmmashzone\server\bin\work\data_ppm\diskstore	Pfad zum temporären Datenspeicher

### 13.9.3 Import-Logging

Für den Prozessimport stehen vier Log-Module zur Verfügung, ein allgemeines für den Prozessimport und für weiter drei Komponenten.

Die Log-Level aller Module können in der Datei **Server\_Log\_settings** eingestellt werden. Die Log-Ausgaben der Module werden in die entsprechende Import-Log-Datei geschrieben.

#### Log-Module

```
#MODULE_PPM_PROCESS_IMPORT
log4j.logger.LOG.PPI=INFO
#MODULE_PPM_PROCESS_IMPORT_EVENT_IMPORT
log4j.logger.LOG.EIP=INFO
#MODULE_PPM_PROCESS_IMPORT_CALCULATION
log4j.logger.LOG.CLC=INFO
#MODULE_PPM_PROCESS_IMPORT_UPDATE
log4j.logger.LOG.UPD=INFO
```

#### Trace-Module

```
#TRACEMODULE_PPM_PROCESS_IMPORT
log4j.logger.TRACE.PPI=OFF
#TRACEMODULE_PPM_PROCESS_IMPORT_EVENT_IMPORT
log4j.logger.TRACE.EIP=OFF
#TRACEMODULE_PPM_PROCESS_IMPORT_CALCULATION
log4j.logger.TRACE.CLC=OFF
#TRACEMODULE_PPM_PROCESS_IMPORT_UPDATE
log4j.logger.TRACE.UPD=OFF
#TRACEMODULE_PPM_PROCESS_IMPORT_DATASTORE
log4j.logger.TRACE.DST=OFF
#TRACEMODULE_PPM_PROCESS_IMPORT_QUEUE
log4j.logger.TRACE.QUE=OFF
```

### 13.9.4 Import mehrere Datenquellen

Mit Hilfe des Arguments **-datasourcelist <datasource>** können mehrere Datenquellen auf einmal importiert werden (siehe Kapitel Argumente runprocessimport (Seite 66)). Zum Import mehrere Datenquellen steht eine Konfigurationsdatei zur Verfügung, in der eine Liste mit Datenquellen angegeben werden kann. Beim Prozessimport werden die Daten, der in der Konfigurationsdatei angegebenen Datenquellen nacheinander importiert, als würde der Prozessimport mehrmals mit dem Argument **-datasource <datasource>** hintereinander aufgerufen. Die Reihenfolge des Imports der Datenquellen wird in der Konfigurationsdatei angegeben.

Das Einlesen und der Import von Datenquellen vom Typ **GRAPH** werden beim Prozessimport mittels Argument **-datasourcelist** nicht unterstützt.

Die Konfigurationsdatei muss der DTD **datasourcelist.dtd** entsprechen, die folgendermaßen aussieht.

```
<!ELEMENT datasourcelist (datasource*)>
<!ELEMENT datasource (#PCDATA)>
<!ATTLIST datasource
    name ID #REQUIRED
    type (EVENT | GRAPH | MYSAP | JDBC | CSV | NIRVANA ) #REQUIRED
>
```



Zu jeder Datenquelle muss eine ID, d. h. im Normalfall der Name der Datenquelle, der auch in CTK verwendet wird, der Datenquellentyp und der Pfad zu der Datenquellendatei angegeben werden.

Eine XML-Datei kann dann zum Beispiel folgendermaßen aussehen.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE datasourcelist SYSTEM "datasourcelist.dtd">
<datasourcelist>
  <datasource name="CLEARING"
type="EVENT">M:/SoftwareAG/ppmmashzone/server/bin/work/data_ppm/custom/umg_en/xml/CLEARING/CLEARING.xml</datasource>
  <datasource name="ACCOUNTING" type="GRAPH">
M:/SoftwareAG/ppmmashzone/server/bin/work/data_ppm/custom/umg_en/xml/ACCOUNTING/ACCOUNTING.xml</datasource>
  <datasource name="BILLING" type="MYSAP">
M:/SoftwareAG/ppmmashzone/server/bin/work/data_ppm/custom/umg_en/xml/BILLING/BILLING.xml</datasource>
  <datasource name="SHIPMENT" type="JDBC">
M:/SoftwareAG/ppmmashzone/server/bin/work/data_ppm/custom/umg_en/xml/SHIPMENT/SHIPMENT.xml</datasource>
  <datasource name="MATERIAL_DOCUMENT"
type="CSV">M:/SoftwareAG/ppmmashzone/server/bin/work/data_ppm/custom/umg_en/xml/MATERIAL_DOCUMENT/MATERIAL_DOCUMENT.xml</datasource>
  <datasource name="PURCHASE_PROCESS" type="NIRVANA">
M:/SoftwareAG/ppmmashzone/server/bin/work/data_ppm/custom/umg_en/xml/PURCHASE_PROCESS/PURCHASE_PROCESS.xml</datasource>
</datasourcelist>
```

## FEHLERVERHALTEN

Wird der Import mit einer gültigen Konfigurationsdatei aufgerufen, die aber keine Datenquellen enthält, so beendet er sich ohne eine Fehlermeldung.

Wird der Import mit einer Konfigurationsdatei aufgerufen, die mehrere Datenquellen enthält und tritt beim Import einer Datenquelle ein Fehler auf, führt das zum Komplettabbruch des Imports.

Tritt beim Import mindestens einer Datenquelle aus einer Konfigurationsdatei ein Fehler auf, der bislang, d. h. beim Import der einzelnen Datenquellen, zu einem Exit-Fehlerstatus (d. h. „-1“) geführt hat, so liefert auch der Import mit der Konfigurationsdatei diesen Exit-Fehlerstatus zurück.

## 13.10 runppmcompress

Das Verdichten von Prozessinstanzen lässt sich mit dem Kommandozeilenprogramm **runppmcompress** automatisieren. Welche Prozessinstanzen verdichtet werden, wird durch ein Paramset bestimmt, das bei Ausführung des Programms als XML-Datei übergeben wird. Dieses Paramset können Sie sich bequem mit der Analysekomponente der PPM-Benutzeroberfläche zusammenstellen und in einer XML-Datei speichern.

Beachten Sie beim Erstellen des Paramset folgende Punkte:

- Die XML-Datei darf nur ein Paramset (XML-Element **paramset**) enthalten.
- Das Paramset darf nur die Abfrage der Kennzahl Prozessanzahl (PNUM) enthalten (XML-Element **kiquery**).
- Es dürfen keine Filter oder Iterationen auf Funktionsdimensionen enthalten sein.

- Es dürfen keine benutzerdefinierten Schrittweiten und keine Benutzerdefinierten Dimensionen enthalten sein.
- Es muss entweder ein Zeitfilter enthalten sein oder die Option **-excluding** angegeben werden.
- Es muss die Dimension **Prozesstyp** enthalten sein. Die Schrittweite wird automatisch auf **2. Stufe (fein)** gesetzt.
- Es können weitere, beliebige Filter zum Einschränken der zu verdichtenden Prozessinstanzmenge definiert werden.

Ist einer der genannten Punkte nicht erfüllt, bricht das Programm mit einer Meldung ab.

### 13.10.1 Argumente `runppmcompress`

#### **-PS <DATEINAME>**

Mit diesem Argument geben Sie die XML-Datei an, die das Paramset enthält.

#### **-MODE <TEST|EXECUTE>**

**test:** Dieses Argument gibt Ihnen die Möglichkeit, in einem Testlauf anhand der ausgegebenen Statistik festzustellen, welche und wie viele Prozessinstanzen verdichtet würden. Hierbei werden keine Änderungen an der Datenbank vorgenommen.

**execute:** Die Prozessinstanzen werden verdichtet. Dieser Vorgang kann nicht rückgängig gemacht werden. Es wird eine Statistik ausgegeben.

Vorgabewert: **test**

#### **-EXCLUDING <N>[Y|Q|M|W|D|H]**

Die in dem angegebenen Zeitraum liegenden Prozessinstanzen werden nicht verdichtet, auch wenn sie im Filter enthalten sind. Der Zeitraum wird als Anzahl bestimmter Zeitabschnitte angegeben. Als Bezugszeitpunkt wird der Startzeitpunkt des Programms verwendet.

Kürzel	Zeitabschnitt	Beispiel
y	Jahr	<b>1y:</b> Prozessinstanzen vom 1. Januar des laufenden Jahres bis zum Startzeitpunkt des Kommandozeilenprogramms werden nicht verdichtet.
q	Quartal	<b>2q:</b> Prozessinstanzen des laufenden und abgelaufenen Quartals werden nicht verdichtet.
m	Monat	
w	Woche	<b>4w:</b> Prozessinstanzen der laufenden und der drei vergangenen Wochen werden nicht verdichtet.

Kürzel	Zeitabschnitt	Beispiel
d	Tag	
h	Stunde	<b>24h:</b> Prozessinstanzen der aktuellen und der 23 vergangenen Stunden werden nicht verdichtet.

### Beispiel

In der Analysekomponente wurde für den Prozesstyp **sonstige Aufträge** die Kennzahl **Prozessanzahl** über die Zeit mit dem Filter **1.1.2003** bis **15.2.2003** wochenweise skaliert dargestellt. Das entsprechende Paramset wurde in der XML-Datei **paramset.xml** gespeichert:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE paramset SYSTEM "paramset.dtd">
<paramset>
  <analysewizparam selectedrow="0" />
  <chartparam>
    <chart-legend-visible>false</chart-legend-visible>
    <chart-scale>1.0,1.0,1.0</chart-scale>
    <chart-zoom>75</chart-zoom>
    <chart-chartposition>0.175,0.1</chart-chartposition>
    <chart-colinfo />
  </chartparam>
  <cpiparam />
  <epkparam />
  <favoritparam />
  <kiquery>
    <keyindicator>
      <criterion name="PNUM">Prozessanzahl</criterion>
    </keyindicator>
    <iteration>
      <criterion name="TIME">Zeit</criterion>
      <refinement name="BY_WEEK">Wochenweise</refinement>
    </iteration>
    <filter>
      <criterion name="TIME">Zeit</criterion>
      <filteritem operator="or">
        <datarange operator="range">
          1. Jan 03 - 15. Feb 03
          <dataitem value="1.1.2003 0:00">
            01.01.03 0:00
            <datatype name="TIME">Zeit</datatype>
            <scale name="MINUTESCALE" factor="60.0">
              Minute(n)
            </scale>
          </dataitem>
          <dataitem value="15.2.2003 23:59">
            15.02.03 23:59
            <datatype name="TIME">Zeit</datatype>
            <scale name="MINUTESCALE" factor="60.0">
              Minute(n)
            </scale>
          </dataitem>
        </datarange>
      </filteritem>
    </filter>
  </filter>
  <criterion name="PROCESSTYPE">Prozesstyp</criterion>
```

```
<filteritem operator="or">
  <dataitem>
    Auftragsabwicklung\sonstige Aufträge
    <datatype name="TEXTPAIR">Textpaar</datatype>
    <scale name="LEVEL2SCALE" factor="2.0">
      Fein
    </scale>
  </dataitem>
</filteritem>
</filter>
</kiquery>
<navparam>
  <nav-view>chart</nav-view>
</navparam>
<tableparam view="listtable">
  <tbl-cross totalresult="yes" />
  <tbl-instance split="horizontal" />
</tableparam>
</paramset>
```

Durch Ausführen der Kommandozeile

```
runppmcompress -user system -password manager -ps paramset.xml -mode execute
```

werden die Prozessinstanzen im Zeitraum vom 1.1.2003 bis 15.2.2003 wochenweise verdichtet.

### **-RECOVERYFILE {YES|NO}**

Mit **-recoveryfile yes** (Standardeinstellung) geben Sie an, dass nach erfolgreichem Abschluss der Verdichtung die für die Prozessanalyse relevanten Wiederherstellungsdateien des Analyseservers unter

<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\recovery\<Mandant> erstellt werden sollen.

## **13.11 runppmdelete**

Mit diesem Kommandozeilenprogramm können Prozessinstanzen testweise oder tatsächlich gelöscht werden. Welche Prozessinstanzen gelöscht werden, wird durch ein Paramset bestimmt, das bei Ausführung des Programms als XML-Datei übergeben wird. Sie können dieses Paramset mit der Analysekomponente der PPM-Benutzeroberfläche zusammenstellen und in einer XML-Datei speichern.

Beachten Sie beim Erstellen des Paramset folgende Punkte:

- Die XML-Datei darf nur ein Paramset (XML-Element **paramset**) enthalten.
- Das Paramset darf nur die Abfrage der Kennzahl **Prozessanzahl** (PNUM) enthalten (XML-Element **kiquery**).
- Es dürfen keine Filter oder Iterationen auf Funktionsdimensionen enthalten sein.
- Es dürfen keine benutzerdefinierten Schrittweiten enthalten sein.
- Es muss entweder ein Zeitfilter enthalten sein oder die Option **-excluding** angegeben werden.
- Es muss ein Prozesstypfilter enthalten sein. Wenn die Iteration über **Prozesstyp** nicht angegeben ist, wird sie automatisch hinzugefügt.

- Es können weitere, beliebige Filter zum Einschränken der zu löschenden Prozessinstanzmenge definiert werden.

Ist einer der genannten Punkte nicht erfüllt, bricht das Programm mit einer Meldung ab.

### 13.11.1 Argumente `runppmdelete`

#### **-PS <DATEINAME>**

Mit diesem Argument geben Sie die XML-Datei an, die das Paramset enthält.

#### **-MODE <TEST|EXECUTE>**

**test:** Dieses Argument gibt Ihnen die Möglichkeit, in einem Testlauf anhand der ausgegebenen Statistik festzustellen, welche und wie viele Prozessinstanzen gelöscht würden. Hierbei werden keine Änderungen an der Datenbank vorgenommen.

**execute:** Die Prozessinstanzen werden gelöscht. Dieser Vorgang kann nicht rückgängig gemacht werden. Es wird eine Statistik ausgegeben.

Vorgabewert: **test**

#### **-EXCLUDING <N>[Y|Q|M|W|D|H]**

Die in dem angegebenen Zeitraum liegenden Prozessinstanzen werden nicht gelöscht, auch wenn sie im Filter enthalten sind. Der Zeitraum wird als Anzahl bestimmter Zeitabschnitte angegeben. Als Bezugszeitpunkt wird der Startzeitpunkt des Kommandozeilenprogramms verwendet.

Kürzel	Zeitabschnitt	Beispiel
y	Jahr	<b>1y:</b> Prozessinstanzen vom 1. Januar des laufenden Jahres bis zum Startzeitpunkt des Kommandozeilenprogramms werden nicht gelöscht.
q	Quartal	<b>2q:</b> Prozessinstanzen des laufenden und abgelaufenen Quartals werden nicht gelöscht.
m	Monat	-
w	Woche	<b>4w:</b> Prozessinstanzen der laufenden und der drei vergangenen Wochen werden nicht gelöscht.
d	Tag	-
h	Stunde	<b>24h:</b> Prozessinstanzen der aktuellen und der 23 vergangenen Stunden werden nicht gelöscht.

## Beispiel

In der Analysekomponente wurde für den Prozesstyp **sonstige Aufträge** die Kennzahl **Prozessanzahl** über die Zeit mit einem Filter auf der Dimension **Auftraggeber/Österreich** dargestellt. Das entsprechende Paramset wurde in der XML-Datei **paramset.xml** gespeichert:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE paramset SYSTEM "paramset.dtd">
<paramset>
  <analysewizparam selectedrow="0"/>
  <chartparam>
    <chart-chartposition>0.175,0.1</chart-chartposition>
    <chart-legend-visible>>false</chart-legend-visible>
    <chart-zoom>75</chart-zoom>
    <chart-scale>1.0,1.0,1.0</chart-scale>
    <chart-colinfo/>
  </chartparam>
  <cpiparam/>
  <epkparam/>
  <favoritparam usecache="yes"/>
  <kiquery usecache="yes" showzero="auto">
    <keyindicator>
      <criterion name="PNUM">Prozessanzahl</criterion>
    </keyindicator>
    <iteration>
      <criterion name="TIME">Zeit</criterion>
      <refinement name="BY_MONTH">Monatsweise</refinement>
    </iteration>
    <filter>
      <criterion name="PROCESSTYPE">Prozesstyp</criterion>
      <filteritem operator="or">
        <datalist usewildcard="no">
          Auftragsabwicklung
          <dataitem>
            Auftragsabwicklung\sonstige Aufträge
            <datatype name="TEXTPAIR">Textpaar</datatype>
            <scale name="LEVEL1SCALE">Grob</scale>
          </dataitem>
        </datalist>
      </filteritem>
    </filter>
    <filter>
      <criterion name="PRINCIPAL">Auftraggeber</criterion>
      <filteritem operator="or">
        <datalist usewildcard="no">
          ÖSTERREICH (A)
          <dataitem>
            A{ÖSTERREICH}
            <datatype name="TEXTPAIR">Textpaar</datatype>
            <scale name="LEVEL2SCALE" factor="2.0">
              Fein
            </scale>
          </dataitem>
        </datalist>
      </filteritem>
    </filter>
  </kiquery>
  <navparam>
    <nav-view>chart</nav-view>
  </navparam>
  <tableparam view="listtable">
```

```
<tbl-cross totalresult="yes"/>
<tbl-instance split="horizontal"/>
</tableparam>
</paramset>
```

Durch Ausführen der folgenden Kommandozeile werden alle Prozessinstanzen mit **Auftraggeber Österreich** gelöscht:

```
runppmdelete -user system -password manager -ps paramset.xml -mode execute
```

### **-RECOVERYFILE {YES|NO}**

Mit **-recoveryfile yes** (Standardeinstellung) geben Sie an, dass nach erfolgreichem Abschluss des Löschens die für die Prozessanalyse relevanten Wiederherstellungsdateien des Analyseservers unter

<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\recovery\**<Mandant>** erstellt werden sollen.

## **13.12 runppmadmin**

Alle Funktionalitäten außer der Hilfe sind nur dem Systemadministrator (PPM-Benutzer **system**) zugänglich und stehen somit für alle Anwender ohne entsprechende Berechtigung nicht zur Verfügung.

Nach Aufruf des Kommandozeilenprogramms erfolgt zunächst eine Authentifizierung des Systemadministrators. Nach erfolgreicher Authentifizierung werden die restlichen angegebenen Parameter ausgewertet. Die einzelnen Aktionen werden mit einer Statusmeldung auf der Konsole abgeschlossen.

### **13.12.1 Argumente runppmadmin**

#### **-KILLSESSION <SESSIONID>**

Mit diesem Argument melden Sie die Sitzung (Session) eines angemeldeten Benutzers ab. Es können nur Benutzeroberflächen-Sitzungen abgemeldet werden. Sitzungen, die über Kommandozeilenprogramme (z. B. runppmconfig, runppmimport) gestartet wurden, können nicht abgemeldet werden.

#### **-SHOWUSERS**

Gibt alle Sitzungen der zur Laufzeit im System angemeldeten Benutzer in tabellarischer Form auf der Konsole aus.

#### **-CLEARCACHE**

Löscht die im System verwendeten Caches.

#### **-FILLCACHE**

Stößt die Neuberechnung der im System verwendeten Caches an. Bereits in den Caches vorhandene Daten werden überschrieben.

Wenn Sie das automatische Füllen des Cache in der mandantenspezifischen Datei **Keyindicator\_settings.properties** mit der Einstellung **AUTO\_FILL\_CACHE=false** ausgeschaltet haben, können Sie bei jeder Änderung Cache-relevanter Daten das erneute Füllen des Cache über die Kommandozeile mit der Option **-fillcache** anstoßen.

#### **-CLEARPCHISTORY**

Der gesamte Historien-Cache von Performance Dashboard wird gelöscht.

#### **-SAVEPCHISTORY**

Der Inhalt des speicherbasierten Historien-Cache von Performance Dashboard wird in der Datenbank gespeichert. Bereits existierende Einträge des Cache gehen hierbei verloren.

#### **-ADDPCHISTORY**

Der Inhalt des speicherbasierten Historien-Cache des Performance Dashboard wird um neu hinzugekommene Einträge erweitert. Bereits existierende Einträge des Cache bleiben erhalten.

#### **-MEMORYINFO <DATEI >**

Folgende Informationen werden ausgegeben und in die angegebene Datei geschrieben:

- Allgemeine Informationen zu PPM und zum Analyseserver
- Informationen zur Laufzeitumgebung
- Information zu Cube- und Dimensionsdaten im Analyseserver
- Information zu konfigurierten Indizes

Diese Informationen sind für den erfahrenen Anwender gedacht, um Optimierungspotential des Analyseservers zu erkennen.

Bei Ausführung dieses Kommandos kann es zu Wartezeiten bei der Analyse kommen, da der Analyseserver vor Ermitteln der Informationen zunächst einen internen Speicherbereinigungsprozess (Garbage Collection) ausführt, währenddessen er keine Analyseanfragen beantworten kann.

#### **-RECOVERYFILE FORCE**

Mit der Option **-recoveryfile force** erzwingen Sie die Neuerstellung der Wiederherstellungsdateien des PPM-Analyseservers, unabhängig vom Status der Servers. Für jeden Mandanten werden jeweils alle Wiederherstellungsdateien erstellt.

Die Wiederherstellungsdateien des Mandanten werden standardmäßig im Verzeichnis **<Installationsverzeichnis>\ppmmashzone\server\bin\work\data\_ppm\recovery\<Mandant >** abgelegt.

Der Pfad zu den Wiederherstellungsdateien ist in der mandantenspezifischen Datei **AnalysisServer\_settings.properties** durch den Schlüssel **RECOVERY\_FOLDER** definiert.



## 13.13 runppmsendmail

Mit dem Kommandozeilenprogramm **runppmsendmail** können Sie einfache Textnachrichten versenden. Das Kommandozeilenprogramm benötigt keinen direkten Bezug zu einem PPM-Mandanten und unterstützt von den bekannten allgemeinen Argumenten nur den Parameter **-protocoloptions**.

### 13.13.1 Argumente runppmsendmail

#### **-MAILHOST**

IP-Adresse des SMTP-Servers. Sollte der Verbindungsaufbau zu dem angegebene SMTP-Server fehlschlagen, wird eine entsprechende Meldung ausgegeben.

#### **-FROM**

Mail-Adresse des Absenders. Diese Adresse wird verwendet, wenn auf die E-Mail geantwortet wird.

Manche SMTP-Server leiten E-Mails nur weiter, wenn die Absenderadresse eine gültige E-Mail-Adresse ist.

#### **-TO**

E-Mail-Adresse des Empfängers. Sie können durch ; getrennt die E-Mail-Adressen mehrerer Empfänger angeben. Wenn eine E-Mail-Adresse ungültig ist oder für diese Adresse keine EMail versendet werden konnte, wird eine entsprechende Meldung ausgegeben.

Die Parameter **-mailhost**, **-from** und **-to** müssen angegeben werden. Anderenfalls wird eine entsprechende Meldung ausgegeben.

#### **-SUBJECT**

Betreff-Zeile der E-Mail. Wenn dieser Parameter nicht angegeben ist, wird als Betreff der Text **PPM test mail** verwendet.

#### **-SALUTATION**

Anrede und Name des Empfängers. Wenn dieser Parameter nicht angegeben ist, beginnt die E-Mail direkt mit dem durch Parameter **-message** bestimmten Inhalt.

#### **-FULLNAME**

Name des Absenders der E-Mail. Der Name wird in der Autorenzeile der E-Mail angezeigt. Wenn dieser Parameter nicht angegeben ist, wird der Text **PPM Server** verwendet.

#### **-MESSAGE**

Textinhalt der E-Mail. Zeilenumbrüche können durch die Zeichenfolge **<br>** eingefügt werden. Wenn dieser Parameter nicht angegeben ist, wird als Inhalt folgender Text verwendet:

This message was sent by PPM in order to verify proper functionality.  
Please do not reply to this message.

**-LANGUAGE <ISO-CODE>**

Sprache, die verwendet werden soll. Auch die Standardtexte der Parameter **-subject**, **-fullname** und **-message** werden, wenn der entsprechende Parameter nicht angegeben wurde, in der angegebenen Sprache ausgegeben.

**-ATTACHEMENT**

Pfad zu einem Dateianhang (Attachment). Es kann nur ein einzelne Datei als Anhang angegeben werden. Um mehrere Dateien beizufügen, können Sie diese zu einem Archiv zusammenfassen, z. B. zu einem zip-Archiv.

## 13.14 **runtableimport**

Das Programm **runtableimport** liest Daten aus einer Datei im Format **XML** oder **ZIP** (XML-Event-Format) und aktualisiert eine Tabelle eines Analyse-raums.

Detaillierte Informationen zu **runtableimport** erhalten Sie im Dokument **PPM Data Analytics**.

## 14 Häufig gestellte Fragen

In diesem Kapitel finden Sie zu verschiedenen Themen häufig gestellte Fragen und entsprechende Antworten.

### Haftungsausschluss /Distanzierung

Die in diesem Kapitel aufgezeigten Lösungen stellen lediglich Hinweise und Anregungen dar. Insbesondere bei Beschreibungen von Eigenschaften und Konfigurationen von 3rd-Party-Software verwenden Sie die Informationen auf eigene Gefahr. Jeder Support durch Software AG ist ausdrücklich ausgeschlossen.

### 14.1 Systemintegration

Q1.01	<p><b>Bei der Installation von PPM unter Windows Vista, Windows Server 2008 oder Windows 7 ist die Installation unvollständig. Der Cloud Agent kann als Windows Systemdienst nicht eingerichtet werden. Was ist die Ursache?</b></p>
	<p>Die genannten Windows Betriebssysteme sind mit erhöhten Sicherheitsrichtlinien versehen. Führen Sie das Installationsprogramm als Administrator aus (Kontextmenüeintrag <b>Ausführen als Administrator</b>). Andere Benutzerkonten, auch wenn diesen Konten Administratorrechte verliehen wurden, funktionieren nicht.</p>
Q1.02	<p><b>Welche Netzwerk-Ressourcen benötigt ein PPM-System?</b></p>
	<p>Als Netzwerkprotokoll unterstützt PPM ausschließlich TCP/IP. Ein PPM-Standardsystem mit http-Zugriffsmöglichkeit (z. B. das mit ausgelieferte Demoszenario) setzt sich aus folgenden Diensten zusammen und benötigt folgende Ressourcen:</p> <ul style="list-style-type: none"> <li>▪ PPM RMI Registry (Standardport <b>17500</b>)</li> <li>▪ PPM RMI-Server (pro Mandant, Standardport ab <b>17501</b> aufsteigend)</li> <li>▪ PPM Analyseserver (pro Mandant, Standardport ab <b>17502</b> aufsteigend)</li> <li>▪ http-Server für PPM</li> </ul> <p>Wenn Sie zusätzlich Performance Dashboard verwenden möchten, werden folgende, weitere Systemressourcen benötigt:</p> <ul style="list-style-type: none"> <li>▪ PPM-Corba Registry (Standardport <b>17590</b>)</li> <li>▪ PPM-Corba Server (pro Mandant, Standardport ab <b>17591</b> aufsteigend)</li> <li>▪ Web-Applikationsserver (WAS)</li> </ul> <p>Die in Klammern angegebenen Standardportbelegungen werden automatisch vom Installationsprogramm vorbelegt.</p>
Q1.03	<p><b>Wie kann man die PPM-Installation vor fremden Zugriff schützen?</b></p>

	Zum Schutz der PPM-Installation vor fremden Zugriff können Sie die Dateizugriffsrechte des Betriebssystems verwenden. Achten Sie in diesem Fall u. U. darauf, dass der Betriebssystembenutzer, mit dessen Kennung die Systemdienste gestartet werden (Windows-Standard ist Benutzer <b>SYSTEM</b> ), volle Zugriffsberechtigung in der Verzeichnisstruktur hat.
Q1.04	<b>Unterstützt PPM NAT (NetworkAddressTranslation)?</b>
	Nein, das von PPM verwendete Java-RMI-Protokoll unterstützt NAT nicht. Somit ist der Betrieb von PPM in einer Public-Cloud-Umgebung nicht möglich, wenn von außen auf das System zugegriffen werden soll.
Q1.05	<b>Kann über https auf PPM zugegriffen werden?</b>
	Ja. Der Load Balancer unterstützt seit PPM 9.0 standardmäßig https. Hierbei ist jedoch zu beachten, dass ein während der Installation erzeugtes und selbst signiertes Zertifikat verwendet wird. Dies führt in den meisten Browsern zu einer Fehlermeldung. Siehe auch Kapitel https-Unterstützung (Seite 41)
Q1.06	<b>Wie kann die Datenübertragung zwischen PPM-Server und -Benutzeroberfläche verschlüsselt werden (SSL-Verschlüsselung)?</b>
	Die SSL-Verschlüsselung können Sie mit Hilfe des Cloud Controller konfigurieren (reconfigure ppm_core ssl.enabled="true" ssl.keystorefile=" <code>&lt;path to keystore file&gt;</code> " ssl.keystorepassword=" <code>&lt;keystorepassword&gt;</code> "). Bitte beachten Sie, dass die Pfadangaben im Java-Standard angegeben werden müssen. Also auch unter Windows Betriebssystemen geben Sie bitte „/" anstelle von „\" an. Ersetzen Sie im Kennwort einen verwendeten Rückwärtsschrägstrich \ durch einen doppelten Rückwärtsschrägstrich \\ Wenn Sie die SSL-Verschlüsselung zwischen PPM-Server und -Client verwenden möchten, müssen Sie einen eigenen Keystore verwenden. Sie können mit Hilfe des Java-Keytool einen eigenen Keystore einrichten und administrieren. Detaillierte Informationen zu diesem Thema erhalten Sie in der Keytool-Dokumentation Keytool - Key and Certificate Management Tool ( <a href="http://docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html">http://docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html</a> ).
Q1.07	<b>Was versteht man unter der Betriebsart server eines PPM-Server?</b>

	Die Option <b>server</b> wird während der Installation eines Mandanten konfiguriert. Wenn die Option aktiviert ist, verwendet der PPM-Server die für Server-Applikationen optimierte Version der Java-VM. Wenn Sie diese Option verwenden möchten, müssen Sie für den PPM-Server eine Java-SDK-Version verwendet.
Q1.08	<b>Wie ändert man die vom PPM-Server verwendete Java-Laufzeitumgebung?</b>
	Das Ändern der Java-Laufzeitumgebung ist nicht möglich. Ab PPM 9 kann nur die mitgelieferte verwendet werden.
Q1.09	<b>Bei Verwendung von PPM mit MS Internet Information Server (IIS) kann nicht auf PPM zugegriffen werden. Warum?</b>
	Ab PPM 9.0 wird IIS nicht mehr unterstützt.
Q1.10	<b>Die Verbindung mit einem PPM-Mandanten-Server ist nicht möglich. In den erweiterten Informationen des Anmeldedialogs wird angegeben, dass der Rechnername des PPM-Mandanten-Servers unbekannt ist. Woran kann das liegen?</b>
	<p>Mögliche Ursachen:</p> <ul style="list-style-type: none"> <li>▪ Der Port des Mandantenserver ist durch eine Firewall blockiert.</li> <li>▪ Der Rechnername des Mandantenservers wird vom DNS-System unterschiedlich erkannt, abhängig davon, ob Sie ihn über den Namen oder IP-Adresse ansprechen. Stellen Sie sicher, dass die Kommandos <b>nslookup &lt;Rechnername&gt;</b> und <b>nslookup &lt;IP-Adresse&gt;</b> denselben Rechner referenzieren.</li> <li>▪ Das DNS-System Ihres Netzwerkes kann den Rechnernamen des PPM Mandanten-Servers nicht aufgelösen. Wenn PPM-Server und -Benutzeroberfläche in verschiedenen Sub-Netzen betrieben werden oder mehrere DNS-Server im Netzwerk existieren, kann es vorkommen, dass der Rechnername in der Datei <b>Registry_settings.properties</b> im Verzeichnis <b>clientjars</b> des http-Server angegeben ist. Vermeiden Sie widersprüchliche Namensauflösungen mittels etc/hosts-Datei und dem DNS-System.</li> </ul>
Q1.11	<b>Die Verbindung zum PPM-Server gelingt nicht. Beispielhafte Meldung in den log-Ausgaben:</b>
	<pre>Der Mandant "umg_de" existiert nicht. Fehler beim Verbindungsaufbau zum Server. Trying to connect PPM-Server //pcppm:16300/ppm_rmi_server_umg_de ppm_rmi_server_umg_de</pre>

	<code>java.rmi.NotBoundException: ppm_rmi_server_umg_de</code>
	Der gewünschte Mandantenserver ist möglicherweise nicht gestartet oder der angegebene Mandantennamen ist nicht korrekt. Wenn der Mandanten-Server gestartet ist, kann es auch sein, dass die PPM-RMI-Registry inzwischen neu gestartet wurde, z. B. durch einen unvorhergesehenen Systemzwischenfall. Starten Sie den Mandantenserver in diesem Fall neu.
Q1.12	<b>Unterstützt PPM die Verwendung eines unternehmensweiten Verzeichnisdienstes?</b>
	Über die in der zentrale Benutzerverwaltung integrierten LDAP-Schnittstelle können bestimmte Benutzer übernommen werden. Die Authentifizierung der aus dem LDAP-System übernommenen Benutzer erfolgt synchron gegen das entsprechende LDAP-System. Detaillierte Informationen finden Sie in der Online-Hilfe der zentralen Benutzerverwaltung.
Q1.13	<b>Was ist bei Umgebungsvariablen zu beachten?</b>
	Seit PPM 9.0 werden keine Umgebungsvariablen mehr unterstützt.
Q1.14	<b>Was ist eine Mandantenspezifische Konfigurationsdatei?</b>
	Für jeden installierten PPM-Mandanten gibt es im Unterverzeichnis <b>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\data_ppm\config</b> ein gleichnamiges Verzeichnis. In diesem Verzeichnis liegen die komponentenspezifischen Konfigurationsdateien des Mandanten- und Analyseservers. Die textbasierten Konfigurationsdateien haben die Endung <b>_settings.properties</b> .
Q1.15	<b>Was ist eine systemweite Konfigurationsdatei?</b>
	Die systemweiten Konfigurationsdateien spezifizieren globale Einstellungen des gesamten PPM-Systems und sind im Verzeichnis <b>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\data_ppm\config</b> gespeichert. Die systemweiten Konfigurationsdateien <b>Registry_settings.properties</b> und <b>Help_settings.properties</b> im Verzeichnis <b>config</b> der PPM-Installation müssen inhaltlich mit denen im Verzeichnis <b>clientjars</b> des Web-Servers übereinstimmen.
Q1.16	<b>Wie ändert man die TCP/IP-Ports der PPM-Dienste?</b>

	<p>Sie können die TCP/IP-Ports mit Hilfe des Cloud Controller ändern.</p> <p>Nähere Informationen zu den einzelnen Befehlen erhalten Sie in der Hilfe der Cloud Controller-Kommandozeile. Die Hilfe können Sie mittels des Befehls <b>help</b> in der Kommandozeile anzeigen.</p> <ul style="list-style-type: none"> <li>▪ <b>PPM Registry</b> reconfigure ppm_core</li> <li>▪ <b>PPM Analyseserver</b> reconfigure &lt;clientname&gt;_as</li> <li>▪ <b>PPM Server</b> reconfigure &lt;clientname&gt;_cs</li> <li>▪ <b>PPM Corba Registry</b> reconfigure ppm_core</li> <li>▪ <b>PPM Corba Server</b> reconfigure &lt;clientname&gt;_cs</li> </ul> <p>Wenn Sie den Corba-Registry-Port ändern, müssen Sie die Änderung in allen login-Modulen der Datei <b>jaas.config</b> (Property corbaurl) und in allen _conf-Dateien Ihres Performance Dashboard sowie Abfrageschnittstelle (jeweils Property corbaurl) nachpflegen (Cloud Controller: reconfigure ppm_web mit entsprechendem Mandantennamen).</p>
Q1.17	<p><b>Was bedeutet der Schlüssel CorbaServerActivationPort in der Konfigurationsdatei Registry_settings.properties?</b></p> <p>Der Corba-Registry-Server verwendet diesen Socket zur Steuerung während der Laufzeit. Dieses Feature kann nicht deaktiviert werden.</p>

### 14.1.1 Datenbankbindung

Q2.01	<p><b>Welche Einstellungen sind bzgl. des verwendeten DB-Schemas zu beachten?</b></p> <p>Die Daten eines PPM-Mandanten werden in einem eigenen DB-Schema des verwendeten Datenbanktyps gespeichert. Genaue Angaben dazu, wie Sie das DB-Schema und die Verbindungsdaten konfigurieren müssen, finden Sie im Benutzerhandbuch <b>PPM Datenbanksysteme</b>.</p> <p>Schlägt der Verbindungsaufbau zur Datenbank beim Starten des Mandantenservers fehl, weisen entsprechende Fehlermeldungen auf mögliche Ursachen hin. Verwenden Sie PPM Customizing Toolkit (<b>Mandanten Bearbeiten</b>, Dialog <b>4. Datenbankeinstellungen</b>, Schaltfläche <b>Datenbankverbindung testen</b>, um</p>
-------	--

	die korrekte Datenbankanbindung des Mandanten zu prüfen und anzupassen.
Q2.02	<p><b>Wie ändert man die Zugangsdaten zur Datenbank?</b></p> <p>Editieren Sie die mandantenspezifische Konfigurationsdatei  <code>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\data_ppm\config\&lt;Mandant&gt;\Database_settings.properties</code> und passen Sie die Schlüsselwerte <b>URL</b>, <b>USER</b> und <b>PASSWD</b> entsprechend an. Auch die Anpassung des im Schlüssel <b>DATABASE_TYPE</b> angegebenen Datenbanktyps wird unterstützt. Verwenden Sie die Kommentare in der Datei als Kurzinformation.</p> <p>Wenn Sie das Datenbankkennwort verschlüsselt angeben möchten, verwenden Sie das PPM-Kommandozeilenprogramm <b>runpassencryptor</b> mit den Parametern <b>-mode db</b> und <b>-password &lt;Neues Kennwort&gt;</b>.</p>
Q2.03	<p><b>Beim Starten des PPM-Servers wird der Datenbankfehler <code>java.lang.ClassNotFoundException</code> ausgegeben. Woran liegt das?</b></p> <p>Die erforderlichen Datenbanktreiber müssen nach der PPM-Installation und vor der Mandanten-Installation von Hand ins Verzeichnis  <code>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\data_ppm\drivers</code> kopiert werden.</p> <p>Existiert der Mandant bereits, kann ein Datenbanktreiber mittels ACC installiert werden (enhance &lt;client name&gt; with dbDriver local file &lt;path to driver jar file&gt;).</p> <p>Weitere Informationen zur Datenbankanbindung finden Sie im Benutzerhandbuch <b>PPM Datenbanksysteme</b>.</p>
Q2.04	<p><b>Beim Starten des PPM-Servers wird die Meldung</b>  DB-Error: Connection not established. (1006) [17002/null] (1034)  <b>ausgegeben. Was ist die Ursache?</b></p> <p>Der Listener der Oracle-Datenbank ist nicht verfügbar. Überprüfen Sie die Angaben Host und Port des JDBC-Connect-String (URL=, Zeile 83) in der mandantenspezifischen Konfigurationsdatei <b>Database_settings.properties</b>.</p>
Q2.05	<p><b>Beim Starten des PPM-Servers wird die Meldung</b>  DB-Error: Connection not established. (1006) [-4499/null] (1034)  Connection not established. (1006) [-4499/null]  [ibm][db2][jcc][t4][2043][11550] Exception java.net.ConnectException:  Error opening socket to server ...  <b>ausgegeben. Was ist die Ursache?</b></p> <p>Die DB2-Datenbank ist nicht verfügbar. Überprüfen Sie die Angaben <b>Host</b> und <b>Port</b> des JDBC-Connect-String (URL=, Zeile 83) in der mandantenspezifischen</p>



	Konfigurationsdatei <b>Database_settings.properties</b> .
Q2.06	<p><b>Beim Starten des PPM-Servers wird die Meldung</b></p> <pre>DB-Error: Error reading Isolation-Level (1500) [-99999/null] (1034)   Error reading Isolation-Level (1500) [-99999/null]   DatabaseMetaData information is not known for server DB2/ ...</pre> <p><b>ausgegeben. Was ist die Ursache?</b></p>
	Die Datenbank- und JDBC-Treiberversion der IBM-DB2-Datenbank passen nicht zueinander.
Q2.07	<p><b>Beim Starten des PPM-Servers wird die Meldung</b></p> <pre>DB-Error: Connection not established. (1006) [0/08S01] (1034)   Connection not established. (1006) [0/08S01]   Es konnte keine TCP/IP-Verbindung mit dem Host hergestellt werden.   java.net.ConnectException: Connection refused   DB-Error: Connection not established. (1006) [17002/null] (1034) ...</pre> <p><b>ausgegeben. Was ist die Ursache?</b></p>
	Der Listener der Oracle-Datenbank ist nicht verfügbar. Überprüfen Sie die Angaben <b>Host</b> und <b>Port</b> des JDBC-Connect-String (URL=, Zeile 83) in der mandantenspezifischen Konfigurationsdatei <b>Database_settings.properties</b> .
Q2.08	<p><b>Wie erkennt man die Version der Datenbank und des JDBC-Treibers, die der PPM-Server verwendet?</b></p> <p>Beim Starten gibt der PPM-Server Meldungen aus, welche Datenbank- und JDBC-Treiberversion verwendet werden. Beispiele:</p> <ul style="list-style-type: none"> <li>▪ <b>Oracle</b></li> </ul> <pre>I: ...: [SRV] Database version used: Oracle9i Enterprise Edition Release 9.2.0.7.0 - Production I: ...: [SRV] JDBC driver used: Oracle JDBC driver (10.2.0.4.0).</pre> <ul style="list-style-type: none"> <li>▪ <b>IBM DB2</b></li> </ul> <pre>I: ...: [SRV] Verwendete Datenbankversion: SQL09013. I: ...: [SRV] Verwendeter JDBC-Treiber: IBM DB2 JDBC Universal Driver Architecture (3.4.65).</pre> <ul style="list-style-type: none"> <li>▪ <b>MS SQL Server</b></li> </ul> <pre>I: ...: [SRV] Verwendete Datenbankversion: 8.00.2039. I: ...: [SRV] Verwendeter JDBC-Treiber: Microsoft SQL Server 2005 JDBC Driver (1.1.1501.101).</pre>

### 14.1.2 Prozessextraktor SAP-2-PPM

Q3.01	<b>Worauf ist bei der Verwendung des SAP-Java-Konnektors zu achten?</b>
-------	---

	Bei der Verwendung einer 64-Bit-Variante des SAP-Java-Konnektors <b>sapjco.jar</b> müssen Sie ebenfalls eine 64-Bit-Java-Version einsetzen. Ansonsten können die erforderlichen Java-Bibliotheken vom Konnektor nicht erkannt werden.
Q3.02	<b>Wie zeigt man Informationen zum verwendeten SAP-Java-Konnektor an?</b>
	Öffnen Sie eine DOS-Eingabeaufforderung auf dem lokalen Verzeichnis, in dem Sie den SAP-Java-Konnektor abgelegt haben, und führen Sie den folgenden Befehl aus: <pre>java -showversion -jar sapjco.jar</pre> Die Informationen werden auch per Doppelklick auf die Konnektordatei grafisch angezeigt.
Q3.03	<b>In welchen Klassenpfaden muss der SAP-Java-Konnektor enthalten sein?</b>
	Stellen Sie sicher, dass der Pfad zur Datei <b>sapjco.jar</b> im Klassenpfad aller PPM-Produktkomponenten enthalten ist, die den SAP-Java-Konnektor verwenden sollen (Prozessextraktor SAP-2-PPM, PPM Customizing Toolkit und PPM, wenn SAP-Transaktionsaufrufe aus der Prozessinstanztafel möglich sein sollen). Die Datei muss im Verzeichnis <b>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\data_ppm\drivers</b> liegen.

### 14.1.2.1 Verwendung von Content Packages

Q4.01	<b>Wo befinden sich die Mandantenkonfigurationsvorlagen Content Package for SAP HR, Content Package for SAP MM beziehungsweise Content Package for SAP SD nach der Installation?</b>
	Die Mandantenkonfigurationsvorlagen, die zusammen mit <b>PPM Process Extractor SAP-2-PPM</b> verwendet werden können, befinden sich nach der Installation im Verzeichnis <b>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-&lt;version&gt;-runnable.zip\ppm\ctk\ctk\examples\custom\...</b>
Q4.02	<b>Welche sprachabhängigen Versionen der Content Packages gibt es?</b>
	Die Mandantenkonfigurationsvorlagen zum Auslesen von Daten aus den SAP-Modulen HR, MM beziehungsweise SD sind in deutsch und englisch verfügbar. Die englische Version der jeweiligen Mandantenkonfigurationsvorlage ist durch das Namenssuffix <b>_en</b> gekennzeichnet. Möchten Sie beispielsweise das Content Package <b>Content Package for SAP SD</b> in

englischer Sprache verwenden, erstellen Sie mithilfe von **PPM Customizing Toolkit** unter Verwendung der entsprechenden Mandantenvorlage den Mandanten **ppm4sd\_en**.

## 14.2 Systemadministration

Q5.01	<p><b>Ein PPM-Benutzer hat sein Kennwort vergessen. Wie ändert man das Kennwort eines fremden PPM-Benutzers?</b></p> <p>Verwenden Sie den Administrationszugang SYSTEM, um in der Benutzerverwaltung das Kennwort des Benutzers neu zu vergeben. Geben Sie im Eingabefeld Administrator-Kennwort das Kennwort des Benutzers SYSTEM an.</p>
Q5.02	<p><b>Wie sperrt man bestimmte Benutzer für das System?</b></p> <p>Entziehen Sie dem Benutzer alle Funktionsrechte (mindestens Anmeldung, Datenberechnung und Datenimport) oder ändern Sie das Kennwort des Benutzers auf einen ihm unbekanntem Wert.</p>
Q5.03	<p><b>Wie ermittele ich die momentan am System angemeldeten Benutzer?</b></p> <p>Führen Sie in einer Eingabeaufforderung für den gewünschten Mandanten das Kommando <b>runppmadmin</b> mit der Option <b>-showusers</b> aus.</p> <p>Beispiel:</p> <pre>C:\&gt;pushd SoftwareAG\ppmmashzone\server\bin\agentLocalRepo\.unpacked\ppm-client-run-prod-95.1.0-RC8-trunk-20130404.122823-4-runnable.zip\ppm\bin D:\SoftwareAG\ppmmashzone\server\bin\agentLocalRepo\.unpacked\ppm-client-run-prod-95.1.0-RC8-trunk-20130404.122823-4-runnable.zip\ppm&gt;runppmadmin -user system -password manager -showusers</pre> <p>Das Programm zeigt Ihnen tabellarisch die Sitzungen aller momentan am PPM-System des default-Mandanten angemeldeten Benutzer an.</p>
Q5.04	<p><b>Warum misslingt das Initialisieren des Systems mittels runinitdb?</b></p> <p>Die Log-Ausgaben enthalten u.a. folgendes:</p> <pre>E: 21.10.08 14:50:22: [STD] Error opening XML file "customsysmonxmlsysmon_locales.xml".       customsysmonxmlsysmon_locales.xml (Das System kann die angegebene Datei nicht finden)</pre>

	[ java.io.FileNotFoundException ]
	<p>Der Verzeichnispfad  <b>&lt;PPM-Installationsverzeichnis&gt;...\custom\sysmon\xml\sysmon_locales.xml</b> in der Konfigurationsdatei  <b>Initdb_settings.properties</b> ist vermutlich in MS-DOS-Format angegeben. Das MS-DOS-übliche Zeichen \ (Rückwärtsschrägstrich) wertet die Java-Laufzeitumgebung als Escape-Zeichen. Ersetzen Sie das Zeichen durch einen doppelten Rückwärtsschrägstrich oder einen einfachen Vorwärtsschrägstrich /.</p>
Q5.05	<p><b>Welchen Notfallbenutzer gibt es im PPM-System?</b></p>
	<p>Der PPM-Benutzer <b>SYSTEM</b> ist als Notfallbenutzer vorgesehen. Er ist immer vorhanden, weder er selbst noch seine uneingeschränkten Systemberechtigungen können gelöscht werden.</p>
Q5.06	<p><b>Welche Authentifizierungen werden durchgeführt, wenn ein PPM-Benutzer ein PPM-Befehlszeilenprogramm ausführt (z. B. runppmconfig)?</b></p>
	<p>Führt ein PPM-Benutzer ein PPM-Befehlszeilenprogramm aus, wird überprüft, ob er für die auszuführende(n) Aktion(en) ausreichende Berechtigungen besitzt. Das Berechtigungskonzept des PPM-Systems baut sich aus folgenden Bestandteilen auf:</p> <ul style="list-style-type: none"> <li>▪ Funktionsrechte</li> <li>▪ Datenzugriffsrechte</li> <li>▪ Prozesszugriffsrechte</li> <li>▪ Datensichtbarkeitsrecht (über Benutzergruppenzuordnung)</li> </ul> <p>Beispiel (fehlendes Funktionsrecht): Benutzer <b>VOGEL</b> möchte die Benutzerverwaltung des PPM-Systems, an welchem er registriert ist, über die Kommandozeile in eine XML-Datei exportieren. Der Aufruf für den Beispielandanten <b>umg_de</b> sieht folgendermaßen aus:</p> <pre>C:\SoftwareAG\ppmmashzone\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-&lt;version&gt;-runnable.zip\ppm\bin&gt;runppmconfig -user vogel -password vogel -mode export -users user.xml -client umg_de</pre> <p>Da Benutzer <b>VOGEL</b> nicht das Funktionsrecht <b>Konfigurationsimport</b> besitzt, erscheint auf der Konsole die folgende Meldung:</p> <pre>I: 23.10.08 10:15:35: [IMP] Exportiere Konfiguration der</pre>

	<p>Komponente "users"...</p> <pre>E: 23.10.08 10:15:35: [IMP] Sie benötigen das Funktionsrecht "Konfigurationsimport". S: 23.10.08 10:15:35: [IMP] Meldungen der Komponente "IMP": S: 23.10.08 10:15:35: [IMP]      1 Fehler</pre> <p>Es werden keine Daten exportiert. Benutzer <b>VOGEL</b> muss sich zunächst vom PPM-Systemadministrator das benötigte Funktionsrecht zuweisen lassen.</p>
Q5.07	<p><b>Wird bei Ausführung von PPM-Kommandozeilenprogrammen im Protokoll auf eingeschränkte Datenzugriffsrechte des PPM-Benutzers hingewiesen (z. B. runppmcompress)?</b></p> <p>Bei Ausführung des Kommandozeilenverdichters werden neben Funktions- und Prozesszugriffrechten auch die Datenzugriffsrechte des ausführenden PPM-Benutzers überprüft. Es werden nur die Prozessinstanzen verdichtet, für die der Benutzer ein Sichtbarkeitsrecht (Datenzugriffsrecht) besitzt.</p> <p>Datenzugriffsrechte werden in der PPM-Benutzerverwaltung für Benutzergruppen definiert. Diese gelten dann für alle der Gruppe zugeordneten Benutzer.</p> <p>Beispiel: Benutzer <b>VOGEL</b> möchte für den Zeitraum März 2008 alle Prozessinstanzen des Prozesstyps <b>Terminauftrag</b> über die Kommandozeile verdichten. Der Aufruf sieht folgendermaßen aus:</p> <pre>C:\SoftwareAG\ppmmashzone\server\bin\agentLocalRepo\.unpacked\ppm-client-run-prod-&lt;version&gt;-runnable.zip\ppm\bin&gt;runppmcompress -user vogel -password vogel -client umg_de -mode execute -ps ps.xml</pre> <p>Von den achtundsechzig im Verdichterparamset (<b>ps.xml</b>) enthaltenen Prozessinstanzen hat Benutzer <b>VOGEL</b> allerdings nur das Datenzugriffsrecht auf die zwanzig Prozessinstanzen mit dem Wert <b>3000 (Frankfurt)</b> bzgl. der Datenzugriffsdimension <b>Verkaufsorganisation</b>. Daher kann Benutzer <b>VOGEL</b> nur diese zwanzig Prozessinstanzen verdichten. Auf der Konsole erscheinen dazu die folgenden Meldungen:</p> <p>[...]</p> <pre>S: 23.10.08 11:05:48: [IMP] Folgende Prozessmengen werden verdichtet: S: 23.10.08 11:05:48: [IMP] 20 Prozesse Auftragsabwicklung-Terminauftrag; Verkaufsorganisation=3000(Frankfurt). I: 23.10.08 11:05:48: [IMP] Das Programm wurde ohne Verdichten der Prozesse beendet. I: 23.10.08 11:05:48: [SRV] Baue Verbindung von Benutzer PPMDB mit jdbc:oracle:thin:@ppmdbsrv1:1521:PPM auf... I: 23.10.08 11:05:48: [SRV] Verwendete Datenbankversion: Oracle9i Enterprise Edition Release 9.2.0.8.0 - Production</pre>

	<p>With the Partitioning, OLAP and Oracle Data Mining options JServer Release 9.2.0.8.0 - Production. I: 23.10.08 11:05:48: [SRV] Verwendeter JDBC-Treiber: Oracle JDBC driver (9.2.0.8.0).</p>
Q5.08	<p><b>Was passiert, wenn ein PPM-Benutzer ohne Benutzergruppenzuordnung und damit ohne Datenzugriffrechte ändernde Tätigkeiten über die Kommandozeile ausführen will?</b></p>
	<p>Datenzugriffsrechte können PPM-Benutzern nur über die Zuordnung zu einer Gruppe zugewiesen werden. Führt ein Benutzer ohne Benutzergruppenzuordnung und somit ohne Datenzugriffrechte ein Befehlszeilenprogramm aus, weist eine Meldung darauf hin, dass der Benutzer über keine Datenzugriffsrechte verfügt.</p> <p>Beispiel:</p> <pre>C:\SoftwareAG\ppmmashzone\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-95.1.0-RC8-trunk-20130404.122823-4-runnable.zip\ppm&gt;runppmcompress -user vogel -password vogel -client umg_de -mode test -ps ps.xml [...]</pre> <p>I: 23.10.08 11:36:22: [IMP] Die Anfrage lieferte keine Daten. [...]</p>
Q5.09	<p><b>Woran kann es liegen, dass ein PPM-Benutzer trotz ausreichender Prozesszugriffs- und Funktionsrechte im PPM-Benutzeroberfläche keine Daten sieht?</b></p>
	<p>Überprüfen Sie als PPM-Administrator, ob der Benutzer einer Benutzergruppe zugeordnet ist und ob diese Gruppe über Datenzugriffrechte verfügt (<b>alle</b> oder <b>eingeschränkte</b> Datenzugriffrechte).</p>
Q5.10	<p><b>Wie können sich eingeschränkte Prozesszugriffsrechte auf Tätigkeiten über die Kommandozeile auswirken?</b></p>

	<p>Hat ein PPM-Benutzer nur Zugriffsrechte auf bestimmte Prozesstypgruppen/Prozesstypen, wie Benutzer <b>VOGEL</b> im folgenden Beispiel nur auf den Prozesstyp <b>Auftragsabwicklung\Barverkauf</b>, kann es sein, dass ihm für bestimmte Aktionen keine Daten zur Verfügung stehen:</p> <pre>C:\SoftwareAG\ppmmashzone\server\bin\agentLocalRepo\.unpacked\ppm-client-run-prod-&lt;version&gt;-runnable.zip\ppm\bin&gt;runppmcompress -user vogel -password vogel -client umg_de -ps ps.xml -mode execute I: 23.10.08 13:25:30: [IMP] Verdichte im Modus "execute"... I: 23.10.08 13:25:30: [SRV] Baue Verbindung von Benutzer PPMDB mit jdbc:oracle:thin:@ppmdbsrv1:1521:PPM auf... I: 23.10.08 13:25:30: [SRV] Verwendete Datenbankversion: Oracle9i Enterprise Edition Release 9.2.0.8.0 - Production With the Partitioning, OLAP and Oracle Data Mining options JServer Release 9.2.0.8.0 - Production. I: 23.10.08 13:25:30: [SRV] Verwendeter JDBC-Treiber: Oracle JDBC driver (9.2.0.8.0). I: 23.10.08 13:25:31: [IMP] Die Komponente(n) "Epk: Keyindicator, KeyindicatorCalculation, KeyindicatorCoreConfig, UserAdmin" wurde(n) vom Benutzer VOGEL gesperrt. I: 23.10.08 13:25:32: [SRV] Die Caches wurden gelöscht. E: 23.10.08 13:25:32: [IMP] Unbekannter Prozesstyp: Auftragsabwicklung-Terminauftrag. I: 23.10.08 13:25:32: [IMP] Die Komponente(n) "Epk: Keyindicator, KeyindicatorCalculation, KeyindicatorCoreConfig, UserAdmin" wurde(n) vom Benutzer VOGEL freigegeben.</pre> <p>PPM-Benutzer <b>VOGEL</b> kann erst dann die im Paramset (<b>ps.xml</b>) enthaltenen Prozessinstanzen des Prozesstyps <b>Terminauftrag</b> verdichten, wenn ihm vom Administrator ein Zugriffsrecht auf diesen Prozesstyp eingeräumt wurde.</p>
Q5.11	<p><b>Beim Datenimport erhalte ich OutOfMemory-Meldungen. Was kann ich tun?</b></p>
	<p>Teilen Sie dem entsprechenden Programm mehr Speicher zu. Sie können die Speicherangaben in PPM Customizing Toolkit über <b>Mandanten bearbeiten</b> (Aufruf der Mandanteninstallation) auf der Seite <b>Servereinstellungen</b> unter <b>JVM-Parameter der Serverprogramme &gt; JVM-Parameter konfigurieren</b> ändern.</p> <p>Passen Sie für runxmlimport den maximalen Systemspeicher an, z. B. <b>1024 MB</b> für max. 1 GB Systemspeicher.</p> <p>Anschließend müssen Sie den Systemdienst des PPM-Servers - falls eingerichtet - bzw. den PPM-Server neu starten.</p>
Q5.12	<p><b>Wie erhöht man den Systemspeicher für den PPM-Server?</b></p>

	<p>Sie können die Speicherangaben jederzeit bequem in PPM Customizing Toolkit über <b>Mandanten bearbeiten</b> (Aufruf der Mandanteninstallation) auf der Seite <b>Servereinstellungen</b> unter <b>Einstellungen des Mandantenservers</b> ändern.</p> <p>Anschließend müssen Sie den Systemdienst des PPM-Servers - falls eingerichtet - bzw. den PPM-Server neu starten.</p>
Q5.121	<p><b>Wie passt man den Systemspeicher für den Analyseserver an?</b></p> <p>Bearbeiten Sie die Angaben zu mini- bzw. maximalem Speicherbedarf bequem in PPM Customizing Toolkit über <b>Mandanten bearbeiten</b> (Aufruf der Mandanteninstallation) auf der Seite <b>Servereinstellungen</b> unter <b>Einstellungen des Analyseservers</b>.</p> <p>Anschließend müssen Sie den Analyseserver neu starten.</p>
Q5.13	<p><b>Welche regelmäßigen Aufgaben (periodical tasks) gibt es für ein PPM-System?</b></p> <p>Die genauen, regelmäßigen Betreuungsaufgaben eines PPM-Systems sind projektspezifisch und werden mit dem Kunden abgestimmt. Wir empfehlen folgende, regelmäßige Tätigkeiten:</p> <ul style="list-style-type: none"> <li>▪ Sicherung des Datenbankschemas des Mandanten</li> <li>▪ Sicherung der Mandantenkonfiguration (Verzeichnisse <b>web</b>, <b>config</b> und <b>custom</b> unter &lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\data_ppm</li> <li>▪ Archivierung der importierten Quellsystemdaten</li> <li>▪ Archivierung der Log-Dateien des Analyse- und PPM-Servers und der Kommandozeilenprogramme</li> </ul> <p>Weitere Informationen finden Sie in den Dokumentationen <b>PPM Datenbanksysteme</b> und <b>PPM Datenimport</b>.</p>

### 14.2.1 PPM-Kommandozeilenprogramme: Protokollausgaben

Q6.01	<b>Wie lassen sich Protokollausgaben konfigurieren?</b>
	Die Protokolloptionen <b>information warning error</b> der



	Kommandozeilenprogramme lassen sich in beliebiger Kombination verwenden (siehe Kapitel <b>Kommandozeilenprogramme</b> (Seite 33)). Ohne Angabe der Protokolloptionen werden ausschließlich die modulspezifischen Log-Level aus den <b>Server_Log_settings.properties</b> verwendet (Empfehlung für XML-Import und PPM-Import).
Q6.02	<b>Können Protokollausgaben in eine Datei umgelenkt werden?</b>
	Mit der Anweisung <b>-logfile &lt;Dateiname&gt;</b> lassen Sie die Protokollausgaben der PPM-Kommandozeilenprogramme in die angegebene Datei schreiben. Sie können die Anweisung mit den zuvor beschriebenen Protokolloptionen beliebig kombinieren.
Q6.03	<b>Können Protokollausgaben sprachabhängig eingestellt werden?</b>
	Mit der Anweisung <b>-language &lt;ISO-code&gt;</b> können Sie die Sprache einstellen, sofern sie vom PPM-System unterstützt wird, z. B. <b>-language en</b> .
Q6.04	<b>Wo befinden sich die Log-Dateien von PPM- und Analyseserver?</b>
	Standardmäßig befinden sich die Log-Dateien von PPM- und Analyseserver im mandantenspezifischen Verzeichnis <b>&lt;Installationsverzeichnis&gt;\ppmmashzone\server\bin\work\data_ppm\log\&lt;Mandat&gt;</b> . Der Dateiname setzt sich aus dem Mandantennamen mit der Erweiterung <b>.log</b> zusammen. Nach einem Neustart von PPM- und Analyseserver werden die Log-Dateien fortgeschrieben und nicht überschrieben.
Q6.05	<b>Kann man den Speicherort der Log-Dateien ändern?</b>
	Nachträglich können Sie den Speicherort von Log-Dateien des PPM- bzw. Analyseservers ändern, indem Sie in den Dateien <b>Server_log_settings</b> und <b>AnalyseServer_log_settings</b> die entsprechenden Appender konfigurieren (siehe Kapitel PPM Systemmeldungen (Seite 20)).
Q6.06	<b>Werden Änderungen der PPM-Konfiguration über die Oberfläche protokolliert?</b>
	Änderungen an der PPM-Konfiguration via GUI, die gespeichert werden, erscheinen als Meldungen im Protokoll auf der Konsole des PPM- bzw. Analyseservers. Zusätzlich erfolgen Ausgaben in die Trace-Dateien von PPM- bzw. Analyseserver, wenn in den Settings von PPM- bzw. Analyseserver das entsprechende Trace-Modul aktiviert ist (siehe Kapitel <b>Systemmeldungen</b> ).  Hat ein PPM-Benutzer z. B. eine neu erstellte, benutzerdefinierte Kennzahl

	<p>gespeichert, geben die bearbeitenden Komponenten entsprechende Meldungen aus (hier: Kennzahlenberechner und Server):</p> <pre>I: 22.10.08 16:30:29: [KZB] Überprüfe die Kennzahlenkonfiguration... I: 22.10.08 16:30:29: [KZB] Die Kennzahlenkonfiguration wurde erfolgreich überprüft. I: 22.10.08 16:30:29: [SRV] Konsistenzprüfung für benutzerdefinierte Kennzahlen wird durchgeführt... I: 22.10.08 16:30:29: [SRV] Konsistenzprüfung für benutzerdefinierte Kennzahlen erfolgreich.</pre>

## 14.2.2 Inkonsistenz der Datenbestände

Q7.01	<p><b>Wie stellt man die Konsistenz der Datenbestände des Analyseservers und PPM-Servers wieder her?</b></p>
	<p>Änderungen an der Konfiguration des PPM-Systems oder erneute Datenimporte können dazu führen, dass die Datenbestände des Analyseservers und PPM-Servers nicht mehr konsistent sind. Entsprechende Fehlermeldungen der PPM-Oberfläche weisen darauf hin.</p> <p>Öffnen Sie eine DOS-Eingabeaufforderung, wechseln Sie in das Verzeichnis <b>&lt;PPM-Installationsverzeichnis&gt;\server\bin\agentLocalRepo\unpacked\ppm-client-run-prod-&lt;version&gt;-runnable.zip\ppm\bin</b> und führen Sie die folgende Befehlszeile aus:</p> <pre>runppmimport -user &lt;ppmuser&gt; -password &lt;password&gt; -client &lt;ppmclient&gt; -reinitanalysissserver auto</pre> <p>Der Analyseserver wird daraufhin neu initialisiert und die veränderten Daten in seine Datenstrukturen geladen. Danach sind die Datenbestände der beiden Server wieder konsistent.</p>