

Archiver 5.1

exportiert am 27-03-2023

Inhaltsverzeichnis

Rechtliche Hinweise	5
Einleitung	6
Was ist neu in Version 5.1	7
Programme des Archiver	9
archiver - Programm	10
Aufruf und Parameter des Archiver	11
arcdir	12
Aufruf und Parameter von ArcDir	13
a2ser	15
Aufruf und Parameter von a2ser	16
a2os	18
Aufruf und Parameter von a2os	20
a2os Version 3.0	22
a2easy	25
Aufruf und Parameter von a2easy	26
a2content, a2saperion	28
Aufruf und Parameter von a2content	29
a2enaio	31
Aufruf und Parameter von a2enaio	33
a2comp (obsolete)	34
Aufruf und Parameter von a2comp (obsolete)	35
a2hyp (obsolete)	37
Aufruf und Parameter von a2hyp (obsolete)	38
a2profile (obsolete)	40
Aufruf und Parameter von a2profile (obsolete)	41
Feldzuordnung innerhalb der Indexdatei (obsolete)	43
Installation (obsolete)	44
Konfiguration des Archivers	45
Einstellungen	46
Pfade	47
Tasks	48
Scannen	50
LogFile	51
Namen, Verfahren und Format	53

PDF Verarbeitung	55
PDF Erzeugung	56
Sprache/Seriennummer	60
Formate	61
Arcis/Teradoc.....	62
Easy	64
FileNET	67
DocuWare	69
IXOS	71
CeyonIQ	72
Transfer	73
SER (obsolete)	76
ProFile (obsolete).....	78
Hinweise zur Dokumentation.....	80
Verwendete Symbole.....	81
Weiterführende Informationen.....	83
Sitemap	84
Rechtliche Hinweise.....	84
Einleitung	84
Was ist neu in Version 5.1	84
Programme des Archiver	84
Konfiguration des Archivers	84
Hinweise zur Dokumentation.....	85
Weiterführende Informationen.....	85
Download der Dokumentation.....	86
Ergänzende Online-Dokumentationen.....	87
Service und Support	88

Herzlich Willkommen auf der Online-Dokumentationsseite des Archiver!


Hier finden Sie alle Informationen zur aktuellen Version 5.1 des Archiver.

Archiver
<input type="checkbox"/> <i>Einleitung</i>
<input type="checkbox"/> <i>Neuerungen</i>
<input type="checkbox"/> <i>Programme</i>
<input type="checkbox"/> <i>Konfiguration</i>

Erweiterte Informationen
<input type="checkbox"/> <i>Verwendete Symbole</i>
<input type="checkbox"/> <i>Sitemap</i>
<input type="checkbox"/> <i>Download der Dokumentation</i>
<input type="checkbox"/> <i>Service & Support</i>

Weitere Online-Dokumentationen

Sie suchen Hilfe für ein anderes Produkt? [Hier](#) gelangen Sie zur Übersicht aller Online-Dokumentationen!

 Beachten Sie, dass Sie zur Darstellung der Online-Dokumentation eine Mindestauflösung von 1280 x 1024, idealerweise eine Auflösung von 1920 x 1200 benötigen. Die Werte können je nach Format des Bildschirms variieren.

Rechtliche Hinweise

Der Inhalt dieser Online-Dokumentation ist das geistige Eigentum der profiforms gmbh. Bei der Erstellung der Texte und Abbildungen dieser Online-Dokumentation wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Die profiforms gmbh übernimmt keinerlei Gewähr für die Aktualität, Korrektheit und Vollständigkeit der bereitgestellten Informationen.

Die profiforms gmbh behält sich das Recht vor, den Inhalt dieser Online-Dokumentation ohne vorherige Ankündigung zu verändern oder ergänzen und übernimmt keine Haftung für Fehler in dieser Online-Dokumentation oder daraus resultierende mögliche Schäden.

Für Verbesserungsvorschläge und Hinweise auf Fehler sind Herausgeber und Autoren dankbar.

Diese Software wird gebündelt mit den Schriftarten des DejaVu-Projekts ausgeliefert. DejaVu ist eine Sammlung von verschiedenen, unter freier Lizenz stehenden Schriftarten, die ihren Ursprung in der Schriftartensammlung Bitstream Vera haben. Die Schriftarten stehen unter dem Bitstream Vera Fonts Copyright und dem Arve Fonts Copyright, welche als Lizenzverträge mit installiert werden. DejaVu ist kein preislicher Bestandteil dieses Produkts. Profiforms übernimmt für diese Schriften weder Garantie noch Wartung. Alle Rechte bezüglich dieser Schriften liegen bei Bitstream und dem DejaVu-Projekt.

SAP, SAP R/3, SAPScript, SmartForms, BC-RDI, BC-XFP und andere sind eingetragene Warenzeichen der SAP AG, Walldorf.

Java ist eine eingetragene Marke der Oracle Corporation.

Adobe, Adobe Present, Adobe Central, Adobe Designer, PostScript, PDF, XDP und weitere Warenzeichen sind eingetragene Warenzeichen der Adobe Systems Incorporated.

Hewlett Packard, HP-PCL sind eingetragene Warenzeichen der Hewlett-Packard Company.

Unix ist ein Warenzeichen der Open Group.

Windows ist ein eingetragenes Warenzeichen der Microsoft Corporation.

TBarcode ist ein eingetragenes Warenzeichen der TEC-IT Datenverarbeitung GmbH.

Alle anderen Firmennamen und Produktbezeichnungen sind Warenzeichen oder eingetragene Warenzeichen der jeweiligen Firmen und unterliegen im Allgemeinen warenzeichen-, marken- und/oder patentrechtlichem Schutz.

Einleitung

Der Archiver ist ein Programmpaket, das als Schnittstelle zwischen **ReportWriter**, **Logistics** und anderen Dokument-erzeugenden Programmen dient. Die Hauptaufgabe ist es, Druckjob-orientierte Dateien in Dokument-orientierte und mit einem Index angereicherte Archiv-Dokument-Dateien zu überführen. Durch den Archiver werden PDF- oder XML-Dateien mit mehreren Dokumenten in einzelne PDFs aufgeteilt und evtl. zu Bitmaps konvertiert, die Index-Feldnamen umgeschlüsselt, ein Archivsystem-spezifischer Index-File angelegt und evtl. ein Transfer-Programm zum Archivsystem aufgerufen.

Das Programmpaket des Archivers besteht aus dem Programm Archiver, einem Standalone-Programm, das eigenständig Verzeichnisse überwacht, und dem ArcDir, ermöglicht Einbindung in andere Ablaufumgebungen wie z.B. dem **Spooler**. Weiterhin können je nach Art des Archivsystems noch Transfer-Programme mit speziellen Transport-Techniken (Java-APIs, WEB-Services, OCGs, DLLs, ActiveX usw.) hinzukommen, welche die direkte Verbindung zum Archivsystem herstellen und meist vom Archivsystem-Hersteller gestellt werden.

Das Konvertieren des PDFs zu einem Bitmap oder zu einem anderen Format ist nicht Bestandteil der Archiver-Lizenz. Diese Module müssen separat erworben werden. Lediglich die Schnittstelle zur Einbindung in den Archiver ist vorhanden. Wenn die Übergabe der Dokumente an das Archivsystem in einem Bitmap-Format erfolgen soll, muss ein Konverter eingesetzt werden, der den PDF-Druckdatenstrom in ein Bitmap überführt. Der Archiver unterstützt hier besonders das **PDFxOut**, natürlich sind auch andere Konverter einsetzbar.

Das Programm Archiver existiert auf allen Plattformen als Single-Thread-Application ohne grafische Oberfläche.

Je nach Spracheinstellung sind das Logging und die Oberfläche in Deutsch oder Englisch verfügbar.

Die Produkte Archiver und ArcDir verarbeiten unterschiedliche Input-Formate, die mit Produkten erzeugt wurden:

- Meta-Format mit internen und externen PDFs
- XML Archive-XML mit internen und externen PDFs
- PDF Archive-PDF mit Archive-Informationen als XMP und vielen Dokumenten
- PDF als Einzeldokumente

Das Meta-Format ist technisch veraltet und nur aus Kompatibilitätsgründen noch vorhanden.

Für die Archivierung wird entweder Archive-XML oder Archive-PDF empfohlen. Beim Archive-XML leistet das Dokument-erzeugende Programm die Arbeit, viele kleine PDFs für das Archiv zu erstellen und ist damit extrem langsam. Die vom Archiver oder ArcDir zu verarbeitenden Daten sind sehr groß. Die Archivierung ist sehr schnell.

Beim Archive-PDF leistet der Archiver oder der ArcDir die Arbeit, viele kleine PDFs für das Archiv zu erzeugen. Das Dokument-erzeugende Programm ist extrem schnell und die zu verarbeitenden Daten sehr klein. Die Archivierung ist etwas langsamer als die von Archive-XML. Archive-PDF stellt damit den besten Kompromiss bezogen auf die Verarbeitungsgeschwindigkeit und die Datengröße für alle beteiligten Systemkomponenten dar.

Was ist neu in Version 5.1

Unterstützung weitere PDF-Versionen für die Ausgabe

Gerade im Bereich der PDF Versionen für die Archivierung hat sich viel getan. Waren in Vorgängerversion des Archivers als Ausgabe die PDF Version 1.4 bis 1.7 und PDF/A1B2005 erlaubt, so erweitert sich das Spektrum nun auf PDF 1.4 bis 2.0 und folgende Substandards:

PDF/A1A2005
 PDF/A1B2005
 PDF/A2A
 PDF/A2B bzw. PDF/A2
 PDF/A2U
 PDF/A3A
 PDF/A3B bzw. PDF/A3
 PDF/A3U

ZUGFeRD Unterstützung für die Archiv-Typen "PRINTFILE" und "ARCHIVEFILE"

Die ZUGFeRD-XML wird als Archiv-Variable EMBEDDED_FILE an den Archiver übergeben und erst im Archiver beim Aufsplitten der Input-PDF-Datei in Ausgabe-PDF-Dateien an die richtige Stelle gerückt. Bedingung ist der ReportWriter bzw. Logistics ab Version 5.9.1041. Dieser Weg der Übergabe der ZUGFeRD-XML beschleunigt den ReportWriter bzw. Logistics enorm beim Archivieren.

Kopieren von Hyperlinks und Farbprofilen aus dem Input-PDF ins Ziel-PDF

Werden aus einem Archiv-PDF heraus einzelne Archiv-Dokumente extrahiert, so gingen bis jetzt alles Annotations verloren. Ab diese Version werden die Annotation-Weblinks auf der Seite gelesen und mit ins neue Archiv-Dokument, sprich das separierte PDF, übernommen. Damit lassen sich die "interaktiven" PDFs jetzt auch über ReportWriter-ArchiveFileType "ARCHIVEFILE" und "PRINTFILE" an den Archiver weitergeben. Alle anderen Annotationen, die nicht vom Typ Weblinks sind, werden auch weiterhin nicht mit übernommen. Unterstützung des Kopierens der OutputIntents vom Input-Dokument zum Ziel-Dokument. Durch diese Änderung ist es möglich auch PDFs mit einem anderen OutputIntent als sRGB richtig zu erzeugen. Dieser Fall kommt nur vor, wenn im ReportW/Logistics ein anderes OutputIntent gesetzt wird als sRGB, was nur sinnvoll ist, wenn PDF/X oder PDF/VT archiviert werden soll. Archiv-Dokumente habe für gewöhnlich sRGB als OutputIntent.

Umstellung der PDFlib Version

Die vorige PDFlib Version 7 war gerade in Bezug auf PDF/A und PDF-ZUGFeRD nicht in allen Fällen fehlerfrei. Die Umstellung auf die aktuelle PDFlib Version 9 erforderte allerdings viele Änderungen. Dieser Wechsel ist nun vollzogen. Die Version Archiver/ArcDir 5.0 ging durch die Fehler der PDFlib nun vollständig außer Support. Kunden müssen nach dem Supportvertrag auf die aktuelle Version 5.1 wechseln um Support zu bekommen.

Wechsel auf neue Compiler für Windows und 64-Bit Support

Der Wechsel des Compilers auf den "C++ Builder XE 10.2" bringt es mit sich, dass nun auch 64-Bit Compilate erzeugt werden können. Ebenso wie es unter Linux schon gewohnt war, wird nun auch für Windows eine 32-Bit und eine 64-Bit Version ausgeliefert.

Beendung des Supports für ältere Unix Plattformen

Ältere Unix-Derivate sind am Auslaufen. Der Support für Solaris, HP-UX und AIX wurde eingestellt. Damit werden aktuell die Plattformen Linux und Windows unterstützt.

neues Programm im Archiver

a2os Version 3.0 basiert auf den neuen Java-Bibliotheken enaio-rpc, enaio-jobs und enaio-beans. Mehr Informationen dazu gibt es [hier](#).

Programme des Archiver

Die Programme gliedern sich in folgende Unterkapitel (bitte auf den jeweiligen Link klicken):

- ***archiver - Programm***
 - ***Aufruf und Parameter des Archiver***
- ***arcdir***
 - ***Aufruf und Parameter von ArcDir***
- ***a2ser***
 - ***Aufruf und Parameter von a2ser***
- ***a2os***
 - ***Aufruf und Parameter von a2os***
 - ***a2os Version 3.0***
- ***a2easy***
 - ***Aufruf und Parameter von a2easy***
- ***a2content, a2saperion***
 - ***Aufruf und Parameter von a2content***
- ***a2enaio***
 - ***Aufruf und Parameter von a2enaio***
- ***a2comp (obsolete)***
 - ***Aufruf und Parameter von a2comp (obsolete)***
- ***a2hyp (obsolete)***
 - ***Aufruf und Parameter von a2hyp (obsolete)***
- ***a2profile (obsolete)***
 - ***Aufruf und Parameter von a2profile (obsolete)***
 - ***Feldzuordnung innerhalb der Indexdatei (obsolete)***
 - ***Installation (obsolete)***

archiver - Programm

Bis auf einige wenige Informationen der Kommando-Zeile bezieht der Archiver fast alle Konfigurationen aus einer Ini-Datei, die ihm entweder beim Aufruf mitgegeben wird oder die er als archiver.ini im aktuellen Arbeitsverzeichnis sucht.

Der Archiver kann entweder als Service-ähnliches Programm gestartet werden und dann bis zu seiner Beendigung laufen oder aber für einen konkreten Auftrag aufgerufen und danach wieder heruntergefahren werden. In allen Fällen ist es aber angeraten, das gewünschte aktuelle Arbeitsverzeichnis mit dem Programmstart zu setzen. Bitte beachten Sie, dass mehrere gleichzeitig laufende Archiver weder Input, Output, Error noch das aktuelle Arbeitsverzeichnis miteinander teilen dürfen.

Hier gelangen Sie zum Aufruf und den Parameter des Archiver (bitte auf den Link klicken):

- [*Aufruf und Parameter des Archiver*](#)

Aufruf und Parameter des Archiver

Aufrufkonventionen

archiver [INIFILE] [-STOP]

Erklärung

INIFILE

Datei-Pfad und -Name der Ini-Datei, die für diesen Start gilt.
Standard ist „archiver.ini“ im aktuellen Arbeitsverzeichnis.

-STOP

Mit dieser Option kann erzwungen werden, dass der Archiver nach der Archivierung aller zum Zeitpunkt des Starts gefundenen Archiv-Dateien kontrolliert herunterfährt.

arcdir

Das Programm ArcDir stellt keine vollständige Ablaufumgebung dar und ist zur Einbindung in andere Rahmenprogramme wie z. B. dem Spooler gedacht. In einem Arbeitsschritt wird exakt eine Input-Datei verarbeitet und dann das Programm beendet. Der ArcDir sollte vor allem dann zum Einsatz kommen, wenn es wichtig ist, dass die Archiv-Datei sofort und inklusive Rückmeldung vom Archivsystem verarbeitet wird. Besonders sinnvoll ist hier die Kombination des ArcDir mit dem Spooler.

Hier gelangen Sie zum Aufruf und den Parameter von ArcDir (bitte auf den Link klicken):

- [*Aufruf und Parameter von ArcDir*](#)

Aufruf und Parameter von ArcDir

Aufrufkonventionen

arcdir [Options] ArchiveFile INIFILE

Erklärung

ArchiveFile

Archiv-Datei im META-, XML- oder PDF-Format.

INIFILE

Datei-Pfad und -Name der Ini-Datei, die für diesen Start gilt. Standard ist „archiver.ini“ im aktuellen Arbeitsverzeichnis.

Options

-v	Programmversion
-all LOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei.
-lan LAN	Reporting-Language: Die Log- und Fehler-Informationen können in Deutsch oder Englisch ausgegeben werden. Beim Start des Programms wird ermittelt, ob die Sprache des Betriebssystems Deutsch ist. In diesem Fall ist auch die Reporting-Language Deutsch, in allen anderen Fällen Englisch. Mit -lan kann die Reporting-Language willkürlich gesetzt werden.
-rsp FILETYPE	Type der Response-Datei: Mit dem Beenden des Programms kann eine Datei ins aktuelle Arbeitsverzeichnis geschrieben werden, die Informationen über den Programmablauf und dessen Erfolg enthält. Dabei sind folgende Werte einstellbar: NONE Es wird keine Response-Datei erzeugt (Standardwert) XML Es wird eine XML-Response-Datei im profiforms-Format erzeugt JETFORM Es wird eine Response-Datei im JetForm-Format erzeugt
-aip PATH	Pfad zu den Ini-Dateien.
-aop PATH	Ausgabepfad
-deb	Debug-Modus, bei dem alle temporären Dateien nicht gelöscht werden.
-adv VARNAME=CONTENT	Definition globaler Variablen, die zum Lesen der Ini-Dateien über den Linereader zur Verfügung stehen. Die Angabe kann mehrfach erfolgen.

-aic CP	CP ist die Code-Page für das Lesen der INI-Dateien.
-adg SEL	Auswahl bestimmter Dokumente für die Archivierung. Dabei wird die Selektion als Dokument-Nummer im Input-Datenstrom interpretiert. Es kann eine kommaseparierte Liste angegeben werden, in der Einzelwerte oder Intervalle stehen. Ein Intervall enthält immer das Zeichen „/“. Beispiel: -adg 1,7/10.
-d SEL	Auswahl bestimmter Dokumente für die Archivierung. Dabei wird die Selektion als DocRef-Wert interpretiert. Es kann eine kommaseparierte Liste angegeben werden.
-sel SEL	Auswahl bestimmter Dokumente für die Archivierung. Dabei kann als Selektion entweder ein Intervall von Dokument-Nummern oder einen DocRef-Wert enthalten sein. Dokument-Nummern werden daran erkannt, dass sie immer ein Intervall, wie z.B. „1/2“, definieren. Wird kein Intervall erkannt, so handelt es sich um einen DocRef-Wert.
-uid UNIQUEID	Diese UID wird Bestandteil des BasisNamens, der zu Eindeutigkeitszwecken in den Namen des Verzeichnisses oder des FileNamens einfließt.

Beispiel

```
arcdir x.pdf archiver.ini -all Logfile.log
```

a2ser

Das Programm a2ser arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung vom Archiver zum ITA-Server der SER über das BlueLine-Interface her. Dieses Programm wurde mit Java erstellt und benötigt eine Java VM der Version 1.4.1 oder höher.

Drei wesentliche Aufgaben werden von a2ser wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente der Index-Datei zum ITA-Server
- Löschen der erfolgreich transportierten Dateien
- Optional die Rückmeldung ans SAP-System

Zur Übergabe der Dokumente an den ITA-Server wird folgendes Index-Feld benötigt:

- FORMNAME (wird automatisch beim Format Transfer generiert)

Das Programm a2ser überprüft anhand des FORMNAME, ob im ITA-Server ein entsprechendes Formular definiert ist. Wird ein entsprechendes Formular gefunden, so ermittelt a2ser die Liste aller Variablen, die der ITA-Server für dieses Formular benötigt und vergleicht sie mit den Variablen des aktuellen Dokuments. Passen die Konfigurationen nicht überein, führt das zu einem Abbruch mit einer vollständigen Erläuterung im LogFile.

Wird a2ser ohne Parameter aufgerufen, so erfolgt die Ausgabe eines Textes zum Gebrauch der Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Hier gelangen Sie zum Aufruf und den Parameter von a2ser (bitte auf den Link klicken):

- [***Aufruf und Parameter von a2ser***](#)

Aufruf und Parameter von a2ser

Aufrufkonventionen

```
java -jar a2ser.jar IndexFile WorkingPath [VarName1=Volume [...]] [Options]
```

Erklärung

IndexFile

Archiver-Index-File im Format Transfer.

WorkingPath

Pfad, in dem die Dokument-Files stehen.

VarNameN=Volume

Liste von Variablen, die fix für diesen Lauf vorgegeben werden und nicht im Index stehen müssen.

Options

-lanGERMAN oder -lanENGLISH	Sprache der Meldungen
-allLOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei
-iniBLUELINEINI	ini-File mit Konfiguration der SER-BlueLine-Schnittstelle
-sysSYSTEM	SER-BlueLine-Systemname
-usrUSER	SER-BlueLine User-Name
-pwdPASSWORD	SER-BlueLine Passwort
-sifSAPINIFILE	SAP Rückmeldung Ini-File (optional)
-deb	Debug Modus (optional)
-edtSEC	SEC ist die Zeit in Sekunden, die im Fehlerfall vergehen müssen, bevor ein Neuversuch unternommen wird. Standardwert ist 6 Sekunden (optional)

Beispiel

```
java -jar a2ser.jar $INDEXFILE$ $PATH$ -all$LOGFILENAME$
      -iniC:\BLUELINE\bli.ini -sysITA1 -usrUSER -pwdPASSWORD
```

Aufrufzeile von a2ser als TransferTask im Archiver.

Platzhalter in Variableninhalt

Folgende Platzhalter können bei der Definition einer Variablen im Variableninhalt verwendet werden:

- \n Zeilenumbruch
- \r Zeilenumbruch
- \" Anführungsstrich
- \' Hochkomma
- \\ Backslash „\“
- \s Leerzeichen (Space)

a2os

Das Programm a2os arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung vom Archiver zum Archivsystem OS:DRT der Firma OPTIMAL SYSTEMS GmbH dar. Dieses Programm wurde mit Java erstellt und benötigt eine Java VM der Version 1.4.1 oder höher. Mindestvoraussetzung ist das Archivsystem in der Version OS:DRT 4.20 SP2. Die von OPTIMAL SYSTEMS zur Verfügung gestellten Java-Klassen sind aufwärtskompatibel zu OS:DRT 4.50.

Drei wesentliche Aufgaben werden von a2os wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente und der Indexinformationen zum OS:DRT
- Löschen der erfolgreich transportierten Dateien

Zur Übergabe der Dokumente an den Archiv-Server werden nicht nur die Index-Variablen des Dokuments benötigt, sondern auch Felder zur Bestimmung des Ablageortes. Der Ablageort eines Dokuments ist ein Ordner und optional darin ein Register, wobei sowohl Ordner als auch Register eigene Attribute haben. Um den Ablageort zu bestimmen, werden Variablen zur Bestimmung des Ordners benötigt. Diese Ordner-Variablen im Indexfile beginnen immer mit dem Prefix "Cabinet.". Register-Variablen zur optionalen Bestimmung des Registers beginnen immer mit dem Prefix "Register.".

Beispiel

```
Cabinet.Name=Ordnername, bzw. Register.Name=Registername
```

Alle anderen Variablen sind Variablen auf Dokumentenebene mit dem Spezialfall der Variable DOCTYPE, die den Dokumenttyp im OS:DRT bestimmt.

Beispiel

Set von Index-Variablen, die zur Ablage in der Objekthierarchie des OS:DRT führen.

```
Cabinet.Name=KUNDE-B1
Cabinet.Jahr=2003
Cabinet.Monat=11
Cabinet.Quartal=3
FormName=Rechnung
KundenNr=123456 Beleg-Nr=789012
```

Die a2os prüft selbständig das Vorhandensein aller in der OS Objektdefinition spezifizierten Pflicht- und Schlüsselfelder und legt Ablageorte bei Bedarf selbständig an.

Hier gelangen Sie zum Aufruf und den Parameter von a2os (bitte auf den Link klicken):

- [***Aufruf und Parameter von a2os***](#)
- [***a2os Version 3.0***](#)

Aufruf und Parameter von a2os

Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Aufrufkonventionen

```
java -jar a2os.jar IndexFile WorkingPath [VarName1=Volume [...]] [Options]
```

Erklärung

IndexFile

Archiver-Index-File im Format Transfer.

WorkingPath

Pfad, in dem die Dokument-Files stehen.

VarNameN=Volume

Liste von Variablen, die fix für diesen Lauf vorgegeben werden und nicht im Index stehen müssen.

Options

-s	Die Arbeit erfolgt ohne Ausschriften auf dem Bildschirm. (Silent)
-lanGERMAN oder -lanENGLISH	Sprache der Meldungen
-allLOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei
-srvSERVER	HOSTNAME/IP des OS:DRT-Applikationsservers
-usrUSER	Anmeldename für OS
-pwdPASSWORT	Benutzerpasswort für OS (im Klartext)

Beispiel

```
java -jar a2os.jar $INDEXFILE$ $PATH$ -all$LOGFILENAME$
      -srvOSDTR -usrPROFIFORMS -pwdPROFIFORMS
```

Aufrufzeile von a2os als TransferTask im Archiver.

Platzhalter in Variableninhalt

Folgende Platzhalter können bei der Definition einer Variablen im Variableninhalt verwendet werden:

- \n Zeilenumbruch
- \r Zeilenumbruch
- \" Anführungsstrich
- \' Hochkomma
- \\ Backslash „\“
- \s Leerzeichen (Space)

a2os Version 3.0

a2os Version 3.0 basiert auf den neuen Java-Bibliotheken enaio-rpc, enaio-jobs und enaio-beans. Am Aufruf und an der Konfiguration ist nichts verändert worden. Hinzugekommen ist der Schalter -eno. Dieser Schalter steht für die enaio-Options und kann Optionen aktivieren.

Der Parameterwert ist eine Semikolon-separierte Liste.

Aufruf
-enoOption1=1;Option2=0;...

Um zum Beispiel die Katalogprüfung zu deaktivieren wurde der Aufruf wie folgt aussehen:

Beispielaufruf Katalogprüfung
-enoCHECKCATALOGUE=0

Parameter "-eno"

Option	Beschreibung	Default
APPENDFILEST OFRONT	Gibt für Update und Varinatenerzeugung an, ob anzuhängende Files vor (1) oder nach dem Bestehenden angehängt werden (0).	Option nicht aktiv (0)
ARCHIVABLE	Gibt an, ob das Dokument den Status archivierbar (1) oder nicht archivierbar (0) erhält .	Option nicht aktiv (0)
ARCHIVEIMME DIATELY	Gibt an, ob das Dokument sofort archiviert wird (1) oder nicht (0). ⚠ Kann das Dokument nicht archiviert werden, wird kein Dokument ohne Seiten angelegt.	Option nicht aktiv (0)
ARCHIVEIMME DIATELYOBJDEF	Gibt an, ob zu einem Dokument, das sofort archiviert wird (ARCHIVEIMMEDIATELY), die Objektdefinition ebenfalls archiviert wird (1) oder nicht (0).	Option nicht aktiv (0)
CHECKACCESS	Für das Objekt werden die Benutzerrechte geprüft (1) oder nicht geprüft (0).	Option aktiv (1)
CHECKCATALOGUE	Es wird geprüft (1) oder nicht geprüft (0), ob alle übergebenen Katalogeinträge auch im Original-Katalog enthalten sind.	Option aktiv (1)
CHECKEXISTENCE	Es wird geprüft (1) oder nicht geprüft (0), ob das angegebene Objekt an der angegebenen Position existiert.	Option aktiv (1)

Option	Beschreibung	Default
CHECKKEYFIELDS	Es wird geprüft (1) oder nicht geprüft (0), ob Schlüsselfelder eindeutig.	Option aktiv (1)
CHECKOBLIGATION	Es wird geprüft (1) oder nicht geprüft (0), ob alle Pflichtfelder gesetzt wurden.	Option aktiv (1)
CHECKPOSITION	Es wird geprüft (1) oder nicht geprüft (0), ob angegebene (Ziel-) Objekte (z. B. Ordner) existieren.	Option aktiv (1)
CHECKREADONLY	Es wird geprüft (1) oder nicht geprüft (0), ob keine Felder verändert wurden, für die keine Schreibrechte bestehen.	Option aktiv (1)
DELETECASCADING	Gibt an, ob Objekte kaskadierend gelöscht werden sollen, auch wenn sie Unterobjekte enthalten (1) oder nicht (0).	Option nicht aktiv (0)
DELETEVARIANTMODE	Gibt an, ob das Löschen einer gegebenen inaktiven Variante zum Löschen des gesamten Variantenbaums führt (1) oder nicht (0).	Option nicht aktiv (0)
COPYCASCADING	Gibt an, ob Objekte kaskadierend kopiert werden sollen, wenn sie Unterobjekte enthalten (1) oder nicht (0).	Option nicht aktiv (0)
COPYCREATEHISTORY	Gibt an, ob Informationen zum Kopieren in die Historie eingetragen werden (1) oder nicht (0).	Option aktiv (1)
COPYINDEXONLY	Gibt an, ob nur die Indexdaten kopiert werden (1) oder nicht (0).	Option nicht aktiv (0)
FULLTEXTFILEATTACHED	Gibt an, ob die letzte übermittelte Datei die Volltextdaten für das Dokument enthalten soll (1) oder nicht (0).	Option nicht aktiv (0)
HARDELETE	Gibt an, ob das Objekt endgültig gelöscht werden soll, also nicht in den Papierkorb verschoben wird (1) oder nicht (0).	Option nicht aktiv (0)
INITFIELDS	Gibt an, ob alle nicht gesetzten Felder mit den Default-Werten belegt werden sollen (1) oder nicht (0).	Option nicht aktiv (0)
INUSERTRAY	Gibt an, ob das Objekt in die Benutzer-Ablage eingefügt werden soll (1) oder nicht (0).	Option nicht aktiv (0)
INWFTRAY	Gibt an, ob das Objekt in die Workflow-Ablage eingefügt werden soll (1) oder nicht (0).	Option nicht aktiv (0)
KEEPLINKWHENEXISTS	Gibt an, ob ein bereits existierender Link als Fehler gewertet werden soll (1) oder nicht (0).	Option nicht aktiv (0)
LINKDOCUMENT	Gibt an, ob ein Dokument nur einen neuen Standort erhalten soll (1) oder nicht (0).	Option nicht aktiv (0)
REPLACEFILES	Gibt an, ob bereits gespeicherten Dateien durch die übergebenen Dateien ersetzt (1) oder angehängt werden (0).	Option aktiv (1)

Option	Beschreibung	Default
REPLACEMULTI FIELDS	Gibt an, ob die übergebenen Multifelder die Originalen ersetzen sollen (1) oder angehängt werden (0).	Option nicht aktiv (0)
REPLACEREMAR KKS	Gibt an, ob die übergebenen Notizen die Originalen ersetzen sollen (1) oder angehängt werden (0).	Option nicht aktiv (0)
REPLACETABLEF IELDS	Gibt an, ob die übergebenen Tabellenfelder die Originalen ersetzen sollen (1) oder angehängt werden (0).	Option nicht aktiv (0)
TRUNCATEVAL UES	Gibt an, ob übergebene Strings abgeschnitten werden sollen, wenn sie länger sind, als definiert (1) oder nicht (0).	Option nicht aktiv (0)
TYPELESS	Gibt an, ob das Objekt als typenlos in die Ablage eingefügt werden soll (1) oder nicht (0).	Option nicht aktiv (0)
UPDATEALLFIEL DS	Gibt an, ob nicht angegebene Felder auf leer gesetzt werden sollen (1) oder nicht (0).	Option nicht aktiv (0)
VARIANTSAMEL EVEL	Gibt an, ob die Variante auf der gleichen Ebene (1) oder als 'Untervariante' (0) eingefügt werden soll.	Option nicht aktiv (0)
VARIANTSETAC TIVE	Gibt an, ob die neue Variante im selben Schritt als 'Aktive' (1) gesetzt wird oder nicht (0).	Option nicht aktiv (0)
VARIANTTRANS FERRETENTION	Gibt an, ob die neue Variante die geplante Retention Zeit vom Original Dokument übernehmen soll (1) oder nicht (0).	Option nicht aktiv (0)
WFTOUSERTRA Y	Gibt an, ob beim Verschieben eines Dokumentes dieses von der Workflowablage in die Benutzerablage verschoben werden soll (1) oder nicht (0).	Option nicht aktiv (0)

a2easy

Das Programm a2easy arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung zum XML-Server des Easy-Archivs der Firma Easy Software AG dar. Dieses Programm wurde mit Java erstellt und benötigt eine Java VM der Version 1.4.1 oder höher.

Vier wesentliche Aufgaben werden von a2easy wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente der Index-Datei zum XML-Server des Easy-Archivs
- Löschen der erfolgreich transportierten Dateien
- Optional die Rückmeldung ans SAP-System

Zur Übergabe der Dokumente an den XML-Server kann das Feld ARCHIVEPATH definiert werden. Ist dieses Feld definiert, so wird der Ablageort aus dem Inhalt dieses Feldes bestimmt und nicht aus dem Standard-Ablageort, der über die Option `-sys` beim Aufruf gesetzt wird.

Easy-Archiv legt die Dokumente in einem Feld ab. Die Zuweisung, in welchem Feld welcher File abgelegt wird, geschieht über die Angabe eines Feld-Routings an jedem Dokument. Ein solches Feld-Routing geschieht über die Angabe des Easy-Feldes gefolgt von den festen Schlüsselwörtern `$FILE1`, `FILE2` usw.

Beispiel

```
2001 $FILE1
2002 $FILE2
```

Wird a2easy ohne Parameter aufgerufen, so erfolgt die Ausgabe eines Textes zum Gebrauch der Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Hier gelangen Sie zum Aufruf und den Parameter von a2easy (bitte auf den Link klicken):

- [Aufruf und Parameter von a2easy](#)

Aufruf und Parameter von a2easy

Aufrufkonventionen

```
java -jar a2easy.jar IndexFile WorkingPath [VarName1=Volume [...]] [Options]
```

Erklärung

IndexFile

Archiver-Index-File im Format Transfer.

WorkingPath

Pfad, in dem die Dokument-Files stehen.

VarNameN=Volume

Liste von Variablen, die fix für diesen Lauf vorgegeben werden und nicht im Index stehen müssen.

Options

-lanGERMAN oder -lanENGLISH	Sprache der Meldungen
-allLOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei
-sysABLAGEORT	Standardablageort im EASY-Archiv z.B: \$SAP_TA1. Kann durch die Variable ARCHIVEPATH das Indexfile verändert werden.
-srvSERVER	Hostname oder IP-Adresse des EASY-Servers
-usrUSER	User im EASY-Archiv, der über die Rechte verfügt, ein Dokument im Archiv abzulegen
-pwdPASSWORD	Passwort des Users
-sifSAPINIFILE	SAP Rückmeldung Ini-File (optional)
-deb	Debug Modus (optional)

Beispiel

```
java -jar a2easy.jar $INDEXFILE$ $PATH$ -all$LOGFILENAME$
      -sys$SAP_TA1 -srv10.1.1.1 -usrUSER -pwdPASSWORD
```

Aufrufzeile von a2easy als TransferTask im Archiver.

Platzhalter in Variableninhalt

Folgende Platzhalter können bei der Definition einer Variablen im Variableninhalt verwendet werden:

- \n Zeilenumbruch
- \r Zeilenumbruch
- \" Anführungsstrich
- \' Hochkomma
- \\ Backslash „\“
- \s Leerzeichen (Space)

a2content, a2saperion

Die Programme a2content und a2saperion sind identisch und unterscheiden sich nur im Namen. Aus diesem Grund wird im Folgenden nur a2content behandelt.

Das Programm a2content arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung vom Archiver zu den Content-Services von SAP oder zu den Content-Services anderer Archivsystem- und DMS-Hersteller her, wenn diese dem SAP-Standard entsprechen. Dieses Programm wurde mit Java erstellt und benötigt eine Java VM der Version 1.4.1 oder höher.

Vier wesentliche Aufgaben werden von a2content wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente der Index-Datei zu den Content-Services
- Löschen der erfolgreich transportierten Dateien
- Optional die Rückmeldung ans SAP-System

Als Erweiterung zur Standard-Lösung der Content-Services liefert a2content neben dem zu transportierenden Dokument noch einen zusätzlichen XML-File ab, der die Index-Felder beinhaltet. Dieser XML-File kann optional vom Archivsystem interpretiert werden und ermöglicht so auch die externe Recherche.

Auf Dokumentebene müssen folgende Felder mitgegeben werden:

- MIMETYPE (MIME-Type des Dokuments)
- CONTREP (Content Repository) Ablageort
- COMID (Component ID) Beschreibung des Dokuments
- INSTANCE Instanz des Archivers (optional)

Beispiel	
MIMETYPE	application/pdf
CONTREP	A1
COMID	data

Wird a2content ohne Parameter aufgerufen, so erfolgt die Ausgabe eines Textes zum Gebrauch der Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Hier gelangen Sie zum Aufruf und den Parameter von a2content (bitte auf den Link klicken):

- [Aufruf und Parameter von a2content](#)

Aufruf und Parameter von a2content

Aufrufkonventionen

```
java -jar a2content.jar IndexFile WorkingPath [VarName1=Volume [...]] [Options]
```

Erklärung

IndexFile

Archiver-Index-File im Format Transfer.

WorkingPath

Pfad, in dem die Dokument-Files stehen.

VarNameN=Volume

Liste von Variablen, die fix für diesen Lauf vorgegeben werden und nicht im Index stehen müssen.

Options

-lanGERMAN oder -lanENGLISH	Sprache der Meldungen
-allLOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei
-srvURL	Erster Teil der URL der Content-Services
-sifSAPINIFILE	SAP Rückmeldung Ini-File (optional)
-deb	Debug Modus (optional)

Beispiel

```
java -jar a2content.jar $INDEXFILE$ $PATH$ -all$LOGFILENAME$
-srvhpt://10.1.1.1/content/content.exe?create&pVersion=5045
```

Aufrufzeile von a2content als TransferTask im Archiver.

Platzhalter in Variableninhalt

Folgende Platzhalter können bei der Definition einer Variablen im Variableninhalt verwendet werden:

- \n Zeilenumbruch
- \r Zeilenumbruch
- \" Anführungsstrich
- \' Hochkomma
- \\ Backslash „\“
- \s Leerzeichen (Space)

a2enaio

Das Programm a2enaio arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung vom Archiver zum Archivsystem enaio der Firma OPTIMAL SYSTEMS GmbH dar. Dieses Programm wurde mit Java erstellt und benötigt eine Java VM der Version 1.8. oder höher. Mindestvoraussetzung ist der enaio-appconnector ab der Version 8.50.

Drei wesentliche Aufgaben werden von a2os wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente und der Indexinformationen zum enaio-appconnector
- Löschen der erfolgreich transportierten Dateien

Zur Übergabe der Dokumente an den Archiv-Server werden nicht nur die Index-Variablen des Dokuments benötigt, sondern auch Felder zur Bestimmung des Ablageortes. Der Ablageort eines Dokuments ist ein Ordner und optional darin ein Register, wobei sowohl Ordner als auch Register eigene Attribute haben. Um den Ablageort zu bestimmen, werden Variablen zur Bestimmung des Ordners benötigt. Diese Ordner-Variablen im Indexfile beginnen immer mit dem Prefix "Cabinet.". Register-Variablen zur optionalen Bestimmung des Registers beginnen immer mit dem Prefix "Register.".

Beispiel

```
Cabinet.Name=Ordnername, bzw. Register.Name=Registername
```

Alle anderen Variablen sind Variablen auf Dokumentebene mit dem Spezialfall der Variable DOCTYPE, die den Dokumenttyp im OS:DRT bestimmt.

Beispiel

Set von Index-Variablen, die zur Ablage in der Objekthierarchie des OS:DRT führen.

```
Cabinet.Name=KUNDE-B1
Cabinet.Jahr=2003
Cabinet.Monat=11
Cabinet.Quartal=3
FormName=Rechnung
Kundennr=123456 Beleg-Nr=789012
```

Die a2anaio prüft selbständig das Vorhandensein aller in der OS Objektdefinition spezifizierten Pflicht- und Schlüsselfelder und legt Ablageorte bei Bedarf selbständig an.

Hier gelangen Sie zum Aufruf und den Parameter von a2enaio (bitte auf den Link klicken):

- ***[Aufruf und Parameter von a2enaio](#)***

Aufruf und Parameter von a2enaio

Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Aufrufkonventionen

```
java -Djava.ext.dirs=lib -cp config de.profiforms.a2enaio.a2enaio IndexFile WorkingPath [Options]
```

Erklärung

IndexFile

Archiver-Index-File im Format Transfer.

WorkingPath

Pfad, in dem die Dokument-Files stehen, welche übertragen werden sollen.

Options


-ini	Ist der Name der Konfigurationsdatei des a2enaio.
-usrUSER	Ist der Anmeldeame des Users, mit welchem sich a2enaio am enaio-appconnector anmelden muss.
-pwdPASSWORT	Ist das Passwort des Users, mit welchem sich a2enaio am enaio-appconnector anmelden muss.

Beispiel

```
java -Djava.ext.dirs=lib -cp config de.profiforms.a2enaio.a2enaio
$INDEXFILE$ $PATH$
    -usrPROFIFORMS -pwdPROFIFORMS -iniconfig/a2enaio.properties
```

Aufrufzeile von a2enaio als TransferTask im Archiver.

a2comp (obsolete)

 a2comp wird nicht mehr genutzt und gilt als veraltet!

Das Programm a2comp arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung vom Archiver zum GlobalSoft Digital working Archiv her. Dieses Programm existiert nur unter Windows NT/2000/XP. Unmittelbar verbunden mit diesem Programm ist die Compaq-DLL „JetFormsArchiver.dll“, die sich im rechtlichen Besitz der Firma GlobalSoft Digital working befindet. Damit a2comp erfolgreich laufen kann, wird auf dem Rechner die Installation des GlobalSoft Digital working Archiv ViewClients benötigt.

Drei wesentliche Aufgaben werden von a2comp wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente der Index-Datei zum GlobalSoft Digital working Archiv
- Löschen der erfolgreich transportierten Dateien

Zur Übergabe der Dokumente an das Compaq Archiv werden folgende Index-Felder benötigt:

- OpticalArchiveld (Bezeichnung des optischen Archivs)
- Barcodeld (Kennzeichnung des Dokuments)
- SapDocumentType (Verarbeitungskennzeichen)


Diese drei Variablen sind pro Dokument zwingend notwendig. Alle anderen Variablen werden ignoriert.

Wird a2comp ohne Parameter aufgerufen, so erfolgt die Ausgabe eines Textes zum Gebrauch der Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Hier gelangen Sie zum Aufruf und den Parameter von a2comp (bitte auf den Link klicken):

- [**Aufruf und Parameter von a2comp \(obsolete\)**](#)

Aufruf und Parameter von a2comp (obsolete)

 a2comp wird nicht mehr genutzt und gilt als veraltet!

Aufrufkonventionen

a2comp IndexFile WorkingPath VarName1=Volume [VarName2=Volume [...]] [Options]

Erklärung

IndexFile

Archiver-Index-File im Format Transfer.

WorkingPath

Pfad, in dem die Dokument-Files stehen.

VarNameN=Volume

Liste von Variablen, die fix für diesen Lauf vorgegeben werden und nicht im Index stehen müssen.

Options

-s	Die Arbeit erfolgt ohne Ausgaben auf dem Bildschirm (Silent).
-lanGERMAN oder -lanENGLISH	Sprache der Meldungen.
-allLOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei.

Beispiel:


```
a2comp indexfile c:\Arch\out\_102 SapDocumentType=YET -allLOG
```

Platzhalter in Variableninhalt

Folgende Platzhalter können bei der Definition einer Variablen im Variableninhalt verwendet werden:

- \n Zeilenumbruch
- \r Zeilenumbruch
- \" Anführungsstrich
- \' Hochkomma
- \\ Backslash „\”
- \s Leerzeichen (Space)

a2hyp (obsolete)

 a2hyp wird nicht mehr genutzt und gilt als veraltet!

Das Programm a2hyp arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung vom Archiver zum HYPARCHIV her. Dieses Programm existiert nur unter Windows NT/2000/XP. a2hyp basiert technisch auf der OLE-Typbibliothek, die mit AutoArchiv von GFT ausgeliefert wird. Vor der Benutzung von a2hyp muss diese Bibliothek installiert worden sein. Um eine Verbindung zum HYPARCHIV aufbauen zu können, muss die Kommunikation zu HYPARCHIV konfiguriert werden. Über den Parameter `-c` ruft a2hyp den Dialog „Department-Verbindungen“ der OLE-Typbibliothek auf. Konfigurieren Sie an dieser Stelle die Zugangsart und die Zugangsberechtigung für Ihr HYPARCHIV.

Drei wesentliche Aufgaben werden von a2hyp wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente der Index-Datei zum HYPARCHIV
- Löschen der erfolgreich transportierten Dateien

Zur Übergabe der Dokumente an das HYPARCHIV wird folgendes Index-Feld benötigt:

- FORMNAME


Beim Aufruf des Programms müssen das Department und der Server richtig angegeben werden. Ebenso muss jedes Dokument ein Indexfeld mit dem Namen FORMNAME enthalten. Das hier angegebene Formular muss tatsächlich in HYPARCHIV existieren. a2hyp archiviert nur die Index-Felder, die zum angegebenen Formular in HYPARCHIV definiert sind.

Wird a2hyp ohne Parameter aufgerufen, so erfolgt die Ausgabe eines Textes zum Gebrauch der Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Hier gelangen Sie zum Aufruf und den Parameter von a2hyp (bitte auf den Link klicken):

- [**Aufruf und Parameter von a2hyp \(obsolete\)**](#)

Aufruf und Parameter von a2hyp (obsolete)

 a2hyp wird nicht mehr genutzt und gilt als veraltet!

Aufrufkonventionen

a2hyp IndexFile WorkingPath [VarName1=Volume [...]] [Options]

Erklärung

IndexFile

Archiver-Index-File im Format Arcis/Teradoc/Compaq.

WorkingPath

Pfad, in dem die Dokument-Files stehen.

VarNameN=Volume

Liste von Variablen, die fix für diesen Lauf vorgegeben werden und nicht im Index stehen müssen.

Options

-s	Die Arbeit erfolgt ohne Ausgabe auf dem Bildschirm (Silent).
-lanGERMAN oder -lanENGLISH	Sprache der Meldungen.
-allLOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei.
-depDEPARTMENT	Name Department, ist eine Pflichtangabe.
-srvSERVER	Servername, ist eine Pflichtangabe.
-d	Debug, ist eine optional Angabe.
-c	Dialog zur Konfiguration des HYPARCHIVs sowie zur Angabe der Anmeldedaten. Ist eine optionale Angabe.

Beispiel

```
a2hyp -c
```

Konfiguration der Anmeldeinformationen.

Beispiel

```
a2hyp -d
```

Konfiguration der Trace- und Debug-Funktion.

Beispiel

```
a2hyp $INDEXFILE$ $PATH$ -depDEPARTMENT1 -srvSERVER1 -all$LOGFILENAME$
```

Aufrufzeile von a2hyp als TransferTask im Archiver.

Platzhalter in Variableninhalt

Folgende Platzhalter können bei der Definition einer Variablen im Variableninhalt verwendet werden:


- \n Zeilenumbruch
- \r Zeilenumbruch
- \" Anführungsstrich
- \' Hochkomma
- \\ Backslash „\“
- \s Leerzeichen (Space)

Jedes Dokument, das dem Archivsystem übergeben wird, muss eine Variable mit dem Namen FORMNAME enthalten.

a2hyp verbindet sich mit HYPARCHIV und informiert sich darüber, welche Variablen beim angegebenen FORMNAMEN zu archivieren sind, sucht diese Variablen im aktuellen Dokument und leitet diese an das Archivsystem weiter. Enthält das Dokument nicht eine einzige der geforderten Variablen, so gibt a2hyp eine Liste aller geforderten Variablen in den LogFile aus.

Hilfe zum Dialog finden Sie in der Beschreibung zu HYPARCHIV.

a2profile (obsolete)

 a2profile wird nicht mehr genutzt und gilt als veraltet!

Das Programm a2profile arbeitet als Transfer-Task im Archiver und stellt die Transportverbindung vom Archiver zum Archivsystem PRO.FILE der Firma ProCAD dar. Dieses Programm existiert nur unter Windows NT/2000/XP. Unmittelbar verbunden mit diesem Programm ist die COM-DLL "checkin.dll", die sich im rechtlichen Besitz der Firma PROCAD befindet und nicht Lieferbestandteil des Archivers ist.

Drei wesentliche Aufgaben werden von a2profile wahrgenommen:

- Lesen einer Index-Datei nach dem Format Transfer
- Transport der Dokumente der Index-Datei zum Archivsystem PRO.FILE
- Löschen der erfolgreich transportierten Dateien

Zur Übergabe der Dokumente an das Archivsystem PRO.FILE wird des folgende Index-Feld pro Dokument benötigt:


- DOCTYPE - PRO.FILE-Dokumenttypkürzel (dieser Parameter ist case-sensitiv)

Wird a2profile ohne Parameter aufgerufen, so erfolgt die Ausgabe eines Textes zum Gebrauch der Kommandozeilen-Parameter. Diese Funktion ermöglicht die Bedienung auch ohne Dokumentation.

Hier gelangen Sie zu den Unterseiten von a2profile (bitte auf den jeweiligen Link klicken):

- [***Aufruf und Parameter von a2profile \(obsolete\)***](#)
- [***Feldzuordnung innerhalb der Indexdatei \(obsolete\)***](#)
- [***Installation \(obsolete\)***](#)

Aufruf und Parameter von a2profile (obsolete)

 a2profile wird nicht mehr genutzt und gilt als veraltet!

Aufrufkonventionen

a2profile IndexFile WorkingPath VarName1=Volume [VarName2=Volume [...]] [Options]

Erklärung

IndexFile

Archiver-Index-File im Format Transfer.

WorkingPath

Pfad, in dem die Dokument-Files stehen.

VarNameN=Volume

Liste von Variablen, die fix für diesen Lauf vorgegeben werden und nicht im Index stehen müssen.

Options

-s	Die Arbeit erfolgt ohne Ausschriften auf dem Bildschirm. (Silent)
-lanGERMAN oder -lanENGLISH	Sprache der Meldungen
-allLOGFILENAME	Protokoll aller Aktivitäten in der angegebenen Datei
-pfdDATABASE	Datenbank-Initialisierung z. B. profile!<servername>
-pffFILESERVER	Name des Fileserver
-pfuUSERNAME	Benutzer zum Einloggen ins PRO.FILE
-pfpPASSWORT	Passwort des Benutzers

Beispiel


```
a2profile C:\arc\indexdatei C:\arc -alllog
          -PFdschl!rs2k-mobil -PFFrs2k-mobil -PFUprocad -PFpprocad
```

Platzhalter in Variableninhalt

Folgende Platzhalter können bei der Definition einer Variablen im Variableninhalt verwendet werden:

- \n Zeilenumbruch
- \r Zeilenumbruch
- \" Anführungsstrich
- \' Hochkomma
- \\ Backslash „\“
- \s Leerzeichen (Space)

Feldzuordnung innerhalb der Indexdatei (obsolete)

 a2profile wird nicht mehr genutzt und gilt als veraltet!

Beispiel

```
DOK=1
DOCTYPE=Ras
FDOC_VFLO1=111001
FDOC_VFLO2=12.12.2002
FDOC_VSTL1=01234567890123456789012345678901234567890123456789
FILE1=beleg1.pdf
```

Die Übergabe des Dokumenttyps ans PRO.FILE erfolgt über die Variable DOCTYPE. Hierbei handelt es sich um das PRO.FILE-Dokumenttypkürzel. Bitte achten Sie auf Groß- und Kleinschreibung, denn der Wert ist case-sensitiv.

Feldbenennung für PRO.FILE-Felder.

Aus der Titelübersicht in der PRO.FILE-MMC können die Datenbankfelder entnommen werden.


Dabei muss folgende Umsetzung berücksichtigt werden:

z. B. wird aus dem Feld DO_FSTL1 der Feldbezeichner FDOC_FSTL1
DV_VFLO1 FDOC_VFLO1

Einschränkung

Auf interne Felder wie Ersteller, Erstellungsdatum des Dokumentes usw. kann nicht geschrieben werden.

Installation (obsolete)

 a2profile wird nicht mehr genutzt und gilt als veraltet!

Folgende Programme werden benötigt:

1. a2profile.exe wird vom Archiver aufgerufen und bildet die Schnittstelle zu PRO.FILE
2. checkin.dll COM-Schnittstelle inklusive PRO.FILE-Funktionen, stellt also das „Arbeitstier“ dar.

Die DLL „checkin“ muss registriert werden. Diese kann z. B. in das Directory

C:\Programme\Gemeinsame Dateien\Procad\procheckin

kopiert werden und mit dem Befehl

```
regsvr32 C:\Programme\Gemeinsame Dateien\Procad\procheckin\checkin.dll
```

registriert werden.

Konfiguration des Archivers

Der Archiver speichert und merkt sich alle Konfigurationen in einer Ini-Datei. Wird der Archiver ohne die Übergabe eines Namens für die Ini-Datei gestartet, so wird standardmäßig die Datei „archiver.ini“ im lokalen Verzeichnis verwendet. Der Inhalt dieser Datei kann erstellt und verändert werden mit einem gewöhnlichen ASCII-Text-Editor.

Der Inhalt der Ini-Datei ist in Reiter bzw. Sektionen gegliedert. Jede dieser Sektionen beinhaltet Einstellungen zu einem bestimmten Thema.

Die Konfiguration gliedert sich in folgende Unterkapitel (bitte auf den jeweiligen Link klicken):

- **Einstellungen**
 - **Pfade**
 - **Tasks**
 - **Scannen**
 - **LogFile**
 - **Namen, Verfahren und Format**
 - **PDF Verarbeitung**
 - **PDF Erzeugung**
 - **Sprache/Seriennummer**
- **Formate**
 - **Arcis/Teradoc**
 - **Easy**
 - **FileNET**
 - **DocuWare**
 - **IXOS**
 - **Ceyoniq**
 - **Transfer**
 - **SER (obsolete)**
 - **ProFile (obsolete)**

Einstellungen

Die Einstellungen gliedern sich in folgende Unterkapitel (bitte auf den jeweiligen Link klicken):

- [*Pfade*](#)
- [*Tasks*](#)
- [*Scannen*](#)
- [*LogFile*](#)
- [*Namen, Verfahren und Format*](#)
- [*PDF Verarbeitung*](#)
- [*PDF Erzeugung*](#)
- [*Sprache/Seriennummer*](#)

Pfade

Quellverzeichnisse (InPath)	Die Quellverzeichnisse geben an, wo der Archiver die zu verarbeitenden Archiver-Dateien sucht. Gesucht werden nur .dat-Dateien. Bei der Syntax ist zu beachten, dass die angegebenen Verzeichnisse durch ein Leerzeichen voneinander getrennt werden müssen.
Dateiendung (Ext)	Die Dateiendung nach der in den Quellverzeichnissen gesucht wird. Standardwert ist „.DAT“
Ausgabeverzeichnis (OutPath)	Das Ausgabeverzeichnis gibt an, in welches Verzeichnis der Archiver die Dokumente für das Archivsystem hinein generieren soll. Wird ohne eine Transfer-Task gearbeitet, so ist dies gleichzeitig das Import-Verzeichnis für das Archivsystem.
Fehlerverzeichnis (ErrorPath)	Das Fehlerverzeichnis gibt an, in welchem Verzeichnis die Informationen über eventuell auftretende Fehler gespeichert werden sollen. Bei einem fehlerhaften Durchlauf wird neben der Input-Datei, die den Fehler verursachte, noch der Auszug aus dem Log gesichert, der die kompletten Informationen der Verarbeitung dieser Datei bis zum Fehler aufzeichnet.

Beispiel

```
Locations {
  InPath c:\archiver\in\
  Ext .pdf
  OutPath c:\archiver\out\
  ErrorPath c:\archiver\error\
}
```

Tasks

Tasks sind externe Programme, die vom Archiver aufgerufen werden können. Im Archiver können zwei Tasks definiert werden. Eine Converter-Task, um PDF-Dokumente vor der Archivierung noch einmal in ein anderes Format zu wandeln oder die PDF-Datei in Einzelseiten aufzuteilen.

Weiterhin gibt es noch eine Transfer-Task, um fertige Archiv-Dokumente direkt in ein Archiv-System zu transportieren.

Die Angabe der Programme für Task muss mit kompletter Pfadangabe erfolgen. Ebenso die Angaben für Ini-Dateien und andere fixe Ressourcen. Vor die Variablen des Archiviers sind grundsätzlich keine Pfade zu schreiben. Alle Variablen kommen mit kompletten Pfadangaben.

ConvertTask	Optionales Programm zur Konvertierung eines Archiv-PDFs in andere Dokument-Formate (z.B. TIFF).
TransferTask	Optionales Programm zum Abtransport der Daten direkt ins Archiv-System hinein.

In jeder Task-Definition gibt es eine Reihe von Variablen, deren Angabe Pflicht oder optional ist.

Variable	ConverterTask	TransferTask
\$PRINTFILE\$	Pflicht	—
\$STARFILES\$	Pflicht	—
\$LOGFILE\$	optional	optional
\$INDEXFILE\$	—	Pflicht
\$PATH\$	—	optional
\$DOCTYPE\$	—	optional

Weiterhin kann jede Variable des Vorab-Index beim Aufruf mit angegeben werden. Dazu muss die Variable groß geschrieben und in Dollarzeichen gesetzt werden, so wie dies von der Syntax auch für alle anderen Variablen gilt.

Zur ConverterTask existiert noch ein weiterer Parameter SinglePages. Dieser ist speziell für die Verwendung von GhostScript als Konverter gedacht. GhostScript ist in der Lage, für jede Seite eines konvertierten Dokuments eine separate Datei zu erstellen. Hierfür benötigt GhostScript im Dateinamen beim Aufruf einen Platzhalter „%d“. Ist die Option SinglePages bzw. „ConverterTask soll Einzelseiten erzeugen“ aktiv, dann arbeitet der Archiver mit dieser Funktionalität.

Beispiel

```
Execute {
  ConvertCommand
  SinglePages 0
  TransferCommand
  ...
}
```

Scannen

Der Archiver ist in der Lage seine Input-Verzeichnisse zyklisch nach neuen Archiv-Dateien zu untersuchen. Durch die Angabe einer Zeit in Sekunden kann vom Nutzer das Intervall der Untersuchung bestimmt werden. Ist das automatische Scannen deaktiviert, so scannt der Archiver nur einmal beim Start des Programms.

Beispiel

```
Scan {  
  Automatic 1  
  Seconds 1  
}
```

LogFile

Der Name des Archiver-LogFiles kann mit Variablen versehen sein, die beim Zugriff auf den LogFile durch konkrete Werte ersetzt werden. Diese Technik erlaubt es, Tages-, Monats- oder Datums-spezifische LogFiles anzulegen. Folgende Variablen sind im Namen des LogFiles erlaubt:

Die Zeitangabe

Syntax	Beschreibung	Beispiel
\$HOUR\$	Stunde, in der die Datei erstellt wurde.	\$15\$
\$MIN\$	Minute, in der die Datei erstellt wurde.	\$23\$
\$SEC\$	Sekunde, in der die Datei erstellt wurde.	\$10\$

Die Tagesangabe

Syntax	Beschreibung	Beispiel
\$DD\$	der Tag als zweistellige Zahl mit Punkt	\$17.\$
\$WD\$	der Tag als zweistellige Zahl Zuweisung erfolgt aufsteigend: 0=So, 1=Mo, ...	\$0\$
\$DAY\$	der englische Tagname als Text	\$Monday\$
\$TAG\$	der deutsche Tagname als Text	\$Montag\$

Die Monatsangabe

Syntax	Beschreibung	Beispiel
\$MM\$	Monatsangabe als zweistellige Zahl	\$03\$
\$MONTH\$	Monatsname in Englisch als Text	\$October\$
\$MONAT\$	Monatsname in Deutsch als Text	\$Oktober\$

Die Jahresangabe

Syntax	Beschreibung	Beispiel
\$YYYY\$	Die Jahreszahl vierstellig.	\$2000\$
\$YY\$	Die Jahreszahl zweistellig.	\$00\$
\$Y\$	Die Jahreszahl einstellig.	\$0\$

Ebenso, wie der Name des LogFiles konfigurabel ist, ist der Inhalt des LogFiles einstellbar. Für den Aufbau des Protokolls stehen folgende Konfigurationsmöglichkeiten zur Verfügung:

Vollständiges Log-Protokoll (Complete)	<p>Alle Aktivitäten und alle Rückmeldungen gestarteter Programme werden mit ins Log übernommen. Diese Option ist vor allem für die Fehlersuche geeignet.</p> <p>Ist die Option nicht aktiviert, so wird ein Kurz-Log ausgegeben, welches normalerweise ausreicht. Entdeckt der Archiver einen Fehler in der Verarbeitung, so schaltet er für dieses Dokument temporär auf das vollständige Log-Protokoll um.</p>
Rückgabewerte des Transfer-Programms werden immer mit ins Log übernommen (Transfer)	<p>Diese Option dient zur Erzeugung eines vollständigen Logs nur für das Transfer-Programm, wenn alle anderen Aktivitäten mit einem Kurz-Protokoll auskommen.</p>
Die erste Variable als Belegnummer im Log-File anzeigen (DocFirstVarInLog)	<p>Viele Anwender haben eigene Belegnummern für Dokumente. Ist die Option aktiviert, so nimmt der Archiver die erste Variable des Vorab-Index und interpretiert diese als Belegnummer. Im anderen Fall verwendet der Archiver eine interne Nummer als Belegnummer.</p>
Die ersten N Variablen des Dokuments werden mit ins Log aufgenommen (DocFirstNVarsInLog)	<p>Zur besseren Übersichtlichkeit kann der Archiver die ersten N Variablen des Vorab-Indexes mit in den LogFile aufnehmen. Dies vergrößert den Speicherbedarf des Log-Files, bringt aber auch mehr Übersichtlichkeit.</p>

Beispiel

```
LogFile {
  Name Archiver_$$DD$. $MM$. $YYYY$.log
  Complete 0
  Transfer 1
  DocFirstVarInLog 1
  DocFirstNVarsInLog 0
}
```

Namen, Verfahren und Format

Namen

Der Archiver generiert die zu archivierenden Dokumente in Dateien. Sowohl der Rumpfname als auch die Extension des Dokument-Namens können vom Anwender bestimmt werden.

File-Rumpf des Ziel-Files (DocName)	Optionaler Rumpfname der Dokument-Datei, dem der Archiver noch weitere Teile hinzufügt
File-Endung des Ziel-Files (Ext)	Optionale File-Ende (Extension) der Dokument-Datei

Verfahren

Unabhängig vom gewählten Archiv-Format kann die Arbeitsweise des Archiviers noch durch allgemeine Verfahrens-Optionen beeinflusst werden:

Eindeutiger Verzeichnis- oder Dateiname über TimeStamp (TimeStamp)	<p>Die Angabe des Verzeichnis- o. Dateinamens über einen TimeStamp kann über einen Datums-Zeitstempel erfolgen. Ist diese Option deaktiviert, so wird eine eindeutige Nummer vergeben.</p> <table border="1" data-bbox="639 1072 1431 1216"> <tr> <td>Beispiel für TimeStamp</td> </tr> <tr> <td>200008111424468</td> </tr> </table>	Beispiel für TimeStamp	200008111424468
Beispiel für TimeStamp			
200008111424468			
Eindeutiger Verzeichnis- oder Dateiname beginnt mit dem Namen der Input-Datei (NameStamp)	Wird von einer übergeordneten Instanz der Erfolg der Archivierung überwacht, so ist es oft notwendig, dass der Datei-Name der Archiver-Datei im Namen des Ausgabeverzeichnis oder der Index-Datei vorkommt.		
Alle Input-Dateien vor der Archivierung zu einer Datei zusammenfassen (CollectInputFiles)	Viele Archivsysteme wollen lieber einen großen Archivlauf als viele kleine. Mit der Option CollectInputFiles werden alle zu einem Scanzeitpunkt gefundenen Archiver-Dateien vor der Archivierung zu einer Datei zusammengefasst.		
Anzahl parallel laufender Threads (NumberOfThreads)	Anzahl der Threads für die parallele Ausführung unter Windows		

Beispiel

```
Execute {
  ...
  DocName beleg
  Ext .pdf
  TimeStamp 0
  NameStamp 0
  CollectInputFiles 0
  NumberOfThreads 1
}
```

Format

Der Archiver bedient unterschiedliche Archivsysteme. Unter Ausgabeformat stehen alle Archiv-Systeme zur Wahl, die zurzeit unterstützt werden.

Beispiel

```
OutputFormat {
  Format ARCIS/TERADOC
}
```

PDF Verarbeitung

Ist das Input-Format PDF, so muss das zu lesende PDF nicht unbedingt frei von einem Passwort sein.

Entweder Sie definieren das Lese-Passwort im Archiver, oder Sie erlauben das Öffnen mit dem System-Passwort. Bitte bedenken Sie, dass PDF/A-Dateien nicht passwortgeschützt sein dürfen.

Passwort zum Öffnen der PDF-Datei (OpenPassword)	<p>Sie können ein Passwort zum Öffnen festlegen. Da Passwörter nicht unverschlüsselt abgelegt werden sollen, konvertiert die Windowsoberfläche das angegebene Passwort sofort in einen NVC-String (None Visible Code) und legt diesen so in der Ini-Datei ab.</p> <p>Anwender, die nicht mit der Oberfläche arbeiten, erreichen den gleichen Effekt mit dem Hilfsprogramm String.</p>
System-Passwort zulassen (AllowSystemPassword)	<p>Die zu lesende Datei kann ohne Passwort oder mit dem System-Passwort versehen sein.</p>

Beispiel
<pre>PDF { OpenPassword %NVC%4aW/4aWn AllowSystemPassword 1 ... }</pre>

PDF Erzeugung

Die PDF-Erzeugung definiert alle Einstellungen, die verwendet werden, wenn das Input-Format Archive-PDF ist. Nur in diesem Fall extrahiert der Archiver oder der ArcDir aus dem Input-PDF einzelne Output-PDFs und nur für die Output-PDFs gelten diese Einstellungen. Bitte bedenken Sie auch die Abhängigkeiten des Output-PDFs vom Input-PDF. So kann die PDF-Versionsnummer des Output-PDFs generell nicht höher sein, als die des Input-PDFs. Wollen Sie als Output-PDF ein PDF/A erstellen, so muss das Input-PDF auch bereits ein PDF/A sein. Bitte bedenken Sie, dass PDF/A-Dateien nicht passwortgeschützt sein dürfen. Sind die Einstellungen des Output-PDFs nicht stimmig oder passen nicht zum Input-PDF, so versucht der Archiver und der ArcDir inkompatible Einstellungen selbständig zu verändern. Entspricht das Output-PDF nicht Ihren Erwartungen, so untersuchen Sie bitte als erstes das Input-Format und als zweites die Einstellungen zur PDF Erzeugung.

MasterPasswort des zu erzeugenden PDFs (MasterPassword)	<p>Das Master-Passwort wird benötigt um PDF-Eigenschaften ändern zu können. Da Passwörter nicht unverschlüsselt abgelegt werden sollen, konvertiert sie Windowsoberfläche das angegebene Passwort sofort in einen NVC-String (None Visible Code) und legt diesen so in der Ini-Datei ab. Anwender, die nicht mit der Oberfläche arbeiten, erreichen den gleichen Effekt mit dem Hilfsprogramm String.</p>
System-Passwort erzeugen (SystemPassword)	<p>Die zu erzeugende PDF-Datei erhält das System-Passwort als Master-Passwort.</p>
UserPassword des zu erzeugenden PDFs	<p>Sie können ein User-Passwort für das zu erzeugende PDF festlegen. Das User-Passwort wird beim Öffnen des PDFs abgefragt. Ohne Master-Passwort kann kein User-Passwort akzeptiert werden. Da Passwörter nicht unverschlüsselt abgelegt werden sollen, konvertiert sie Windowsoberfläche das angegebene Passwort sofort in einen NVC-String (None Visible Code) und legt diesen so in der Ini-Datei ab. Anwender, die nicht mit der Oberfläche arbeiten, erreichen den gleichen Effekt mit dem Hilfsprogramm String.</p>
Version des zu erzeugenden PDFs (PDFVersion)	<p>Version der zu erzeugenden PDF. Die kann die Angabe als Zahl sein oder auch eine spezielle Versionsbezeichnung: Zahl: 1.4 - 2.0 Versionbezeichnungen:</p> <p>PDFA1A2005 PDFA1B2005 PDFA2A PDFA2B bzw. PDFA2 PDFA2U PDFA3A PDFA3B bzw. PDFA3 PDFA3U</p> <p>Die Version des Output-PDFs kann nicht kleiner sein des Input-PDFs. Für PDF/A muss das Input-PDF bereits PDF/A vom selben Typ sein.</p>

MetaDaten erzeugen (MetaData)	<p>In das zu erzeugende PDF werden XMP-basierte Meta-Daten mit den Archivvariablen generiert. Für XMP-fähige Archivsystem erspart das eine zusätzliche Datei mit Index-Variablen.</p>																		
Ticketing-Variablen mit Seitensteuerung kopieren (Ticketing)	<p>Sollen Archivdateien so im Archivsystem abgelegt werden, dass diese wieder mit allen Eigenschaften druckbar sind, so spielen die Ticketing-Variablen eine wichtige Rolle und sollten mit ins Archiv-PDF generiert werden.</p>																		
Kompression (Compression)	<p>Legt die Kompression des PDFs fest. Zulässig sind numerische Werte zwischen 0 und 9. 0=keine Kompression 9=maximale Kompression</p>																		
Zugelassene Rechte (Permissions)	<p>Wenn ein Master-Passwort definiert wurde, ist es möglich, die Rechte an zulässigen Aktionen für den Leser des PDFs einzuschränken. Folgende Einschränkungen sind möglich:</p> <table border="1" data-bbox="687 882 1433 1742"> <tr> <td data-bbox="687 882 965 972">noprint</td> <td data-bbox="965 882 1433 972">Gibt an, ob der Druck des Dokuments verboten ist.</td> </tr> <tr> <td data-bbox="687 972 965 1061">nomodify</td> <td data-bbox="965 972 1433 1061">Gibt an, ob die Veränderung des Dokuments verboten ist.</td> </tr> <tr> <td data-bbox="687 1061 965 1151">nocopy</td> <td data-bbox="965 1061 1433 1151">Gibt an, ob es verboten ist aus dem Dokument herauszukopieren.</td> </tr> <tr> <td data-bbox="687 1151 965 1279">noannots</td> <td data-bbox="965 1151 1433 1279">Gibt an, ob es verboten ist, Bemerkungen zu erstellen oder zu verändern.</td> </tr> <tr> <td data-bbox="687 1279 965 1368">noforms</td> <td data-bbox="965 1279 1433 1368">Gibt an, ob es verboten ist, Formularfelder auszufüllen.</td> </tr> <tr> <td data-bbox="687 1368 965 1458">noaccessible</td> <td data-bbox="965 1368 1433 1458">Gibt an, ob es verboten ist, Texte und Grafiken zu extrahieren.</td> </tr> <tr> <td data-bbox="687 1458 965 1547">noassemble</td> <td data-bbox="965 1458 1433 1547">Gibt an, ob das Hinzufügen von Elementen zum PDF verboten ist.</td> </tr> <tr> <td data-bbox="687 1547 965 1637">nohighresprint</td> <td data-bbox="965 1547 1433 1637">Gibt an, ob der Druck des Dokuments in hoher Auflösung verboten ist.</td> </tr> <tr> <td data-bbox="687 1637 965 1727">plainmetadata</td> <td data-bbox="965 1637 1433 1727">Gibt an, ob Metadaten unverschlüsselt bleiben.</td> </tr> </table> <p>Um Einschränkungen zu kombinieren, werden gewählten Einschränkungen durch Leerzeichen separiert hintereinander geschrieben.</p>	noprint	Gibt an, ob der Druck des Dokuments verboten ist.	nomodify	Gibt an, ob die Veränderung des Dokuments verboten ist.	nocopy	Gibt an, ob es verboten ist aus dem Dokument herauszukopieren.	noannots	Gibt an, ob es verboten ist, Bemerkungen zu erstellen oder zu verändern.	noforms	Gibt an, ob es verboten ist, Formularfelder auszufüllen.	noaccessible	Gibt an, ob es verboten ist, Texte und Grafiken zu extrahieren.	noassemble	Gibt an, ob das Hinzufügen von Elementen zum PDF verboten ist.	nohighresprint	Gibt an, ob der Druck des Dokuments in hoher Auflösung verboten ist.	plainmetadata	Gibt an, ob Metadaten unverschlüsselt bleiben.
noprint	Gibt an, ob der Druck des Dokuments verboten ist.																		
nomodify	Gibt an, ob die Veränderung des Dokuments verboten ist.																		
nocopy	Gibt an, ob es verboten ist aus dem Dokument herauszukopieren.																		
noannots	Gibt an, ob es verboten ist, Bemerkungen zu erstellen oder zu verändern.																		
noforms	Gibt an, ob es verboten ist, Formularfelder auszufüllen.																		
noaccessible	Gibt an, ob es verboten ist, Texte und Grafiken zu extrahieren.																		
noassemble	Gibt an, ob das Hinzufügen von Elementen zum PDF verboten ist.																		
nohighresprint	Gibt an, ob der Druck des Dokuments in hoher Auflösung verboten ist.																		
plainmetadata	Gibt an, ob Metadaten unverschlüsselt bleiben.																		

Voreinstellungen für den Acrobat (ViewerPreferences)	ViewerPreferences steuert die Einstellungen, die das PDF dem Acrobat mitteilt, um sich zu konfigurieren. Folgende Werte sind möglich:	
	centerwindow und nocenterwindow	Gibt an, ob das Dokumentfenster am Bildschirm zentriert wird.
	directionl2r und directionr2l	Definiert die Lesereihenfolge eines Dokuments, welche sich auf das Blättern in der Doppelseitenansicht auswirkt: l2r = von links nach rechts r2l = von rechts nach links
	displaydoctitle und nodisplaydoctitle	Gibt an, ob der Titel des PDFs aus der Info-Struktur in der Acrobat Fensterzeile angezeigt werden soll.
	fitwindow und nofitwindow	Gibt an, ob das Fenster von Acrobat an die Größe der ersten Seite angepasst wird.
	hidemenubar und nohidemenubar	Gibt an, ob die Menüleiste von Acrobat sichtbar ist.
	hidetoolbar und nohidetoolbar	Gibt an, ob die Werkzeugleiste von Acrobat sichtbar ist.
	hidewindowui und nohidewindowui	Gibt an, ob die Fenstersteuerelemente von Acrobat sichtbar sind.
Um Werte zu kombinieren, werden gewählten Werte durch Leerzeichen separiert hintereinander geschrieben		

Beispiel

```
PDF {
  SystemPassword 0
  MasterPassword %NVC%4Zab4Zao4Zec4Ze5
  UserPassword %NVC%4oCd4oCZ
  PDFVersion 1.4
  MetaData 1
  Ticketing 1
  Compression 1
  Permissions noprint nomodify nocopy noannots
  ViewerPreferences centerwindow displaydoctitle
  ...
}
```

Sprache/Seriennummer

Die Oberfläche von Windows präsentiert sich sprachabhängig in Deutsch oder Englisch. Das gilt auch für die Texte des Log-Files.

Beispiel

```
Language {
  Language German
}

oder

Language {
  Language English
}
```

Seriennummer. Die Angabe der Seriennummer ist Pflicht.

Beispiel

```
Execute {
  SerNo XXXXX-XXXXX-XXXXX-XXXXX
  ...
}
```

Formate

Die Formate gliedern sich in folgende Unterkapitel (bitte auf den jeweiligen Link klicken):

- [*Arcis/Teradoc*](#)
- [*Easy*](#)
- [*FileNET*](#)
- [*DocuWare*](#)
- [*IXOS*](#)
- [*Ceyoniq*](#)
- [*Transfer*](#)
- [*SER \(obsolete\)*](#)
- [*ProFile \(obsolete\)*](#)

Arcis/Teradoc

Die Archivsysteme Arcis und Teradoc werden hier zusammengefasst, da diese dasselbe Import-Format benutzen.

Rumpfname des Index-Files (IndexName)	Der Index-File wird immer in ein SubDirectory hinein generiert. Diese Angabe ist Pflicht. Der Index-File entsteht erst am Ende eines Archivlaufes. Die Erzeugung dieses Files ist atomar. Wenn dieser File erzeugt wurde, kann das Archivsystem mit dem Import der Daten beginnen.
Schlüsselwort FILE im IndexFile wird inkrementell hochgezählt (IncFilePostFix)	Erzeugt ein Beleg mehrere Files, so kann gewählt werden, ob das Schlüsselwort FILE mehrfach vorkommt oder je ein FILE1, FILE2 usw. erzeugt wird.
Trailer anhängen (Trailer)	An das Ende der Index-Datei wird ein Trailer mit Statusinformationen generiert.
In Unterverzeichnis generieren (InSubDir)	Die Ausgabedateien werden pro Lauf in ein eigenes Unterverzeichnis generiert.

Umschlüsselung der Variablen

Für dieses Format ist es nicht unbedingt notwendig, die Variablen des Vorab-Index nach Arcis und Teradoc umzuschlüsseln. Fehlen die Angaben zum Umschlüsseln gänzlich, so werden die Variablen des Vorab-Index direkt in den Index für Arcis und Teradoc übernommen. Soll umgeschlüsselt werden, so kann dies unter dem Eintrag Vars geschehen.

Syntax

ARCISVAR VARNAME[,VARNAME[...]]

Variablen, die FileNamen mit in den Index-File bringen, sind nicht notwendig; dies ist durch das Format sichergestellt. Das Arcis- und Teradoc-Interface unterstützt die Arbeit eines Transfer-Programms.

Beispiel

```
ARCIS/TERADOC {
  IndexName indexdatei
  IncFilePostFix 0
  Vars {
    ARCISVAR1 VORABVAR1A,VORABVAR1B
    ARCISVAR2 VORABVAR2
  }
}
```

Easy

Der Archiver unterstützt die File-Schnittstelle von EASY-Archiv bis einschließlich der Version 3.3. Damit ist der Archiver in der Lage, auch mehrere Dokument-Typen in ein Index-File zu generieren. Das EASY-Interface unterstützt die Arbeit eines Transfer-Programms. Der Archiver ermittelt den Dokument-Typ entsprechend des Eintrages, der im Feld Dokument-Typ definiert wurde. Das Feld Dokument-Typ enthält einen String, der eine oder mehrere Variablen beinhalten kann, die zum Zeitpunkt der Erkennung aufgelöst werden. Für jeden Dokument-Typ lassen sich im Archiver zwei Tabellen anlegen. In der ersten Tabelle können Variablen umgeroutet werden, in der zweiten Tabelle wird die Ausgabe der Index-Information für das Archivsystem beschrieben.

Wird der Archiver auf einer Windows-Plattform betrieben, so kann als Transfer-Task das Programm EZCBIP32.EXE der Firma EASY verwendet werden. Hierbei werden die Dokumente über das Transferprogramm direkt ins EASY-Archiv geliefert. Leider liefert EZCBIP32.EXE keine Rückmeldungen über fehlerhafte Archivierungsversuche.

Rumpfname des Index-Files (IndexName)	Name des Index-Files. Wird hier nichts gesetzt, dann muss zumindest die File-Endung gesetzt sein. Der Index-File entsteht erst am Ende eines Archivlaufes. Die Erzeugung dieses Files ist atomar. Wenn dieser File erzeugt wurde, kann das Archivsystem mit dem Import der Daten beginnen.
File-Endung des Index-Files (IndexExt)	Name der Extension des Index-Files. Wird hier nichts gesetzt, dann muss zumindest der Rumpfname gesetzt sein.
ins Sub-Directory generieren (InSubDir)	Die Ausgabe der Dokumente erfolgt in ein Unterverzeichnis des Aufgabepfades. Der Index-File wird in ein Sub-Directory generiert.
Pro Dokument-Typ ein Ausgabeverzeichnis (InTypeDir)	Pro erkanntem Dokument-Typ wird ein Ausgabeverzeichnis angelegt, in das die zugehörigen Dokumente generiert werden.
Header Title (HeaderTitle)	Jeder Header einer EASY-Index-Datei benötigt einen Titel, anhand dessen EASY feststellen kann, um welche Art von Dokumenten es sich handelt, und dementsprechend die Dokumente klassifizieren kann.
Lokaler EASY-Pfad (LocalPath)	Pfad-Angabe des OutPath aus Sicht des Easy-Archivsystems (DOS-Angabe, kein UNC-Name)

Umschlüsselung der Variablen

Für dieses Format ist es notwendig, die Variablen des Vorab-Indexes nach Easy-Feldern umzuschlüsseln.

Syntax

EASYFieldNo VARNAME[,VARNAME[...]]

Im Easy-Archiv ist die Größe eines Textfeldes auf 32k Byte begrenzt. Felder, die größer sind, werden in mehrere kleinere Felder gleichen Namens aufgeteilt, da Mehrfachnennung von Easy-Archiv unterstützt wird. Dies ist ein Automatismus in der Schnittstelle, der Fehlarchivierungen und Abbrüche verhindern hilft.

Die Namen der generierten Files sind nicht automatisch Bestandteil des Easy-Index-Files. Diese müssen separat angegeben werden und nennen sich FILE1 bis FILEn. Wird nicht in mehrere Seiten gesplittet, so reicht die Angabe von FILE1.

Beispiel

```

EASY {
  LocalPath d:\jetform\in\
  HeaderTitle Archiver
  IndexName index
  IndexExt .idx
  InSubDir 1
  InTypeDir 0
  DocType $MODULE_ID$
  DocTypes {
    DocType RECHNUNG {
      HeaderTitle IS-U Rechnung
      Vars {
        1001 OPBLT
        1002 LNDAT
        9001 FILE1
      }
    }
    DocType LIEFERSCHEIN {
      HeaderTitle Material Lieferschein
      Vars {
        1001 BELEGNUMMER
        1002 DRUCKDATUM
        1003 KUNDENNUMMER
        9001 FILE1
      }
    }
    DocType DEFAULT {
      HeaderTitle Allgemeines Dokument
      Vars {
        1001 BELNR
        1002 DAT
        9001 FILE1
      }
    }
  }
}

```

FileNET

Das FileNET-Interface arbeitet nach einem zweistufigen Verfahren. Im ersten Schritt werden die Dokumente erzeugt und der Index-File angelegt. Im zweiten Schritt erfolgt der direkte Abtransport dieser Dateien zum Archivsystem über das Transfer-Programm Batchlt. Batchlt wird von FileNET zur Verfügung gestellt und existiert mit Sicherheit für WindowsNT und einige UNIX-Plattformen wie AIX und HP-UX. Batchlt darf vom Archiver nur singular aufgerufen werden. Aus diesem Grund muss die Anzahl der Threads auf 1 stehen.

Dokumente pro Paket (PackageSize)	Anzahl der Dokumente in einem Index-File 0 = keine Beschränkung		
Files pro Paket (PackageFiles)	Anzahl der Files in einem Index-File 0 = keine Beschränkung		
DocClass (DocClass)	<p>Es kann für die Verarbeitung optional eine Dokumentenklasse gesetzt werden. Diese kann hier als fixer Text gesetzt werden oder als Verweis auf eine Variable des Vorab-Index geschehen.</p> <table border="1" data-bbox="336 1025 1431 1211"> <tr> <th data-bbox="336 1025 1431 1093">Beispiel</th> </tr> <tr> <td data-bbox="336 1093 1431 1211">Angenommen, es gibt eine Vorab-Index-Variable ARCHIVETYPE. So wäre \$ARCHIVETYPE\$ eine mögliche Angabe für die DocClass.</td> </tr> </table>	Beispiel	Angenommen, es gibt eine Vorab-Index-Variable ARCHIVETYPE. So wäre \$ARCHIVETYPE\$ eine mögliche Angabe für die DocClass.
Beispiel			
Angenommen, es gibt eine Vorab-Index-Variable ARCHIVETYPE. So wäre \$ARCHIVETYPE\$ eine mögliche Angabe für die DocClass.			
Ready-File mit Protokoll anlegen (ReadyFileWithProtocol)	Am Ende eines jeden Archivlaufes wird eine Datei mit dem Namen „ready“ erzeugt, die das Protokoll dieses Archivlaufes enthält. Vorsicht: Wird diese Option gewählt, so kann der Archiver nach erfolgreichem Transport der Daten ins Archivsystem das ansonsten leere Unterverzeichnis nicht automatisch löschen.		

Umschlüsselung der Variablen

Für dieses Format ist es notwendig, die Variablen des Vorab-Index nach FileNET-Feldern umzuschlüsseln.

Syntax

FileNETVariable VARNAME[VARNAME[...]]

Variablen, die FileNamen mit in den Index-File bringen, sind nicht notwendig; dies ist durch das Format sichergestellt.

Beispiel

```
FileNET {
  PackageSize 0
  PackageFiles 9000
  DocClass
  ReadyFileWithProtocol 0
  Vars {
    TITEL_RECHNUNG JF_TITEL_RECHNUNG
    RE_NR JF_RE_NR
    RE_DATUM JF_RE_DATUM
    RE_NAME1 JF_RE_NAME1
    RE_NAME2 JF_RE_NAME2
    KUNDEN_NR JF_KUNDEN_NR
    WAEHRUNG JF_WAEHRUNG
    P_ENDBETRAG JF_P_ENDBETRAG
  }
}
```

DocuWare

Die Anbindung an das DocuWare-Archivsystem geschieht über die Schnittstelle ACTIVE IMPORT. Diese Schnittstelle unterstützt nur Dokumente, die nicht gesplittet sind. Auch die Arbeit mit Unterverzeichnissen wird nicht von ACTIVE IMPORT unterstützt. Aus diesem Grund werden alle Dokumente und Index-Files direkt in das Verzeichnis von OutPath generiert. Jedes erzeugte Dokument benötigt für die Schnittstelle eine eigene Index-Datei.

File-Endung des MetaIndexFiles (IndexExt)	File-Endung (Extention) des MetaIndexFiles, der für jedes Dokument erzeugt wird.
--	--

Das Index-Format listet hintereinander in einer Zeile alle Variablenwerte in Trennzeichen geklammert. Die Reihenfolge, in der die Variablen vom Archiver geschrieben werden, muss mit der Reihenfolge übereinstimmen, die das Archivsystem erwartet. Stimmen Sie die Ende-Kennung (Extention) des Index-Files und die Variablen-Reihenfolge mit dem DocuWare-Systembetreuer ab. Die Angabe einer Variable zur Ausgabe erfolgt wie in folgender Syntax gezeigt.

Syntax

VARNAME[,VARNAME[...]] [N]

Erklärung

Das N steht für Numeric und bedeutet, dass dieser Wert ohne Anführungszeichen in den Index-File geschrieben wird. Diese Angabe ist optional. Standard ist nicht numerisch. Variablen, die den File-Namen beinhalten, sind in diesem Format nicht notwendig; dies wird durch die Namensgleichheit des Dokumenten- und des Index-Files erreicht.

Beispiel

```
DocuWare {
  IndexExt .idx
  OneIndexFile 0
  Separator #
  Quotes
  Prefix #
  PostFix #
  Vars {
    X_ADRESS_NAME
    WA_DOC_HEADER_OPBEL
    X_GPNR
    X_VKNR
    XX_DATUM
    FILE
  }
}
```

IXOS

Mit dieser Schnittstelle wird das Meta-Dokument-Interface des Archivsystems IXOS bedient. Diese Schnittstelle unterstützt nur Dokumente, die nicht gesplittet sind. Pro Importsatz werden folgende vier Dateien erzeugt: COMMANDS, IXATTR, data.pdf bzw. data.tif und log. In einer IXATTR-Datei bzw. data.-Datei können ein oder mehrere Dokumente vorkommen. Wenn mehrere Dokumente vorkommen, ist der Batch-Mode aktiv. Andernfalls wird pro Dokument ein Ausgabesatz erstellt. Für den produktiven Einsatz wird der Batch-Mode empfohlen, da dieser einen schnelleren Import erlaubt.

Das Feld COMMANDS beinhaltet den Text-Inhalt der COMMANDS-Datei. Das Feld IXATTR enthält den Text-Inhalt eines Dokumentes in der IXATTR-Datei. Beide Felder können Variablen beinhalten, die in der üblichen Form \$VarName\$ zu schreiben sind und beim Generieren aufgelöst werden.

Der Inhalt beider Felder entspricht Syntaxvorgaben von IXOS und sollte in deren Dokumentation nachgelesen werden.

Beispiel

```
IXOS {
  BatchMode 1
  DocumentMax 9000
  COMMANDS {
    DOCTYPE MTA
    COMP META_DOCUMENT PDF $.DATAFILENAME$
    R3_CLIENT $R3_CLIENT$
    R3_DESTINATION $R3_DESTINATION$
    R3_SAP_OBJ $R3_SAP_OBJ$
    R3_AR_OBJ PDF
  }
  IXATTR {
    NEWDOC
    R3_CLIENT TOAV0|MANDT|CC|%s|
    R3_SAP_OBJ TOAV0|SAP_OBJECT|CC|%s|
    R3_AR_OBJ TOAV0|AR_OBJECT|CC|%s|
    ARCHIVIDATTR TOAV0|ARCHIV_ID|CC|%s|
    DOCIDATTR TOAV0|ARC_DOC_ID|CC|%s|
    ATTRIBUTES TOAV0|OBJECT_ID|CC|$R3_OBJ_ID$|
  }
}
```

Ceyoniq

Die Anbindung an das Archivsystem Ceyoniq geschieht über die File-Schnittstelle des COLD-Import-Programms von Ceyoniq. Ceyoniq ist ein Primär-Index-Archivsystem. Das COLD-Import-Programm wird von Ceyoniq nur angeboten, wenn eine zusätzliche Index-Datenbank in Benutzung ist.

Filename des Index-Files (IndexExt)	Der Index-File wird immer in ein SubDirectory hinein generiert. Diese Angabe ist Pflicht. Der Index-File entsteht erst am Ende eines Archivlaufes. Die Erzeugung dieses Files ist atomar. Wenn dieser File erzeugt wurde, kann das Archivsystem mit dem Import der Daten beginnen.
In Sub-Directory generieren (SubDir)	Die Ausgabe der Dokumente erfolgt in ein Unterverzeichnis des Aufgabepfades. Der Index-File wird in ein Sub-Directory generiert.

Umschlüsselung der Variablen

Für dieses Format ist es nicht unbedingt notwendig, die Variablen des Vorab-Index nach Ceyoniq umzuschlüsseln. Fehlen die Angaben zum Umschlüsseln gänzlich, so werden die Variablen des Vorab-Index direkt in den Index übernommen. Soll umgeschlüsselt werden, so kann dies unter dem Eintrag Vars geschehen.

Syntax

```
ARCHIVINDEXNAME VARNAME[,VARNAME[...]]
```

Variablen, die FileNamen mit in den Index-File bringen, sind nicht notwendig; dies ist durch das Format sichergestellt. Das Arcis-, Teradoc-, Compaq- und HYPARCHIV-Interface unterstützt die Arbeit eines Transfer-Programms.

Beispiel
<pre>CEYONIQ { IndexExt .AU\$ InSubDir 0 Vars { MODULE_ID MODULE_ID BRANCH_CODE BRANCH_CODE CORRESPONDENT CORRESPONDENT IDENTIFIER IDENTIFIER } }</pre>

Transfer

Viele Archivsysteme stellen zur direkten Ablage der Archivdokumente Bibliotheken zur Verfügung. Diese Bibliotheken sind meist plattformabhängig und werden in den Transferprogrammen wie ein Treiber aufgerufen. Der Archiver ist aufgrund seiner integrierten Transferprogramme in der Lage, die verschiedenen Bibliotheken der Archivsystemhersteller in den plattformunabhängigen Archiver einzubinden. Im Archiver sind Treiber für die Archivsysteme SER-BlueLine, GlobalSoft Digital working (früher Compaq Archive), GFT HYPARCHIV und Optimal Systems integriert.

Die Transferschnittstelle unterstützt unterschiedliche Dokumenttypen. Aus diesem Grund können in einem Archivjob unterschiedliche Dokumenttypen vorkommen und archiviert werden. Der Archiver ermittelt den Dokument-Typ entsprechend des Eintrages, der im Feld Dokument-Typ definiert wurde. Wird der angegebene Dokumenttyp nicht gefunden, sucht der Archiver einen Dokumenttyp mit dem Namen DEFAULT. Findet er einen solchen, erfolgt das Umrouten der Variablen über diesen Dokumenttyp.

Das Feld Dokument-Typ enthält einen String, der eine oder mehrere Variablen beinhalten kann, die zum Zeitpunkt der Erkennung aufgelöst werden. Für jeden Dokument-Typ lassen sich im Archiver zwei Tabellen anlegen. In der ersten Tabelle können Variablen umgeroutet werden, in der zweiten Tabelle wird die Ausgabe der Index-Information für das Archivsystem beschrieben.

Max. Anzahl von Dokumenten pro Transfer (DocumentMax)	Maximale Anzahl von Dokumenten, die in einem Transfer übertragen werden. 0 bedeutet keine Beschränkung.
Ausgabeverzeichnis pro Typ (InTypeDir)	Es wird für jeden Dokument-Typ ein eigenes Ausgabeverzeichnis. Der Verzeichnisname wird aus dem Namen des Dokument-Typs gebildet.
Ausgabe im MetaFormat (MetaFormat)	Die Index-Datei wird im Standard als Archive-XML erzeugt. Für ältere Transfer-Programme kann das alte Meta-Format erzeugt werden.

Zum Umrouten der internen Variablen des Archivers auf die Variablen-Namen des Archivsystems gilt die folgende Syntax.

Syntax

```
INDEXNAME VAR[,VAR[...]]
```

Beispiel


```

Transfer {
  DocumentMax 0
  DocType $BEZ3$
  MetaFormat 1
  DocTypes {
    DocType RECHNUNG {
      Adds {
        SYSTEM_DATE=20020424
        FORMNAME=1003
      }
      Vars {
        1009 SYSTEM_DATE
        1012 BEZ3
        1013 K023_DATUM_WERT
        1014 K022_BELEGNR_WERT
        1038 K022_BELEGNR_WERT
        1030 K024_KUNDENNR_WERT
        1028 K011_ADRESSE
        1011 UP05_BEST_LIEF_WERT
        1032 UP06_BEST_LIEF_WERT
      }
    }
  }
  DocType DEFAULT {
    Adds {
      SYSTEM_DATE=20020424
      FORMNAME=1003
      BEZ3=UNKNOWN-DOCTYPE
    }
    Vars {
      1009 SYSTEM_DATE
      1012 BEZ3
      1013 K023_DATUM_WERT
      1014 K022_BELEGNR_WERT
      1030 K024_KUNDENNR_WERT
      1028 K011_ADRESSE
      1011 UP05_BEST_LIEF_WERT
      1032 UP06_BEST_LIEF_WERT
    }
  }
}

```

```
}  
  }  
    }
```

SER (obsolete)

 SER-File-Interface gilt als veraltet! Bitte nutzen Sie den Weg über das Transferformat und **a2ser** als Transportprogramm.

Das SER-Archivsystem arbeitet mit einem festen Satzformat ohne Header. Welche Variablen mit welcher Breite vom Archivsystem erwartet werden, ist unbedingt mit dem Administrator des Archivsystems abzusprechen.

Rumpfname des Index-Files (IndexName)	Name des Index-Files. Wird hier nichts gesetzt, dann muss zumindest die File-Endung gesetzt sein. Der Index-File entsteht erst am Ende eines Archivlaufes. Die Erzeugung des Files ist atomar. Wenn dieser File erzeugt wurde, kann das Archivsystem mit dem Import der Daten beginnen.
File-Endung des Index-Files (IndexExt)	Name der Extention des Index-Files. Wird hier nichts gesetzt, dann muss zumindest der Rumpfname gesetzt sein.
In Sub-Directory generieren (IndexInSubDir)	Der Index-File und alle Dokumente werden in ein Sub-Directory generiert. Andernfalls werden diese direkt in das OutPath Verzeichnis erstellt.

Da das SER-Format keine Variablen-Namen enthält, kann der Vorgang des Aufbaus der Index-Zeile für ein Dokument schwerlich als Umschlüsseln bezeichnet werden. An die Stelle des Variablen-Namens zur Identifikation tritt vielmehr die Position der Daten in der Index-Zeile.

Die Index-Zeile wird so aufgebaut, dass alle Variablen der Var-Liste hintereinander in die Index-Zeile geschrieben werden. Um dem festen Satzformat mit der damit ebenfalls festen Position der Daten in der Zeile Rechnung zu tragen, muss jede Variable mit einer Breitenangabe versehen werden. Ist der Inhalt einer Variable größer als die angegebene Länge, so wird der Inhalt gekürzt. Ist er kleiner, so wird der Inhalt aufgefüllt. Aus diesem Grund muss zwingend die Auffüllrichtung angegeben werden.

LEFT bedeutet, dass der Variablenwert linksbündig geschrieben wird, das Auffüllen mit Füllzeichen geschieht an der rechten Seite.

RIGHT meint dementsprechend dann rechtsbündig mit Auffüllen auf der linken Seite des Variablenwertes. Das Füllzeichen kann optional angegeben werden.

Fehlt die Angabe des Füllzeichens, so wird das Leerzeichen hierfür verwendet. Bitte beachten Sie, dass bei der Position im Satzformat auch die Trennzeichen eine Rolle spielen.

Eine Index-Zeile beginnt immer mit der Raute („#“) und vor jedem Variablenwert wird ein Ausrufezeichen („!“) ausgegeben.

Syntax der Variablen-Ausgabe

VARNAME[,VARNAME[...]] Size LEFTIRIGHT [FillChar]

Erklärung

VARNAME

Name der Vorab-Index-Variable.

Size

Breitenangabe für den Inhalt der Variable.

LEFT|RIGHT

Orientierung des Variablenwertes beim Auffüllen.

FillChar


Füllzeichen (Standard ist Leerzeichen)

Der Name des generierten Files ist nicht automatisch Bestandteil des SER-Index-Files. Dieser muss separat angegeben werden und nennt sich FILE. Das SER-Interface unterstützt heute keine aufgesplitteten Dokumente. Aus diesem Grund kommt das Schlüsselwort FILE auch nur einmal vor. Um die Arbeit für das Archivsystem zu erleichtern, gibt es noch eine zusätzliche Variable UNIQUEJOBNO. Jeder Archivlauf bekommt eine einmalige Nummer generiert, die mit in den Archiv-File aufgenommen werden kann.

Beispiel

```
SER {
  IndexName indexdatei
  IndexExt .idx
  InSubDir 1
  Vars {
    UNIQUEJOBNO 9 RIGHT 0
    DATUM 10 LEFT
    KUNDENNO 16 RIGHT 0
    FILE 30 LEFT
  }
}
```

ProFile (obsolete)

 Das PRO.FILE-File-Interface gilt als veraltet! Bitte nutzen Sie den Weg über das Transferformat und **a2profile (obsolete)** als Transportprogramm.

ProFile ist ein Archivsystem der Firma ProCAD aus Karlsruhe. Die Anbindung an das ProFile-Archivsystem geschieht über die File-Schnittstelle. Diese Schnittstelle unterstützt nur Dokumente, die nicht gesplittet sind. Typischerweise erwartet ProFile, dass die generierten Dokumente abhängig vom Dokument-Typ in verschiedene Verzeichnisse abgelegt werden. Der Archiver ermittelt den Dokument-Typ entsprechend des Eintrages, der im Feld Dokument-Typ definiert wurde. Das Feld Dokument-Typ enthält einen String, der eine oder mehrere Variablen beinhalten kann, die zum Zeitpunkt der Erkennung aufgelöst werden. Im Ausgabeverzeichnis muss ein Unterverzeichnis angelegt werden, das denselben Namen wie der Dokument-Typ trägt. Für jeden Dokument-Typ lassen sich im Archiver zwei Tabellen anlegen. In der ersten Tabelle können Variablen umgeroutet werden, in der zweiten Tabelle wird die Ausgabe der Index-Information für das Archivsystem beschrieben.

File-Endung des AttributFiles (IndexExt)	File-Endung (Extention) des AttributFiles, der für jedes Dokument erzeugt wird.
---	---

Reihenfolge, in der die Variablen vom Archiver geschrieben werden, muss mit der Reihenfolge übereinstimmen, die das Archivsystem erwartet. Stimmen Sie die Ende-Kennung (Extention) des Index-Files und die Variablen-Reihenfolge mit dem DocuWare-Systembetreuer ab. Die Angabe einer Variable zur Ausgabe erfolgt wie in folgender Syntax gezeigt.

Syntax

OUTVAR VAR[,VAR[...]] Size LEFT|RIGHT|DATE [FillChar]

Die Angabe der OUTVAR ist Pflicht, der Inhalt ist momentan noch nicht von Bedeutung. Bei der Angabe von DATE wird ein Datum im Format TT.MM.JJ oder TT.MM.JJJJ in ein ProFile-Datumsformat gewandelt.

Beispiel

```

ProFile {
  IndexExt .txt
  SubDirPerType 1
  UseTransferTask 0
  DocType $DOCTYPE$
  DocTypes {
    DocType RECHUNG {
      Vars {
        DOCNR WA_DOC_HEADER_OPBEL 15 LEFT
        CONNR WA_CONT_ACCT_VKONT 15 LEFT
        DATE SY_DATUM 15 DATE
      }
    }
    DocType LIEFERSCHEIN {
      Adds {
        XADD=Sonderkunde Firma $FIRMA$
      }
      Vars {
        DOCNR WA_DOC_HEADER_OPBEL 15 LEFT
        CONNR WA_CONT_ACCT_VKONT 15 LEFT
        DATE SY_DATUM 8 DATE
        XADD XADD 30 LEFT
      }
    }
  }
}

```

Hinweise zur Dokumentation

Teile dieser Online-Dokumentation können die Nutzer und Administratoren zur eigenen Verwendung ausdrucken. Es gelten dabei die **rechtlichen Hinweise**.

Die verwendeten Symbole, Schriftarten und deren Bedeutung werden **hier** näher erklärt.

Alle Abbildungen, Grafiken und Diagramm wurden teilweise aus Platzgründen in ihrer Größe bearbeitet. Für eine optimale Darstellung der Online-Dokumentation sollte Ihre Bildschirmauflösung 1680x1050 Pixel oder höher betragen, mindestens jedoch 1440x900.




Falls Sie weitere Fragen haben, die in der Dokumentation nicht thematisiert werden, finden Sie **hier** eine Liste ergänzender Dokumentationen.

Sie haben natürlich auch die Möglichkeit sich jederzeit an unseren **Support** zu wenden.

Verwendete Symbole

In der Nutzer-Dokumentation gibt es diverse Symbole und Zeichen. Um Ihnen einen Überblick zu verschaffen, sind hier die wichtigsten Symbole erklärt.

Symbol	Beschreibung
	Dieses Symbol gibt zusätzliche Informationen zu einem Textabschnitt.
	Dieses Symbol stellt eine Warnung dar, welche unbedingt beachtet werden müssen.
	Dieses Symbol gibt einen Hinweis, welcher beachtet werden sollte.
Beispiel	Dieses Symbol enthält ein Beispiel für den darüber beschriebenen Textabschnitt.
Klicken Sie hier, um den Text auszuklappen. Wir wünschen Ihnen einen schönen Tag.	Beinhaltet Informationen, welche durch einen Klick ausgeklappt werden können.
Archiver 5.1	Verlinkung auf eine andere Seite der Server-Dokumentation.
	Dies entspricht den Symbolen 1-4, Sie können aber auch als reines Symbol in der Tabelle verwendet werden.
	Diese Symbole kennzeichnen Vor- und Nachteile oder geben zusätzliche Funktionen, Merkmale bzw. Warnungen an.
	Zwingende Angabe bzw. Aktion. Es handelt sich um ein Pflichtfeld.
	Die Nichteingabe erzeugt einen Fehler. Es ist aber nicht wichtig für die Funktionalität des BC-XOM Servers.
	Es handelt sich um eine optionale Angabe.

Symbol	Beschreibung
	<p>Wenn dieses Symbol in den Screenshots auftaucht, gibt es unter der Abbildung eine Erklärung zu der Nummer bzw. zu den Nummern.</p>
	<p>Dieses Symbol fasst mehrere Elemente in der Abbildung zusammen.</p>
	<p>Dieses Symbol hebt Bildausschnitte hervor.</p>
<p>Abb. A (1)</p>	<p>Gibt im Text an, dass sich eine Erklärung auf eine Abbildung mit dem jeweiligen Buchstaben bzw. der jeweiligen Nummer bezieht.</p> <p>Enthält eine Seite nur eine Abbildung, kann statt <u>Abb. A (1)</u> auch nur (1) stehen. Enthält eine Seite mehrere Abbildungen,</p> <p>beziehen sich Angaben ohne explizite Abbildungsangabe wie (1) immer auf Abbildung A.</p>
<p>STRG + ALT + ENTF</p>	<p>Hier handelt es sich um eine Tastenkombination.</p>

Weiterführende Informationen

Die weiterführenden Informationen gliedern sich in folgende Unterkapitel (bitte auf den jeweiligen Link klicken):

- [*Sitemap*](#)
- [*Download der Dokumentation*](#)
- [*Ergänzende Online-Dokumentationen*](#)
- [*Service und Support*](#)

Sitemap

Rechtliche Hinweise

Einleitung

Was ist neu in Version 5.1

Programme des Archiver

- **archiver - Programm**
 - **Aufruf und Parameter des Archiver**
- **arcdir**
 - **Aufruf und Parameter von ArcDir**
- **a2ser**
 - **Aufruf und Parameter von a2ser**
- **a2os**
 - **Aufruf und Parameter von a2os**
 - **a2os Version 3.0**
- **a2easy**
 - **Aufruf und Parameter von a2easy**
- **a2content, a2saperion**
 - **Aufruf und Parameter von a2content**
- **a2enaio**
 - **Aufruf und Parameter von a2enaio**
- **a2comp (obsolete)**
 - **Aufruf und Parameter von a2comp (obsolete)**
- **a2hyp (obsolete)**
 - **Aufruf und Parameter von a2hyp (obsolete)**
- **a2profile (obsolete)**
 - **Aufruf und Parameter von a2profile (obsolete)**
 - **Feldzuordnung innerhalb der Indexdatei (obsolete)**
 - **Installation (obsolete)**

Konfiguration des Archivers

- **Einstellungen**
 - **Pfade**
 - **Tasks**
 - **Scannen**
 - **LogFile**
 - **Namen, Verfahren und Format**
 - **PDF Verarbeitung**
 - **PDF Erzeugung**
 - **Sprache/Seriennummer**
- **Formate**
 - **Arcis/Teradoc**
 - **Easy**
 - **FileNET**
 - **DocuWare**
 - **IXOS**
 - **Ceyoniq**
 - **Transfer**
 - **SER (obsolete)**

- *ProFile (obsolete)*


Hinweise zur Dokumentation

- *Verwendete Symbole*

Weiterführende Informationen

- *Sitemap*
- *Download der Dokumentation*
- *Ergänzende Online-Dokumentationen*
- *Service und Support*

Download der Dokumentation

Format	Erstellungsdatum	Größe	Download-Datei
PDF	 27.03.2023	< 1,0 MB	PDF-Datei



Bitte beachten Sie unsere **rechtlichen Hinweise**, bevor Sie die Dateien herunterladen!
 Die Dateien entsprechen dem Datum in der Spalte "Erstellungsdatum". Die Online-Dokumentation ist nur zu diesem Zeitpunkt der Erstellung aktuell.
 Wir weisen Sie darauf hin, dass interaktive Multimedia-Inhalte in der Online-Dokumentation im PDF-Format nicht angezeigt werden können.

Ergänzende Online-Dokumentationen

Hier gelangen Sie zu anderen Online-Dokumentationen der Firma profiforms gmbh.
Bitte beachten Sie, dass der jeweilige Link eine neue Seite öffnet und Sie die aktuelle Dokumentation verlassen.

Spooler
<input type="checkbox"/> <i>Spooler 4.0</i>

PDFxOut
<input type="checkbox"/> <i>PDFxOut 2.0</i>

ReportWriter
<input type="checkbox"/> <i>ReportWriter 5.9</i>

Logistics
<input type="checkbox"/> <i>Logistics 5.9</i>

Service und Support

Sie haben Fragen oder Probleme zu/mit einem unserer Produkte und verfügen über einen gültigen Support- und Update-Vertrag?

Dann kontaktieren Sie uns bitte:

- ... über unsere Webseite: <http://www.profiforms.de>
- ... über den Ihnen bekannten/zugeordneten Projekt-/Vertriebs-Mitarbeiter
- ... über unseren Service Desk: <https://support.profiforms.de/servicedesk/customer/portals>