



**Autodesk Productstream**  
**COMPASS 5 Job Server**  
**Systemhandbuch**

März 2006

© Copyright 2006 Autodesk, Inc.

Alle Rechte vorbehalten.

Dieses Werk oder Teile davon dürfen in keiner Form, mit keiner Methode und zu keinem Zweck vervielfältigt werden.

AUTODESK, INC. GIBT KEINERLEI GARANTIE, WEDER AUSDRÜCKLICH NOCH IMPLIZIT, AUF DIESE MATERIALIEN, SCHLIESST JEDLICHE IMPLIZITE GARANTIE AUF DIE VERKAUFSEIGNUNG ODER TAUGLICHKEIT EINER BESTIMMTEN ANWENDUNG DIESER MATERIALIEN AUS UND STELLT DIESE MATERIALIEN LEDIGLICH IN DER AUGENBLICKLICHEN FORM ZUR VERFÜGUNG.

AUTODESK, INC. KANN KEINESFALLS UND GEGENÜBER NIEMANDEM FÜR BESONDERE, KOLLATERALE, ZUFÄLLIGE ODER INDIREKTE SCHÄDEN, DIE SICH AUS DEM KAUF ODER DER BENUTZUNG DIESER MATERIALIEN ERGEBEN, HAFTBAR GEMACHT WERDEN. IM FALLE EINER HAFTUNG SEITENS AUTODESK, INC., IST AUTODESK, INC. AUSSCHLIESSLICH UND HÖCHSTENFALLS FÜR DIE RÜCKERSTATTUNG DES KAUFPREISES DER HIER BESCHRIEBENEN MATERIALIEN HAFTBAR.

Bestimmungen zur Verwendung und die Genehmigung zur Veröffentlichung dieses Materials in einer anderen Sprache als Deutsch müssen bei Autodesk, Inc. angefordert werden. Sämtliche Rechte an der Übersetzung der vorliegenden Publikation liegen bei Autodesk, Inc. Autodesk, Inc. behält sich das Recht vor, ihre Produkte nach eigenem Gutdünken zu revidieren oder zu verändern. Diese Publikation beschreibt den Zustand dieses Produkts zum Zeitpunkt der Veröffentlichung und muss nicht mit zukünftigen Versionen des Produkts übereinstimmen.

#### Autodesk-Marken

3D Studio, 3D Studio MAX, 3D Studio VIZ, 3ds max, ActiveShapes, ActiveShapes (Logo), Actrix, ADI, AEC-X, ATC, AUGI, AutoCAD, AutoCAD LT, Autodesk, Autodesk Envision, Autodesk Inventor, Autodesk Map, Autodesk MapGuide, Autodesk Streamline, Autodesk WalkThrough, Autodesk World, AutoLISP, AutoSketch, backdraft, Biped, bringing information down to earth, Buzzsaw, CAD Overlay, Character Studio, Cinepak, Cinepak (Logo), cleaner, Codec Central, combustion, Design Your World, Design Your World (Logo), EditDV, Education by Design, gmax, Heidi, HOOPS, Hyperwire, i-drop, IntroDV, lustre, Mechanical Desktop, ObjectARX, Physique, Powered with Autodesk Technology (Logo), ProjectPoint, RadioRay, Reactor, Revit, VISION\*, Visual, Visual Construction, Visual Drainage, Visual Hydro, Visual Landscape, Visual Roads, Visual Survey, Visual Toolbox, Visual Tugboat, Visual LISP, Volo, *WHIP!* und *WHIP!* (Logo) sind eingetragene Marken von Autodesk, Inc., in den USA und/oder anderen Ländern.

AutoCAD Learning Assistance, AutoCAD LT Learning Assistance, AutoCAD Simulator, AutoCAD SQL Extension, AutoCAD SQL Interface, AutoSnap, AutoTrack, Built with ObjectARX (Logo), burn, Buzzsaw.com, CAiCE, Cinestream, Civil 3D, cleaner central, ClearScale, Colour Warper, Content Explorer, Dancing Baby (Image), DesignCenter, Design Doctor, Designer's Toolkit, DesignKids, DesignProf, DesignServer, Design Web Format, DWF, DWFit, DWG Linking, DXF, Extending the Design Team, GDX Driver, gmax (Logo), gmax ready (Logo), Heads-up Design, jobnet, mass, ObjectDBX, onscreen onair online, Plasma, PolarSnap, Productstream, Real-time Roto, Render Queue, Visual Bridge, Visual Syllabus und Where Design Connects sind Marken von Autodesk, Inc., in den USA und/oder anderen Ländern.

#### Autodesk Canada Inc.-Marken

discreet, fire, flame, flint, flint RT, frost, glass, inferno, MountStone, riot, river, smoke, sparks, stone, stream, vapour und wire sind eingetragene Marken von Autodesk Canada, Inc., in den USA und/oder Kanada und/oder anderen Ländern.

backburner und Multi-Master Editing sind Marken von Autodesk Canada, Inc., in den USA, Kanada und/oder anderen Ländern.

#### Marken anderer Firmen

TList™ 5 Active X control und Bennet-Tec Information Systems.

Alle übrigen Markennamen, Produktnamen oder Marken sind Eigentum der jeweiligen Inhaber.

#### Anwendungen von Drittanbietern

HTML Help © 1995-2002 Microsoft Corp. Alle Rechte vorbehalten.

Internet Explorer © 1995-2001 Microsoft Corp. Alle Rechte vorbehalten.

Microsoft® SQL Server™ 2000. Copyright © 2003 Microsoft Corporation. Alle Rechte vorbehalten.

Integrated Viewing Technology © 1993-2003 Cimmetry Systems, Inc.. Alle Rechte vorbehalten

List & Label © 1992-2003 combit GmbH. Alle Rechte vorbehalten.

Veröffentlicht von: Autodesk, Inc.  
111 McInnis Parkway  
San Rafael, CA 94903  
USA

# Inhaltsverzeichnis

<b>Inhaltsverzeichnis .....</b>	<b>3</b>
<b>Vorwort .....</b>	<b>4</b>
Zielgruppe.....	4
Voraussetzungen.....	4
Inhalt.....	5
Neuerungen.....	5
<b>Grundlagen .....</b>	<b>6</b>
Allgemeines Konzept.....	6
Begriffsdefinitionen.....	6
Einschränkungen des Jobserver im Productstream Compass Easy Umfeld.....	6
Anwendungsbeispiele.....	7
Anwendungsszenarien.....	7
Tipps & Tricks.....	8
Mehrere Productstream Compass Job Server im selben Umfeld.....	11
Productstream Compass Job Server in einer replizierten Umgebung.....	11
<b>Installation.....</b>	<b>12</b>
Lizenzierung.....	12
Installation des Client.....	12
Parameter der Productstream Compass Job Server- Verknüpfung.....	12
<b>Die Komponenten.....</b>	<b>13</b>
Die Productstream Compass Job Server- Ordner.....	13
Der Jobspooler.....	14
Der „Watchdog“ überwacht die Jobs.....	15
Die Productstream Compass Job Server Statusverwaltung.....	16
Bedeutung der Felder.....	16
<b>Konfiguration und Administration.....</b>	<b>18</b>
Benutzer, Gruppen und Rechte.....	18
Einen neuen Jobtyp definieren.....	18
Arbeitsweise des Productstream Compass Job Server.....	19
Konfiguration.....	20
Beispiele.....	22
<b>Referenz .....</b>	<b>27</b>
Interne Funktionen.....	27
DBP-Prozeduren.....	27
Rückgabewerte aus der AutoCAD-Schnittstelle.....	29

## Vorwort

Willkommen bei Autodesk Productstream Compass Job Server für Productstream Compass 5, dem leistungsstarken PDM-System von Autodesk Inc.

Der Productstream Compass Job Server ermöglicht es, Arbeitsprozesse rund um Productstream Compass zu automatisieren. Damit kann Zeit gespart und die Produktivität im Unternehmen erhöht werden. Der Productstream Compass Job Server ist ein Zusatzmodul der Autodesk Productstream Compass Produktfamilie und muss eigens lizenziert werden.

## Zielgruppe

Die Productstream Compass Job Server Dokumentation richtet sich an Systempartner und Administratoren. Das Dokument umfasst Informationen rund um Installation, Konfiguration und Wartung. Informationen für den Anwender zur Bedienung sind im Anwenderhandbuch enthalten.

## Voraussetzungen

- Zur Anwendung und Konfiguration des Productstream Compass Job Server wird Grundwissen über Productstream Compass und über die Applikationen mit denen sich der Productstream Compass Job Server beschäftigt vorausgesetzt. Zur Implementierung komplexer Jobs sind gegebenenfalls Programmierkenntnisse zu Productstream Compass und den betroffenen Applikationen nötig.
- Es wird empfohlen, den Productstream Compass Job Server auf einem dedizierten Rechner (Server oder Workstation) zu betreiben. Dadurch kann er kontinuierlich und ungestört seine Jobs ausführen. Es ist abzuraten den Productstream Compass Job Server auf dem Datenbankserver zu installieren. Das kontinuierliche Arbeiten mit Applikationen nimmt viel Rechnerleistung in Anspruch, was sich negativ auf die Performance der anderen Arbeitsplätze auswirken könnte. Der Productstream Compass Job Server könnte auch auf einem lokalen Arbeitsplatz installiert, dann jedoch nur außerhalb der regulären Bürozeiten aktiviert werden.
- Die Systemvoraussetzungen für den Productstream Compass Job Server -Arbeitsplatz richten sich nach den zu behandelnden Applikationen.
- Der Productstream Compass Job Server ist sowohl für Autodesk Productstream Compass Easy als auch für Productstream Compass Pro verfügbar. Über eventuelle Einschränkungen im Productstream Compass Easy-Umfeld, lesen Sie bitte das Kapitel „Einschränkungen des Productstream Compass Job Server im Productstream Compass Easy-Umfeld“
- Für den korrekten Betrieb des Productstream Compass Job Server benötigen Sie einen eigenen Lizenzcode. Dieser ist kostenpflichtig und auf Ihrer Registrationskarte zu finden.
- Damit der Productstream Compass Job Server die unterschiedlichen Jobs ausführen kann, müssen auch die dazu benötigten Applikationen auf dem Productstream Compass Job Server Arbeitsplatz installiert sein. Muss zum Beispiel, ein CAD Dokument gedruckt, konvertiert o. ä. werden, so muss auch die entsprechende CAD-Applikation installiert sein.

## Inhalt

Die Dokumentation des Productstream Compass Job Server ist in mehrere Teile aufgliedert und richtet sich an unterschiedliche Zielgruppen und Arbeitsweisen.

Administratoren und Systempartner finden in der vorliegenden Dokumentation zum Productstream Compass Job Server alle Informationen, die sie bei der Installation, Konfiguration und Administration des Productstream Compass Job Server benötigen.

**Hinweis!** Diese Dokumentation ergänzt die Dokumentation von Productstream Compass mit den Informationen, die für die Konfiguration und Administration des Productstream Compass Job Server notwendig sind. Autodesk ist bemüht, sie möglichst vollständig und auf dem aktuellen Stand zu halten. Wir sind jedoch für Hinweise und Verbesserungsvorschläge dankbar.

## Neuerungen

In die aktuelle Version des Productstream Compass Job Server wurden folgende Neuerungen eingebracht:

- Feld Prio wurde durchgängig auf allen Masken bereit gestellt. Im Standard wird das Feld über eine Auswahlliste mit den Prios A, B, C bedient.
- Vor der Ausführung des Jobs, lädt der Productstream Compass Job Server jetzt die Daten des zu verarbeitenden Elementes neu. Damit wird gesichert, dass der Productstream Compass Job Server immer auf aktuelle Informationen zugreift.
- Restart des Productstream Compass Job Server wurde optimiert. Ist der automatische Restart des Productstream Compass Job Server eingestellt, so ist nur eine einmalige Anmeldung nötig. Ab diesem Zeitpunkt meldet sich der Productstream Compass Job Server nach jedem automatischen Restart mit den selben Benutzerdaten wieder an, unabhängig davon, ob Autologin gesetzt ist oder nicht.

## Grundlagen

### Allgemeines Konzept

Der Productstream Compass Job Server dient zur Automatisierung von Arbeitsprozessen, die im Zusammenhang mit Productstream Compass erledigt werden sollen. Jeder Arbeitsprozess wird in Form eines Jobtyps definiert. Dies geschieht im Unterordner „Konfiguration“ des Ordners „Jobserver“ und wird im Allgemeinen vom Systempartner oder geschulten Administrator durchgeführt. Der Anwender kann beliebige Elemente aus Productstream Compass per Drag&Drop an den Ordner „Jobserver“ übergeben. Bei der Übergabe der Elemente wird überprüft, welche Jobtypen zu den übergebenen Elementen passen. Die passenden Jobtypen werden dem Anwender zur Auswahl angeboten. Wählt der Anwender einen passenden Jobtyp aus, so entsteht ein Eintrag (Job) im Ordner „Aktive Jobs“.

Der Jobserver läuft normalerweise auf einem dedizierten Rechner. Über die vereinfachte Oberfläche des Productstream Compass Job Server, auch Jobspooler genannt, können die auszuführenden Jobtypen ausgewählt werden. Ein Notizfeld im unteren Bereich protokolliert die laufenden Aktivitäten mit. Nach dem Start des Productstream Compass Job Server werden die Jobs in zyklischen Abständen abgearbeitet. Der Verlauf jedes Jobs ist im Notizfeld zu sehen. Am Jobstatus ist zu erkennen, ob dieser erfolgreich war oder nicht. Ist ein Job bearbeitet, so ist er im Ordner „erledigte Jobs“ zu finden. Zu jedem Job gibt es eine eigene Historie, welche den Verlauf dokumentiert.

### Begriffsdefinitionen

Job	Ist eine durch den Anwender ausgelöste Aufgabe für den Productstream Compass Job Server
Jobtyp	Ist der vom Administrator definierte Typ der Aufgabe. Von dieser Definition werden die Jobs abgeleitet
Jobspooler	Ist die Oberfläche des Productstream Compass Job Server, welche die einzelnen Jobs abarbeitet
Jobserver	Ist die gesamte Umgebung, welche sich sowohl um die Ausführung als auch um die Verwaltung rund um die Jobs und Jobtypen kümmert.

### Einschränkungen des Jobserver im Productstream Compass Easy Umfeld

Der Productstream Compass Job Server ist ein eigenständiges Modul. Als solches darf und kann er in jedem Productstream Compass-Umfeld programmiert werden. Wird der Productstream Compass Job Server in einer Productstream Compass Easy-Umgebung eingesetzt, so ist es erlaubt, Funktionen, welche vom Spooler am Server ausgeführt oder Funktionen, die bei der Erzeugung des Jobs am Productstream Compass Client über die Funktion `___JobCreate` ausgeführt werden zu implementieren. Diese Funktionen dürfen ausschließlich vom Productstream Compass Job Server ausgeführt werden

Ein konkretes Beispiel: Muss ein Job mehrere Aktionen ausführen, so können diese Funktionen am Job selbst oder als Modul implementiert werden. Abhängig vom Kontext der Funktionen, muss das DBP-Modul in der Konfiguration entsprechend eingebunden werden. Damit ist der Funktionsumfang der Productstream Compass Easy erweitert. Diese neu eingebrachten Funktionen dürfen jedoch nur vom Productstream Compass Job Server bzw. *Productstream Compass Job Server*-Funktionen (wie `___JobCreate`) ausgeführt werden.

Wir behalten uns vor, jederzeit weitere technische Vorkehrungen zu treffen, um die Ausführung von nicht durch Autodesk implementierte Funktionen im Productstream Compass Easy-Client zu unterbinden.

## Anwendungsbeispiele

Um einen Überblick über die Leistungsfähigkeit und die vielfältigen Einsatzmöglichkeiten zu geben, werden hier einige Beispiele für den Einsatz des Productstream Compass Job Server angeführt. Diese Beispiele können von Systempartnern oder geschulten Administratoren implementiert werden, sind jedoch nicht in der Auslieferung enthalten. Einige dieser Beispiele werden in den Folgenden Kapiteln genauer beschrieben.

- Aktualisierung der Dokumente bei der Freigabe: Wird ein Dokument freigegeben, so soll dieses mit den zuletzt gültigen Informationen versorgt und eventuell vorher „bereinigt“ werden.
- Automatische Benachrichtigung: Lassen Sie sich vom Productstream Compass Job Server per e-mail über die Aktivitäten in Ihrem System informieren. Lassen Sie den Productstream Compass Job Server die betroffenen Mitarbeiter informieren, wenn z. B. Dokumente zu prüfen sind oder freigegeben werden. Oder lassen Sie Ihre Einkaufsabteilung benachrichtigen, wenn eine Stückliste in Productstream Compass freigegeben wurde.
- Drucken: Wieviel Zeit wird zum Drucken von Dokumenten verwendet?. Wieviele Dokumente werden pro Tag in Ihrer Konstruktionsabteilung gedruckt? Wie oft kommen Kollegen aus anderen Abteilungen und wollen ein Dokument gedruckt haben? In allen diesen Situationen hilft der Productstream Compass Job Server. Wenn alle Druckaufträge zentral abgewickelt werden, können alle Abteilungen Druckaufträge absetzen, auch wenn sie selbst die Applikation (zum Beispiel CAD) nicht installiert haben. Das erspart Zeit und erhöht die Qualität, denn es wird immer die richtige Version des Dokumentes gedruckt.
- Stapeldrucken: Was passiert am Ende eines Projektes? Was tun wenn Sie die komplette Dokumentation einer Maschine ausgedruckt haben möchten? Der Productstream Compass Job Server kann so konfiguriert werden, dass per Knopfdruck alle Dokumente zu einem Projekt oder alle relevanten Dokumente einer Stückliste, auch über mehrere Ebenen hinweg, gedruckt werden.
- Konvertieren: Dokumente müssen heutzutage in einem neutralen Format vorliegen. Dieses ist von jedermann lesbar, zum Verschieben geeigneter und zeitbeständiger. Konvertieren Sie automatisch bei der Freigabe oder bei Bedarf Ihre Dokumente in ein neutrales Format wie DWF, TIFF, PDF oder ähnliches.
- Reports: Reports geben eine Übersicht des aktuellen Zustandes Ihrer Informationen. Sie können zum Beispiel ausdrucken, welche Dokumente noch zu prüfen oder in Arbeit sind, oder wie die Stücklisten aussehen, etc. Lassen Sie sich zum Beispiel jeden Montagmorgen automatisch einen Report über die für Sie wichtigen Daten zukommen, entweder ausgedruckt, oder per Mail. Reports halten den momentanen Zustand fest, damit können sie auch gut als feste Historie dienen.
- Datenaustausch: Sie müssen in regelmäßigen Zeitabständen Informationen mit anderen Systemen austauschen? Der Productstream Compass Job Server kann diesen Arbeitsgang übernehmen. Entsprechend konfiguriert und programmiert, kann er Daten aus Productstream Compass exportieren und sie eventuell auch wieder importieren.

Haben diese Beispiele Ihr Interesse geweckt? Sprechen Sie mit Ihrem Systempartner über Ihre Prozessprobleme; eine Lösung lässt sich sicher finden.

## Anwendungsszenarien

Die Anwendungsszenarien zum Productstream Compass Job Server sind im Productstream Compass Handbuch im Kapitel „Anwendungsszenarien“ zu finden .

## Tipps & Tricks

Die Bedienung und Konfiguration des Productstream Compass Job Server ist einfach, wenn man sich überlegt was man erreichen möchte und auf ein paar seiner Eigenschaften achtet. Hier folgen Tipps, die wir aus unserer eigenen Erfahrung gesammelt haben.

## Einige Grundsätze

Grundsätzlich gilt, dass der Productstream Compass Job Server jene Aufgaben abarbeiten soll, die sonst ein Mitarbeiter mit hohem Zeitaufwand erledigen müsste. Damit ist der Anspruch auf Zuverlässigkeit und Schnelligkeit recht hoch.

Um Aufgaben schnell und sicher erledigen zu können, ist es bei Menschen wie auch bei Software wichtig, dass die Aufgaben genau definiert sind. Das heißt, dass bei der Definition eines Jobtyps, dessen Zweck und Ziel genau erfasst werden muss. Bei der Implementierung des Jobtyps muss darauf geachtet werden, dass sich dieser nur um das Wesentliche kümmert und von nichts anderem gestört wird.

## Keep it small and simple (KISS)

Versuchen Sie die Jobtypen so „schlank“ wie möglich zu gestalten. Sollen mehrere verschiedene Aufgaben, wie Drucken, Konvertieren, Statuswechsel, etc. auf einmal erledigt werden, so empfiehlt es sich, diese Aufgaben nicht in einem Job abzuwickeln sondern auf verschiedene Jobs aufzuteilen. Nach erfolgreicher Abarbeitung des ersten Jobs, kann dieser dann den nächsten auslösen.

Damit erreichen Sie mehr Transparenz. Jeder Job hat exakt eine Aufgabe und muss nur diese abarbeiten. Schlägt diese fehl, so kann die Analyse des Problems auf diesen Teil beschränkt werden. Läuft er erfolgreich durch, so löst er den Folgejob aus, und die korrekte Abfolge der Jobs ist gewährleistet.

Jobs die „schlank“ sind, haben auch eine kurze Laufzeit. Damit kann das Timeout für diesen Job kürzer gesetzt werden. Ein umfangreicher Job braucht seine Zeit bis er abgearbeitet ist. Damit muss das Timeout größer sein. Tritt ein Problem bereits am Anfang der Ausführung des Jobs auf, so bleibt der Productstream Compass Job Server möglicherweise für längere Zeit stehen (bis das Timeout eintritt) und keine weiteren Jobs kommen zum Zuge. Das Laufzeitverhalten des Productstream Compass Job Server wäre also in diesem Fall nicht optimal.

Ein weiterer Vorteil von „schlanken“ Jobtypen ist, dass man für jeden einzelnen Jobtyp eine eigene Fehlerbehandlungsroutine definieren kann. Hat man einen umfangreichen Jobtyp, so kann man den Fehlerfall nur für den gesamten Job behandeln.

## Weg mit dem „Speck“

Vermeiden Sie unnötigen Ballast. Überlegen Sie sich was Sie wirklich brauchen, um den Job auszuführen. Ein Beispiel: Sie möchten über MS-Word ein Bürodokument drucken. Dazu wollen Sie dieses Dokument mit Hilfe der Productstream Compass-Schnittstelle (CAI) öffnen und bearbeiten. Sie müssen dabei bedenken, dass eine Schnittstelle meist mehr leistet, als eine Applikation zu starten. Die CAI-Schnittstelle versucht, einen Kommunikationskanal zwischen Productstream Compass und der Applikation aufzubauen, um z.B. Informationen auszutauschen. Wenn Sie jedoch nur ein Dokument drucken wollen, so benötigen Sie keine Kommunikation mittels einer Schnittstelle. Ein einfaches `___Shell(shellexecute(print „#DOCNAME0“))` würde also reichen. Die Applikation startet schneller weil keine Initialisierung nötig ist und die ganz normalen Druckmechanismen des Betriebssystems genutzt werden. Der Job wird auf diese Weise schneller abgearbeitet und wenn ein Fehler auftritt, können Sie ein Schnittstellenproblem ausschließen.

Ähnlich verhält es sich mit AutoCAD. Möchten Sie beispielsweise AutoCAD starten, ein Productstream Compass-Dokument laden, einige Einstellungen verändern und dann einen Druck veranlassen, so könnten Sie das mit Hilfe der CAI-Befehle tun. Dafür muss jedoch die Schnittstelle initialisiert werden. Dies benötigt Zeit und führt zu unnötigem Ballast in der Applikation. Sie könnten das Ergebnis statt dessen durch direkten Aufruf der Applikation (C:\Programme\AutoCAD.exe) und Übergabe einer Script-Datei als Parameter erreichen. In der Script-Datei sind dann die auszuführenden Befehle enthalten. Damit umgehen Sie die Schnittstelle, haben eine geringere Joblaufzeit, und die übergebenen Befehle sind unter Kontrolle. Tritt ein Problem auf, sind die mitwirkenden Stellen klar ersichtlich.

Denken Sie also daran. Wenn Sie einen Jobtyp definieren, fragen Sie sich, ob sie wirklich eine weitere Applikation/Schnittstelle verwenden wollen oder ob es nicht einen einfacheren Weg gibt.



## Wann ist der richtige Zeitpunkt für die Ausführung eines Jobs?

Nehmen wir an, Sie müssen einen Konvertierungsjob (in DWF, PDF, DXF, TIFF) implementieren. Sie möchten sicher gehen, dass der Inhalt des zu konvertierenden Dokumentes aktuell ist. Deshalb entscheiden Sie, vor der Konvertierung immer eine Aktualisierung des Dokumentes vorzuschalten. Es wäre nun sinnvoll, sich zu überlegen, wann eigentlich die Konvertierung durchzuführen ist. Oft stellt sich heraus, dass nur freigegebene Dokumente konvertiert werden sollen. Wenn man gewährleistet, dass bei der Freigabe die Inhalte des Dokumentes aktualisiert werden, kann man davon ausgehen, dass bei der Konvertierung das Dokument bereits aktuell ist. Stellt sich heraus, dass sowohl freigegebene Dokumente als auch in Arbeit befindliche Dokumente konvertiert werden müssen, so könnte man sich überlegen, zwei Typen von Jobs zu haben: einen der immer nur für freigegebene Dokumente ausgeführt wird und damit keine Aktualisierung benötigt, den anderen mit einer vorgeschalteten Aktualisierung.

Überlegen Sie sich daher, ob der Job wirklich zu diesem Zeitpunkt ausgeführt werden muss, oder ob es eventuell einen günstigeren gibt. Damit sparen Sie wieder Jobzeit ein und reduzieren die Jobs auf das Notwendigste. Ist Ihnen das Risiko der Konvertierung an einem nicht aktuellen Dokument zu hoch, könnten Sie prüfen, ob es ausreicht, nach der Aktualisierung in einem Feld des Datensatzes einen Zeitstempel einzufügen. Darüber können Sie prüfen, ob und wann eine Aktualisierung stattgefunden hat und ob diese notwendig ist.

## Verwenden Sie Bedingungen!

Angenommen Sie haben eine Reihe von Jobtypen definiert. Sie installieren diese beim Kunden und nach wenigen Tagen treten Fehlermeldungen auf. Das könnten Probleme mit den Jobs sein, doch nach einer Analyse stellen Sie fest, dass die Jobs korrekt sind aber der Anwender diese Jobs falsch nutzt. Jetzt haben Sie zwei Möglichkeiten:

- 1.) den Anwender schulen, damit er weiß, wie und wann er die Jobs richtig anwendet
- 2.) die Jobtypen „wasserdicht“ machen

Angenommen, Sie haben einen Druck-Job definiert. Es kann nun vorkommen dass Anwender Dokumente an den Druck-Job geben, denen keine Datei zugeordnet ist. Dies führt natürlich zu einer Fehlermeldung. Die Lösung ist aber recht einfach. Nutzen Sie die Bedingungsfelder „für die Anlage“, „für die Aktivierung“, „für die Ausführung“. Hier können Sie festlegen welche Bedingungen erfüllt sein müssen, um den Job anlegen, starten oder abarbeiten zu können.

- Erstellen Sie Jobs für Dateien (Drucken, Konvertieren, Kopieren, etc.)? Dann prüfen Sie, ob überhaupt eine Datei existiert: `exist(„#DOCNAME0“) oder exist(„#(DTY:Container0“)`
- Ist der Job für eine bestimmte Applikation bzw. für einen oder mehrere Dokumenttypen bestimmt? Überprüfen Sie folgendes:  
 „Bedingung für Anlage“ `wcomp(„#(FILE_TYPE).“,“.AX.PX.“)`  
 „Bedingung für Aktivierung/Ausführung“ `exist(„C:\programme\AutoDesk\acad.exe“)`
- Ist ein Job nur für einen bestimmten Benutzer bzw. Gruppe oder Status bestimmt? Dann:  
`comp(„Müller“,“$USERID“)`  
`wcomp(„Administrators“,“$ALL_GROUPS“)`  
`comp(„00005“,“$(STATUSKEY“)`
- Muss ein bestimmtes Feld belegt sein, damit der Job korrekt durchläuft (z. B. das Format einer Zeichnung)? Prüfen Sie dies ab:  
`lempty(„#FORMAT“)`

Vermeiden Sie Fehler indem Sie Bedingungen definieren!

## Fehlerbehandlung

Was tun wenn trotzdem Fehler auftreten? Überlegen Sie sich zunächst, was sie vor der eigentlichen Ausführung des Jobs erledigt haben. Vielleicht haben Sie Dateien kopiert, Templates oder Skripte vorbereitet, oder Felder mit Werten belegt. Das alles sollte im Falle eines Fehlers rückgängig gemacht werden, da eine bereinigte Umgebung die Fehlersuche vereinfacht.

Um die Fehleranalyse zu erleichtern, ist es deswegen sinnvoll, eine Routine zu erstellen, welche die „Umgebung“ in ein eigenes Verzeichnis kopiert, damit das Arbeitsumfeld sauber bleibt, jedoch die „Beweise“ gesichert sind.

## Meldungen im Fehlerfall verschicken

Um zu erkennen, ob der ausgeführte Job problemlos durchgelaufen ist, müsste man nur in die Liste der erledigten Jobs sehen. Dies wird jedoch meist nicht geschehen. Daher ist es sinnvoll, wenn im Fehlerfall ein bestimmter Benutzer eine Meldung erhält. Beispielsweise kann man über Net Send dem Administrator oder dem Benutzer, der den Job ausgelöst hat, eine Nachricht zukommen lassen. Dazu reicht es aus, in der Zeile „Befehl bei Fehler“ zum Beispiel folgendes einzutragen:

```
Net send Administrator „Ein Plot-Job ist fehlgeschlagen“
```

Noch besser ist es, wenn in der Nachricht auch die Informationen zum Job stehen. Hierzu muss diese Zeile bei der Anlage mittels `JBSSubstJobField(CMD_ON_ERROR)` substituiert werden, damit die aktuellen Daten verwendet werden.

```
net send $COMPUTERNAME "Job ‚$JBSIdent‘ ist fehlgeschlagen"
```

Damit wird dem Arbeitsplatz, von dem der Job ausgelöst wurde, eine Nachricht mit der Job-ID zugeschickt.

## Ein sauberer Datenbestand

Sie möchten CAD-Daten drucken, jedoch hat der Kunde nicht alle Dokumente auf die neueste Version aktualisiert? Das kann zu Problemen führen. Die Applikation gibt beim Öffnen dieser Dokumente Meldungen aus, die vom Anwender bestätigt werden müssen. Daraufhin bleibt der Jobserver stehen. Es muss dafür gesorgt werden, dass der Productstream Compass Job Server reibungslos durchlaufen kann. Wenn also bereits bekannt ist, dass Dokumente ungültig sind, oder erst aktualisiert werden müssen, damit sie korrekt geöffnet werden können, muss man zunächst den Datenbestand aktualisieren bzw. vermeiden, dass solche Dokumente zum Jobserver gelangen.

Um zu vermeiden dass solche Dokumente zum Jobserver gelangen, kann wieder mittels Bedingungen gearbeitet werden. Beispielsweise durch eine Bedingung, die nur die Übergabe von Dokumenten, welche jünger sind als ein bestimmtes Datum, zulässt. Ist dies nicht möglich, so muss dem Kunden die Situation geschildert und auf die vorauszusehenden Fehlern hingewiesen werden. Durch klare Arbeitsrichtlinien können Probleme vermieden werden.

## Mehrere Productstream Compass Job Server im selben Umfeld

In größeren Projekten kann es sein, dass ein einziger Productstream Compass Job Server nicht ausreicht, um mit der Arbeitslast fertig zu werden. In diesem Fall können mehrere Productstream Compass Job Server parallel betrieben werden. Dabei ist auf folgendes zu achten:

- Die Productstream Compass Job Server müssen auf getrennten PCs laufen. Ansonsten kann es zu Konflikten zwischen den Productstream Compass Job Servern kommen, bzw. die Applikationen könnten von beiden Productstream Compass Job Servern gleichzeitig in Anspruch genommen werden.
- Man sollte sich überlegen, welche Jobtypen auf welchem Productstream Compass Job Server ausgeführt werden sollen. Grundsätzlich verwaltet der Productstream Compass Job Server die Warteschleife eigenständig, damit könnten parallel betriebene Productstream Compass Job Server auch alle Jobtypen ausführen. Wenn jedoch gewünscht wird, dass nur bestimmte Jobtypen von einem der Productstream Compass Job Server ausgeführt werden, müssen entweder diese Jobtypen an den verschiedenen Productstream Compass Job Servern aktiviert bzw. deaktiviert, oder über die Bedingung „Bedingung für Ausführung“ gesteuert werden.  
Zum Beispiel könnte ein Vergleich mit dem Computernamen des jeweiligen Productstream Compass Job Server Abhilfe leisten:  
`comp("PC-JOBSRV01", "$COMPUTERNAME")`

## Productstream Compass Job Server in einer replizierten Umgebung

Wird der Productstream Compass Job Server in einer replizierten Umgebung betrieben, so muss beachtet werden, dass die Warteschleife ebenfalls auf allen Standorten repliziert wird. Das heißt, dass -egal auf welchem Standort ein Job ausgelöst wird- dieser zentral verwaltet und von allen Productstream Compass Job Servern erkannt wird. Jetzt muss überlegt werden, ob der an einem Standort ausgelöst Job auch auf dem selben Standort ausgeführt werden soll. Es könnte ja auch erwünscht sein, dass ein Job standortübergreifend ausgeführt wird.  
Empfehlungen:

- Will man sicher gehen, dass ein Job immer auf dem selben Standort ausgeführt wird, so kann man über die Bedingung für die Ausführung eine Standortprüfung einbauen:  
`comp("SO1", "$LOCATION")`
- Will man einen Job auf einem Standort auslösen, die Abarbeitung soll jedoch auf einem anderen Standort erfolgen, so muss man prüfen, ob wegen der Replikation auch alle Informationen und Dokumente zur Ausführung dieses Jobs am anderen Standort verfügbar sind.
- Möchte man verhindern, dass auf allen Standorten die Jobs aus allen anderen Standorten sichtbar sind, so kann man das über die flexible Rechteverwaltung von Productstream Compass Pro steuern. Hierzu müssen verschiedene Productstream Compass Job Server-Administratorengruppen für die unterschiedlichen Standorte eingerichtet werden. An jedem Standort muss in der Konfiguration die jeweilige Gruppe entsprechend eingetragen werden. Das Leserecht muss der Welt-Gruppe entzogen werden.

**Hinweis: In einer replizierten Umgebung sollte an jedem Standort ein Productstream Compass Job Server vorhanden sein. Werden an einzelnen Standorten die Dienste des Productstream Compass Job Server nicht benötigt, so müssen diese Standorte keinen Productstream Compass Job Server besitzen. Der Productstream Compass Job Server ist ein Werkzeug mit dem Prozesse optimiert und beschleunigt werden. Damit muss an jenen Standorten wo solche Prozessverbesserungen stattfinden sollen auch ein dedizierter Productstream Compass Job Server vorhanden sein. Es ergibt keinen Sinn, im replizierten Umfeld den Productstream Compass Job Server zentral für alle Standorte zu nutzen. Die Zeit die für den Transfer der Dokumente und die Abarbeitung der Jobs über die Distanz der Standorte verwendet wird, macht den Nutzen des Productstream Compass Job Server zunichte!**

## Installation

### Lizenzierung

Productstream Compass Job Server wird automatisch mit jedem Productstream Compass installiert, bedarf jedoch eines eigenen Lizenzschlüssels. Der Lizenzschlüssel für den Productstream Compass Job Server muss separat erworben werden und wird dann mittels Netzwerk Lizenzmanager in das System eingegeben.

Nachdem der Lizenzschlüssel eingegeben und Productstream Compass neu gestartet wurde, erscheint im Productstream Compass - Hauptmenü, ein neuer Menüeintrag „Job auslösen“. Zudem ist es jetzt auch möglich, Elemente auf den Productstream Compass Job Server-Ordner zu ziehen (Drag&Drop) ohne dass die Meldung „Kein Productstream Compass Job Server installiert! Funktion abgebrochen!“ erscheint.

### Installation des Client

An demjenigen Arbeitsplatz wo der Productstream Compass Job Server (der Spooler) arbeiten wird, muss eine Productstream Compass-Client-Installation erfolgen. Dabei muss während des Setup die Option für den Productstream Compass Job Server ausgewählt werden. Der Productstream Compass Client wird lokal installiert, und eine Verknüpfung erstellt. Diese Verknüpfung kann noch konfiguriert werden. Dazu mehr im nächsten Abschnitt.

### Parameter der Productstream Compass Job Server- Verknüpfung

Damit der Productstream Compass-Client als Productstream Compass Job Server startet, muss an der Programmverknüpfung in der „Ziel“-Zeile der Parameter „-jobserver“ folgen. Dieser wird dann vom Setup automatisch hinzugefügt. Weitere Varianten oder zusätzliche Parameter werden in der Folgenden Tabelle aufgezählt.

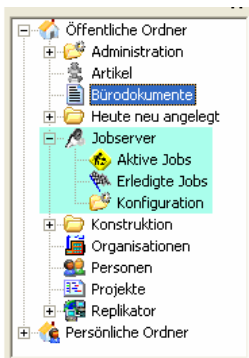
Parameter	Bedeutung
-jobserver	Lässt Productstream Compass im Productstream Compass Job Server-Modus starten. Hierbei startet die gewohnte Productstream Compass- Oberfläche, jedoch wird die Lizenz des Productstream Compass Job Server verwendet. Um den Productstream Compass Job Server ( <i>Spooler</i> ) zu starten, muss im Ordner „Jobserver/Konfiguration“, im Werkzeugkasten „Jobserver starten“ gewählt werden.
-jobserver=start	Lässt Productstream Compass im Productstream Compass Job Server-Modus starten. Hierbei wird jedoch sofort der Productstream Compass Job Server gestartet und es erscheint der Spooler. Nach 10 Sekunden Wartezeit, startet der Productstream Compass Job Server mit der Abarbeitung der Jobs. Diese Einstellung wird für den automatischen Betrieb des Productstream Compass Job Server auf dem dedizierten Arbeitsplatz empfohlen.

## Die Komponenten

Der Productstream Compass Job Server besteht aus verschiedenen Komponenten. Einige davon sind im Standard-Productstream Compass-Client sichtbar, andere nur am Productstream Compass Job Server Arbeitsplatz. Grundsätzlich wird zwischen der Funktionalität, welche dem Anwender zur Verfügung steht und jener welche am Server ausgeführt wird, Unterschieden. Im Folgenden werden die Komponenten einzeln erläutert.

### Die Productstream Compass Job Server- Ordner

Auf der Standard-Productstream Compass-Oberfläche liegen einige Ordner für den Productstream Compass Job Server bereit. Diese Ordner sind im öffentlichen Ordnerbereich in der Sektion „Jobserver“ zu finden. Hier die Bedeutung der verschiedenen Ordner:



- Der Ordner „Aktive Jobs“. Hier finden Sie die vom Anwender ausgelösten Jobs, welche vom Productstream Compass Job Server noch nicht ausgeführt wurden. In diesem Ordner bleiben die Jobs solange, bis sie abgearbeitet wurden. Über diesen Ordner kann der Status der „Warteschlange“ beobachtet werden.
- Der Ordner „Erledigte Jobs“. Hier finden Sie alle Jobs die bereits vom Productstream Compass Job Server abgearbeitet wurden. Über dessen Status kann man erkennen ob die Ausführung erfolgreich war oder fehlgeschlagen ist. Für jeden Job ist in der Registerkarte „Job Verlauf“ der Ausführungsstatus zu sehen. Die erledigten Jobs müssen von Hand aus dem Ordner entfernt werden.
- Der Ordner „Konfiguration“. Dieser Ordner ist nur für jene Benutzer sichtbar, welche der Benutzergruppe JOBSERVER zugeordnet sind. In diesem Ordner werden die Jobtypen definiert.

Der Anwender kann einen Job auf zwei unterschiedliche Arten auslösen. Dazu muss er zunächst eines oder mehrere Elemente markieren und dann:

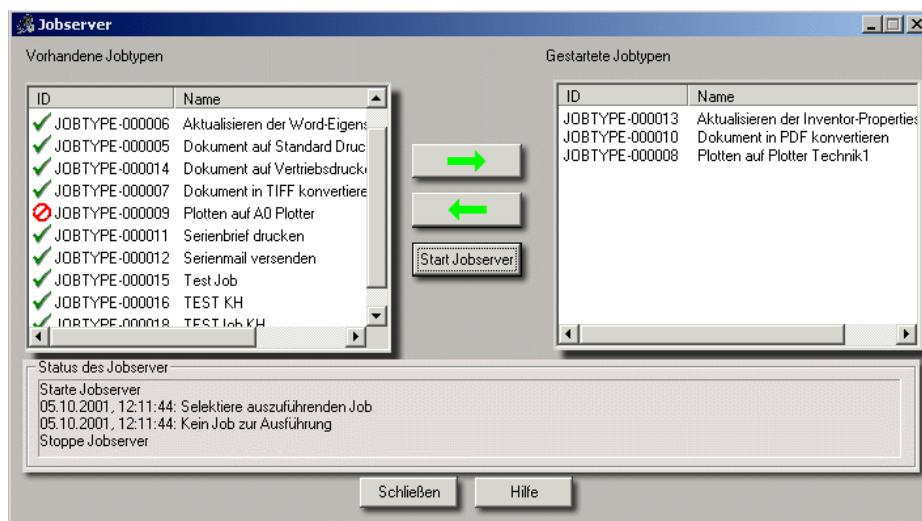
- Entweder: über das Menü Compass ->Job auslösen den gewünschten Job auslösen  
Oder: per Drag&Drop die markierten Elemente auf den Ordner „aktive Jobs“ ziehen und fallen lassen.

## Der Jobspooler

Der Jobspooler ist die Oberfläche des Productstream Compass Job Server auf dem Server. Über diese Oberfläche kann man den Productstream Compass Job Server starten oder anhalten. Hier wird auch gesteuert, welche Jobtypen auf einem bestimmten Productstream Compass Job Server ausgeführt werden sollen. Das kann zum Beispiel notwendig werden, wenn mehrere Productstream Compass Job Server im selben Unternehmen im Einsatz sind, um die Arbeitslast zu verteilen, oder bestimmte Jobtypen nicht ausgeführt werden sollen.

Um die in der Warteschlange („Aktive Jobs“) vorhandenen Jobs abzuarbeiten, muss der Jobspooler aktiviert werden. Beim Starten des Jobspoolers verschwindet die Productstream Compass-Oberfläche und der Spooler erscheint. Dies geschieht automatisch wenn man den Productstream Compass Job Server mit dem Argument –jobserver=start startet.



Die Oberfläche des Spoolers ist dreigeteilt. Im oberen Bereich sind links und rechts zwei Listen definiert. Die linke Liste „Vorhandene Jobtypen“ beschreibt die für diesen Arbeitsplatz verfügbaren Jobtypen. Auf der rechten Seite ist die Liste der „Gestarteten Jobtypen“ zu finden. Im unteren Bereich ist der „Status des Spooler“ definiert. Die drei Bereiche haben folgende Funktionen:



- **Vorhandene Jobtypen.** Alle im Ordner „Konfiguration“ definierten Jobtypen sind in dieser Liste aufgeführt. Ein Status-Icon beschreibt, für welche Jobtypen die Ausführung erlaubt ist. Das Feld „Bedingung für Aktivierung“ am jeweiligen Jobtyp, enthält die Bedingung welche für diesen Zeck ausgewertet wird. Für jeden definierten Jobtyp wird das Feld ausgewertet und je nach Rückgabewert (TRUE oder FALSE) wird das entsprechende Icon dargestellt. Werden neue Jobtypen definiert, so muss der Spooler nochmals gestartet werden, damit dieser die neuen Jobtypen erfassen kann.
- **Gestartete Jobtypen.** In dieser Liste befinden sich die für diesen Spooler auszuführenden Jobtypen. Sie wird mit den entsprechenden Buttons (Pfeil nach rechts, Pfeil nach links) erweitert oder verkürzt. Dabei werden die Elemente aus der Liste „Vorhandene Jobtypen“ zur Liste „Gestartete Jobtypen“ übertragen und umgekehrt. Nur vorhandene Jobtypen mit dem grünen Haken können zur Liste der gestarteten Jobtypen hinzugefügt werden.
- **Status des Spooler.** In diesem Fenster wird der Status des aktuell ausgeführten Job angezeigt. Das selbe Protokoll wird auch im Feld „Job Verlauf“ beim jeweiligen Job mitgeführt. Damit steht zu jedem Job auch ein Ausführungsprotokoll zur Verfügung.

Der Spooler arbeitet die Warteschlange (Ordner „Aktive Jobs“) geordnet nach Priorität und Jobnummer (IDENT) ab. Kann ein Job derzeit nicht ausgeführt werden, weil zum Beispiel seine „Bedingung für Ausführung“ nicht TRUE ist, so wird er übersprungen. Nach jeder Ausführung eines Jobs wird die Liste der abzuarbeitenden Jobs neu aufgebaut.

## So starten Sie den Jobserver

1. Wählen sie den Ordner „Jobserver Konfiguration“
2. Klicken Sie zum Starten des Jobserver das entsprechende Icon aus dem Werkzeugkasten an. Die Productstream Compass Oberfläche verschwindet und statt dessen erscheint der Spooler.
3. Wählen Sie aus der linken Liste des Spoolers die von diesem Spooler auszuführenden Jobtypen. Verwenden Sie dazu den Button () mit dem Pfeil nach rechts um die gewünschten Jobtypen zur Liste der auszuführenden Jobtypen hinzuzufügen.
4. Dann kann mit dem Button „Start Jobserver“ () der Jobserver gestartet werden.

**Hinweis!** Wenn der Productstream Compass Job Server über den Parameter `-jobserver=start` gestartet wurde, so sind die obigen Schritte nicht notwendig. Der Productstream Compass Job Server arbeitet dann automatisch alle Jobtypen ab, die als „Bedingung für Ausführung“ TRUE haben. Der Productstream Compass Job Server startet automatisch nach 10 Sekunden mit der Ausführung der Jobs.

## Der „Watchdog“ überwacht die Jobs

Neben dem sichtbaren Spooler existiert auch ein so genannter „Watchdog“; unsichtbar für den Anwender aber notwendig für den korrekten Verlauf der Jobs. Der Watchdog prüft, ob die Jobs in der angegebenen Zeit abgearbeitet werden (Feld Timeout am Jobtyp). Dauert die Ausführung eines Jobs länger als die angegebene maximale Laufzeit, so tritt er in Aktion.

Beim Start jedes Jobs, teilt der Spooler dem Watchdog mit, welches Timeout für diesen Job definiert ist. Die Aufgabe des Watchdog ist es nun, dafür zu sorgen, dass der stehen gebliebene Job wieder zum Laufen kommt, um ihn dann als fehlgeschlagen zu beenden. Dazu wird versucht, die vom aktuellen Job zuletzt geöffnete Applikation zu terminieren. Gelingt das, so kann der Job weiterlaufen, wird jedoch sofort als fehlerhaft beendet. Der Watchdog hat also eine Überwachungs- und Bereinigungsfunktion. Kann trotz aller Versuche der Watchdog den Job nicht zum Laufen bringen, startet er eine neue Productstream Compass-Session und beendet das stehen gebliebene Productstream Compass. Die neue Productstream Compass-Session beginnt dann mit der Abarbeitung des nächsten Jobs.

Wird eine Applikation über Productstream Compass mittels der Befehle `___Shell(...)` oder `___Shell(shellexecute(...))` gestartet, so merkt sich Productstream Compass einen so genannten Prozesshandle, und die Variable `STARTED_PROCESSNAME` enthält den Prozessnamen (z.B.: `autocad.exe`). Ist der Prozesshandle bzw. Prozessname bekannt, so kann dieser im Notfall terminiert werden. Wird eine Applikation über `___Shell(system(...))` oder über einen `GetObject` Befehl aus einer CAI-DLL gestartet, so wird kein Prozesshandle und damit auch keine `STARTED_PROCESSNAME` Variable gesetzt. Der Watchdog kennt also die zuletzt gestartete Applikation nicht und kann sie deshalb nicht beenden.

**Hinweis:** Wir empfehlen, Applikationen im Jobserver-Umfeld immer über ein `___Shell(...)` bzw. `___Shell(shellexecute(...))` zu starten. Wenn dies nicht möglich ist, so ist zumindest das vorherige manuelle Setzen der Variable `STARTED_PROCESSNAME` mit dem Prozessnamen zu empfehlen.

Läuft der Timeout eines Jobs also ab, so wird der im Prozesshandle bekannte Prozess vom Watchdog terminiert. Es erfolgt eine Meldung in der Datei `Errlog.err`: *“WatchDog-Thread: last created Process (notepad.exe) not running any more (self-terminated or killed) !!”*  
Aus dieser Meldung ersieht man, wie der Watchdog die zuletzt gestartete Applikation (in diesem Falle `notepad.exe`) erkannt hat und -um den Job fortzusetzen- diese terminiert wurde.

Konnte der Prozesshandle nicht ermittelt werden und ist der Timeout abgelaufen, so meldet sich der Watchdog wieder in der `Errlog.err` und teilt mit, dass der Prozesshandle nicht erkennbar war und der Prozess nicht beendet werden kann. Jetzt wartet der Watchdog weitere 60 Sekunden. Sollte der Spooler dann noch immer nicht wieder mit dem Arbeiten begonnen haben (weil der Job noch blockiert), so werden alle Instanzen des Prozesses aus der Variable `STARTED_PROCESSNAME` beendet. Folgende Meldungen stehen dann in der `Errlog.err`:

```
WatchDog-Thread: one more minute over time and Job still not ready!!!
WatchDog-Thread: trying to kill all instances of notepad.exe (60)
Process "notepad.exe" successfully killed!
```






Konnten weder Prozesshandle noch Prozessname ermittelt werden, so weiß der Watchdog nicht, welche Applikation beendet werden soll. Er meldet sich regelmäßig alle 60 Sekunden in der `Errlog` mit der Fehlermeldung:

```
WatchDog-Thread: timeout, but no last created Process to kill (hProcess == 0)!!
WatchDog-Thread: one more minute over time and Job still not ready!!!
WatchDog-Thread: $STARTED_PROCESSNAME is empty! (60)
```

Um zu verhindern, dass der Jobserver jetzt blockiert bleibt, ohne dass dies bemerkt wird, kann eingestellt werden, dass eine Nachricht über Net Send an einen Arbeitsplatz geschickt wird. Damit wird ein Mitarbeiter auf das Fehlverhalten aufmerksam gemacht und kann die Ursache des Problems analysieren und eventuell beheben. Wie diese Nachricht konfiguriert werden kann, wird im Folgenden erläutert.

## Die Productstream Compass Job Server Statusverwaltung

Der Productstream Compass Job Server besitzt für die eigenen Jobs eine interne Statusverwaltung. Diese kann nicht von außen beeinflusst werden. Derzeit sind 4 Statuszustände definiert:

Status	Icon	Bedeutung
Angelegt (STATUSID=JOB01)		Job.ico Die Instanz des Jobs liegt in der Warteschlange. Die Instanz wurde noch nicht vom Spooler in Angriff genommen.
In Arbeit (STATUSID=JOB02)		Init.ico Der Job wurde vom Spooler für die Abarbeitung aufgenommen und reserviert. Die PreDo Methoden werden ausgeführt.
In Ausführung (STATUSID=JOB03)		Wip.ico Der Spooler arbeitet Methoden wie Do, Post u. ä. ab.
Erledigt (STATUSID=JOB04)		Ok.ico Der Job wurde erfolgreich abgearbeitet.
Fehler (STATUSID=JOB05)		Error.ico Während der Ausführung des Jobs ist ein Fehler aufgetreten.

## Bedeutung der Felder

Feldbezeichnung	Feldname	Format	Zwingend	Beschreibung	Beispiel
Job Nummer	IDENT	Text	Ja	Eindeutige Jobnummer. Wird automatisch von Productstream Compass generiert.	
Kategorie	CATEGORY	Text	Nein	Dient zur Gruppierung der Jobtypen. Feld und Inhalt stehen beliebig zur Verfügung	Dient nur zur optischen Gruppierung.
Bezeichnung	SHORT_DESC	Text	Ja	Bezeichnung des Jobtyps. Wird sowohl im Spoolerfenster als auch dem Anwender angezeigt und in den Job übernommen.	
Beschreibung	LONG_DESC	Text	Nein	Hier kann eine ausführlichere Beschreibung des Jobs eingegeben werden.	
Bedingung bei der Anlage	COND_CREATE_JOB	Bool	Ja	Dieses Feld wird bei der Übergabe von Elementen an den Jobserver ausgewertet. Ist der Ausdruck TRUE so steht dieser Jobtyp für das übergebene Element zur Verfügung	isEType(„AIM.DOC.OFF“) Hiermit steht der Jobtyp nur für Bürodokumente zur Verfügung. Hier sind Productstream Compass-Bedingungs- und logische Operatoren erlaubt.
Befehl bei der Anlage	CMD_CREATE_JOB	Cmd	Nein	Hier werden Aktionen definiert, welche unmittelbar bei der Anlage des Jobs stattfinden sollen.	Über einen Eingabedialog soll der Anwender noch über Papierformat und Drucker abgefragt werden. Das kann in einer eigenen DBP-Prozedur programmiert werden.
Bedingung für Aktivierung	COND_EXEC_JOBTYPE	Cmd	Ja	Hiermit wird festgelegt ob der Jobtyp für den Spooler zur Verfügung steht. Der Rückgabewert ist TRUE oder FALSE.	exist(„C:\Programme\AutoCAD\acad.exe“) Damit ist ein installiertes AutoCAD Voraussetzung für diesen Jobtyp.
Job Timeout	JOB_TIMEOUT	Num	Ja	Hiermit wird die maximale Laufzeit des Jobs definiert. Ein Job muss innerhalb der hier definierten Zeit fertig werden. Die Angabe erfolgt in Minuten.	5 Damit ist definiert, dass der Job maximal 5 Min. Zeit hat um durchzulaufen, andernfalls wird er terminiert.
Bedingung für Ausführung	COND_START_JOB	Bool	Ja	Ist die hier definierte Bedingung TRUE, so wird der Job ausgeführt. Andernfalls bleibt er in der Warteschleife.	TaskRunning(„acad.exe“) Wenn die Applikation AutoCad läuft wird der Job ausgeführt.



Befehl vor Ausführen	CMD_PRE_JOB	Cmd	Nein	Befehl vor dem Ausführen des Jobs. Kann dazu verwendet werden, Vorbereitungen zu erledigen	___Prepare(„c:\myTemplate.txt“ „test.lsp“) Hiermit wird eine Datei substituiert.
Befehl für Ausführung	CMD_DO_JOB	Cmd	Nein	Productstream Compass-Befehl der den Job ausführt	MyJobProcedure Das könnte der Name einer selber programmierten Funktion sein.
Befehl nach Ausführung	CMD_POST_JOB	Cmd	Nein	Befehl nach dem Ausführen des Jobs. Kann dazu verwendet werden, Aufräumarbeiten zu erledigen.	rm(„test.lsp“) Hiermit wird die angegebene Datei gelöscht
Befehl bei Erfolg	CMD_ON_OK	Cmd	Nein	Der hier beschriebene Befehl wird bei Erfolg des Jobs ausgeführt. Er kann dazu verwendet werden, dem Anwender eine Nachricht zu schicken.	Net send \$computername „Job Succesfull terminated“ Hiermit wird eine Nachricht per Net Send verschickt
Befehl bei Fehler	CMD_ON_ERROR	Cmd	Nein	Im Falle eines Fehlers wird diese Funktion ausgeführt.	Net send \$computername„Job ERROR“ Nachricht wird per Net Send verschickt

## Konfiguration und Administration

### Benutzer, Gruppen und Rechte

Für den Productstream Compass Job Server wurde eine eigene Benutzergruppe eingerichtet: JOBSERVER. Die Benutzergruppe JOBSERVER erteilt Berechtigungen für die Administration des Productstream Compass Job Server. Die Standard-Administratorenberechtigungen reichen für die Administration des Productstream Compass Job Server nicht aus. Der Ordner „Konfiguration“ ist ebenfalls der Gruppe JOBSERVER zugeordnet und damit nur für jene Benutzer sichtbar, die dieser Gruppe angehören.

Die Arbeitsgruppe (DefaultGroup) JOBSERVER ist Jobtypen und Jobs als Standardgruppe zugeordnet. Damit können Jobtypen grundsätzlich nur von Benutzern angelegt, bearbeitet oder gelöscht werden, die Mitglieder dieser Gruppe sind. Vom Anwender ausgelöste Jobs sind also dem Auslöser und der Gruppe JOBSERVER zugeordnet. Andere Benutzer können hier ausgelöste Jobs nicht beeinflussen, da nur der Besitzer und die Mitglieder der Gruppe JOBSERVER diese Rechte haben. Jobs können jedoch grundsätzlich von jedem Benutzer ausgelöst werden.

In der Standardauslieferung ist auch ein Benutzer „Productstream Compass Job Server“ angelegt, welcher der Gruppe JOBSERVER zugeordnet ist. Dieser dient als Beispiel oder kann als Productstream Compass Job Server-Benutzer verwendet werden.

Vergewissern Sie sich, dass der Benutzer, der mit dem Productstream Compass Job Server arbeiten wird, auch allen benötigten Benutzergruppen zugeordnet ist. Andernfalls kann der Productstream Compass Job Server eventuell Operationen nicht ausführen oder auf Informationen nicht zugreifen.

### Einen neuen Jobtyp definieren

Jobtypen definieren die Art der Aufgabe die abgewickelt werden soll. Am Jobtyp werden die Funktionen hinterlegt, welche zur Abarbeitung des Jobs benötigt werden. Dies können DBP-Prozeduren oder DLL-Funktionen sein.

### So legen Sie einen neuen Jobtyp an

1. Wählen sie den Ordner **Jobserver Konfiguration**
2. Klicken Sie mit der rechten Maustaste in die Listendarstellung im Übersichtsbereich. Es erscheint ein Kontextmenü.
3. Wählen Sie **Neuer Jobtyp**. Es erscheint das Dialogfenster zur Neuanlage eines Jobtyps.
4. Geben Sie in die Eingabefelder die Angabe zum Jobtyp ein. (Die Bedeutung der einzelnen Felder ist unten beschrieben)
5. Wenn Sie einen weiteren Jobtyp anlegen wollen, wählen Sie **Datei > Speichern und neuer Jobtyp** oder das entsprechende Symbol. Wenn Sie keine weiteren Jobtypen anlegen wollen, wählen Sie **Datei > Speichern und Schließen** oder das entsprechende Symbol. Der neue Jobtyp wird nun angelegt.

## Arbeitsweise des Productstream Compass Job Server

### Ein Job wird durch den Anwender ausgelöst

Löst der Anwender einen Job aus, werden folgende Arbeitsschritte ausgeführt:

1. Aus der Liste der definierten Jobtypen, welche zum betroffenen Element passen, wird der Jobtyp ermittelt. Dies geschieht durch das Auswerten des Feldes „Bedingung bei der Anlage“. Die in diesem Feld definierten Bedingungen beziehen sich auf das übergebene Element. Liefert die Bedingung ein TRUE, so wird dieser Jobtyp in die Liste aufgenommen. Ist sie fertig gestellt, wird die Liste dem Anwender präsentiert.
2. Der Anwender wählt den gewünschten Jobtyp aus. Dabei wird die im Feld „Befehl bei Anlage“ definierte Funktion ausgeführt. Die hier definierte Funktion dient dazu, eventuell nötige Vorbereitungen zu treffen oder den Anwender nach weiteren Informationen zu fragen. Der Kontext dieser Funktion ist der neue Job selbst. Diese Funktion wird am jeweiligen Client Arbeitsplatz ausgeführt, damit stehen ihr Variablen wie Benutzerinformationen zur Verfügung. Dies ist der richtige Zeitpunkt um z.B. bestimmte Dateien zu kopieren oder spezielle Vorkehrungen zu treffen bevor der Job angelegt wird. Der Job ist jetzt noch nicht in die Datenbank eingefügt worden, also kann man am `__CURRENT__` RecordBuffer noch Änderungen durchführen.
3. Jetzt wird der Job in der Warteschlange angelegt.

### Ein Job wird durch den Spooler ausgeführt

Damit ein Job ausgeführt werden kann, muss die im Feld „Bedingung für Ausführung“ definierte Bedingung einen gültigen Rückgabewert (TRUE) ergeben.

Läuft der Spooler und kommt ein Job zum Zuge so wird folgende Reihenfolge bei der Ausführung der Funktionen eingehalten:

1. Die Funktion „Befehle vor Ausführung“ (Feld: `CMD_PRE_JOB`) wird ausgeführt. Ist diese erfolgreich wird die
2. Funktion „Befehle für Ausführung“ (Feld: `CMD_DO_JOB`) ausgeführt. Ist diese erfolgreich wird die
3. Funktion „Befehle nach Ausführung“ (Feld: `CMD_POST_JOB`) ausgeführt. Ist diese erfolgreich wird die
4. Funktion „Befehle bei Erfolg“ (Feld: `CMD_ON_OK`) ausgeführt

Treten in der oben genannten Reihenfolge Fehler auf, so wird die Abarbeitung des Jobs unterbrochen und sofort die Funktion „Befehle bei Fehler“ (Feld: `CMD_ON_ERROR`) ausgeführt.

### Arbeitskontext der Funktionen

Alle am Job definierten Funktionen, mit Ausnahme des „Befehl bei Anlage“, werden am verlinkten Element ausgeführt. Bei der Anlage eines Jobs durch den Anwender wird am erzeugten Job der Folder und der AIMKEY des „Gast“-Elementes gespeichert (in den Feldern `ELEMENT_FOLDER` und `ELEMENT_AIMKEY`). Das bedeutet, dass die zur Ausführung des Jobs notwendigen Funktionen am Element (oder seinem Parent in der EntityType Struktur) definiert sein müssen. Hierbei gilt die Standard-Productstream Compass-Programmierschnittstelle (DBP, VB, C/C++). Die Funktionen können in einem eigenen Modul gekapselt sein.

Zu jedem Funktionsfeld können mehrere Funktionen angegeben werden. Diese müssen entweder durch Semikolon (;) oder durch Zeilenumbruch (`{Strg}+[Enter]`) getrennt werden.

Die Bedingung "Bedingung bei der Anlage" wird im Kontext des "Gast" Elementes ausgewertet. Die Bedingungen "Bedingung für Aktivierung" und "Bedingung für Ausführung" werden im Kontext des Jobs selber ausgewertet.

## Neustart, "wenn nichts mehr geht"

Bleibt der Productstream Compass Job Server stehen und kann der Watchdog nicht feststellen, welche Applikation dafür verantwortlich ist, so wird ein Neustart des Productstream Compass Job Server eingeleitet. Sobald der neue Productstream Compass Job Server läuft, terminiert der Watchdog den stehen gebliebenen Productstream Compass Job Server. Damit wird gewährleistet, dass die anstehenden Jobs weiter ausgeführt werden.

Beim Neustart wird auch eine Nachricht -sofern konfiguriert- verschickt, um den Vorfall zu melden. Mit großer Wahrscheinlichkeit befindet sich der Job, welcher für den Stillstand verantwortlich war, noch in der Warteschleife (Status: „in Ausführung“).

## Konfiguration

Rund um den Productstream Compass Job Server gibt es eine Reihe von Schaltern, welche das Verhalten des Productstream Compass Job Server beeinflussen. In diesem Kapitel wollen wir auf die einzelnen Schalter näher eingehen und Empfehlungen für die Benutzung geben.

### Jobserver-Intervall

Der Productstream Compass Job Server besitzt ein konfigurierbares Zeitintervall. Solange der Spooler keine Jobs auszuführen hat (die Warteschlange ist leer), wird dieses Zeitintervall abgewartet. Damit wird verhindert, dass der Jobserver kontinuierlich auf der Datenbank nach auszuführenden Jobs sucht und damit unnötigen Netz- und Datenbankverkehr produziert.

Sobald ein Job gefunden wird, beginnt der Spooler mit der Abarbeitung dieses Jobs. Ist der Job durchgelaufen, wird geprüft, ob weitere Jobs in der Warteschlange vorhanden sind. Liegt ein weiterer Job vor, wird dieser nach 10% des Zeitintervalls ausgeführt. Dieses kurze Zeitintervall ermöglicht es dem Rechner, die Ressourcen wieder frei zu geben, bevor der nächste Job zum Zuge kommt. Ist die Warteschlange leer, wird auf „idle“ (Schlafmodus) geschaltet und die Intervall-Zeit abgewartet, bevor nach neuen Jobs gesucht wird.

Somit wird bei leerer Warteschlange der nächste Job erst nach Ablauf des Jobserver-Intervalls ausgeführt. Trotzdem ist es sinnvoll, dieses Intervall größer/gleich 60 Sekunden zu setzen. Andernfalls wird unnötiger Verkehr auf der Datenbank produziert, welcher Performance-Verluste im sonstigen Productstream Compass-Betrieb verursacht.

Das Intervall kann in der COMPASS.INI konfiguriert werden. Der Schalter `JobserverInterval` ist in der Sektion `[PATH]` zu finden (defaultwert 60 Sek.). Der angegebene Wert ist in Sekunden angegeben.

Beispiel: `JobserverInterval=90` bedeutet 1,5 Minuten

## Nachrichten im Fehlerfall

Tritt der Fall auf, dass der Jobserver (wie oben beschrieben) stehen bleibt, und der Watchdog es nicht schafft, die Ursache des Problems zu beheben, so bleibt der Jobserver stehen bis ihm ein Mitarbeiter weiter hilft. Aus diesem Grund ist es möglich, den Jobserver so zu konfigurieren, dass dieser nach 90 Sekunden Stillstand eine Meldung (Net Send) an eine/n bestimmten Rechner, Benutzer, Gruppe oder Domäne verschickt. Dafür stehen zwei Variable zur Verfügung. Die Variable `JobserverAlertUser` definiert, an welche Arbeitsstationen, Benutzer, Gruppen oder Domänen die Meldung verschickt werden soll. Während die Variable `JobserverAlertInfo` die Meldung enthält. Diese Variablen können in der COMPASS.INI einmalig gesetzt oder auch zur Laufzeit am Job definiert werden.

Wurden diese beiden Variablen definiert, wird nach 90 Sekunden Stillstand über Net Send dem gewünschten Gremium eine Meldung geschickt. Werden die Variablen zu Beginn des Jobs (z. B. im Feld „Befehl vor Ausführung“) gesetzt, so können diese für jeden Job unterschiedlich definiert werden. Damit könnte man Fehlermeldungen abhängig vom Job mit unterschiedlichem Text an verschiedene Benutzer übermitteln.

Die Nachricht wird erstmals nach 90 Sekunden Stillstand (nach dem Timeout des Jobservers) geschickt und dann ca. alle 30 min. wiederholt.

In der COMPASS.INI

`[PATH]`

....

`JobserverAlertUser=Administrator Maier Müller`

`JobserverAlertInfo=Jobserver ist stehen geblieben. Bitte kontrollieren!`

Am Job im „Befehl vor Ausführung“

`set JobserverAlertUser=Administrator Maier Müller;`

`set JobserverAlertInfo=Jobserver ist stehen geblieben Bitte kontrollieren!;`

## Log-Möglichkeiten zur Fehleranalyse

bleibt der Jobserver öfter stehen und sind die Ursachen unbekannt oder schwierig zu ermitteln, benötigt man genauere Informationen zum Geschehen. Aus diesem Grund wurde eine erweiterte Protokollierungsmöglichkeit eingebaut. Über die Variable `_AIM_TEST_=2048` (zu setzen in COMPASS.INI oder während der Laufzeit) kann eine Protokollierung des Geschehens erreicht werden. Ist dieser Schalter aktiv, so wird im Work-Verzeichnis des Productstream Compass Job Server eine Datei `Jobserver.log` erstellt. In dieser Datei wird der Beginn und das Ende eines jeden Jobs mitgeschrieben:

```
Start Job: JobIDENT, Beschreibung, errno=0; WatchDog: Running=1 Ende=0 Idle=1, Watching=0 Timeout=1200
Set Timeout=60, errno=0; WatchDog: Running=1 Ende=0 Idle=1, Watching=0 Timeout=60
Finished Job: JobIDENT, Beschreibung WatchDog: Running=1 Ende=0 Idle=1, Watching=0 Timeout=51
```

Damit kann man erkennen, welche Jobs problemlos durchgelaufen sind und welche Probleme verursacht haben.

Tritt ein Problem auf, so werden in der selben Datei auch die Meldungen des Watchdog, welche sonst in der Erlog.err enthalten sind, mitgeschrieben. Zusätzlich werden die letzten 5 Befehle des Jobs eingetragen. Der letzte hier stehende Befehl, muss derjenige sein, welcher die Blockade verursacht hat. Beispiel:

```
Start Job: JobIDENT, Beschreibung, errno=0; WatchDog: Running=1 Ende=0 Idle=1, Watching=0 Timeout=51
Set Timeout=60, errno=0; WatchDog: Running=1 Ende=0 Idle=1, Watching=0 Timeout=60
2. Command: set JobserverAlertUser=server; , errno=0; WatchDog: Running=1 Ende=0 Idle=0, Watching=1 Timeout=0
1. Command: set JobserverAlertInfo=Meine Meldung; , errno=0; WatchDog: Running=1 Ende=0 Idle=0, Watching=1 Timeout=0
0. Command: system(notepad.exe) , errno=0; WatchDog: Running=1 Ende=0 Idle=0, Watching=1 Timeout=0
Kill LCP "???"-> retval=-2, errno=0; WatchDog: Running=1 Ende=0 Idle=1, Watching=1 Timeout=0
WatchDog-Thread: timeout, but no last created Process to kill (hProcess == 0)!!, errno=0; WatchDog: Running=1 Ende=0 Idle=1, Watching=1 Timeout=0
Finished Job: JobIDENT, Beschreibung (86815 msec), errno=0; WatchDog: Running=1 Ende=0 Idle=1, Watching=0 Timeout=0
```

Wie man aus diesem Beispiel sehen kann, ist der Jobserver beim Befehl `system(...)` stehen geblieben. Vorher hat er zwei Befehle erfolgreich ausgeführt.

## Automatischer Neustart des Productstream Compass Job Server

Um den Productstream Compass Job Server automatisch neu zu initialisieren, kann ein programmierter Neustart des Productstream Compass Job Server eingerichtet werden. Dazu wird in der COMPASS.INI über `JOBS_RESTART_JOBSEVER` (=Zeit in Minuten) ein Zeitintervall eingestellt. Der automatische Neustart ist nur dann erfolgreich, wenn (in der Programmverknüpfung) `-jobserver=start` eingestellt ist. Ist dies nicht der Fall, so wird bereits beim Start des Jobserver eine Fehlermeldung ausgegeben.

Ab Version COMPASS 2005 muss für den automatischen Neustart kein Autologin mehr gesetzt sein. Productstream Compass merkt sich beim Login die Benutzerinformationen und wiederholt im Falle eines Neustarts selbständig das Login mit diesen Informationen.

## Job bei Statuswechsel auslösen

Möchte man, dass beim Statuswechseln automatisch ein Job erzeugt wird, so kann man das in der Productstream Compass Konfiguration einstellen. Am entsprechenden EntityType (z.B. AIM.DOC.ENG für die Konstruktionsdokumente) können Attribute mit dem Namen `OnStatusXXXXX` definiert werden. Hier gilt XXXXXX für die Statuskennung an der der Job ausgelöst werden soll.

Man kann also zum Beispiel bei der Freigabe automatisch einen Aktualisierungsjob auslösen. Dazu nutzt man den Befehl `___JobCreate`. Siehe hierzu auch die Entwicklerdokumentation für `___JobCreate` und den Abschnitt „Zusätzliche Aktionen konfigurieren“ im Kapitel „Statusverwaltung – Prinzipielle Abläufe“

## Beispiele

### Schriftfeldupdate

#### Einleitung:

Bei der Aktualisierung der Eigenschaften von Dokumenten ist es wichtig, dass diese an der Originaldatei vorgenommen wird, sonst gehen die Änderungen verloren. Dabei muss die vom Anwender dem Productstream Compass Job Server übergebene Datei über die Applikation geöffnet und innerhalb der Applikation die Funktion angestoßen werden, welche sich um das Aktualisieren kümmert. Um sicher zu stellen, dass der Job wirklich erfolgreich war, muss die Applikation ordnungsgemäß beendet worden sein.

Die von uns implementierte JBSPPropUpdate Funktion, nützt CAI-Befehle um die Applikation anzusprechen. Diese Befehle sind modal, das heißt, Productstream Compass wartet, bis die Befehle ausgeführt wurden. So muss nach diesem Befehl nicht zusätzlich auf das Schließen der Applikation gewartet werden.

#### Beispiel

Folgende Felder sind wie folgt zu belegen

Feld	Wert	Kommentar
Bezeichnung	Schriftfeldupdate	
Bedingung bei Anlage	isEtype(„AIM.DOC.ENG“) && wcomp(„#(FILE_TYPE:A)“, „A.AX.PX“) && exist(„#(DTY:Container0)“)	Es wird überprüft ob das übergebene Element ein Konstruktionsdokument vom Typ AutoCAD, MDT-Baugruppe oder Komponenten ist
Job Timeout	5	Definiert (in Minuten) wie lange ein Job ausgeführt werden darf. Wird die Zeit überschritten, so wird der Job terminiert und die Applikation beendet.
Befehl für Ausführung	JBSPPropUpdate	Dieser Befehl ist oben beschrieben

## Plotten bzw. Drucken mit AutoCAD

### Einleitung

Bei der Anlage eines Jobs durch den Anwender wird die Datei (#DOCNAME0) in ein zentral abgelegtes temporäres Verzeichnis kopiert. Diese Datei wird dann vom Jobserver geöffnet und geplottet. Damit ist gewährleistet, dass die Datei während des Plottens nicht durch einen anderen Anwender in Verwendung genommen wird. Zudem ist sicher gestellt, dass der gerade aktuelle Zustand der Zeichnung gedruckt wird.

Der Dateiname der kopierten Datei setzt sich aus der ID des Jobs und dem Namen der Datei zusammen. Damit ist sicher gestellt, dass eine bereits vorhandene lokale Kopie nicht überschrieben wird. Hierzu wird die DBP Prozedur „JBSCopyFileLocal“ verwendet; diese übernimmt den gesamten Kopiervorgang. Dabei belegt sie die Variable \$TargetFile, die dann wieder verwendet werden kann, um für jeden einzelnen Job zu ermitteln, welche Datei gedruckt werden soll.

Die lokale Kopie wird über die Applikation geöffnet. In diesem Falle wird eine Lisp- oder Script-Routine gestartet, die sich um das effektive Plotten kümmert. (diese ist nicht beigelegt und muss vom Händler geschrieben werden).

Nach Beendigung des Jobs kann die temporäre Kopie gelöscht werden

Da für das reine Drucken der Datei keine Productstream Compass CAD-Schnittstelle nötig ist, wird die Applikation direkt gestartet und als Argumente Dateiname und Script/Lisp übergeben.

### Beispiel

Feld	Wert	Kommentar
Bezeichnung	Plotten mit AutoCAD	
Bedingung bei Anlage	isEtype(„AIM.DOC.ENG“) && wcomp(„#(FILE_TYPE:A)“,“A.AX.PX“)	Es wird überprüft, ob das übergebene Element ein Konstruktionsdokument vom Typ AutoCAD, MDT-Baugruppe oder Komponenten ist.
Befehl bei Anlage	JBSCopyFileLocal(\\Server\JBSTemp); JBSSubstJobField(CMD_DO_JOB); JBSSubstJobField(CMD_ON_OK); JBSSubstJobField(CMD_ON_ERROR);	Hier wird bei der Anlage des Jobs eine lokale Kopie der Datei erstellt. Dazu werden die genannten Felder substituiert.
Job Timeout	5	Definiert (in Minuten) wie lange ein Job ausgeführt werden darf. Wird die Zeit überschritten, so wird der Job abgesetzt und die Applikation beendet.
Befehl für Ausführung	C:\programme\autodesk\autocad\acad.exe „\$(TargetFile:!)“ /b “C:/MeinScr.scr”	AutoCAD wird geöffnet und die über TargetFile übergebene Datei geladen. Über /b wird ein script geladen, welches dann das Drucken/Plotten übernimmt.
Befehl nach Ausführung	JBSLastCreatedProcessWait	Es wird abgewartet bis die Applikation sich selbständig schließt. Dafür ist es wichtig, in der Script/Lisp-Routine ein „end“ oder „quit“ anzugeben.
Befehl bei Erfolg	JBSRemoveLocalCopy(\$TargetFile)	Im Falle des Erfolgs wird die lokale Kopie entfernt.
Befehl bei Fehler	JBSRemoveLocalCopy(\$TargetFile)	Im Falle eines Misserfolgs wird die lokale Kopie entfernt.

## Plotten mit Inventor

Um über Inventor plotten zu können, stellen wir eine neue Funktion zur Verfügung. Über Steuerungsdateien kann der Plot konfiguriert werden. Syntax und Möglichkeiten werden im Folgenden erläutert.

### Änderungen in der aimdinv.dll

Implementierung der Funktion JBSPrint.

#### Aufruf:

```
JBSPrint <Inventor-Dokument> <Konfigurations-Datei>
```

#### Ablauf:

- Inventor wird gestartet, bzw. ein bereits laufender Inventor verwendet.
- Die spezifizierte Konfigurationsdatei wird in eine interne Struktur eingelesen und auf Gültigkeit (Syntax) überprüft. Im Fehlerfall wird eine Meldung in eine Log-Datei geschrieben und der Plot-Vorgang abgebrochen. Falls keine Konfigurationsdatei angegeben ist, werden Standard-Einstellungen verwendet.
- Das spezifizierte Inventor-Dokument wird geöffnet.
- Der Dokumenttyp wird ermittelt. Bei IDWs wird zum Plotten das API Objekt *DrawingPrintManager* verwendet. Bei allen anderen Dokumenten das Objekt *PrintManager*.
- Im jeweiligen PrintManager werden die Einstellungen aus der internen Struktur gesetzt.
- Über den PrintManager wird der Plot-Vorgang gestartet.
- Das Inventor Dokument wird geschlossen.

### Syntax der Konfigurationsdatei

#### Allgemeine Einstellungen

In der Sektion *General* werden allgemeine Einstellungen für alle Dokumenttypen vorgenommen:

Attribut und Wert	Beschreibung
Printer=<Name des Druckers>	Druckername definieren. Falls leer, wird der Standard-Drucker verwendet.
ColorMode=<Farbmodus>	Farbmodus definieren. Mögliche Werte: <i>kPrintGrayScale</i> , <i>kPrintColorPalette</i> , <i>kPrintDefaultColorMode</i> . Falls leer, wird <i>kPrintGrayScale</i> verwendet.
NumberOfCopies=<Anzahl>	Anzahl der Kopien. Falls leer wird der Wert 1 gesetzt
Orientation=<Orientierung>	Orientierung: Mögliche Werte: <i>kDefaultOrientation</i> , <i>kLandscapeOrientation</i> , <i>kPortraitOrientation</i> Falls leer, wird <i>kDefaultOrientation</i> verwendet.
PaperSize=<Papiergröße>	Papiergröße: mögliche Werte: <i>kPaperSize10x14</i> , <i>kPaperSize11x17</i> , <i>kPaperSizeA2</i> , <i>kPaperSizeA3</i> , <i>kPaperSizeA4</i> , <i>kPaperSizeA4Small</i> , <i>kPaperSizeA5</i> , <i>kPaperSizeB4</i> , <i>kPaperSizeB5</i> , <i>kPaperSizeCSheet</i> , <i>kPaperSizeDSheet</i> , <i>kPaperSizeESheet</i> , <i>kPaperSizeExecutive</i> , <i>kPaperSizeFolio</i> , <i>kPaperSizeLedger</i> , <i>kPaperSizeLegal</i> , <i>kPaperSizeLetter</i> , <i>kPaperSizeQuarto</i> , <i>kPaperSizeCustom</i> , <i>kPaperSizeDefault</i> Falls leer, wird <i>kPaperSizeDefault</i> verwendet.
PaperHeight=<Papierhöhe>	Papierhöhe: Diese Einstellung wird nur verwendet, wenn die Papiergröße auf <i>kPaperSizeCustom</i> gesetzt ist. Die Angabe erfolgt in Zentimeter.
PaperWidth=<Papierbreite>	Papierbreite: Diese Einstellung wird nur verwendet, wenn die Papiergröße auf <i>kPaperSizeCustom</i> gesetzt ist. Die Angabe erfolgt in Zentimeter.
PaperSource=<Nummer der Papierquelle>	Papierquelle: Druckerabhängig. Standardmäßig (Wert: -1) wird die Papierquelle verwendet, welche die gewählte Papiergröße unterstützt.



## Spezielle Einstellungen für Zeichnungsableitungen (IDWs)

In der Sektion *IDW* werden Einstellungen vorgenommen, die nur für Zeichnungsableitungen gültig sind:

Attribut und Wert	Beschreibung
PrintRange=<Seiten>	Seiten: mögliche Werte: <i>kPrintAllSheets</i> , <i>kPrintCurrentSheet</i> , <i>kPrintSheetRange</i> Falls leer, wird <i>kPrintCurrentSheet</i> verwendet.
FromPage=<Seitenzahl>	Erste Seite: diese Einstellung wird nur verwendet, wenn Seiten auf <i>kPrintSheetRange</i> gesetzt ist.
ToPage=<Seitenzahl>	Letzte Seite: diese Einstellung wird nur verwendet, wenn Seiten auf <i>kPrintSheetRange</i> gesetzt ist.
ScaleMode=<Skaliermodus>	Skaliermodus: mögliche Werte: <i>kPrintBestFitScale</i> , <i>kPrintCurrentWindow</i> , <i>kPrintCustomScale</i> , <i>kPrintFullScale</i> Falls leer, wird <i>kPrintBestFitScale</i> verwendet.
Scale=<Maßstab>	Maßstab: diese Einstellung wird nur verwendet, wenn der Skaliermodus auf <i>kPrintCustomScale</i> gesetzt ist. Zum Beispiel: 0.25 für Maßstab 1:4; 2.0 für Maßstab 2:1
Rotate90Degrees=<0   1>	Drehung um 90°. Falls leer, wird 0 verwendet.

### Beispiel

Da Inventor mit Referenzen arbeitet, ist es nicht möglich, eine lokale Kopie zu erstellen. Jedoch kommt Inventor mit dem mehrfachen Öffnen von Dokumenten besser zurecht als AutoCAD. Deswegen kann hier die Originaldatei gedruckt werden.

Da hier das Drucken wieder über CAI-Befehle erfolgt und diese modal sind, ist kein abschließendes Warten auf das Beenden der Applikation notwendig.

Feld	Wert	Kommentar
Bezeichnung	Plotten mit Inventor	
Bedingung bei Anlage	<code>isEtype(„AIM.DOC.ENG“) &amp;&amp; wcomp(„#(FILE_TYPE:A)“,„IDW.IAM.IPT“) &amp;&amp; exist(„#DOCNAME0“)</code>	Es wird überprüft, ob das übergebene Element ein Konstruktionsdokument vom Typ Inventor ist, sowie ob eine Datei vorhanden ist.
Job Timeout	5	Definiert (in Minuten) wie lange ein Job ausgeführt werden darf. Wird die Zeit überschritten, wird der Job abgesetzt und die Applikation beendet.
Befehl für Ausführung	<code>JBSPrint(„#DOCNAME0“ „c:\myPrint.ini“)</code>	Inventor wird geöffnet, die Datei geladen und der Druckvorgang laut Einstellungen der myPrint.ini ausgeführt.

## Drucken von allgemeinen Dokumenten

Um allgemeine Dokumente wie Textdateien, Bilder, PDF-Dateien, aber auch Office-Dokumente zu drucken, kann man den einfachen Druckbefehl verwenden. Dieser druckt die Datei über die entsprechende Applikation (wie am Betriebssystem konfiguriert) auf dem Standarddrucker aus. Der Standarddrucker kann über ein internes Productstream Compass-Kommando umgestellt werden.

Der Druckvorgang ist hier je nach Applikation unterschiedlich. Manche Applikationen schließen sich nach dem Drucken selbständig andere bleiben geöffnet. Da wir hier nicht über CAI-Befehle arbeiten, haben wir die Applikation nicht im Griff. Das bedeutet, dass das Beenden der Applikation abgewartet werden muss, um zu erkennen, ob der Druckvorgang erfolgt ist.

### Beispiel

Feld	Wert	Kommentar
Bezeichnung	Drucken	
Bedingung bei Anlage	isEType(„AIM.DOC.ENG“) && exist(“#DOCNAME0”)	Es wird überprüft ob das übergebene Element ein Konstruktionsdokument und ob eine Datei vorhanden ist.
Job Timeout	5	Definiert (in Minuten) wie lange ein Job ausgeführt werden darf. Wird die Zeit überschritten, so wird der Job abgesetzt und die Applikation beendet.
Befehl vor Ausführung	___SetPrinter(MyPrinter)	Hiermit wird der gewünschte Drucker als Standarddrucker gesetzt.
Befehl für Ausführung	JBSShellPrint(„#DOCNAME0“)	Die Datei wird über die Standardapplikation gedruckt.
Befehl nach Ausführung	JBSLastCreatedProcessWait(360000)	Wartet 6 Minuten (360 Sekunden) auf das Schließen der Applikation. Diese Zeit kann nie erreicht werden, denn nach 5 Min. ist das Timeout abgelaufen und der Watchdog terminiert die Applikation.

## Referenz

### Interne Funktionen

#### Interne Funktion: `___Jobserver`

Siehe Entwicklerdokumentation, Kapitel Befehlsreferenz `___Jobserver`

#### Interne Funktion: `___JobCreate`

Siehe Entwicklerdokumentation, Kapitel Befehlsreferenz `___JobCreate`

### DBP-Prozeduren

Um die Definition von Jobs zu erleichtern liefern wir einen Satz von DBP-Prozeduren, welche im Umfeld von Jobs nützlich sein könnten. Zudem dienen diese Prozeduren als Beispiele für die Implementierung eigener Jobs

#### **JBSShellOpen(„PfadUndDateiNameMitExtension“)**

Damit kann eine Datei von jedem beliebigen Ort geöffnet werden. Hier wird die Applikation explizit mittels Shell geöffnet, so wird auch keine Schnittstelle in der Applikation geladen.

#### **JBSOpen(„PfadUndDateiNameMitExtension“)**

Damit kann eine Datei von jedem beliebigen Ort geöffnet werden. Ist für eine spezielle Applikation ein besonderer Open-Befehl implementiert (CAI), so wird dieser verwendet. Ansonsten verweist die Funktion auf JBSShellOpen.

Von dieser Funktion gibt es auch die Varianten JBSOpen1 und JBSOpen2, welche die jeweilige Nebendatei öffnen.

#### **JBSPropUpdate**

Damit kann „Schriftfeld aktualisieren“ ausgelöst werden. Der Unterschied zur „Procedure“ (welche ein Schriftfeld-Update in Productstream Compass auslöst) ist, dass in JBSPropUpdate nicht überprüft wird, ob ein Element markiert ist (#nummark).

#### **JBSShellPrint(„PfadUndDateiNameMitExtension“)**

Mittels dieser Funktion wird die angegebene Datei über das Betriebssystem gedruckt. Hier werden keine CAI-Befehle verwendet, damit ist die Funktion asynchron (nicht modal) gegenüber der Applikation. Unmittelbar nach dem Aufruf dieser Funktion arbeitet Productstream Compass weiter. Gegebenenfalls kann die Funktion JBSLastCreatedProcessWait dazu verwendet werden.

#### **JBSPrint(„PfadUndDateiNameMitExtension“ „DruckKonfigurationsdatei“)**

Diese Funktion ist derzeit nur für Inventor implementiert. Damit wird die Datei mittels Inventor gedruckt. Für den Druck werden die Einstellungen der beigegebenen Konfigurationsdatei verwendet. Wird das zweite Argument nicht angegeben, so wird die Standarddruckdatei von Productstream Compass (cai\Autodesk\Inventor\aimdprint.ini) verwendet.

Siehe hierzu auch das Beispiel „Plotten mit Inventor“.

## **JBSSubstJobField(FeldName)**

Über diese Funktionen können bei der Anlage des Jobs Felder am Job selbst substituiert werden. Dies wird verwendet, wenn man bei der Definition mit Variablen gearbeitet hat und diese bei der Erzeugung (Instanz) des Jobs mit Klartext ersetzt werden müssen.

Beispiel:

Feld "Befehle vor Ausführung" (CMD\_PRE\_JOB) = read()Hello \$USERID

Feld „Befehle bei Anlage“ = JBSSubstJobField(CMD\_PRE\_JOB)

Das Ergebnis ist, dass nach der Anlage des Jobs im Feld "Befehle vor Ausführung" das \$USERID mit dem aktuell eingeloggt Benutzernamen ersetzt worden ist: read()Hello Mustermann

## **JBSCopyFileLocal(ZielVerzeichnis)**

Über diese Funktion wird das #DOCNAME0 in das Zielverzeichnis kopiert und mit einem eindeutigen Dateinamen versehen. Zielpfad und neuer Dateiname stehen dann in der Variablen \$TargetFile zur Verfügung.

Hiermit kann man zum Zeitpunkt der Anlage eines Jobs die zu behandelnde Datei in ein temporäres Verzeichnis kopieren, damit der Jobserver dann mit genau diesem Stand der Datei arbeitet. Es besteht so nicht die Gefahr, dass ein anderer Anwender die Originaldatei in Bearbeitung hat, während der Jobserver sie öffnen möchte.

## **JBSRemoveLocalCopy(PfadUndDateiname)**

Damit wird die angegebene Datei entfernt.

Es ist günstig, als Parameter eine Variable wie z.B. \$TargetFile (siehe JBSCopyFileLocal) einzusetzen, welche beim Erzeugen des Jobs substituiert wird (siehe JBSSubstJobField).

## **JBSLastCreatedProcessWait(ZeitInMillisekunden)**

Diese Funktion wartet ab, bis der zuletzt über Productstream Compass gestartete Prozess beendet ist. Wird beispielsweise über die Shell ein NOTEPAD.EXE gestartet und dann in der Shell die obige Funktion eingegeben, bleibt Productstream Compass in einem Wartemodus bis die Applikation notepad geschlossen wird.

Die Zeitangabe ist optional. Wird nichts angegeben, so wartet die Funktion 240000 Millisekunden (=240 Sekunden =4 Minuten) und bricht dann ab. In dieser Funktion wird die Bedingungsfunktion „LastCreatedProcessWait“ verwendet. Näheres dazu finden Sie im Befehl-Referenzbereich.

## Vorbelegung von Variablen bei der Anlage eines Jobs

Bei der Anlage eines Jobs durch den Anwender können Variablen vorbelegt werden, die im Rahmen der Neuanlage verwendet werden können. Diese Variablen enthalten die Werte der verschiedenen Job Felder. Damit kann z.B. der aktuelle IDENT sehr einfach ermittelt werden:

JBSCmdPreJob, JBSCmdDoJob, JBSCmdPostJob, JBSCmdOnOk, JBSCmdOnError, JBSCmdCreateJob, JBSCmdStartJob, JBSCmdCreateJob, JBSCmdIdent

## Rückgabewerte aus der AutoCAD-Schnittstelle

### Rückgabewert bei Schriftfeld aktualisieren

Um AutoCAD-Schriftfelder über den Jobserver zu aktualisieren, kann die Funktion JBSPropUpdate verwendet werden. Diese stützt sich auf die Funktion enumPropUpdate, welche im normalen Productstream Compass Umfeld für das Aktualisieren von Dokumenten verwendet wird. Ab der Version AutoCAD 2004 liefert unsere Schnittstelle nach der Aktualisierung des Schriftfeldes einen Rückgabewert, der über den Erfolg bzw. Misserfolg berichtet. Der Rückgabewert steht in der Variable ARXErrorCode, die nach dem Aktualisieren belegt wird. Folgende Werte sind möglich

Rückgabewert	Beschreibung
0	Kein Fehler
1	Unbekannter Fehler
2	Kein Schriftfeld in der Zeichnung gefunden
3	Mindestens 1 Schriftfeld kann nicht aktualisiert werden, weil der Blockname nicht in der amtitle.ini definiert ist

Es wird empfohlen, insbesondere beim Jobserver die Rückgabewerte zu überprüfen und entsprechend darauf zu reagieren.