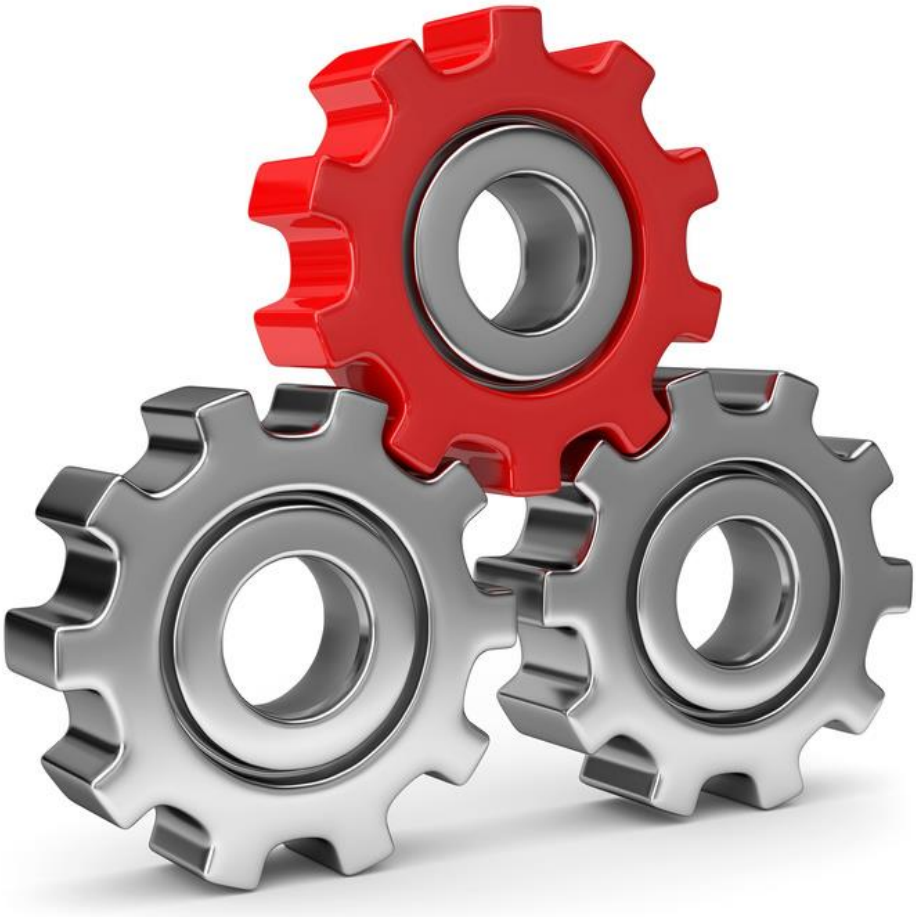
ERPAdmin

für büro+



Version 3.46

SYSTEMHAUS **HORN**
Eibenstraße 6
63322 Rödermark
Fon +49 60 74 / 88 10 46
Fax +49 60 74 / 88 10 47
www.Systemhaus-HORN.de
ERPAdmin@Systemhaus-HORN.de

microtech Gold Partner 2024 microtech Solution Partner 2024

Inhaltsverzeichnis

1 Einleitung 7
2 Installation und Aktivierung ERPAdmin 9
3 ERPAdmin -SETUP - Vorbelegung der wichtigsten VAR.INI Variablen STD 12
4 Installation der Chilkat-DLL's 15
5 Die Datei „VAR.INI“ STD PRO 20
6 Rückmeldungen an den Benutzer 26
7 Aufrufen von ERPAdmin aus Windows 38
8 Aufruf von ERPAdmin durch den Benutzer direkt aus büro+ 39
9 Verschlüsselung von Passworten mit ERPAdmin BAS 42
10 Historisierung 43
11 Optionen und Parameter von ERPAdmin 44
12 Benutzung einer Bereichsauswahl /Von: .... /Bis: 50
13 Benutzung von Filtern STD PRO 52
14 Drucken mit ERPAdmin PRO 54
15 Platzhalter und variable Parameter BAS 56
16 Doubletten-Prüfung 61
17 Die Option „/HELP“ BAS 63
18 Der Befehl „-Init“ BAS 64
19 Der Befehl „-AdressenUstIDCheck“ PRO 66
20 Der Befehl „-ANSI2UNICODE“ STD 70
21 Der Befehl „-ANSI2UTF8“ STD 70
22 Der Befehl „-APPENDANSIFILE“ STD 71
23 Der Befehl „-ArtikelBildLinkCheck“ PRO 72
24 Der Befehl „-Artikelweglegen“ PRO 74
25 Der Befehl „-AusgabeverzeichnisSafeToFile“ PRO 76
26 Der Befehl „-BackupRestoreVorgangArten“ BAS 78
27 Der Befehl „-BelegNrCheck“ BAS 80
28 Der Befehl „-CalcMinArtikelEK“ PRO 82
29 Der Befehl „-CalcStkLstMinVK“ STD 83
30 Der Befehl „-CopyFromFile“ PRO 85
31 Der Befehl „-CopyLinesToFile“ PRO 86

32 Der Befehl „-ConCatFiles“ STD 87
33 Der Befehl „-CreateSN“ PRO 88
34 Der Befehl „-CryptPwd“ BAS 90
35 Der Befehl „-CSV2HTML“ STD 92
36 Der Befehl „-CSV2XLS“ PRO 93
37 Der Befehl „-CSV2XML“ STD 95
38 Der Befehl „-CSVKonverter“ PRO 98
39 Der Befehl „-DateDiff“ STD 136
40 Der Befehl „-DeleteData“ STD 138
41 Der Befehl „-DeleteDataFromFile“ PRO 139
42 Der Befehl „-DeleteFiles“ BAS 140
43 Der Befehl „-DeleteFilesByDate“ STD 141
44 Der Befehl „-DeleteFoldersByDate“ STD 142
45 Der Befehl „-DeleteLF“ PRO 143
46 Der Befehl „-Dir2CSV“ PRO 144
47 Der Befehl „-DokumentDoublettenCheck“ PRO 146
48 Der Befehl „-DokumentLinkCheck“ PRO 149
49 Der Befehl „-DokumentSearchAndReplace“ PRO 151
50 Der Befehl „-Email“ BAS 154
51 Der Befehl „-EmailFromFile“ PRO 156
52 Der Befehl „-EmailNotify“ PRO 159
53 Der Befehl „-ExcelCMD“ PRO 164
54 Der Befehl „-Export“ STD 171
55 Der Befehl „-ExportLtzAend“ PRO 175
56 Der Befehl „-FilenameSearchAndReplace“ STD 178
57 Der Befehl „-FileReplace“ STD 179
58 Der Befehl „-FileReplaceCol“ PRO 181
59 Der Befehl „-FTPDelete“ PRO 182
60 Der Befehl „-FTPDir“ STD 183
61 Der Befehl „-FTPPut“ STD 184
62 Der Befehl „-FTPGet“ STD 187
63 Der Befehl „-HTTPDownload“ PRO 189
64 Der Befehl „-HTTPDownloadFromFile“ PRO 190
65 Der Befehl „-Import“ STD 191

66 Der Befehl „-InsertDataFromFile“ PRO 193
67 Der Befehl „-InsertDocument“ PRO 194
68 Der Befehl „-InsertDocumentFromDir“ PRO 197
69 Der Befehl „-InsertDocumentFromFile“ PRO 201
70 Der Befehl „-KillProcess“ BAS 204
71 Der Befehl „-KontaktSenden“ PRO 205
72 Der Befehl „-LagerAusbuchen“ PRO 207
73 Der Befehl „-LagerbestandUmbuchen“ PRO 209
74 Der Befehl „-ListAdressVerteiler“ BAS 211
75 Der Befehl „-ListAll“ BAS 212
76 Der Befehl „-ListBuilds“ BAS 215
77 Der Befehl „-ListDatenbankFelder“ BAS 217
78 Der Befehl „-ListEinheiten“ BAS 220
79 Der Befehl „-ListExports“ BAS 221
80 Der Befehl „-ListFilter“ BAS 223
81 Der Befehl „-ListImports“ BAS 224
82 Der Befehl „-ListLayouts“ BAS 226
83 Der Befehl „-ListPOPConnect“ PRO 228
84 Der Befehl „-ListPOPControl“ PRO 229
85 Der Befehl „-ListPrinter“ BAS 230
86 Der Befehl „-ListReguläreausdrücke“ BAS 232
87 Der Befehl „-ListRules“ BAS 233
88 Der Befehl „-ListSelektionsFelder“ BAS 235
90 Der Befehl „-ListTextbausteine“ BAS 237
91 Der Befehl „-ListUmsatzsteuer“ BAS 238
92 Der Befehl „-ListVorgangArten“ BAS 239
93 Der Befehl „-ListWechselkurse“ BAS 241
94 Der Befehl „-ListeDrucken“ PRO 242
95 Der Befehl „-ListeDruckenDatei“ PRO 247
96 Der Befehl „-Login“ BAS 251
97 Der Befehl „-MassMail“ STD 254
98 Der Befehl „-MKDir“ PRO 259
99 Der Befehl „-OPAusgleich“ PRO 260
100 Der Befehl „-OffenePostenInArchiv“ PRO 262

101 Der Befehl „-OffenePostenInArchivDatei“ PRO 263
102 Der Befehl „-OpenURL“ STD 265
104 Der Befehl „-OutlookImportKontakte“ PRO 266
105 Der Befehl „-POP2CSV“ PRO 271
106 Der Befehl „-POPConnect“ PRO 272
107 Der Befehl „-POPControl“ PRO 276
108 Der Befehl „-PrintFile“ PRO 283
109 Der Befehl „-ProjekteInArchiv“ PRO 284
110 Der Befehl „-ReadData“ PRO 285
111 Der Befehl „-ReadDataSetValue“ PRO 287
113 Der Befehl „-ReplaceQuotation“ PRO 289
114 Der Befehl „-RTFExport“ PRO 290
115 Der Befehl „-RTFImport“ PRO 291
116 Der Befehl „-SendAtOnce“ PRO 292
117 Der Befehl „-SendMsg“ und „/SendMsg“ PRO 293
118 Der Befehl „-SetAktDate“ PRO 295
119 Der Befehl „-SFTPConvertPuttyKey“ PRO 296
120 Der Befehl „-SFTPDelete“ PRO 297
121 Der Befehl „-SFTPDir“ PRO 298
122 Der Befehl „-SFTPPut“ PRO 301
123 Der Befehl „-SFTPGET“ PRO 305
124 Der Befehl „-SplitCSVToFiles“ PRO 307
125 Der Befehl „-Update“ BAS 309
126 Der Befehl „-UpdateExchangeRate“ PRO 311
127 Der Befehl „-UTF82ANSI“ STD 314
128 Der Befehl „-UTF8OB2ANSI“ STD 314
129 Der Befehl „-UNICODE2ANSI“ STD 314
130 Der Befehl „-VersandInArchiv“ STD 315
131 Der Befehl „-VorgangAusArchiv“ STD 316
133 Der Befehl „-VorgangBuchen“ STD 317
134 Der Befehl „-VorgangArchivDrucken“ PRO 319
135 Der Befehl „-VorgangArchivDruckenDatei“ PRO 320
136 Der Befehl „-VorgangDrucken“ PRO 321
137 Der Befehl „-VorgangDruckenDatei“ PRO 324

138 Der Befehl „-VorgangErzeugen“ PRO 327
139 Der Befehl „-VorgangImport“ PRO 328
140 Der Befehl „-VorgangInArchiv“ STD 332
141 Der Befehl „-VorgangPositionAdd“ PRO 334
142 Der Befehl „-VorgangPositionAddFromFile“ PRO 337
143 Der Befehl „-VorgangPositionCount“ PRO 339
144 Der Befehl „-VorgangPositionMengeCount“ PRO 341
145 Der Befehl „-VorgangStornieren“ STD 342
146 Der Befehl „-VorgangSVArtikelAdd“ PRO 344
147 Der Befehl „-VorgangUstIDCheck“ PRO 349
148 Der Befehl „-VorgangWandeln“ STD 356
149 Der Befehl „-WriteData“ PRO 360
150 Der Befehl „-WriteDataFromFile“ PRO 362
151 Der Befehl „-XLS2CSV“ PRO 364
152 Die Befehle „-ZVEAdresseZuweisen“ und „ZVEEmpfaengerSpeichern“ PRO 365
153 Weitere Anwendungsbeispiele 368
154 Automatisierung von ERPAdmin mit dem Windows Aufgabenplaner 370
155 Tipps & Tricks 375
156 Systemvoraussetzungen 378
157 Anregungen, Kritik und Verbesserungsvorschläge 378

1 Einleitung

ERPAdmin unterstützt Administratoren und Power-User bei ihren wiederkehrenden Arbeiten wie z. B. dem Export/Import von **büro+**, automatischer Benachrichtigung von Benutzern oder Kunden per E-Mail über Systemzustände, automatisierten Konvertierungen von Excel-Dateien, beim Transfer von Dateien per FTP/SFTP, dem automatische buchen, wandeln und drucken von Vorgängen, dem Verschieben von Vorgängen oder Versand-Datensätzen ins Archiv und noch vieles mehr.

E-Mail-Postfächer können überwacht werden und erkannte E-Mails automatisch bei einer Adresse in den Dokumenten hinterlegt werden. Dokumenten- und Bild-Verlinkungen können auf nicht mehr existierenden Dateien geprüft und auf Wunsch entfernt werden.

Vertreten können Vorgänge von unterwegs per E-Mail freigeben und Mitarbeiter werden bei Unterschreitung des Lagermindestbestandes oder bei Überschreitung des Liefertermins per E-Mail informiert.

Damit diese wiederkehrenden Aufgaben schnell und effizient erledigt werden können, wurde auf eine grafische Benutzer-Oberfläche verzichtet. Alle Ausgaben erfolgen in eine Logdatei und in das Protokoll von **büro+**. Zur Rückmeldung und Überwachung versendet **ERPAdmin** nach Ausführung einen E-Mail-Report getrennt nach **OK**- und **Fehler-**Meldungen.

Dazu wird **ERPAdmin** vom Anwender nach Bedarf von Hand oder über den Windows Scheduler in festen Intervallen aufgerufen (Siehe Kapitel „**Automatisierung von ERPAdmin mit dem Windows Aufgabenplaner**“).

Achtung:
Beim Aufrufen von **ERPAdmin** ist darauf zu achten, dass keine Instanzen von **ERPAdmin** parallel ausgeführt werden. Alle Aufrufe müssen sequenziell erfolgen.

Highlights von ERPAdmin

- **Funktioniert mit allen aktuellen büro+ Versionen**
- Ablage von E-Mails aus einem POP-Postfach in den Dokumenten
- Aktualisierung der Wechselkurse mit Daten der Bundesbank
- Anbindung individueller WebShops z.B. B2B
- Ändern von Feldwerten in **büro+** per E-Mail
- Abrechnung SV-Artikel mit einem einzigen SV-Artikel (auch Stücklisten, Varianten, etc.)
- Dynamische Benennung aller Ergebnis-Dateien mit ca. 40 Platzhaltern z.B. Berechnung des Primus und des Ultimos von beliebigen Monaten, KW-Anfang und -Ende, etc.
- Artikel in Bestellungen in Kundenlager umbuchen (Artikel Weglegen)
- Aufbereitung und Konvertierung von CSV-, Text- und Excel-Dateien
- Aus- und Umbuchen von Lagerbeständen
- Automatische Zuordnung und Ablage von E-Mails in den Dokumenten
- Benachrichtigung von Benutzern über kritische Systemzustände (Min. Bestand, Liefertermine, fällige SEPA-Einzüge, Anstehenden SV-Abrechnungen, etc.)
- Benutzer oder Vertreter per E-Mail über anstehende Termine informieren
- Doubletten-Prüfung zur Vermeidung von Doppel-Übertragungen

- Druck von Listen und Auswertungen und Ausgabe in den Formaten PDF, XLS, XLSX, PPT, DOC, HTML, JPG, BMP, TIF, EMF, XML, LL, etc.
- Ermittlung Mindestbestand auf Stückumsatz-Basis des Vormonats
- Erstellung von Artikel- oder Kunden-Stammdatenblättern z.B. als PDF
- Erstellung von Auswertungen aller Regeln, Filter und Layouts zu Dokumentationszwecken
- Erzeugen einer Dokumentation aller Selektionsfelder inkl. Auswahl und Feldlängen
- Erzeugt automatisiert Seriennummern nach Vorgaben und schreibt diese in **büro+**
- Erzeugt Bestellungen um den Lagerbestand aufzufüllen (Beachtung min. max. Bestandes)
- Export/Import (auch zwischen Mandanten) und Anpassung des Dateinamens
- Freigabe von Vorgängen per E-Mail von unterwegs (4-Augen Prinzip)
- FTP/FTPS/SFTP-Transfer von Daten, z.B. Artikel-Daten zum Webshop und Download und Import von Kunden-Bestellungen
- Import von Artikeln oder Adressen mit autom. Vergabe von Artikel- oder Adress-Nummer
- Import von Dokumenten und automatische Zuordnung anhand des Dateinamens
- Umbuchung von Lagerbeständen
- Umsatzsteuer-ID Prüfung der Vorgänge inkl. Prüfung der Lieferanschrift
- Versand von Serien-E-Mails aus **büro+**
- Verwaltung von Service-Verträgen ohne SV-Artikel (auch für Stücklisten und Varianten-Artikel)
- Wandeln, Buchen und Drucken von Vorgängen mit Bedingungen (Vorkasse bezahlt, Bestand vorhanden, etc.)
- Warnung bei Differenzen von Lager- und Stellplatzbeständen
- Zusammenfassen von Vorgängen eines Kunden (ähnlich Sammel-Vorgang)

Bevor Sie **ERPAdmin** in Ihrem Produktiv-Mandanten in Betrieb nehmen, testen Sie bitte alle Funktionen ausgiebig in einem **Test-Mandanten**.

Hinweis:
Beim Daten-Import oder bei einem schreibenden Zugriff auf **büro+**-Datenbanken z.B. beim „**Vorgang-Wandeln**“ kann es zu Blockaden (Meldungen) kommen, wenn Datensätze von einem anderen Benutzer zum Ändern geöffnet wurden und **ERPAdmin** zeitgleich versucht eine Änderung vorzunehmen.

Importe im Bereich der Stammdaten sollten deshalb immer außerhalb der normalen Geschäftszeiten ausgeführt werden damit es hier nicht zu Verklemmungen kommt.

Weiterhin empfehlen wir, in den Parametern im Bereich **Sonstige->Globale-Einstellungen** die folgende Einstellung zu aktivieren, sofern diese zur Verfügung steht:

2 Installation und Aktivierung ERPAdmin

Eine Installation in Form einer Setup-Routine die Dateien kopiert etc. gibt es nicht, weil es nicht notwendig ist. Es reicht aus, **ERPAdmin** in ein Verzeichnis zu kopieren und zwei mitgelieferte MSI-Dateien zu installieren (chilkatax-9.5.0-x64.msi und chilkatax-9.5.0-win32.msi). Für die Installation der DLLs lesen Sie bitte das Kapitel „**Installation der Chilkat-DLL's**".

Für die Vorbelegung der wichtigsten Variablen in der **VAR.INI** gibt es den Befehl **ERPAdmin -Setup** STD. Diese ist weiter unten im nächsten Kapitel beschrieben.

ERPAdmin steht Ihnen in drei Ausbaustufen zur Verfügung: Die **Basic** Version welche kostenfrei ist und die **Standard**- und die **Professional**-Version welche kostenpflichtig sind.

Alle Befehle, Parameter und Optionen die in der **Basic**-Version kostenfrei zur Verfügung stehen sind mit einem BAS markiert, die der **Standard**-Version mit STD und die der **Professional**-Version mit PRO. Alle Funktionen die in einer kleinen Version zur Verfügung stehen, sind auch in allen größeren Versionen enthalten.

Für das Ausführen der Basic-Befehle BAS wird die Datei VAR.INI nicht benötigt und sollte gelöscht werden!

Achtung beim Einsatz eines Virenscanners

ERPAdmin ver- und entschlüsselte seine Kennworte. Ergebnis-Dateien z.B. aus einem Export werden zur Historisierung in ein anderes Verzeichnis verschoben. Dieses verhalten kann bei einigen Virenscannern dazu führen, dass **ERPAdmin** als potentielle Schadsoftware eingestuft wird und dadurch die Ausführung verhindert oder **ERPAdmin** in Quarantäne genommen wird. Damit dieses nicht passiert definieren Sie alle Verzeichnisse in denen sich eine **ERPAdmin.exe** befindet als Ausnahme im Virenscanner.

Aktivierung von ERPAdmin STD PRO

Zur Aktivierung Ihrer **ERPAdmin**-Version bekommen Sie einen Lizenzschlüssel zugesendet.

Bitte beachten Sie, dass die Aktivierung der beiden Versionen (**Standard** und **Professional**) unterschiedlich von statten geht. Die Lizenzschlüssel sind nicht austauschbar und passen nur zu der von Ihnen erworbenen Ausbaustufe.

Aktivierung von ERPAdmin Standard-Version STD

Zur Aktivierung von **ERPAdmin** in der Standard Version tragen Sie den Lizenzschlüssel in der Datei **VAR.INI** in der folgenden Weise ein: **C_LICENSE="XXXXX-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX".**

Haben Sie in der **VAR.INI** eine Verlinkung zu einer externen Datei angegeben (z.B. **C_LICENSEPATH= "LIZENZ.INI"**) die den Lizenzschlüssel enthält, hinterlegen Sie den Lizenzschlüssel auf gleiche Weise in dieser Datei.

Aktivierung der Professional-Lizenz von ERPAdmin PRO

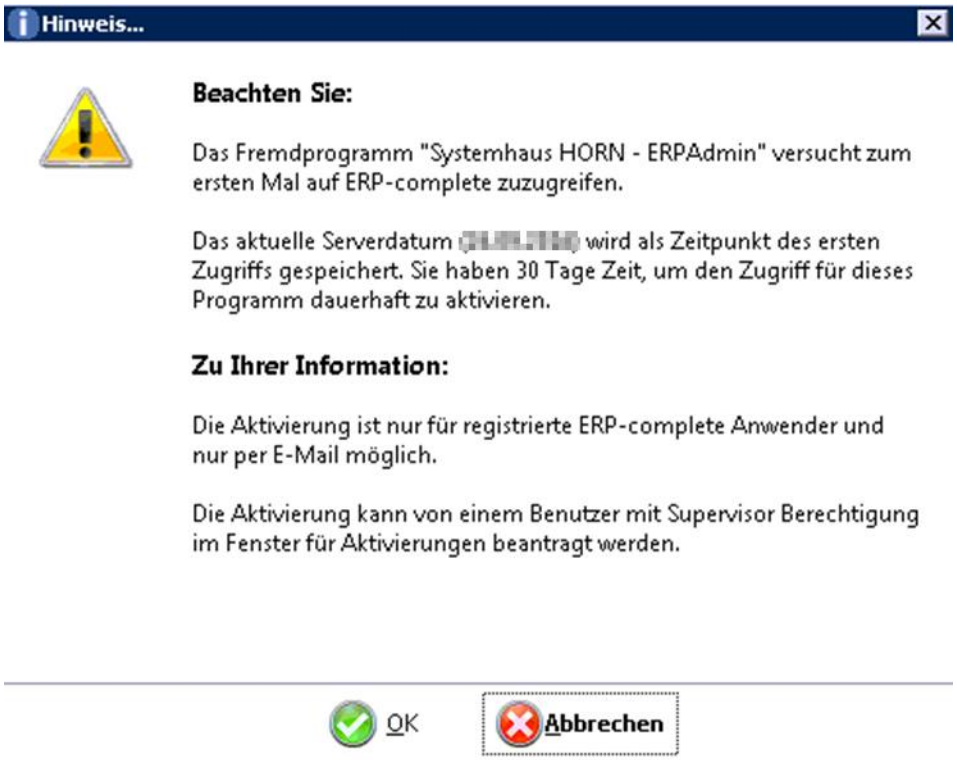
Für die Verwendung der Druck-Funktion und einiger anderer Befehle wird eine **Professional**-Lizenz von **ERPAdmin** benötigt. Diese muss gesondert erworben werden. Sofern Sie eine **ERPAdmin Standard**-Lizenz besitzen können Sie jeder Zeit ein Update auf die **Professional**-Lizenz erwerben.

Im Gegensatz zur **Standard** Version darf der Lizenz-Schlüssel der **Professional**-Version **NICHT** in der Datei **VAR.INI** eingetragen werden. Dort darf **kein** Eintrag **C_LICENSE=** existieren. Sollte das dennoch der Fall sein, löschen Sie diese Zeile einfach aus der Datei **VAR.INI**.

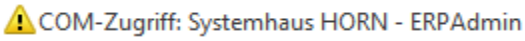
Wenn Sie die **ERPAdmin-Professional** zum ersten Mal verwenden ist bereits ein 30-Tage-Testschlüssel enthalten.

Damit die **Professional**-Lizenz zum Testen aktiviert wird, rufen Sie **ERPAdmin** mit dem **-INIT** Befehl auf:
ERPAdmin -INIT

Es erscheint folgende Meldung die Sie bitte mit „**OK**“ bestätigen.



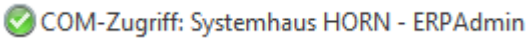
Danach finden Sie einen neuen Lizenzeintrag der COM-Anwendung **ERPAdmin** mit einem gelben Ausrufezeichen im Bereich Ihrer Aktivierungen.



Eine Beantragung eines finalen Schlüssels ist nicht notwendig. Diesen bekommen Sie automatisch zugesendet, wenn Sie eine Professional-Version von **ERPAdmin** gekauft haben.

Der Lizenzschlüssel befindet sich dann auf der Rechnung.

Tragen Sie diesen im Bereich **Hilfe->Lizenz->Aktivierungen…** am Server über den Button „**Hinzufügen**“ ein. Wenn ein gültiger Schlüssel hinterlegt wurde verschwindet das gelbe Ausrufezeichen und ein grüner Hacken erscheinen vor der **ERPAdmin-**Lizenz:



Hinweis 1:
Die Lizenzschlüssel aus der **Standard**-Version können **NICHT** zur Aktivierung der **Professional**-Version in **büro+** verwendet werden!

Hinweis 2:
Erfolgt nach dem ersten Aufruf von **ERPAdmin**-Professional nicht innerhalb von 30 Tagen die Eintragung eines finalen Lizenz-Schlüssels, kann **ERPAdmin** nach Ablauf dieser Frist nicht mehr benutzt werden ohne einen gültigen Lizenzschlüssel.

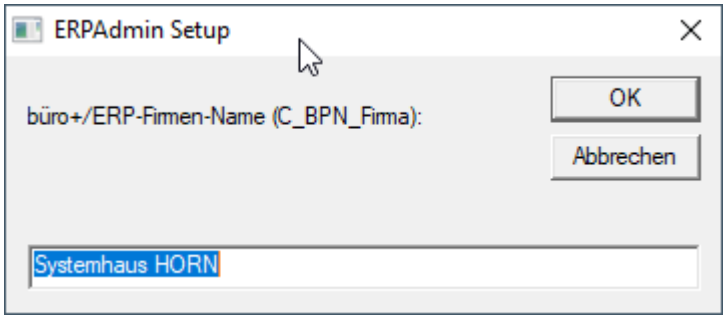
3 ERPAdmin -SETUP - Vorbelegung der wichtigsten VAR.INI Variablen STD

Um mit **ERPAdmin** zu arbeiten muss ein gültiger Lizenz-Schlüssel vorhanden, die Kommunikation mit dem Anwender per E-Mail gewährleistet sein und die Verbindung zu **büro+** funktionieren. Die Hinterlegung der hierfür benötigten **VAR-INI**-Variablen kann über den Aufruf **ERPAdmin -Setup** durchgeführt werden.

Rufen Sie dazu in der Commando-Zeile (CMD.exe) **ERPAdmin** wie folgt auf:

ERPAdmin -Setup

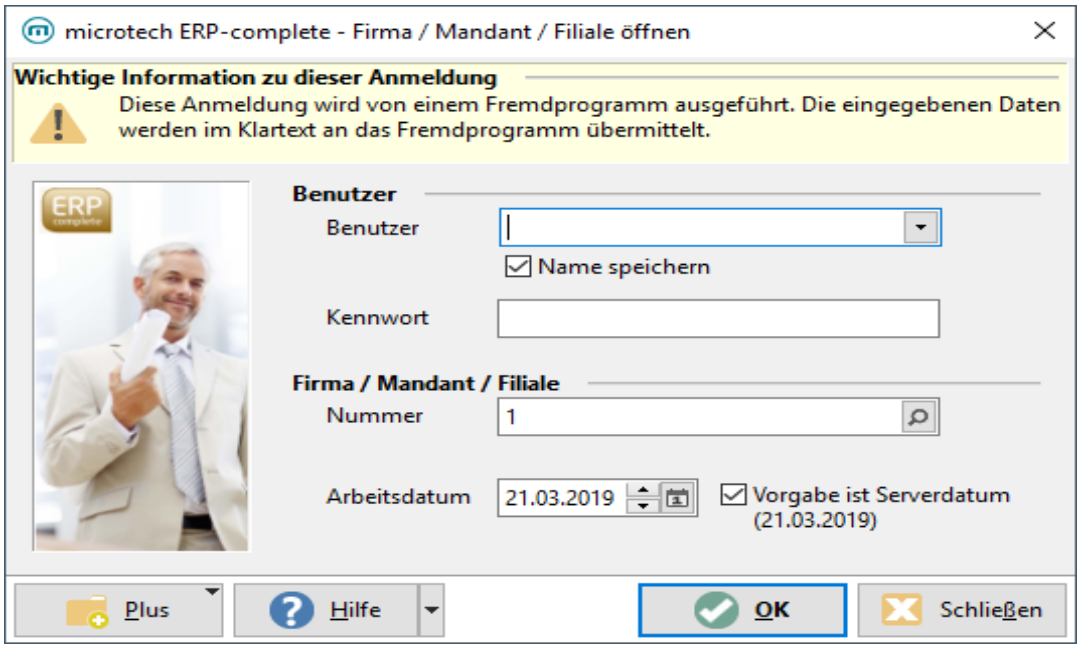
ERPAdmin fragt dazu die entsprechenden Parameter beim Benutzer ab. Ausgenommen von dem Lizenzschlüssel für die Standard-Version sind alle Felder in diesem Dialog Pflichtfelder und dürfen nicht leer sein. Die Eingabe eines Leerstrings beendet die Erfassung umgehend. Soll ein bereits vorhandener Wert erhalten bleiben, Klickt der Benutzer auf **OK**. Die Werte werden umgehend in der **VAR.INI** gespeichert.



[Setup-Dialog der wichtigsten **VAR.INI** Variablen]

Die Eingabewerte werden im Dialogfenster untereinander angezeigt. Kennworte werden dabei automatisch verschlüsselt abgespeichert.

Die Anmeldedaten für **büro+** werden mit den Anmeldedialog abgefragt sofern noch keine Daten in der **VAR.INI** vorhanden waren. Nur bei gültiger Anmeldung werden diese auch gespeichert.

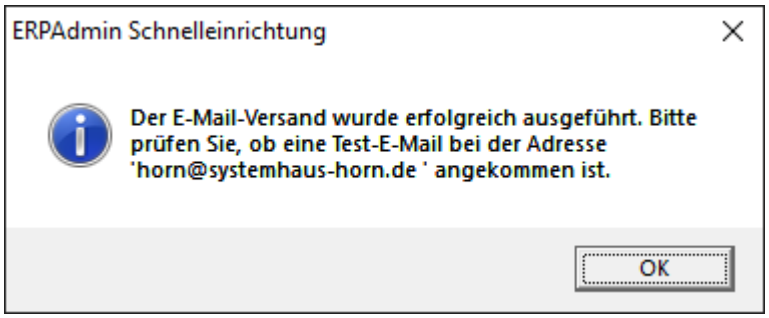


Zuletzt werden SMTP-Versand-Daten für die E-Mail-Kommunikation abgefragt. Wenn Sie hier keine Angaben hinterlegt haben, versendet **ERPAdmin** seine E-Mails über den **ERPAdmin** -Server und Sie bekommen bei jedem Aufruf von **ERPAdmin** eine Warnung mit einem entsprechenden Hinweis.

Diese Meldung erscheint nicht mehr, wenn gültige SMTP-Versand-Daten in der **VAR.INI** hinterlegt wurden. Dazu sind die folgenden Variablen mit Werten zu belegen:

C_SMTPSERVER	**' SMTP-Server/IP-Adresse**
C_SMTPSERVERPORT	**' SMTP-Server-Port (25/465/587)**
C_SMTPSSL	**' SMTP-Versand mit SSL-Verschlüsselung TRUE oder FALSE**
C_SMTPTLS	**' SMTP-Versand mit TLS-Verschlüsselung TRUE oder FALSE**
C_SMTPUSER	**' SMTP-Benutzer**
C_SMTPPWD	**' SMTP-Passwort**
C_ERROR_FROM	**' Absenderadresse für Fehler-E-Mails**
C_ERROR_To	**' Empfängeradresse für Fehlerbenachrichtigungen**
C_Warning_To	**' Empfängeradresse für Warnmeldungen**
C_OK_To	**' Empfängeradresse für OK-Meldungen**

Haben Sie Ihre SMTP-Einstellungen korrekt angegeben wird die E-Mail-Anbindung getestet und dabei eine E-Mail an die angegebene OK-E-Mail-Adresse gesendet. Nach einem erfolgreichen E-Mail-Versand erscheint die folgende Meldung:



[Setup Dialog nach erfolgreichem E-Mail-Versand]

E-Mail-Versand mit Office 365

Erfolgt der E-Mail-Versand über einen Office 365 Server ist zunächst das Setup wie oben beschrieben durchzuführen. Der E-Mail-Versand-Test wird aber fehlschlagen.

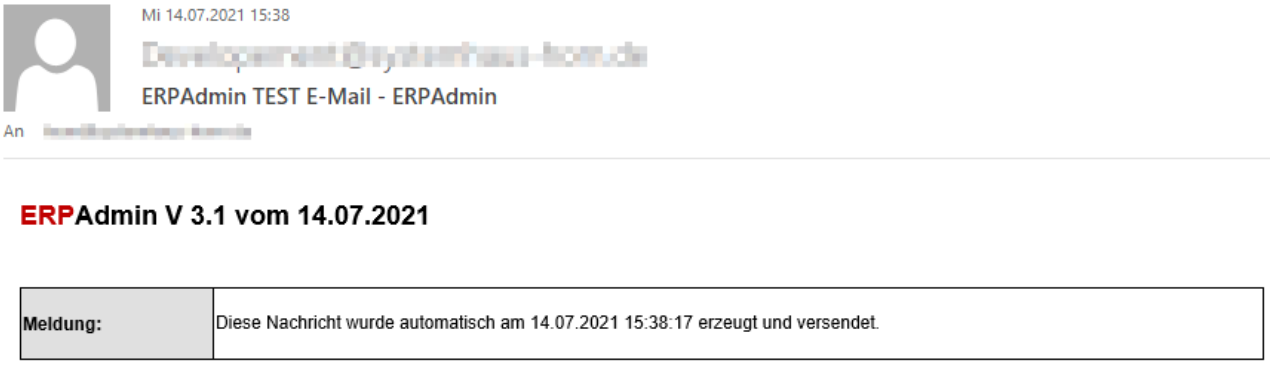
Öffnen Sie die vom Setup erzeuge **VAR.INI** und ändern/ergänzen Sie die folgenden Einträge:

C_SMTPSERVER	**= "smtp.office365.com"**	' SMTP-Server/IP-Adresse
C_SMTPSERVERPORT	**= "587"**	' SMTP-Server-Port (25/465/587)
C_SMTPSSL	**= FALSE**	' SMTP-Versand mit SSL-Verschlüsselung
C_SMTPTLS	**= TRUE**	' SMTP-Versand mit TLS-Verschlüsselung

Testen Sie danach den SMTP-Versand mit folgendem Aufruf:

ERPAdmin -Email <Ihre Email-Adresse>

Prüfen Sie anschließend Ihren Posteingang. War der E-Mail-Versand erfolgreich, finden Sie eine neue E-Mail mit folgendem Inhalt im Postfach:



> **Hinweis**:
> Die **OAuth2**-Authentifizierung für Office 365 wird aktuell noch nicht unterstützt. Bei Bedarf sprechen Sie uns bitte an.

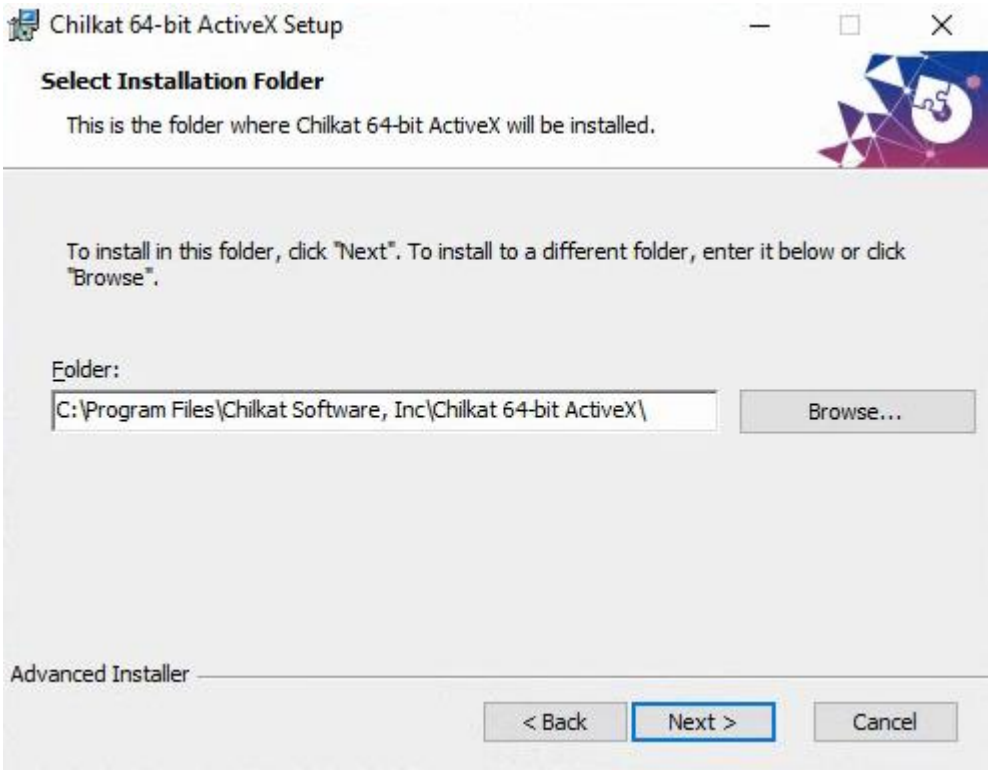
4 Installation der Chilkat-DLL's

Für die Verwendung von **ERPAdmin** wird die Unterstützung der Chilkat-DLLs benötigt. Damit diese genutzt werden können müssen zuerst die beiden mitgelieferten Dateien „**chilkatax-9.5.0-win32.msi**“ und „**chilkatax-9.5.0-x64.msi**“ installiert werden.

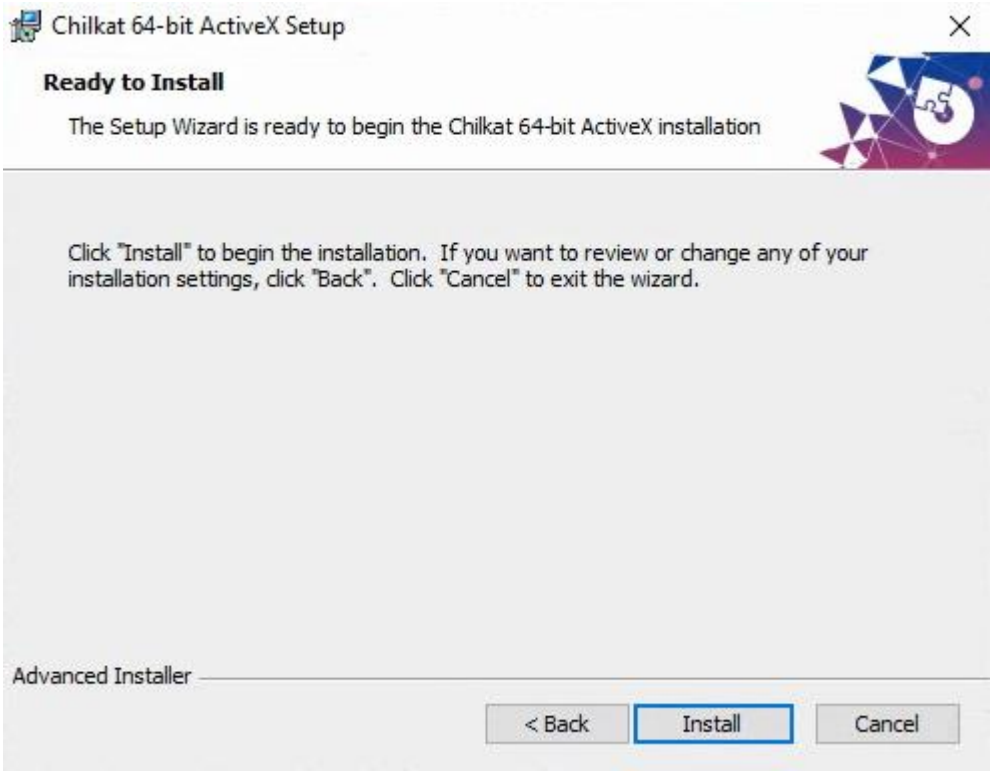
Dazu müssen Sie mit Administrator-Rechten unter Windows angemeldet sein. Wechsel Sie im Explorer in das Verzeichnis in das Sie **ERPAdmin** entpackt haben und in dem sich die beiden MSI-Dateien befinden.

Führen Sie nun nacheinander die beiden Dateien per Doppelklick aus. Es startet dann der nachfolgende Assistent zur Installation.

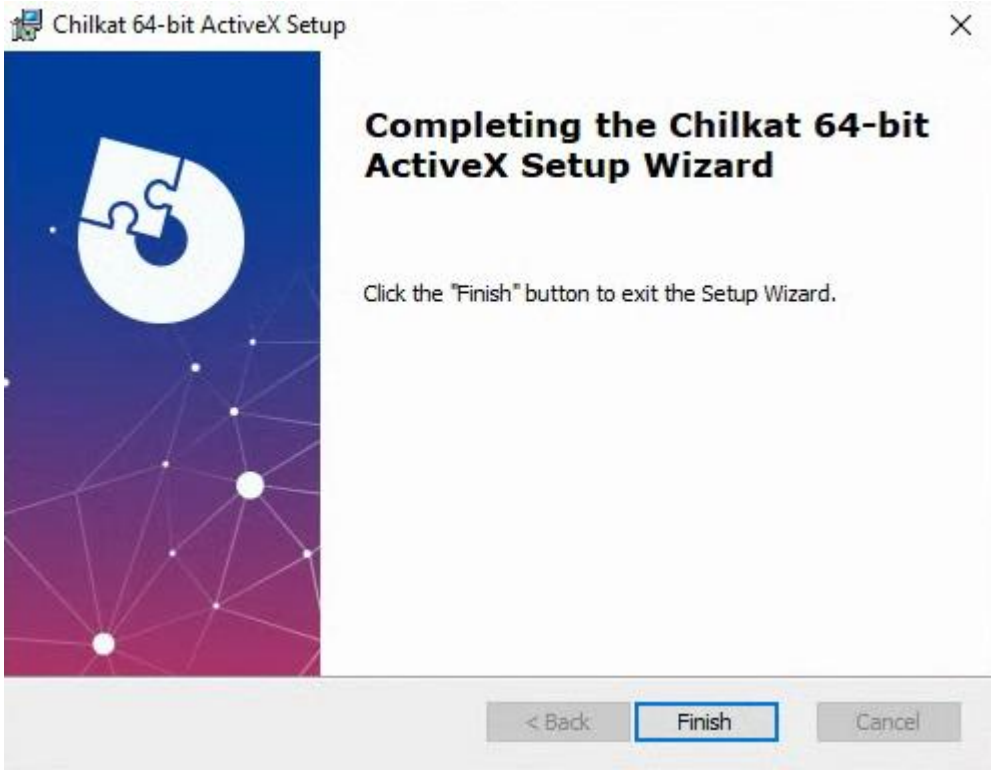
Klicken Sie auf „**Next**“ um mit der Installation zu starten.



Wählen Sie hier das Verzeichnis aus in das die Dateien entpackt werden sollen.

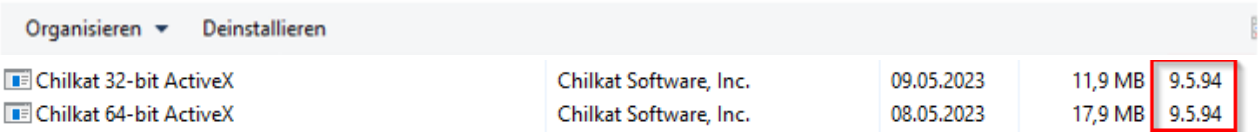


Klicken Sie auf „**Install**“ um den Installationsvorgang zu starten. Sollten Sie nicht über ausreichende Rechte verfügen erscheint ein Dialog in dem Sie Zugangsdaten eines Administrativen Benutzers eingeben müssen.



Nach erfolgter Installation Klicken Sie auf „**Finish**“.

In der Systemsteuerung unter „Programme und Features“ finden sich dann beide Installationen wieder. Hier kann auch überprüft werden welche Version der Chilkat DLLs auf dem PC installiert ist.



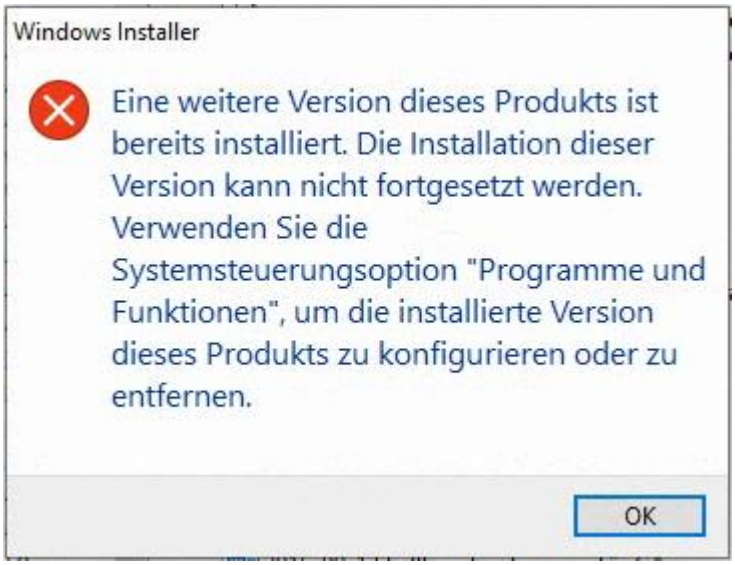
Rufen Sie nun **ERPAdmin** auf um zu prüfen ob die Chilkat-Komponenten korrekt erkannt werden.

ERPAdmin -init

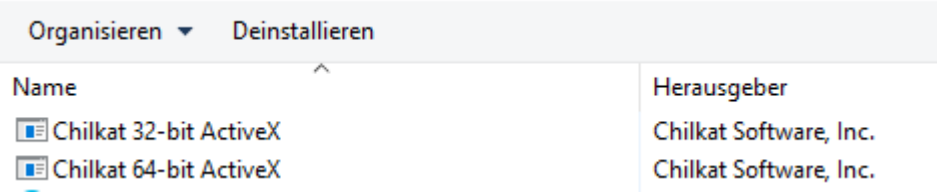
Wenn die DLLs korrekt registriert wurde, sollte im Meldungsfenster die Versionen für **FTP, SFTP, CSV** und **Mailman** angezeigt werden.

Update der vorhandenen Chilkat-DLLs

Bevor eine neue Version der Chilkat-DLLs installiert werden kann, muss eine bereits vorhandene Version deinstalliert werden. Sie erhalten sonst die folgende Meldung:



Wechseln Sie dazu in die **Systemsteuerung** von Windows und dort in den Bereich „**Programme und Features**“. Suchen Sie die beiden Einträge in der Liste und deinstallieren Sie beide.



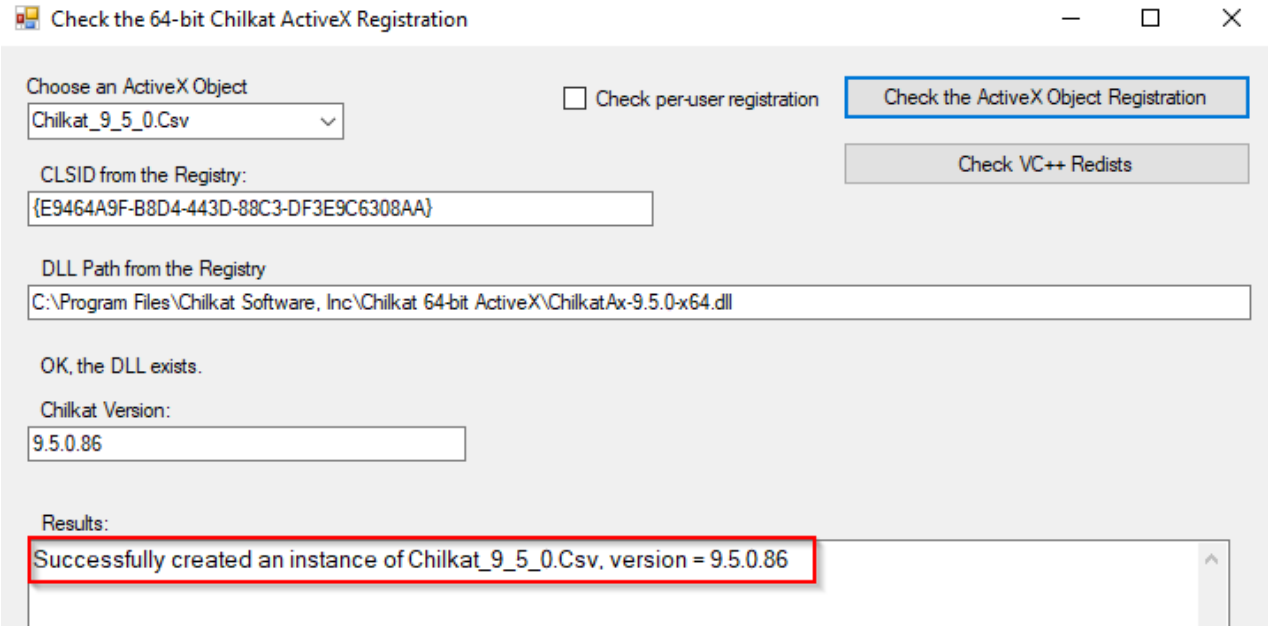
Nach erfolgreicher Deinstallation können Sie die neue Version über den MSI-Installer installieren.

<u>Überprüfung der Installation der Chilkat-DLLs</u>

Nach erfolgreicher Installation der beiden Chilkat-DLLs befinden sich die DLL-Dateien (wenn kein anderer Pfad bei der Installation ausgewählt wurde) in folgenden Verzeichnissen:

- c:\Program Files (x86)\Chilkat Software, Inc\Chilkat 32-bit ActiveX\
- c:\Program Files\Chilkat Software, Inc\Chilkat 64-bit ActiveX\

In jedem der beiden Verzeichnisse befindet sich auch das Programm „**CheckChilkatActiveX.exe**“ zur Überprüfung der Installation. Nach dem Aufruf und nach Auswahl eines beliebigen Active-X Objektes unter „**Choose an ActiveC Object**“ und anschließender Betätigung des Buttons „**Check the ActiveX Registration**“ sollte dann eine positive Bestätigungsmeldung erscheinen.



5 Die Datei „VAR.INI“ STD PRO

In der Datei **VAR.INI** hinterlegen Sie alle Voreinstellungen für die Benutzung von **ERPAdmin**. Hier eine Auswahl der Parameter die möglich sind:

Parameter	Beschreibung
C_LICENSE	Lizenzschlüssel **NUR** für **ERPAdmin** -Standard Version
C_LICENSEPATH	Pfad zur Datei mit weiteren Angaben **(optional)**
C_SMTPSERVER	SMTP-Server/IP-Adresse
C_SMTPSERVERPORT	SMTP-Port z. B. 25, 465, 587
C_SMTPUSER	SMTP-Benutzername
C_SMTPPWD	SMTP-Kennwort
C_SMTPSSL	SMTP-Versand mit SSL-Verschlüsselung (TRUE oder FALSE)
C_SMTPTLS	SMTP-Versand mit TLS-Verschlüsselung (TRUE oder FALSE)
C_OK_FROM	Absenderadresse für Bestätigungsmeldungen
C_OK_TO	E-Mail-Empfänger für Bestätigungsmeldungen
C_OK_SUBJECT	E-Mail-Betreff für Bestätigungsmeldungen
C_WARNING_FROM	Absenderadresse für Warnmeldungen **(optional)**
C_WARNING_TO	E-Mail Empfänger für Warnmeldungen (**(optional)**)
C_WARNING_TO2	E-Mail-Empfänger2 für Warnmeldungen **(optional)**
C_WARNING_SUBJECT	E-Mail-Betreff für Warnmeldungen **(optional)**
C_ERROR_FROM	Absenderadresse für Fehlermeldungen
C_ERROR_TO	E-Mail-Empfänger für Fehlermeldungen
C_ERROR_TO2	E-Mail-Empfänger2 für Fehlermeldungen **(optional)**
C_ERROR_SUBJECT	E-Mail-Betreff für Fehlermeldungen
C_EMAIL_Signatur_HTML	Übergibt eine E-Mail-Signatur im HTML-Format an **ERPAdmin**. Soll keine Signatur verwendet werden kann ein Leerzeichen (Kein Leerstring) übergeben werden. **(optional)**
C_BPN_Firma	Name der Firma auf die **büro+** registriert ist
C_BPN_User	Benutzername für Anmeldung in Mandant 1
C_BPN_Pwd	Passwort für Benutzeranmeldung in Mandant 1
C_BPN_MANDNR	Mandanten-Nummer 1
C_BPN_User2	Benutzername für Anmeldung in Mandant 2 **(optional)**
C_BPN_Pwd2	Passwort für Benutzeranmeldung in Mandant 2 **(optional)**
C_BPN_MANDNR2	Mandanten-Nummer 2 **(optional)**
C_BPN_User3	Benutzername für Anmeldung in Mandant 3 **(optional)**
C_BPN_Pwd3	Passwort für Benutzeranmeldung in Mandant 3 **(optional)**
C_BPN_MANDNR3	Mandanten-Nummer 3 **(optional)**
C_BPN_User4	Benutzername für Anmeldung in Mandant 4 **(optional)**

C_BPN_Pwd4	Passwort für Benutzeranmeldung in Mandant 4 **(optional)**
C_BPN_MANDNR4	Mandanten-Nummer 4 **(optional)**
C_BPN_User5	Benutzername für Anmeldung in Mandant 5 **(optional)**
C_BPN_Pwd5	Passwort für Benutzeranmeldung in Mandant 5 **(optional)**
C_BPN_MANDNR5	Mandanten-Nummer 5 **(optional)**
C_FTPSERVER	FTP-Server oder IP-Adresse 1
C_FTPUSER	FTP-Benutzername 1
C_FTPPWD	FTP-Passwort 1
C_FTPPASSIV	"0" = AKTIV Mode, "1" = PASSIV Mode
C_FTPTLS	"0"=Standard FTP, "1"=TLS explicite (AUTH SSL) **(optional)**
C_FTPSSL	"0"=Standard FTP, "1"=SSL implicite (AUTH SSL) **(optional)**
C_FTPPort	FTP-Port, Default="21" **(optional)**
C_FTPSERVER2	FTP-Server oder IP-Adresse 2 **(optional)**
C_FTPUSER2	FTP-Benutzername 2 **(optional)**
C_FTPPWD2	FTP-Passwort 2 **(optional)**
C_FTPPASSIV2	"0" = AKTIV Mode, "1" = PASSIV Mode **(optional)**
C_FTPTLS2	"0"=Standard FTP, "1"=TLS explicite (AUTH SSL) **(optional)**
C_FTPSSL2	"0"=Standard FTP, "1"=SSL implicite (AUTH SSL) **(optional)**
C_FTPPort2	FTP-Port, Default="21" **(optional)**
C_FTPSERVER3	FTP-Server oder IP-Adresse 3 **(optional)**
C_FTPUSER3	FTP-Benutzername 3 **(optional)**
C_FTPPWD3	FTP-Passwort 3 **(optional)**
C_FTPPASSIV3	"0" = AKTIV Mode, "1" = PASSIV Mode **(optional)**
C_FTPTLS3	"0"=Standard FTP, "1"=TLS explicite (AUTH SSL) **(optional)**
C_FTPSSL3	"0"=Standard FTP, "1"=SSL implicite (AUTH SSL) **(optional)**
C_FTPPort3	FTP-Port, Default="21" **(optional)**
C_FTPSERVER4	FTP-Server oder IP-Adresse 4 **(optional)**
C_FTPUSER4	FTP-Benutzername 4 **(optional)**
C_FTPPWD4	FTP-Passwort 4 **(optional)**
C_FTPPASSIV4	"0" = AKTIV Mode, "1" = PASSIV Mode **(optional)**
C_FTPTLS4	"0"=Standard FTP, "1"=TLS explicite (AUTH SSL) **(optional)**
C_FTPSSL4	"0"=Standard FTP, "1"=SSL implicite (AUTH SSL) **(optional)**
C_FTPPort4	FTP-Port, Default="21" **(optional)**
C_FTPSERVER5	FTP-Server oder IP-Adresse 5 **(optional)**
C_FTPUSER5	FTP-Benutzername 5 **(optional)**
C_FTPPWD5	FTP-Passwort 5 **(optional)**
C_FTPPASSIV5	"0" = AKTIV Mode, "1" = PASSIV Mode **(optional)**
C_FTPTLS5	"0"=Standard FTP, "1"=TLS explicite (AUTH SSL) **(optional)**
C_FTPSSL5	"0"=Standard FTP, "1"=SSL implicite (AUTH SSL) **(optional)**
C_FTPPort5	FTP-Port, Default="21" **(optional)**

C_SFTPServer	SFTP-Server oder IP-Adresse 1
C_SFTPUser	SFTP Benutzer 1
C_SFTPPwd	SFTP Kennwort 1
C_SFTPPort	SFTP Port 1
C_SFTPPKFILE	SFTP-Public-Key-File 1 (als alternative Authentifizierung zum Password) **(optional)**
C_SFTPServer2	SFTP-Server oder IP-Adresse 2 **(optional)**
C_SFTPUser2	SFTP Benutzer 2 **(optional)**
C_SFTPPwd2	SFTP Kennwort 2 **(optional)**
C_SFTPPort2	SFTP Port 2 **(optional)**
C_SFTPPKFILE2	SFTP-Public-Key-File 2 (als alternative Authentifizierung zum Password) **(optional)**
C_SFTPServer3	SFTP-Server oder IP-Adresse 3 **(optional)**
C_SFTPUser3	SFTP Benutzer 3 **(optional)**
C_SFTPPwd3	SFTP Kennwort 3 **(optional)**
C_SFTPPort3	SFTP Port 3 **(optional)**
C_SFTPPKFILE3	SFTP-Public-Key-File 3 (als alternative Authentifizierung zum Password) **(optional)**
C_SFTPServer4	SFTP-Server oder IP-Adresse 4 **(optional)**
C_SFTPUser4	SFTP Benutzer 4 **(optional)**
C_SFTPPwd4	SFTP Kennwort 4 **(optional)**
C_SFTPPort4	SFTP Port 4 **(optional)**
C_SFTPPKFILE4	SFTP-Public-Key-File 4 (als alternative Authentifizierung zum Password) **(optional)**
C_SFTPServer5	SFTP-Server oder IP-Adresse 5
C_SFTPUser5	SFTP Benutzer 5 **(optional)**
C_SFTPPwd5	SFTP Kennwort 5 **(optional)**
C_SFTPPort5	SFTP Port 5 **(optional)**
C_SFTPPKFILE5	SFTP-Public-Key-File 5 (als alternative Authentifizierung zum Password) **(optional)**
C_SFTPUploadDir	SFTP Default Verzeichnis für Upload **(optional)**
C_SFTPDownloadDir	SFTP Default Verzeichnis für Download **(optional)**
C_LayoutGruppe_GL	Modul aus dem exportiert (importiert werden soll z.B. Au, Fi, Lo **(optional)**
C_HISTORYDIR	Speichert in diesem Verzeichnis die History-Dateien. **(optional)**
C_HistoryDirDeleteFolderByDays	Löscht alle Unterverzeichnisse der Historie die älter sind als xxx Tage. Achtung: Erhöht die Laufzeit erheblich. Nicht für mehrfach tägliche Intervall-Aufrufe geeignet. Pro Tag wird nur ein CleanUp auf Performanz Gründen ausgeführt

C_LOGFILEDIR	Legt in diesem Verzeichnis eine Kopie des Logfiles ab **(optional)**
C_LogfileDirDeleteFilesByDays	Lösche alle Logfiles die älter sind als xxx Tage. Pro Tag wird nur ein CleanUp auf Performanz Gründen ausgeführt. **(optional)**
C_BADDIR	Verzeichnis für die Speicherung fehlerhafter Daten. **(optional)**
C_MaxLogFileSize	Maximale Größe der Logdatei in Byte bevor diese autom. gelöscht wird. Die Löschung erfolgt ggf. bei Programmstart **(optional)**
C_HALT_ON_ERROR	**TRUE**: Tritt ein Fehler bei der Ausführung von ERPAdmin auf, muss die Datei „**-ERROR-**“ vom Benutzer gelöscht werden damit ERPAdmin wieder ausgeführt werden kann. So lange die Datei „-**ERROR-**“ existiert, wird kein weiterer Befehl von ERPAdmin ausgeführt. Der Benutzer wird per E-Mail darüber informiert **(optional)**
C_KillERPAdminTimediff	Angabe in Sekunden. Beendet autom. alle laufenden Instanzen von ERPAdmin welche vor dieser Zeit gestartet wurden und noch aktiv sind. Standardwert=**43200** (12 Stunden) **(optional)**
C_KillBpNexTTimediff	Angabe in Sekunden. Beendet autom. alle laufenden Instanzen von BpNexT.exe welche vor dieser Zeit gestartet wurden und noch aktiv sind. Kein Standardwert. Achtung: Achten Sie darauf, dass auf dieser Maschine keine Automatisierung läuft und auch keine anderen COM-Anwendungen. **(optional)**
C_KillCMDTimediff	Angabe in Sekunden. Beendet autom. alle laufenden Instanzen von CMD.exe welche vor dieser Zeit gestartet wurden und noch aktiv sind. Kein Standardwert. **(optional)**
C_BP_PROTOCOL	**TRUE** wenn ein Protokoll in **büro+** geschrieben werden soll, **FALSE** sonst. **(optional)**
C_VorgangImportEinhErsetzung	Liste mit Ersetzungen durch Semikolon getrennt. Beispiel: "**10=kg;20=Stk.;30=liter**"
C_DoubletFilesMax	Gibt die max. Anzahl an Einträgen vor die eine Doubletten-Datei haben darf. Wird diese max. Anzahl überschritten wird eine Hinweis-Email an den Administrator (**C_ERROR_TO**) versendet. **(optional)**
C_DoubletFilesCutTo	Ist der angegebene Wert größer 0 und wurde die Anzahl **C_DoubletFilesMax** der Einträge in einer Doubletten-Datei überschritten, wird diese Doubletten-Datei auf die Anzahl **C_DoubletFilesCutTo** an Einträge reduziert. Dabei bleiben immer die neusten Einträge erhalten. **(optional)**
C_BuchenMeldeMinusBestand	**TRUE** wenn beim Buchen der Lagerbestand negativ werden würde und eine **Fehlermeldung** ausgegeben

	werden soll. **FALSE** es wird nur eine normale Meldung ausgegeben. **(optional)**
C_STOP	**TRUE** wenn die Ausführung von **ERPAdmin** wegen Wartungsarbeiten unterbunden werden soll. **(optional)**
C_NoOkEmail	**TRUE** unterdrückt alle OK-Emails an den Benutzer. Die Option **/NoOKEmail** wird dann nicht mehr benötigt. **(optional)**
C_DiskSpaceLowDrive	Laufwerksbuchstabe bei dem der freie Speicherplatz überwacht werden soll
C_DiskSpaceLow	Angabe freier Speicherplatz im MB. Bei Unterschreitung erfolgt eine Fehlermeldung per E-Mail. **(optional)**
C_DiskSpaceLowEmail	E-Mail-Adresse an die eine Fehlermeldung bei Unterschreitung des feien Speicherplatzes gesendet werden soll. Ist keine E-Mail angegeben erfolgt der Versand an '**C_ERROR_TO**', **(optional)**
C_CSV_Delimiter	Übergibt ein abweichendes Trennzeichen für die Datensätze. Standard ist Semikolon (;). **(optional)**
C_CSV_EnableQuotes	**TRUE:** Beim Speichern einer CSV-Datei werden Text-Felder in Anführungszeichen gesetzt, **FALSE** sonst. **(optional)**
C_CSV_EscapeBackslash	**TRUE:** Aktiviert die Erkennung von Escape-Sequenzen innerhalb der Datei. **FALSE** sonst. **(optional)**

Weitere Einträge für die **VAR.INI** Datei finden Sie bei den jeweiligen **ERPAdmin** Befehlen.

Beispiel für eine VAR.INI Datei:

C_SMTPSERVER	= "smtp.systmehaus-horn.de"
C_SMTPSERVERPORT	= "465"
C_SMTPUSER	= "ERPAdmin@systemhaus-horn.de"
C_SMTPPWD	= "123kennwort"
C_SMTPSSL	= TRUE
C_SMTPTLS	= FALSE
C_BPN_Firma	= "Systemhaus HORN"
C_BPN_User	= "Supervisor"
C_BPN_Pwd	= "123kennwort"
C_BPN_MANDNR	= "100"
C_ERROR_FROM	= "ERPAdmin@systemhaus-horn.de "
C_ERROR_TO	= "support@systemhaus-horn.de"
C_OK_FROM	= "ERPAdmin@systemhaus-horn.de"
C_OK_TO	= "horn@systemhaus-horn.de"
C_OK_SUBJECT	= "[ERPAdmin] OK: "
C_WARNING_SUBJECT	= "[ERPAdmin] WARNING: "
C_ERROR_SUBJECT	= "[ERPAdmin] ERROR: "

C_FTPSERVER = "ftp.systemhaus-horn.de"
C_FTPUSER = "Benutzername"
C_FTPPWD = "123kennwort"
C_FTPPASSIV = "1"
C_FTPTLS = "0"

C_SFTPSERVER = "sftp.systemhaus-horn.de"
C_SFTPUSER = "SFTP-Benutzername"
C_SFTPPWD = "123kennwort"
C_SFTPPORT = "2222"
C_SFTPPKFILE = ""

C_HISTORYDIR = "C:\ERPAdmin\History\"
C_MaxLogFileSize = "100000"
C_LICENSE = "8U7Z6-438TQ-5V35C-GS5PP-WPWNG-I89AK" Nur Standard Version

C_BP_PROTOCOL = TRUE

C_KillERPAdminTimediff ="3600"
C_KillBpNexTTimediff ="43200"

C_EMAIL_Signatur_HTML = "<SMALL>© by Systemhaus HORN • Eibenstr. 6 • D-63322 Rödermark • Telefon: +49 6074 / 88 10 46 • Fax: +49 6074 / 88 10 47 • www.ERPAdmin.de</SMALL>"

Hinweis:
Die Werte der Variablen müssen in **Anführungszeichen** gesetzt werden. Ausnahmen bilden die **Boolean**-Werte **TRUE** und **FALSE**. Diese dürfen **NICHT** in Anführungszeichen gesetzt sein!

Sind doppelte Anführungszeichen (") in einer Zeichenkette vorhanden, müssen diese maskiert werden. Ersetzen Sie jedes doppelte Anführungszeichen durch zwei doppelte Anführungszeichen ("") innerhalb der Zeichenkette (Siehe Beispiel **C_EMAIL_Signatur_HTML)**.

6 Rückmeldungen an den Benutzer

ERPAdmin erzeugt standardmäßig keine Bildschirmmeldungen. Alle Rückmeldungen an den Benutzer erfolgen per E-Mail. Darüber hinaus werden alle wichtigen Informationen in einer LOG-Datei gespeichert, die im **ERPAdmin**-Verzeichnis erzeugt wird und unter dem Namen **ERPAdmin <PC-Name> <Timestamp>-<ERPAdmin-Befehl>.log** gespeichert wird. Trat eine Warnmeldung oder ein Fehler währen der Ausführung von **ERPAdmin** auf, wird der Logfile-Dateiname um den Zusatz **_WARNING** oder **_ERROR** erweitert.

Hinweis:
Mehrere E-Mail-Adressen können mit einem Semikolon getrennt übergeben werden. So kann überall bei **ERPAdmin** verfahren werden. In der **VAR.INI,** aber auch bei der Übergabe von Parametern oder Optionen wie z.B**. /Email**, **/EmailCC**, **/EmailBcc**.

Beispiel:
Wird auf dem PC \\BPServer am **20.01.2019** um **11:15 Uhr** der Befehl „**ERPAdmin -DokumentLinkCheck**“ ausgeführt wird eine Logfiledatei mit dem Namen „**ERPAdmin BPServer 2019-01-20 11.15.23-DokumentLinkCheck.log**“ erzeugt. Trat während der Ausführung ein Fehler auf trägt die Datei den Namen „**ERPAdmin BPServer 2019-01-20 11.15.23-DokumentLinkCheck_ERROR.log**“

In der **VAR.INI** kann mit der Variablen **C_LOGFILEDIR** ein Verzeichnis angegeben werden in das alle Logfiles bei Beendigung von **ERPAdmin** kopiert werden.

In diesem Verzeichnis sammelt sich so mit jedem Aufruf von **büro+** eine weitere Logdatei. Mit der Variable **C_LogfileDirDeleteFilesByDays** in der **VAR.INI** kann ein Zeitfenster in Tagen definiert werden nach dem die Logdateien automatisch gelöscht werden. Die Prüfung wird aus Performanz Gründen von **ERPAdmin** nur einmal am Tag ausgeführt. Manuell kann das Verzeichnis mit folgendem Befehl beliebig oft aufgeräumt werden.

Beispiel:
Lösche alle Dateien im Verzeichnis **c:\ERPAdmin_Logfiles** die älter sind als 30 Tage.

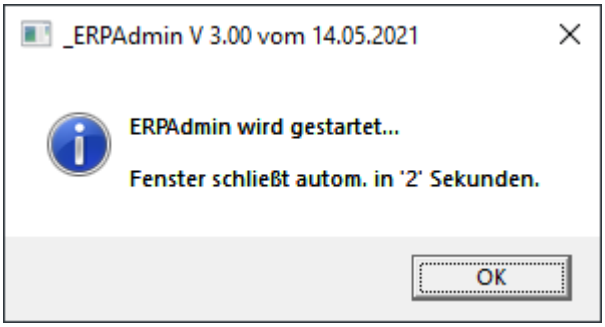
ERPAdmin -DeleteFilesByDate "c:\ERPAdmin_Logfiles\" @LAST30DAYS@

Hinweis:
Bleibt eine Instanz von **ERPAdmin** stehen und warten auf eine Benutzereingabe z.B. weil eine Meldung nicht deaktiviert wurde, erkennt die nachfolegnde **ERPAdmin**-Instanz dieseses und erzeugt eine Kopie des Logfiles der vorherigen Version im Verzeichnis **C_LogFileDir** (sofern dieses existiert) mit der Endung **.err**

Die Option /MsgBox PRO

Soll **ERPAdmin** eine Rückmeldung am Bildschirm ausgeben, z.B. wenn **ERPAdmin** direkt aus **büro+** heraus aufgerufen wird und um dem Anwender den Start und das Ende der Ausführung zu signalisieren, kann folgende Option verwendet werden: **/MsgBox**

Nach dem Start von **ERPAdmin** erscheint dann die folgende Meldung:

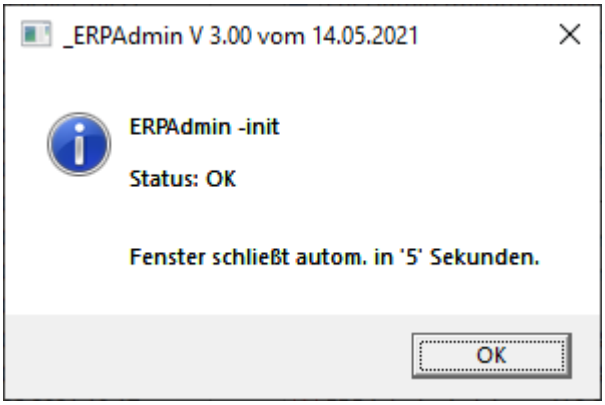


[Signalisiert den Programmstart von **ERPAdmin**]

Das Fenster wird automatisch nach 2 Sekunden geschlossen, wenn der Benutzer nicht auf **OK** klickt.

Nach Beendigung der Ausführung gibt **ERPAdmin** dann eine Meldung aus in der der **ERPAdmin**-Aufruf und der Status zurückgegeben wird, so dass der Benutzer weiß ob es zu einem Fehler gekommen ist oder nicht.

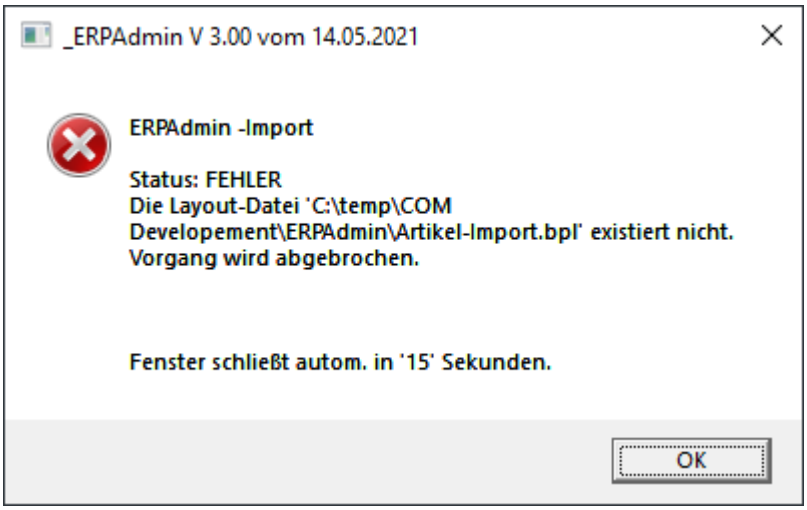
Folgende Meldung erscheint, wenn kein Fehler aufgetreten ist:



[Signalisiert das Ausführungs-Ende von **ERPAdmin -Init** ohne Fehler]

Das Fenster mit der OK-Meldung wird nach 5 Sekunden automatisch geschlossen.

Kam es bei der Ausführung von **ERPAdmin** zu einem Fehler, wird diese in der Meldung mit ausgegeben.



[Signalisiert das Ausführungs-Ende von **ERPAdmin -Import** mit Fehler]

Das Fenster mit der Fehlermeldung wird nach 15 Sekunden automatisch geschlossen.

Individual-Text

Es besteht die Möglichkeit, bei der Ende-Meldung einen individuellen Text anstelle des ausgeführten **ERPAdmin** -Befehls auszugeben. Dazu kann wie folgt ein Text übergeben werden:

"/MsgBox:<Individual-Text>"

Die Option /MsgBoxStart PRO

Soll **ERPAdmin** nur beim Starten eine Rückmeldung an den Benutzer geben, weil z.B. mehrere **ERPAdmin**-Befehle nacheinander ausgeführt werden, kann die Option **/MsgBoxStart** verwendet werden.

Es besteht die Möglichkeit, bei der Start-Meldung einen individuellen Text anstelle des ausgeführten **ERPAdmin** -Befehls auszugeben. Dazu kann wie folgt ein Text übergeben werden:

"/MsgBoxStart:<Individual-Text>"

Die Option /MsgBoxEnd PRO

Soll **ERPAdmin** nur beim Beenden eine Rückmeldung an den Benutzer geben, weil z.B. mehrere **ERPAdmin**-Befehle nacheinander ausgeführt werden, kann die Option **/MsgBoxEnd** verwendet werden.

Es besteht die Möglichkeit, bei der Ende-Meldung einen individuellen Text anstelle des ausgeführten **ERPAdmin** -Befehls auszugeben. Dazu kann wie folgt ein Text übergeben werden:

"/MsgBoxEnd:<Individual-Text>"

Hinweis:
Die Optionen **/MsgBoxStart**, **/MsgBox** oder **/MsgBoxEnd** dürfen nicht beim automatischen Aufruf von **ERPAdmin** verwendet werden.

Abfrage eines Wertes beim Benutzer /Abfrage1:<Abfrage-Text> PRO

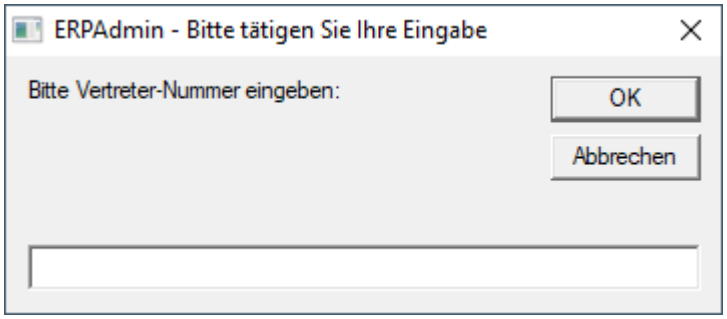
Wird **ERPAdmin** direkt aus **büro+** heraus aufgerufen, kann eine Eingabe mit der Option **/Abfrage1:<Abfrage-Text>** vom Benutzer abgefragt und an **ERPAdmin** übergeben werden. Das Ergebnis kann dann mit dem Platzhalten **@ABFRAGE1@** an **ERPAdmin** übergeben werden.

Weiterhin stehen die Optionen **/Abfrage2:<Abfrage-Text>** und **/Abfrage3:<Abfrage-Text>** mit den dazugehörigen Platzhaltern **@ABFRAGE2@** und **@ABFRAGE3@** zur Verfügung.

Beispiel:
Für die Ausgabe einer Vorgangs-Auswertung für die Vertreter über **ERPAdmin** soll die Vertreternummer abgefragt werden.

ERPAdmin -LISTEDRUCKEN Vorgang AuVogDrListe "Vorgangsliste-PDF" /Von:VtrNr=@ABFRAGE1@ /Bis:VtrNr=@ABFRAGE1@ "/Abfrage1:Bitte Vertreter-Nummer eingeben:"

Nach dem Aufruf von **ERPAdmin** erscheint dann folgende Abfrage:



[Abfrage der Vertreter-Nummer]

Hinweis:
Soll eine Abfrage abgebrochen werden und die Ausführung des aktuellen **ERPAdmin**-Befehls abgebrochen werden, muss die Zeichenfolge **@@@** eingegeben werden.

Insgesamt können drei Werte abgefragt werden mit den Optionen **/Abfrage1:<…>, /Abfrage2:<…>** und **/Abfrage3:<…>.** Als Platzhalter sind dann **@ABFRAGE1@, @ABFRAGE2@** und **@ABFRAGE3@** zu verwenden.

Wichtig:
Die Optionen **/Abfrage1:<Abfrage-Text>** dürfen nicht beim automatischen Aufruf von **ERPAdmin** verwendet werden da es sonst zu einer Verklemmung kommt.

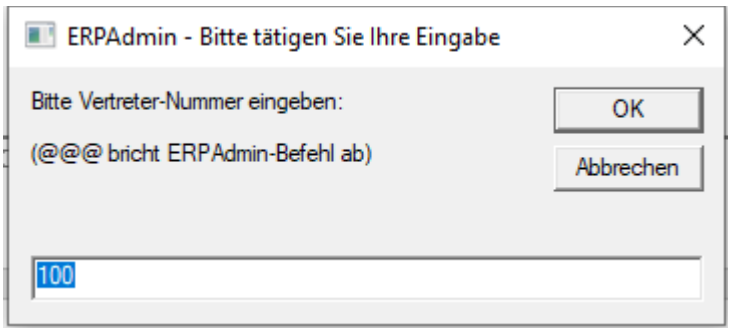
Vorbelegung einer Abfrage

Soll für eine Abfrage ein Vorgabewert in dem Eingabefeld erscheinen, kann dieser Wert mit einem doppelten Pipe-Zeichen (**||**) getrennt hinter der Frage übergeben werden.

Beispiel:

ERPAdmin -LISTEDRUCKEN Vorgang AuVogDrListe "Vorgangsliste-PDF" /Von:VtrNr=@ABFRAGE1@ /Bis:VtrNr=@ABFRAGE1@ "/Abfrage1:Bitte Vertreter-Nummer eingeben:||100"

Nach dem Aufruf von **ERPAdmin** erscheint dann folgende Abfrage mit Vorbelegungswert **100**:



[Abfrage der Vertreter-Nummer]

@ABFRAGE1@ bei späterem ERPAdmin Aufruf verwenden PRO

Es ist möglich, die abgefragten Werte über die Parameter **/Abfrage1:<…>** auch in einem späteren **ERPAdmin** Aufruf zu verwenden. Dazu muss in der **VAR.INI** der Parameter **C_Abfrage_Memory=TRUE** hinzugefügt werden. **ERPAdmin** merkt sich dann die Werte so lange bis in der **VAR.INI** der Eintrag **C_Abfrage_Memory=FALSE** gesetzt wird. So können Benutzereingabe beim ersten Aufruf eines **ERPAdmin** -Befehl gesammelt abgefragt werden und dann bei einer Folge von **ERPAdmin**-Aufrufen nach und nach verarbeitet werden.

Die Zwischengespeicherten Werte können auch gelöscht werden. Dazu muss bei einem **ERPAdmin** -Aufruf die Abfrage-Option ohne Werte angegeben werden.

Beispiel: Es wurden alle drei **Abfragen** ausgeführt und die Werte durch Angabe **C_Abfrage_Memory=TRUE** in der **VAR.INI** zwischen gespeichert. Nun soll nur der Abfrage-Wert2 gelöscht werden. Die Option sieht dann wie folgt aus: **"/Abfrage2: "**

Werte für einen späteren ERPAdmin Aufruf übergeben /SetVar1, /SetVar2, /SetVar3 PRO

Sollen Werte eines **ERPAdmin** Aufrufes für einen späteren Aufruf gespeichert werden kann dies über die Option **/SetVar1:<Wert>** erfolgen. **ERPAdmin** speichert dann den angegebenen Wert. Auf dieser kann dann zu einem späteren Zeitpunkt über den Platzhalter **@SETVAR1@** zugegriffen werden. Voraussetzung hierfür ist, dass **ERPAdmin** auf demselben PC und mit demselben Windows Benutzer ausgeführt wird.

Insgesamt können drei Werte **/SetVar1:<Wert1>**, **/SetVar2:<Wert2>** und **/SetVar3:<Wert3>** gespeichert und zu einem späteren Zeitpunkt mit den Platzhaltern **@SETVAR1@, @SETVAR2@** und **@SETVAR3@** verwendet werden.

Beispiel:

Eine Umsatzauswertung aus büro+ soll mehrmals täglich mit dem Tagesdatum und der Uhrzeit (**@TIMESTAMP@**) im Dateinamen erzeugt werden (**Export**). Da die Auswertung längere Zeit läuft, liefert

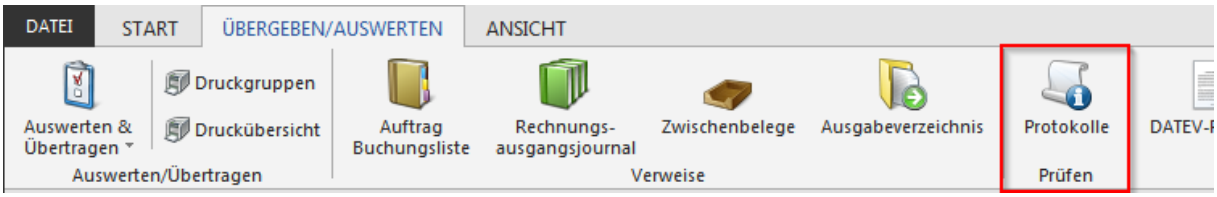
der Platzhalter **@TIMESTAMP@** bei verschiedenen **ERPAdmin**-Aufrufen evtl. verschiedene Zeiten zurück. Anschließend soll der Inhalt noch mit dem Befehl **ERPAdmin -CSVKonverter** angepasst werden (ISO-Land-Nr ‚276' in Spalte 14 durch ‚DE' ersetzen). Die Aufrufe könnten dann wie folgt aussehen:

ERPAdmin -Export Umsatz.bpl Umsatz@TIMESTAMP@.csv "/SetVar1: Umsatz@TIMESTAMP@.csv"

ERPAdmin -CSVKonverter @SETVAR1@ SearchAndReplace 14 276 DE

Anzeige der ERPAdmin -Protokolle in büro+

Zusätzlich können auch Protokoll-Einträge in **büro+** erzeugt werden, welches dann wie folgt eingesehen werden kann:



Zur Aktivierung der Protokollierung in **büro+** muss die Datei **VAR.INI** um folgenden Eintrag ergänzt werden: **C_BP_PROTOCOL=TRUE**

Sammeln von Rückmeldungen

Werden mehrere **ERPAdmin-**Befehle nacheinander ausgeführt erhält der Anwender nach jeder Ausführung von **ERPAdmin** eine Email als Rückmeldung, vorausgesetzt der SMTP-Versand wurde konfiguriert und der Aufruf von **ERPAdmin** wurde ohne die Option **/NoOKEMail** ausgeführt.

Dies kann unterbunden werden so dass pro Aufruf einer CMD-Datei nur eine Rückmeldung mit dem gesammelten Inhalt an den Benutzer verwendet wird. Weitere Informationen finden Sie weiter unten in der Dokumentation zum Befehl:
ERPAdmin -SendAtOnce <Email-Subject>

<Email-Subject>
Gibt den Inhalt der Betreff-Zeile an, die dem Benutzer zurückgemeldet wird.

Semaphoren

Damit **ERPAdmin** seinen Dienst im Hintergrund verrichten kann, sollten alle Meldungen deaktiviert sein (**C_BEDUG=FALSE**). Damit man dennoch sehen kann, ob das Programm gestartet wurde und ob es noch läuft, wird bei Programmstart die Datei „**___RUNNING_XXX__**" im Verzeichnis von **ERPAdmin** erzeugt. Nach Beendigung von **ERPAdmin** wird die Datei „**---END-XXX---**" erzeugt. Tritt ein Fehler auf, wird die Datei „**---ERROR-XXX---**" erzeugt. **XXX** steht dabei für den Maschinennamen von wo aus **ERPAdmin** aufgerufen wurde. Es ist immer nur eine der drei Dateien im Verzeichnis vorhanden. In der Datei ist der Zeitstempel enthalten, wann diese Datei erzeugt wurde.

Autom. Beenden von noch laufenden ERPAdmin Instanzen

Es kann vorkommen, dass **ERPAdmin (ERPAdmin.exe)** nicht korrekt beendet wird. Solche noch nicht beendeten Instanzen im Speicher können bei erneutem Start von **ERPAdmin** zu Fehlern führen. Um das Problem zu lösen, kann in der Datei **VAR.INI** eine Variable **C_KillERPAdminTimediff** gesetzt werden mit einer Angabe in Sekunden, welche alle noch laufende Instanzen bei Start von **ERPAdmin** automatisch beendet, wenn diese Instanzen seit **C_KillERPAdminTimediff** Sekunden läuft. Der Wert sollte ausreichend groß gewählt werden, so dass eine Instanz nur dann beendet wird, wenn diese Ihren Job schon längst ausgeführt haben sollte. Befindet sich eine Instanz von **ERPAdmin** seit mehr als 12 Stunden (43200 Sekunden) im Speicher, wird diese auch ohne eine Angabe in der **VAR.INI** beendet. Dieser Wert kann vom Anwender verlängert oder verkürzt werden.

Autom. Beenden von noch laufenden büro+ oder **CMD.exe Instanzen**

Es kann auch vorkommen, dass noch alte Instanzen von **büro+** (**BpNexT.exe**) im Speicher vorhanden sind. Diese können in gleicher Weise wir **ERPAdmin** beim Start von **ERPAdmin** beendet werden. In der Datei **VAR.INI** eine Variable **C_KillBpNexTTimediff** gesetzt werden mit einer Angabe in Sekunden, welche alle noch laufende Instanzen bei Start von **BpNexT.Exe** automatisch beendet, wenn diese Instanzen seit **C_KillBpNexTTimediff** Sekunden läuft. Einen Vorgabewert wie bei **KillERPAdminTimediff** gibt es nicht.

Wird **ERPAdmin** über eine **.CMD** Datei ausgeführt wird hierfür ein Prozess der CMD.exe verwendet. Auch diese Prozesse können autom. beendet werden, wenn in der Datei **VAR.INI** die Variable **C_KillCMDTimediff** gesetzt wurde. Der angegebene Wert entspricht den Sekunden seit dem Start von CMD.exe.

Hinweis:
Wenn auf dem PC die microtech Automatik oder eine andere COM-Anwendung arbeitet, sollte beachtet werden, dass auch diese ggf. auch beendet werden bei ausführen eines Kill-Befehls.

Rückmeldung an den Benutzer per E-Mail

Das Programm kann nach jedem erfolgreich ausgeführten Befehl automatisiert eine E-Mail an einen Benutzer versenden. In der Datei **VAR.INI** muss dafür unter **C_OK_FROM, C_OK_TO** und **C_OK_SUBJECT** mit gültigen Werten gefüllt sein. Weiterhin müssen die Einstellungen für den SMTP-Versand hinterlegt werden.

Achtung: Ab Version 2.0 ist die Einrichtung der E-Mail Benachrichtigung Voraussetzung für die Benutzung von ERPAdmin.

Bei auftretenden Fehlern kann ebenfalls eine automatische E-Mail an einen Benutzer versendet werden. Dazu müssen die Variablen **C_ERROR_FROM, C_ERROR_TO** und **C_ERROR_SUBJECT** in der Datei **VAR.INI** befüllt werden.

C_ERROR_FROM	Absenderadresse für Fehlermeldungen
C_ERROR_TO	E-Mail-Empfänger für Fehlermeldungen
C_ERROR_SUBJECT	Präfix für die Betreff-Zeile einer Fehler-E-Mail
C_ WARNING _FROM	Absenderadresse für Warnmeldungen
C_ WARNING _TO	E-Mail-Empfänger für Warnmeldungen
C_WARNING_SUBJECT	Präfix E-Mail-Betreff für Warnmeldungen

C_OK_FROM	Absenderadresse für Bestätigungsmeldungen
C_OK_TO	E-Mail-Empfänger für Bestätigungsmeldungen
C_OK_SUBJECT	Präfix E-Mail-Betreff für Bestätigungsmeldungen
C_EMAIL_CC	E-Mail-Adresse für CC (Kopie)
C_EMAIL_BCC	E-Mail-Adresse für BCC (Blinde-Kopie)

Einstellungen für den E-Mail-Versand per SMTP

Damit das Programm überhaupt E-Mails versenden kann, müssen folgende Angaben in der **VAR.INI** mit gültigen Werten befüllt sein:

C_SMTPSERVERSMTP-	Server/IP-Adresse
C_SMTPSERVERPORT	SMTP-Port z. B. 25, 465 oder 587
C_SMTPUSER	SMTP-Benutzername
C_SMTPPWD	SMTP-Kennwort
C_SMTPSSL	SMTP-Versand mit SSL-Verschlüsselung (TRUE oder FALSE)
C_SMTPTLS	SMTP-Versand mit TLS-Verschlüsselung (TRUE oder FALSE)

Beispiel:

Bei erfolgreicher Programmausführung soll eine E-Mail an den Chef (**chef@test.de**) versendet werden. Tritt ein Fehler auf, soll eine E-Mail an den Abteilungs-Leiter (**abteilungsleiter@test.de**) und an Sie als **büro+**-Betreuer (**betreuer@edv.de**) versendet werden.

Beispiel einer **VAR.INI** Datei:

```
### Einstellungen für den SMTP E-MAIL-Versand
C_SMTPSERVER        = "mailserver@test.de"        ' SMTP-Server/IP-Adresse
C_SMTPSERVERPORT    = "465“                       ' SMTP-Server-Port (25/465/587)
C_SMTPUSER          = "Benutzername"              ' SMTP-Benutzer
C_SMTPPWD           = "123Kenwort"                ' SMTP-Passwort
C_SMTPSSL           = TRUE                        ' SSL-Verschlüsselung (TRUE/FALSE)
C_SMTPTLS           = FALSE                       ' TLS-Verschlüsselung (TRUE/FALSE)


'### Vorbelegungen für E-Mail Benachrichtigungen bei auftretenden Fehlern
C_ERROR_FROM        = "erpadmin@test.de"          ' Absender der Error-E-Mails
C_ERROR_TO          = "abteilungsleiter@test.de"  ' 1. Empfängeradresse bei Fehlern
C_ERROR_TO2         = "betreuer@edv.de"           ' 2. Empfängeradresse bei Fehlern
C_ERROR_SUBJECT     = "[ERPAdmin] Error: "        ' Betreff der Error-E-Mails

'### E-Mail Bestätigung nach Programmabarbeitung
C_OK_FROM           = "erpadmin@test.de"          ' Absenderadresse für OK-E-Mails
C_OK_TO             = "chef@test.de"              ' Empfänger für OK-E-Mail
C_OK_SUBJECT        = "[ERPAdmin] OK: "           ' Betreff der OK-E-Mails
```

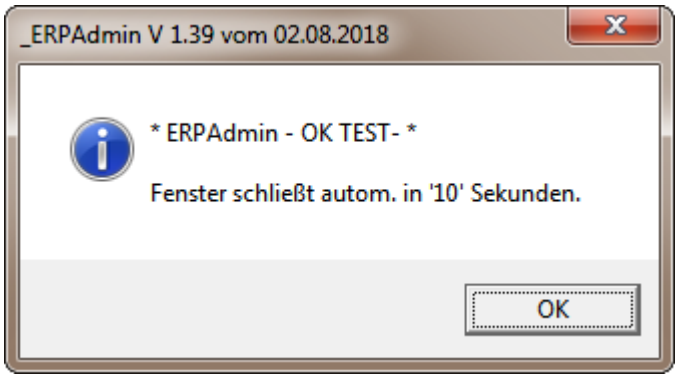
Hinweis:
Prüfen Sie innerhalb von **büro+** ob mit dem Benutzer mit dem sich **ERPAdmin** bei **büro+** anmeldet auch E-Mails versendet werden können.

Testen der Meldungen:
Sie können die drei verschiedenen Formen der E-Mail-Rückmeldung auch Testen. Rufen Sie **ERPAdmin** dazu wie folgt auf:

ERPAdmin -OK	Erzeugt eine OK-E-Mail
ERPAdmin -WARNING	Erzeugt eine Warning-E-Mail
ERPAdmin -ERROR	Erzeugt eine Error-E-Mail

Sie erhalten neben einer Bildschirmmeldung auch die dazugehörige E-Mail-Rückmeldung.

Beispiel für Bildschirm-Rückmeldung für **ERPAdmin -OK**



Beispiel für eine E-Mail Rückmeldung OK:

Mi 20.03.2019 17:28

Developement@systemhaus-horn.de

[Developer] OK: -OK

An OK@ERPAdmin.de

ERPAdmin V 2.1c vom 18.03.2019

Status:	**OK**

Verlauf:	* ERPAdmin - OK TEST- *

ERPAdmin-Befehl:	-ok

Start-Zeitpunkt: **Ende-Zeitpunkt:**	20.03.2019 17:27:45 20.03.2019 17:27:47
Programm-Laufzeit:	2 Sek.

Beispiel für eine E-Mail Rückmeldung WARNUNG:

Mi 20.03.2019 17:28

Developement@systemhaus-horn.de

[Developer] Warnung: -WARNING

An Error@ERPAdmin.de

ERPAdmin V 2.1c vom 18.03.2019

Status:	**WARNUNG**

Warn-Meldung:	* ERPAdmin - WARNING TEST- *

Verlauf:	* ERPAdmin - WARNING TEST- *

ERPAdmin-Befehl:	-warning

Start-Zeitpunkt: **Ende-Zeitpunkt:**	20.03.2019 17:27:55 20.03.2019 17:27:57
Programm-Laufzeit:	2 Sek.

Beispiel für eine E-Mail Rückmeldung FEHLER:

Mi 20.03.2019 17:28

Developement@systemhaus-horn.de

[Developer] ERROR: -ERROR

An Error@ERPAdmin.de

ERPAdmin V 2.1c vom 18.03.2019

Status:	**FEHLER**

Fehler-Meldung:	*ERPAdmin - ERROR TEST- *

Verlauf:	*ERPAdmin - ERROR TEST- *

ERPAdmin-Befehl:	-error

Start-Zeitpunkt: **Ende-Zeitpunkt:**	20.03.2019 17:28:02 20.03.2019 17:28:04
Programm-Laufzeit:	2 Sek.

An jeder **ERROR**-Meldung wird das aktuellen Logfile mit angehängt.

Timestamp:	**LogFile-Eintrag:**
Programm:	ERPAdmin V 2.1c vom 18.03.2019
Programmpfad:	C:\temp\COM Developement\ERPAdmin\
ComputerName:	PHW10
Username:	Peter.Horn
Install Date:	07.12.2018 12:57:29
Code Set:	1252
Country Code:	49
Betriebssystem:	Microsoft Windows 10 Pro
Version:	10.0.17134
20.03.2019 17:28:02	+----------+
20.03.2019 17:28:02	--- (C) 2019 by Systemhaus HORN - www.Systemhaus-HORN.de - Tel.: +49 6074 881046
20.03.2019 17:28:02	--- Programm ERPAdmin V 2.1c vom 18.03.2019 ---
20.03.2019 17:28:02	--- Programmpfad: **'C:\temp\COM Developement\ERPAdmin\'**
20.03.2019 17:28:02:226	SYSTEM_INIT: ----------
20.03.2019 17:28:02:226	SYSTEM_INIT: **'1'** Argument gefunden
20.03.2019 17:28:02:226	SYSTEM_INIT: G_ARG1="-error"
20.03.2019 17:28:02:226	SYSTEM_INIT: ----------
20.03.2019 17:28:02:226	SYSTEM_INIT: Lizenziert für **'Systemhaus HORN'**
20.03.2019 17:28:02:226	SYSTEM_INIT: Lizenz '[redacted] gültig bis: **'31.12.2019'**
20.03.2019 17:28:02:226	SYSTEM_INIT: PRINTENABLED - Professional-Lizenz in büro+/ERP wird benötigt!
20.03.2019 17:28:02:226	SYSTEM_INIT: ----------
20.03.2019 17:28:02:226	SYSTEM_INIT: /Mand1 (TEST)

Werden mehrere Parameter und Optionen beim Aufruf an **ERPAdmin** übergeben, werden die erkannten Werte in den E-Mail-Rückmeldungen in tabellarischer Form angezeigt. So können Sie schnell sehen, ob die Reihenfolge der übergebenen Parameter und Optionen korrekt erkannt wurde.

Mi 20.03.2019 09:55
ERPAdmin@ERPAdmin.de
[ERPAdmin] OK: -READDATA Adressen
An OK@ERPAdmin.de

ERPAdmin V 2.1b vom 12.03.2019

Status:	**OK**

Verlauf:	**'104'** Datensätze in **'c:\ERPAdmin\Beispiele\ReadData\Adressenliste.csv'** gespeichert. Die Datei **'c:\ERPAdmin\Beispiele\ReadData\Adressenliste.csv'** wurde per E-Mail an **'Demo@ERPAdmin.de'** versendet.

ERPAdmin-Befehl:	-ReadData **Adressen AdrNr;Suchbeg;Status;Branche /Datei:Adressenliste.csv /HEADER /Email:Demo@ERPAdmin.de "/Subject:Adressenliste (@FILENAME)"**
	ERP-Tabelle: Adressen **ERP-Feldliste:** AdrNr;Suchbeg;Status;Branche **/Datei:** Adressenliste.csv **/Email:** Demo@ERPAdmin.de **/Subject:** Adressenliste (@FILENAME) **/Header:** Wahr

Start-Zeitpunkt: **Ende-Zeitpunkt:**	20.03.2019 09:54:22 20.03.2019 09:54:30
Programm-Laufzeit:	8 Sek.

7 Aufrufen von ERPAdmin aus Windows

Der Aufruf von **ERPAdmin** erfolgt in der Kommandozeile von Windows wie folgt:

ERPAdmin -Befehl Parameter1 … Parameter14 /Option1 ….. /OptionX

-Befehl
Hier wird einer der in diesem Handbuch beschriebenen **ERPAdmin** -Befehle übergeben. Befehle beginnen immer mit einen Minus-Zeichen und stehen mit einem Leerzeichen getrennt hinter der **ERPAdmin** Aufruf.

Parameter1 … Parameter14
Je nach **ERPAdmin -Befehl** können verschiedene Parameter an **ERPAdmin** zur Steuerung des **Befehls** übergeben werden. Die Einhaltung der Reihenfolge ist dabei sehr wichtig. Enthält ein **Parameter** keinen Wert muss ein **"" (Leerstring)** übergeben werden damit die Reihenfolge der Nachfolge **Parameter** erhalten bleibt.

Die einzelnen **Parameter** und **Optionen** werden durch ein Leerzeichen getrennt an **ERPAdmin** übergeben.

Doppelte Anführungszeichen (") um die einzelnen Parameter und Optionen sind immer dann zu verwenden, wenn Leerzeichen, **"<", ">"** oder "**%**" innerhalb der Parameter oder der Optionen vorhanden sind. Sie dürfen nur dann weggelassen werden, wenn es sich bei dem Parameter oder der Option um eine Zeichenkette ohne Leerzeichen und ohne obige Sonderzeichen handelt. Im Zweifel können Sie immer Doppelte-Anführungszeichen verwenden.

Hinweis:
Windows kennt beim Programmaufruf sogenannte ESCAPE-Sequenzen. Dies sind Zeichenfolgen die ein bestimmtes Verhalten bei der Interpretation auslösen. Eine solche ESCAPE-Sequenz ist z.B. **\"**

Enthält ein Parameter Leerzeichen muss dieser in Anführungszeichen gesetzt werden. Endet dieser dann vor dem schließenden Anführungszeichen mit einem Backslash (****) wird das schließende Anführungszeichen bei der Übergabe an **ERPAdmin** ignoriert. Verwenden Sie in diesem Fall einen doppelten Backslash (****) vor dem schließenden Anführungszeichen.

Beispiel:

ERPAdmin -Update "" "" "c:\ERPAdmin\Test Beispiele\\"

/Option1 …/OptionX
Nach den Parametern folgen die Optionen. Diese beginnen immer mit einem Slash (**/**). Enthält eine Option Leer- oder Sonderzeichen muss diese Option, genau wie entsprechende Parameter, in doppelte Anführungszeichen (") gesetzt werden.

Wie automatisiere ich ERPAdmin Aufrufen?
Für die Automatisierung von **ERPAdmin** lesen Sie bitte das Kapitel „**Automatisierung von ERPAdmin mit dem Windows Aufgabenplaner**“ am Ende dieser Dokumentation.

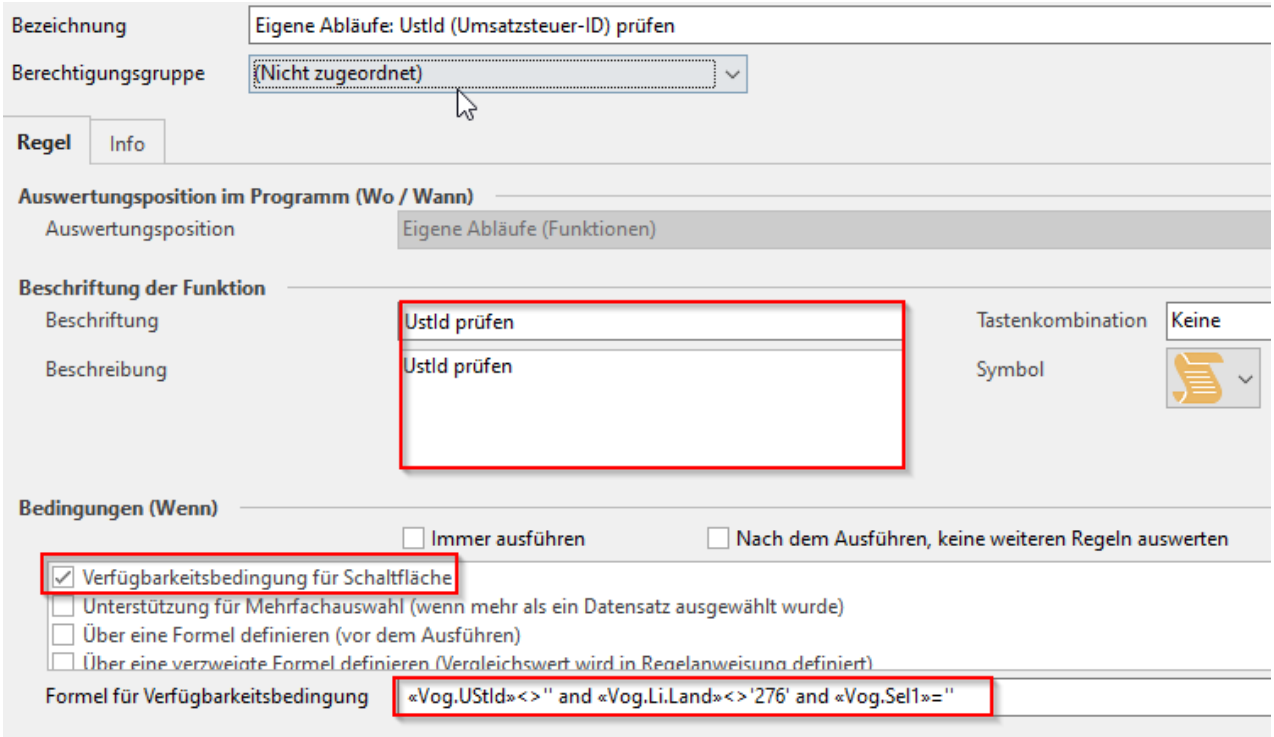
8 Aufruf von ERPAdmin durch den Benutzer direkt aus büro+

Bei einigen Anwendungs-Szenarien kann es notwendig sein, dass der Aufruf von **ERPAdmin** nicht automatisiert erfolgt, sondern vom Benutzer direkt aus **büro+** erfolgt.

Ein Anwendungsszenario wäre z.B. die Prüfung der Umsatzsteuer-ID in einem einzelnen Vorgang. Dazu verwenden wir den Befehl **ERPAdmin -VorgangUstIDCheck Sel1 Sel2 /Von:BelegNr=<BelegNrVon> /Bis:<BelegNrBis>**

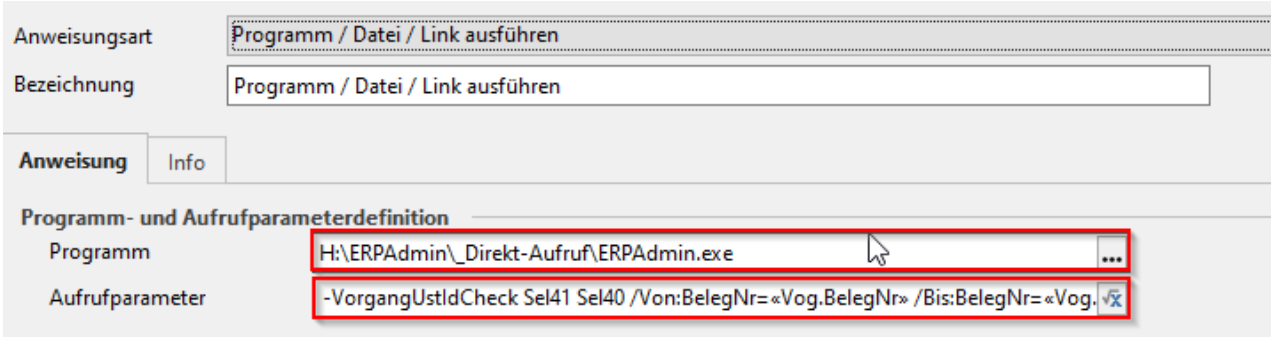
Die Rückmeldungen werden in unserem Beispiel in den Vorgangs-Selektionen **Sel1** und **Sel2** gespeichert.

Für den Aufruf von **ERPAdmin** eignet sich die Vorgangs-Regel „**Eigene Abläufe (Funktionen)**“. Diese wurde wie folgt erstellt:



Folgende „**Verfügbarkeitsbedingung für Schaltfläche**“ wurde hinterlegt:
«Vog.UStId»<>'' and «Vog.Li.Land»<>'276' and «Vog.Sel1»=''

Dann wurde die folgende **Anweisung** erstellt:

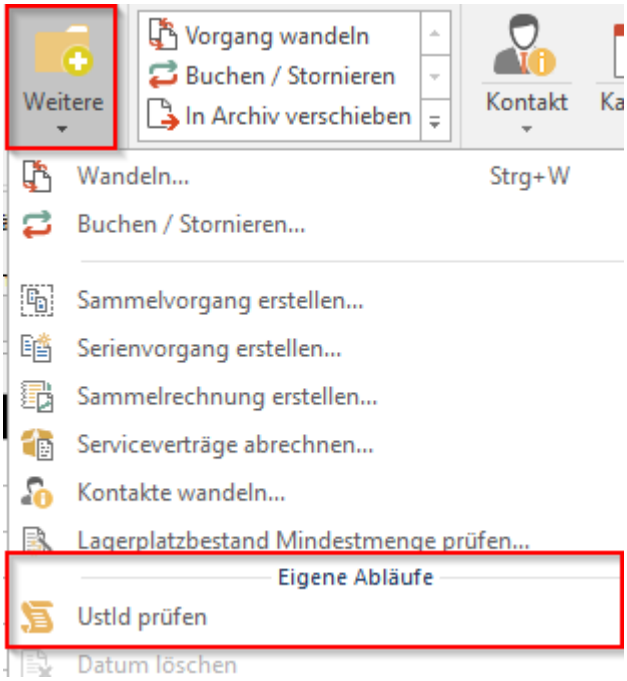


Bei „**Programm**“ wurde die **ERPAdmin.exe** inkl. Pfad hinterlegt.
H:\ERPAdmin_Direkt-Aufruf\ERPAdmin.exe

Beim „**Aufrufparameter**“ wurde folgenden Eintrag hinterlegt:

-VorgangUstIdCheck Sel1 Sel2 /Von:BelegNr=«Vog.BelegNr» /Bis:BelegNr=«Vog.BelegNr» /MsgBox

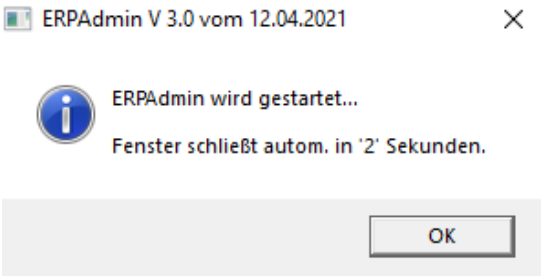
Nach einem Neustart von **büro+** findet der Benutzer dann den folgenden Eintrag im Bereich „**Vorgänge->Weitere**“:



Sofern die „**Verfügbarkeitsbedingung für Schaltfläche**“ erfüllt ist, kann der Benutzer nun die Prüfung der Umsatzsteuer-ID für den aktuell-markierten Vorgang durch Klick auf den Eintrag „**UstID prüfen**“ starten (Mehrfachselektion ist nicht möglich).

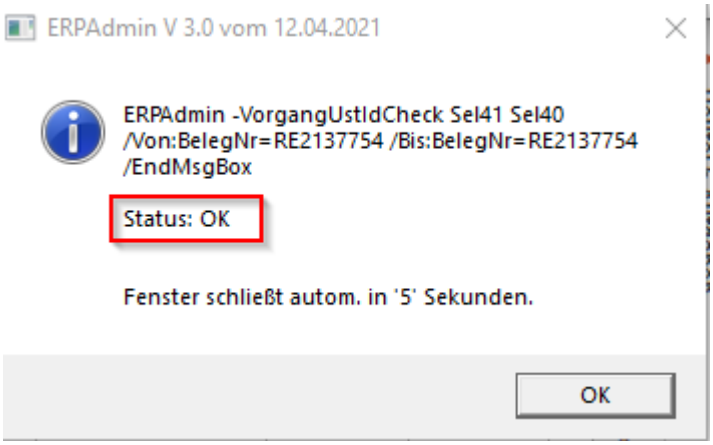
> **Hinweis:**
> Die Funktionen der „**Eigenen Abläufe**“ werden **NICHT** bei bereits gebuchten Vorgängen angezeigt.
> Für gebuchte Vorgänge können diese nicht verwendet werden.

Nach dem Aufruf erscheint eine kurze Rückmeldung das **ERPAdmin** gestartet wurde:



[Rückmeldung nach dem Aufruf]

Ein wenig später quittiert **ERPAdmin** dann das Ende seiner Tätigkeit mit folgendem Fenster:



In der Mitte wird ein Status zurückgegeben. In diesem Fall ein **OK**. Die Umsatzsteuer-ID konnte also erfolgreich geprüft werden und das Ergebnis findet sich in den beiden angegebenen Selektionsfeldern im Vorgang wieder.

Für die Aktivierung der beiden Fenster beim Start und beim Beenden von **ERPAdmin** ist die Option **/MsgBox** verantwortlich die beim Aufruf mitgegeben wurde.

Hinweis:
Wenn **ERPAdmin** von einem Arbeitsplatz aus aufgerufen werden soll, müssen auf diesem PC die Chilkat-DLLs installiert sein.

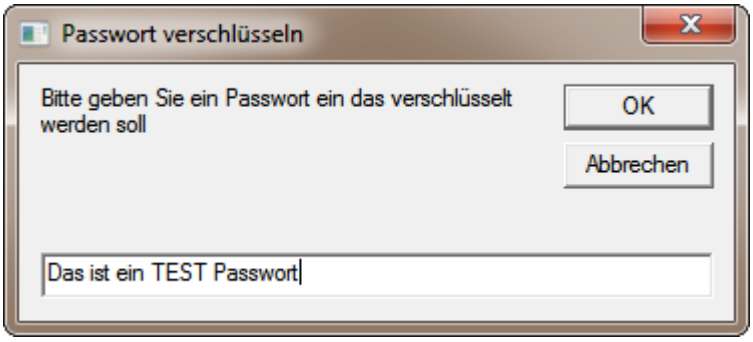
9 Verschlüsselung von Passworten mit ERPAdmin BAS

In der **VAR.INI** werden die Zugangsdaten für **büro+**, für die **FTP**- und **SFTP**-Server und für den E-Mail-Versand per **SMTP** abgespeichert. Sofern vom Benutzer nicht anders angegeben werden diese Daten im Klartext hinterlegt.

Sollen die Passworte in der **VAR.INI** nicht für jeden Benutzer direkt sichtbar sein, können diese verschlüsselt werden. Rufen Sie **ERPAdmin** dazu wie folgt auf:

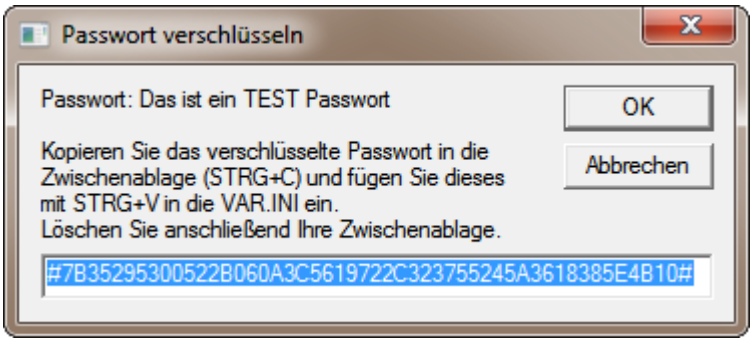
ERPAdmin -CryptPWD

Es erscheint der folgende Dialog. Tragen Sie in das Eingabefenster Ihr Passwort ein welches verschlüsselt werden soll.



Klicken Sie dann auf **OK.**

Sie bekommen jetzt das verschlüsselte Passwort angezeigt.



Kopieren Sie das verschlüsselte Passwort mit **STRG+C** in die Zwischenablage und tragen Sie es mit **STRG+V** an der entsprechenden Stelle in die **VAR.INI** ein. Löschen Sie anschließend Ihre Zwischenablage.

Wiederholen Sie den Vorgang für alle **büro+** Zugangsdaten sowie für die **FTP**-Passworte, **SFTP**-Passworte und für den **SMTP**-Versand.

Verschlüsselte Passworte in der **VAR.INI** erkennen Sie daran, dass diese immer mit einem **#** beginnen und enden. Beispiel: #5A262A120D4C36485D65090107#

Um Klartext-Passworte die bereits in der **VAR.INI** hinterlegt wurden nachträglich zu verschlüsseln lesen Sie bitte weiter unten die Details zu dem Befehl **ERPAdmin -CryptPWD**.

10 Historisierung

Ist die Variable **C_HISTORYDIR** in der Datei **VAR.INI** mit einem Verzeichnis belegt, erzeugt **ERPAdmin** in diesem Verzeichnis ein weiteres Unterverzeichnis mit Zeitstempel als Verzeichnis-Namen (Beispiel: **2015-03-11 10.38.55-Befehl**). In diesem Verzeichnis wird dann automatisch eine Sicherungskopie der exportierten und importierten Dateien abgelegt wobei die Zeichenkette **Befehl** durch den jeweiligen **ERPAdmin** Befehl ersetzt wird. Die Pfadangaben in der **VAR.INI** müssen immer mit einem "****" enden.

Ist in der **VAR.INI** die Variable **C_HISTORYDIR** mit einem Namen belegt, prüft **ERPAdmin** bei der Ausführung als erstes ab, ob dieses Verzeichnis existiert und beendet ggf. die Ausführung von **ERPAdmin.**

Verschiedene Befehle unterstützen eine Historisierung. Dies sind z.B.:

- Export
- FTPGet
- FTPPut
- Import
- SFTPGet
- SFTPPut

Der in der **VAR.INI** hinterlegte Wert **C_HISTORYDIR** kann überschrieben werden. Die Option sieht dazu wie folgt aus: **/HisDir:<Verzeichnis>**

Tritt bei der Verarbeitung von Dateien ein Fehler auf können diese Dateien in einem gesonderten Verzeichnis abgespeichert werden. Dazu ist in der **VAR.INI** der Variablen **C_BADDIR** ein Verzeichnis zu hinterlegen.

Weitere Informationen finden Sie auch im Kapitel „**Optionen und Parameter**“ für die Optionen **/Delete**, **/HIS**, **/MoveToHis, /MoveToBad, /MovetoHis:<Verzeichnis>, /MoveToBad:<Verzeichnis>**

Automatisches Aufräumen des History-Verzeichnisses

Mit der **VAR.INI**-Variable **C_HistoryDirDeleteFolderByDays** kann ein Zeitfenster in Tagen definiert werden nach dem die Unterverzeichnisse des History-Verzeichnisses automatisch gelöscht werden. Aus Performanz Gründen führt **ERPAdmin** diese Bereinigung pro Kalendertag nur einmal aus.

Manuell kann das Verzeichnis mit folgendem Befehl beliebig oft aufgeräumt werden.

Beispiel:
Es sollen alle History-Unterverzeichnisse gelöscht werden die älter sind als 30 Tage.

ERPAdmin -DeleteFoldersByDate "c:\ERPAdmin_History\" @LAST30DAYS@

11 Optionen und Parameter von ERPAdmin

Alle Parameter müssen an **ERPAdmin** durch Leerzeichen getrennt übergeben werden. Befindet sich innerhalb eines Parameters ein Leerzeichen oder ein „**<**“ oder „**>**“, muss der ganze Parameter in Anführungszeichen gesetzt werden.

Ist das letzte Zeichen einer Zeichenkette die in Anführungszeichen steht ein Backslash „****“ muss dieser verdoppelt werden, weil es sich sonst um eine Escape-Sequenz handelt und Windows ignoriert das schließende Anführungszeichen.

Beispiel: **"c:\ERPAdmin\Bilder\" => "c:\ERPAdmin\Bilder\\"**

Zusätzlich zu den Parametern können verschiedene Optionen an **ERPAdmin** übergeben werden welche entweder Einfluss auf die Verarbeitung durch **ERPAdmin** haben oder an **büro+** weitergegeben werden. Auch hier gilt die Regel mit den Leerzeichen. Optionen die Leerzeichen enthalten müssen in Anführungszeichen gesetzt werden. Optionen beginnen immer mit einen Forslah (**/**)

An **ERPAdmin** können max. 12 Parameter oder Optionen in der Summe übergeben werden.

Möchten Sie für einen Parameter keinen Wert übergeben, ihn also weglassen, verwenden Sie doppelte Anführungszeichen: **""** (Leerstring)**.** Sie können den Parameter auch weglassen, wenn dieser der letzte in der Befehlszeile ist oder danach nur noch Optionen kommen die mit einem Slash **„/“** beginnen.

Die Reihenfolge der Optionen spielt keine Rolle.

Nicht alle Optionen stehen jedem **ERPAdmin** Befehl zur Verfügung. Sie finden bei jedem Befehl in der Dokumentation eine Auflistung welche Optionen verfügbar sind.

Nachfolgend eine Liste aller Optionen:

/Administrator

Mit dem Aufruf **/Administrator** wird **ERPAdmin** mit erweiterten Rechten ausgeführt. Je nach Einstellung der „**Benutzerkontensteuerung**“ erscheint eine Sicherheitsabfrage.

/Von:<Bedingung>
Übergibt einen Wert für den Anfangsbereich einer Eingrenzung

/Bis:<Bedingung>
Übergibt einen Wert für den Ende Bereich einer Eingrenzung. Wird kein**/Bis:** Wert übergeben wird der **/Von:** Wert verwendet analog zu den Eingabemasken der Sortierungen in **büro+**.

/CharSet:<Character-Set>
Gibt den Zeichensatz an, in dem **ERPAdmin** eine Datei erzeugen soll. Mögliche Werte sind z.B.: **ANSI**, **UNICODE**, etc.

/Datei:<Dateiname>
Übergibt einen Datennamen inkl. Pfadangaben.

/Debug
Aktiviert den DEBUG-Modus. Im DEBUG-Modus werden die Rückmeldungen an den Benutzer nicht

unterdrückt, sondern als Meldungen am Bildschirm ausgegeben. Das kann bei einer Fehlersuche hilfreich sein.

/Delete
Übergibt den Befehl zur Löschung einer Datei oder eines Datensatzes oder dem Inhalt eines Feldes während oder nach der Befehlsausführung.

Sollen die Quell-Dateien nach einer erfolgreichen Verarbeitung gelöscht werden. kann die Option **/Delete** angegeben werden. Bei erfolgloser Verarbeitung bleibt die Datei im Verzeichnis bestehen.

Nach einem FTP-Download wird die Datei nach erfolgreicher Übertragung auf dem FTP-Server gelöscht.

/Drucker:<Druckernamen>
Übergibt den Namen eines abweichenden Druckers für die Druckausgabe.

/Email:<E-Mail-Adresse>, /EMailBCC: <E-Mail-Adresse>, /EmailBCC:<E-Mail-Adresse>
Übergibt eine oder mehrere E-Mail-Adressen. Mehrere E-Mail-Adressen sind durch Semikolons zu trennen.

/Filter:<Filterausdruck>
Übergibt eine Filterbedingung. Weitere Infos zur Verwendung von Filtern finden Sie in dem Kapitel „**Benutzung von Filtern**“.

/FTP1, /FTP2, /FTP3… /FTP10
Sie können in der **VAR.INI** für fünf verschiedene FTP-Server Zugangs Daten hinterlegen. Erflogt ein FTP-Aufruf z.B. **-FTPPUT** oder **-FTPGET** kann durch Angabe der Option **/FTP1, /FTP2, FTP3, … /FTP10** angegeben werden, welche der Zugangsdaten aus der **VAR.INI** verwendet werden sollen.

Für die Verwendung der Optionen **/FTP4** bis **/FTP10** benötigen Sie eine Professional-Lizenz. PRO

Beispiel:

ERPAdmin -FTPGet BestellungenServer1.xls /FTP3
Führt einen FTP-Abruf der Datei „**BestellungenServer1.xls**“ mit den Zugangsdaten für FTP-Server 3 aus.

/GL:<Modul>

Beim Export von Daten, die in mehreren Programm-Modulen von der **büro+** zur Verfügung stehen z.B. Offene-Posten, Dokumente, Bilder, Projekte, etc. muss angegeben werden, aus welchem Modul (Auftrag (Au), Fibu (Fi) oder Lohn (Lo)) der Export erfolgen soll. Zulässig sind hier die folgenden Angaben:

- **Au**ftrag = **Au**
- **Fi**bu = **Fi**
- **Lo**hn = **Lo**

Wird vom Benutzer keine Angabe gemacht, geht **ERPAdmin** von keinem speziellen Modul aus und es bleibt bei der Gruppe „**Gl“**.

Beispiel: Wenn ein Export aus den **Offenen-Posten** durchgeführt werden soll, sehen die Vorgaben in **büro+** folgt aus:

Druckposition	
Modul	GI
Gruppe	OPoExpListe
Layout ID	2954

Da die Offenen-Posten sowohl im Modul **Auftrag** als auch im Modul **Fibu** existieren, benötigt **büro+** einen Hinweis, aus welchem Modul der Export ausgeführt werde soll. Durch Angabe der Option **/GL:Au** wird der Export aus dem Modul **Auftrag** ausgeführt.

/Header
ERPAdmin erzeugt in der Ergebnisdatei eine Kopfzeile (z.B. bei einem Export) oder ignorierte den ersten Datensatz (z.B. bei einem Import), weil es eine Kopfzeile ist. Details zum verhalten entnehmen Sie bitte dem jeweiligen Befehl.

/HELP
Gibt eine Hilfefenster zu einem Befehl aus in dem die Notation und die Reihenfolge der Parameter und Optionen beschrieben sind. In den meisten Fällen wird auch ein Anwendungs-Beispiel ausgegeben.

/HIS
Beim Import oder Export von Daten in die Archiv-Bereiche **VorgangArchiv** und **KontaktArchiv** von **büro+** muss dieser Schalter gesetzt werden. Wird der Schalter nicht gesetzt, erfolgt der Import bzw. Export immer in die Bereiche **Vorgang** oder **Kontakt**.

/HisDir:<Verzeichnis>
Nach der Ausführung von bestimmten **ERPAdmin**-Befehlen können verarbeitete oder erzeugte Dateien in das History-Verzeichnis kopiert werden um eine Historisierung vorzunehmen. In der **VAR.INI** wird das History-Verzeichnis mit der Variable **C_HistoryDir** dazu vorbelegt. Die Ergebnis-Dateien werden dann nach Ausführung in das History-Verzeichnis (**HISTORYDIR\TIMESTAMP)** kopiert. Dies passiert z.B. nach jedem **büro+**-Export oder nach einem FTP-Download.

Das History-Verzeichnis kann direkt mit dem Parameter **/HisDir:** übergeben werden und überschreibt damit die Vorgabe aus der **VAR.INI**.

/LogDir:<Verzeichnis> PRO
Überschreibt den in der **VAR.INI** mit der Variablen **C_LogfileDir** hinterlegten Wert für das Logfile-Verzeichnis für die Dauer der Ausführung des aktuellen **ERPAdmin** Befehls.

/Mand1, /Mand2, /Mand3, /Mand4 PRO **, /Mand5** PRO
Mit diesen Optionen werden die Anmeldungen an verschiedene Mandanten gesteuert. Die hierfür benötigten Anmeldedaten müssen vorher in der **VAR.INI** hinterlegt werden. Damit haben Sie die Möglichkeit, sich nacheinander an verschiedenen Mandanten der **büro+** anzumelden, um z. B. Daten aus Mandant 1 zu exportieren und das Ergebnis direkt in Mandant 2 zu importieren.

Für die Verwendung der Optionen **/Mand4** und **/Mand5** benötigen Sie eine Professional-Lizenz.

Folgende Einstellungen können in der **VAR.INI** hinterlegt werden:

C_BPN_Firma	Name der Firma auf die **büro+** registriert wurde

C_BPN_User	Benutzername für Anmeldung in Mandant 1
C_BPN_Pwd	Passwort für Benutzeranmeldung in Mandant 1
C_BPN_MANDNR	Mandanten-Nummer 1
C_BPN_User2	Benutzername für Anmeldung für Mandant 2
C_BPN_Pwd2	Passwort für Benutzeranmeldung in Mandant 2
C_BPN_MANDNR2	Mandanten-Nummer 2
C_BPN_User3	Benutzername für Anmeldung für Mandant 3
C_BPN_Pwd3	Passwort für Benutzeranmeldung in Mandant 3
C_BPN_MANDNR3	Mandanten-Nummer 3
C_BPN_User4	Benutzername für Anmeldung für Mandant 4
C_BPN_Pwd4	Passwort für Benutzeranmeldung in Mandant 4
C_BPN_MANDNR4	Mandanten-Nummer 4
C_BPN_User5	Benutzername für Anmeldung für Mandant 5
C_BPN_Pwd5	Passwort für Benutzeranmeldung in Mandant 5
C_BPN_MANDNR5	Mandanten-Nummer 5

/MoveToHis - Verschieben erfolgreich verarbeiteter Dateien

Einige **ERPAdmin-**Befehle können Dateien nach einer erfolgreichen Verarbeitung in das History-Verzeichnis verschieben damit diese bei einem Folgeaufruf nicht erneut verarbeitet werden. Dazu kann der Parameter **/MoveToHis** angegeben werden.

Die Dateien werden dann in das History-Verzeichnis (**HISTORYDIR\TIMESTAMP)** verschoben. Die Einstellung erfolgt über die Datei **VAR.INI** über die Variable **C_HistoryDir**.

Das History-Verzeichnis kann auch direkt mit dem Parameter übergeben werden wie folgt: **/MoveToHis:<Verzeichnisname>.** Wird ein History-Verzeichnis mit übergeben wird der in der VAR.INI vorbelegte Wert für diesen **ERPAdmin** -Befehl überschrieben.

> **Hinweis:** Dieser Parameter steht nicht für alle Befehl zur Verfügung. Schauen Sie bei den jeweiligen Befehlen ob der Parameter unterstützt wird.

/MoveToBad - Verschieben NICHT verarbeiteter Dateien:

Einige **ERPAdmin-B**efehle können Dateien nach einer **NICHT** erfolgreichen Verarbeitung in ein Verzeichnis verschieben damit diese bei einem Folgeaufruf nicht erneut verarbeitet werden. Dazu kann der Parameter **/MoveToBad** angegeben werden.

Die Dateien werden dann in das Bad-Verzeichnis (**C_BADDIR\TIMESTAMP)** verschoben. Die Einstellung erfolgt über die Datei **VAR.INI**. Das Bad-Dir-Verzeichnis kann auch direkt mit dem Parameter übergeben werden in folgender Form: **/MoveToBad:<Verzeichnisname>**

/NoOkEmail und **C_NoOkEmail =TRUE**
ERPAdmin versendet nach jedem erfolgreich ausgeführten Befehl automatisiert eine E-Mail an den Benutzer. In der Datei **VAR.INI** kann unter **C_OK_TO** angegeben werden, an welche Adresse diese E-Mail versendet werden soll. **/NoOkEmail** unterdrückt das Versenden einer Bestätigungsmail an den Benutzer für den aktuellen Aufruf. Sollen die OK-Emails dauerhaft unterbunden werden kann in der **VAR.INI** auch der Parameter **C_NoOkEmail=TRUE** hinterlegt werden. E-Mails mit Fehlerreport werden auch bei Unterdrückung der OK-Bestätigung versendet, sofern die benötigen Werte **C_ERROR_TO** mit einer gültigen E-Mail-Adresse korrekt vorbelegt sind.

C_OK_TO	E-Mail-Adresse des Empfängers für den Status-Bericht
C_ERROR_TO	E-Mail-Adresse des Empfängers für den Fehler-Bericht

Für den E-Mail-Versand müssen alle anderen benötigen Angaben in der **VAR.INI** mit gültigen Werten vorbelegt sein.

/NoLockError
Kann ein Datensatz nicht geschrieben werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

/NoSendAtOnce PRO
Wurde in der **VAR.INI** mit der Variablen **C_EMAIL_SENDATONCE** das Sammeln der Rückmeldungen aktiviert, kann mit diesem Parameter eine sofortige Rückmeldung des aktuellen **ERPAdmin**-Befehles erzeugt werden. Bisher gesammelte **SendAtOnce** Rückmeldungen bleiben davon unberührt.

/NoWarningEmail
Kommt es während der Ausführung von **ERPAdmin** zu einer Warnung, versendet **ERPAdmin** eine entsprechende E-Mail an den Benutzer. Sollen diese Benachrichtigungen unterdrückt werden, kann die Option **/NoWarningEmail** verwendet werden. In der **VAR.INI** kann eine abweichende Absender- und Empfänger-E-Mail-Adresse für Warnungen hinterlegt werden sowie ein spezieller Betreff:

C_WARNING_FROM	Absender-E-Mail-Adresse
C_WARNING_TO	Empfänger-E-Mail-Adresse
C_WARNING_SUBJECT	E-Mail-Betreff der Vorangestellt wird

/SFTP1, /SFTP2, /SFTP3, /SFTP4, /SFTP5 PRO

Sie können in der **VAR.INI** für drei verschiedene SFTP-Server Zugangs Daten hinterlegen. Erflogt ein SFTP-Aufruf z.B. **-SFTPPUT** oder **-SFTPGET** kann durch Angabe der Option **/SFTP1** oder **/SFTP2** oder **/SFTP3** angegeben werden, welche der Zugangsdaten aus der **VAR.INI** verwendet werden sollen.

Für die Verwendung der SFTP-Funktionen benötigen Sie eine Professional-Lizenz.

ERPAdmin -SFTPGet BestellungenServer1.xls /SFTP3
Führt einen SFTP-Abruf mit den Zugangsdaten für SFTP-Server 3 aus.

/Subject:<E-Mail-Betreff>
Bekommt ein Benutzer das Ergebnis eines **ERPAdmin**-Befehlt mit der Option **/Email:<E_Mail-Adresse>**

zugesendet kann auch eine individuelle Betreffzeile übergeben werden. Erzeugt der **ERPAdmin**-Befehl eine Ergebnisdatei, kann der Dateiname mit dem Platzhalter **@FILENAME@** in den Betreff eingefügt werden. Es können alle Platzhalter verwendet werden welche im Kapitel „**Platzhalter und variable Parameter**“ beschrieben sind.

/VAR.INI:<Andere-INI-Datei>
Um unterschiedliche Arbeiten mit **ERPAdmin** durchzuführen, können Sie verschiedene Verzeichnisse erzeugen, um dort in **VAR.INI**-Dateien die unterschiedlichen Einstellungen abzulegen.

Sie können aber auch im selben Verzeichnis mehrere **INI**-Dateien erzeugen und **ERPAdmin** mitteilen, welche dieser **INI**-Dateien verwendet werden sollen. Dabei wird zuerst die **VAR.INI**-Datei von **ERPAdmin** verarbeitet und anschließend die zusätzlich angegebene Datei. Die Werte aus der **VAR.INI** werden dann mit den Werten aus der zusätzlichen INI-Datei überschrieben.

In der überschreibenden INI-Datei müssen daher nur die Werte stehen, welche Sie gegenüber der **VAR.INI** ändern möchten. Sollen Werte entfernt werden, muss eine leere Zeichenfolge übergeben werden.

Hinweis:
Es können nicht alle Werte der **VAR.INI** durch eine übergebene **INI**-Datei überschrieben werden!

12 Benutzung einer Bereichsauswahl /Von: …. /Bis:

Einige **ERPAdmin**-Befehle unterstützen eine Bereichsauswahl. Diese kann über die beiden Parameter **/Von:** und **/Bis:** an **ERPAdmin** übergeben werden. Auf diese Weise kann die Datenbasis eines Exportes oder einer Liste eingeschränkt werden. Zusätzlich kann noch ein Filterwert **/Filter:** mit übergeben werden („Benutzung von Filtern“ siehe nächstes Kapitel).

Für die Bereichseinschränkung sind nur Felder erlaubt, welche auch eine Sortierung besitzen. Mit dem Befehl **ERPAdmin -ListDatenbankFelder /Index** kann eine solche Liste mit allen Suchindexfelder aus **büro+** erstellt werden.

Für die Verwendung der Parameter **/Von:** und **/Bis:** wird bei einigen Befehlen die Professional-Version von **ERPAdmin** benötigt!

Wird ein Datenbereich von **büro+** durch eine Bereichsauswahl oder einen Filter eingeschränkt, werden damit die in **büro+** bereits hinterlegten Filter oder Sortiereingrenzungen vollständig überschrieben. Die Angaben sind **nicht** additiv. Dabei ist es egal ob nur eine Bereichsauswahl oder nur ein Filter übergeben wird. Sobald einer der Parameter **/Von:**, **/Bis:** oder **/Filter:** verwendet werden, werden die Werte in Auswertungen oder Exporten in **büro+** ignoriert.

> **Hinweis:**
> Es können nur die Suchfelder zur Eingrenzung und Sortierung verwendet werden, die auch als Sortierung in der jeweiligen Datenbank zur Verfügung stehen. Eine Liste alle Sortierungen kann mit dem Befehl **ERPAdmin -ListDatenbankFelder /INDEX** ausgegeben werden.

Beispiele für eine Bereichsauswahl:

ERPAdmin -Export c:\ERPAdmin\Adressen.bpl c:\ERPAdmin\Kunden.xls /Von:AdrNr=10000 /Bis:AdrNr=19999 /Filter:Status=‘Kunde‘
Es werden nur die Adressen aus dem Adress-Bereich **10000-19999** mit dem Status „**Kunde**“ exportiert.

> **Hinweis:**
> Werden in einer Bereichsauswahl die Zeichen **<** oder **>** verwendet, muss die Bereichsauswahl in doppelte Anführungszeichen gesetzt werden.
>
> Beispiel: **"/Von:AdrNr>10000" "/Bis:AdrNr<20000"**

Sonderfall Felder mit Timestamp
Bei der Eingrenzung von Tabellen über ein Feld mit einem **Timestamp** z.B. im **Ausgabearchiv** über das Feld „**Dat**“ muss folgende bei der Eingrenzung über das Datum beachtet werden:

Soll nach einem Tagesdatum selektiert werden und wird „**nur**“ ein Datum übergeben (anstelle eines Timestamp“ ergänzt büro+ den Datumswert wie folgt:
"/Von:Dat=06.02.2023" wird interpretiert als: **"/Von:06.02.2023 00:00:00"**

Soll nur ein einzelner Tag selektiert werden z.B. der **06.02.2023** müssen die beiden Bereich **/Von:…./Bis:…** wie folgt lauten:

"/Von:Dat=06.02.2023" **"/Bis:Dat=07.02.2023" (Folgetag 0 Uhr)**

oder mit Angabe der Uhrzeit:

"/Von:Dat=06.02.2023 00:00:00" **"/Bis:Dat=06.02.2023 23:59:59"**

13 Benutzung von Filtern STD PRO

Mit der Option **/Filter:** können in der Notation von **büro+** Filter gesetzt werden. Filter werden in verschiedenen Befehlen von **ERPAdmin** unterstützt. Diese sind z.B. „**Export**“, „**VorgangBuchen**“ oder „**VorgangDrucken**“ und noch einige mehr.

In den Filterbedingungen dürfen nur Datenbankfelder verwendet werden welche nicht berechnet werden. Mit dem Befehl **ERPAdmin -ListDatenbankfelder** kann eine Liste aller Datenbankfelder erstellt werden. Felder die mit einem Stern (*****) markiert sind, stehen für die Filter deshalb **nicht** zur Verfügung.

Bei Filtern sind folgende Operatoren erlaubt:

<	Linker Operator kleiner als der Rechte
>	Linker Operator größer als der Rechte
>=	Linker Operator größer oder gleich dem Rechten
<=	Linker Operator kleiner oder gleich dem Rechten
=	Linker und Rechter Operator gleich
<>	Linker und Rechter Operator sind ungleich
AND	Wahr, wenn Linker und rechter Ausdruck dasselbe Ergebnis liefern (Wahr oder Falsch Ausdruck)
NOT	Wahr, wenn Linker und rechter Ausdruck ein unterschiedliches logisches Ergebnis liefern (Wahr oder Falsch Ausdruck)
OR	Entweder ist der Linke oder der Rechte Ausdruck wahr (Wahr oder falsch Ausdruck)

Falls im Filterausdruck Leerzeichen vorkommen, muss der ganze Filter-Ausdruck inkl. der Angabe /Filter: in Anführungszeichen gesetzt werden. Falls Feldnamen mit Leerzeichen verwendet werden, müssen diese in eckige Klammern gesetzt werden**: "/Filter:[Meine Bezeichnung] = 'Filter Wert'"**

Datumsangaben und Zeichenketten müssen in Hochkommas gesetzt werden.

Hier einige Beispiele für Verwendung der Filter für **Datum**- oder **String**-Felder:

- **"/Filter:Dat<>'13.10.2016'"**
- **"/Filter:Dat>'30.12.1899'"**
- **"/Filter:BelegNr <> 'RE000001' OR ArtNr = NULL"**
- **"/Filter:VrtNr> ' 100 ' and VtrNr < ' 200 '"**

> **Hinweis:**
> Werden in einem Filter-Ausdruck die Zeichen **<** oder **>** verwendet, muss der gesamte Filterausdruck in doppelte Anführungszeichen gesetzt werden.

Sollen im Filter Vergleiche mit leer ('''), also ohne Inhalt OHNE Inhalt, erfolgen muss mit dem Wert **NULL** verglichen werden. **Verwenden Sie deshalb bei einem Vergleich ob ein Feld leer ist immer die folgende Notation (Siehe Hinweis im COM Kompendium Seite 93 ff.):**

- Anstelle von **"/Filter:Sel1=''"**
- Verwenden Sie: **"/Filter:Sel1=NULL"**

Wenn das Filter-Feld vom Typ **Integer** ist, darf die Wertzuweisung entgegen der Filter-Notation in **büro+** **nicht** in Hochkommas stehen:

- **/Filter:VsdArt=10**

- **/Filter:ReLand=276**

Filter für **Boolean**-Feldern sind wie folgt anzuwenden:

- **/Filter:BuchKZ=Wahr** oder **/Filter:BuchKz=1**
  "/Filter:BuchKZ<>Wahr" oder **"/Filter:BuchKz<>1"**

Tipp: Verwenden Sie bei Boolean-Feldern besser den Vergleich „**<>1**“ anstelle von „**=0**“. Filter die ein „**<>**“ enthalten müssen in Anführungszeichen gesetzt werden!

Die Punkte innerhalb der Feldvariablen müssen „**manchmal**“ weggelassen werden. Probieren Sie es einfach aus. **ERPAdmin** gibt Ihnen eine Rückmeldung ob **büro+** den Filterausdruck akzeptiert oder nicht.

Hinweis:
Felder die Sie in Filtern verwenden dürfen **nicht** zeitgleich in Wert-Zuweisungen verwendet werden!

Die Übergabe von Filtern an **ERPAdmin** überschreibt bereits vorhandene **Bereichsfilter**-Ausdrücke in Auswertungen oder Exporten. **Ausgabefilter** sind davon nicht betroffen.

Filter arbeiten immer Case-Sensitiv. Achten Sie deshalb unbedingt auf **Groß-/Kleinschreibung**.

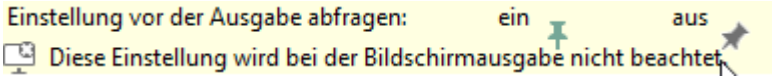
14 Drucken mit ERPAdmin PRO

Für die Ausgabe einer Liste, eines Vorgangs oder den Versand einer E-Mail aus **büro+** wird die Professional-Version von **ERPAdmin** benötigt.

Bevor ein Druck mit **ERPAdmin** angestoßen werden kann, muss die Druckausgabe in **büro+** auf dem jeweiligen Rechner auf dem **ERPAdmin** ausgeführt wird konfiguriert werden. Dazu ist es unbedingt notwendig, dass die Anmeldung auf diesem Rechner an **büro+** mit dem Benutzer erfolgt, der in der **VAR.INI** für den zum Druck verwendeten Mandanten hinterlegt wurde.

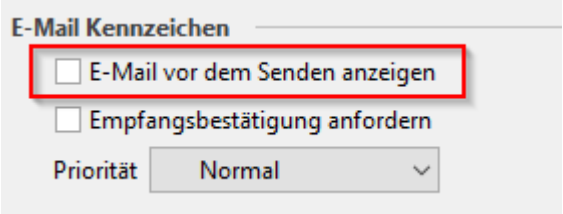
Wird der Aufruf von **ERPAdmin** über den Windows Scheduler ausgeführt, muss der verwendete Benutzer derselbe Windows-Benutzer sein mit dem auch die Einrichtung und Programmierung von **ERPAdmin** erfolgt ist.

In den Drucklayouts ist darauf zu achten, dass sämtliche Benutzer-Dialoge deaktiviert wurden. Da **ERPAdmin** die Druckausgabe über die COM-Schnittstelle aufruft, würde ein Benutzerdialog nicht angezeigt werden können. Prüfe Sie deshalb in allen Drucklayouts, dass alle Pinnadeln **deaktiviert** wurden.



Im Idealfall erstellen Sie ein eigenes Drucklayout für alle Druckausgabe mit **ERPAdmin**.

Bei E-Mail-Layouts muss das Kennzeichen „**E-Mail vor dem Senden anzeigen**“ unbedingt deaktiviert sein.



Nachfolgende **ERPAdmin** -Befehle können unter anderem eine Druckausgabe erzeugen:

- **ListeDrucken**
- **ListeDruckenDatei**
- **VorgangDrucken**
- **VorgangDruckenDatei**
- **VorgangArchivDrucken**
- **VorgangArchivDruckenDatei**
- **VorgangWandeln …. /Drucken**

Hinweis:
Wenn Sie den Druck über Regeln auslösen z.B. „**nach dem Wandeln**“ ist ebenfalls darauf zu achten, dass kein Benutzer-Dialog erscheint.

Achtung:
Jede Druckausgabe verändert das Datum „**LtzAend**“ des entsprechenden Datensatzes. Das passiert auch bei Druckausgaben direkt aus dem büro+-Client.

15 Platzhalter und variable Parameter BAS

Soll in regelmäßigen Abständen z.B. ein Export erzeugt werden und die Daten des vorherigen Exportes sollen nicht überschrieben werden, muss sich der Dateiname automatisch anpassen. Dies könnte z.B. der Fall sein, wenn man täglich einen Umsatzexport erstellen möchte um die finanzielle Entwicklung zu dokumentieren.

Weiterhin können auch Datumswerte automatisch ermittelt werden um z.B. die Daten des letzten oder aktuellen Monats zu selektieren. Dazu können bei allen **ERPAdmin**-Befehlen die folgenden Platzhalter verwendet werden welche einmalig zu Beginn des Aufrufes von **ERPAdmin** durch den entsprechenden Wert ersetzt werden.

Nachfolgende Beispiele sind für das Datum **Dienstag**, den **13.10.2015**, **10:30:23 Uhr**.

@DATETIME@	Zeitstempel mit Datum und Uhrzeit z.B.: **13.10.2015 10.30.23**	
@DATE@	Aktuelles Datum z.B.: **13.10.2015**	
@DATEREV@	Aktuelles Datum rückwärts z.B.: **2015.10.13**	
@DATEREVSHORT@	Aktuelles Datum rückwärts z.B.: **20151013**	
@TIME@	Aktuelle Zeit z.B.: **10.30.23**	(Punkte als Trennzeichen)
@TIMESTD@	Aktuelle Zeit z.B.: **10:30:23**	(Doppelpunkte als Trennzeichen)
@TIMESHORT@	Aktuelle Zeit z.B.: **103023**	
@YYYY@	Jahreszahl 4-Stellig z.B.: **2015**	
@YYYX@	Jahreszahl 4-Stellig des Vormonats z.B.: **2015**	
@YY@	Jahreszahl 2-Stellig z.B.: **15**	
@MM@	Monat 2-Stellig z.B.: **10**	
@DD@	Tag 2-Stellig: z.B.: **13**	
@h@	Stunde 2-Stellig z.B.: **10**	
@m@	Minute 2-Stellig z.B.: **30**	
@s@	Sekunde 2-Stellig z.B.: **23**	
@KW@	Aktuelle Kalenderwoche z.B. **42**	
@KWA@	Datum des ersten Tages der aktuellen Woche z.B.: **12.10.2015**	
@KWE@	Datum des letzten Tages des Monats z.B.: **18.10.2015**	
@Q1A@	**A**nfangsdatum des 1. Quartals aus dem aktuellen Jahr z.B.: **01.01.2015**	
@Q1E@	**E**nddatum des 1. Quartals aus dem aktuellen Jahr z.B.: **31.03.2015**	
@Q2A@	**A**nfangsdatum des 2. Quartals aus dem aktuellen Jahr z.B.: **01.04.2015**	
@Q2E@	**E**nddatum des 2. Quartals aus dem aktuellen Jahr z.B.: **30.06.2015**	
@Q3A@	**A**nfangsdatum des 3. Quartals aus dem aktuellen Jahr z.B.: **01.07.2015**	
@Q3E@	**E**nddatum des 3. Quartals aus dem aktuellen Jahr z.B.: **30.09.2015**	
@Q4A@	**A**nfangsdatum des 4. Quartals aus dem aktuellen Jahr z.B.: **01.10.2015**	
@Q4E@	**E**nddatum des 4. Quartals aus dem aktuellen Jahr z.B.: **31.12.2015**	
@QA-1@	Anfangsdatum des vorherigen Quartals z.B. **01.07.2015**	
@QE-1@	Enddatum des vorherigen Quartals z.B. **30.09.2015**	
@TIMESTAMP@	Zeitstempel im Format YYYY.MM.DD_h.m.s, z.B.: **2015.10.13_10.30.23**	
@TIMESTAMPSHORT@	Zeitstempel im Format YYYYMMDD_hms, z.b.: **20151013_103023**	
@VORMONAT@	Vormonat 2-Stellig z.B.: **09**	
@VORMONATNAME@	Name des Vormonats z.B.: **September**	
@MONAT@	Name des aktuellen Monats z.B.: **Oktober**	

@MM.YYYY@	Monat 2-Stellig und Jahreszahl 4-Stellig z.B.: **10.2015**
@MM.YYYY-1@	Vormonat 2-Stellig und Jahreszahl 4-Stellig z.B.: **09.2015**
@DAY@	Name des Wochentages z.B.: **Dienstag**
@PRIMO@	Datum des ersten Tages des aktuellen Monats z.B.: **01.10.2015**
@ULTIMO@	Datum des letzten Tages des aktuellen Monats z.B.: **31.10.2015**
@FILENAME@	Funktionsabhängig wird der Dateiname der Ergebnisdatei eingefügt
@COMPUTER@	Maschinenname des Computers auf dem **ERPAdmin** ausgeführt wird.
@YEARBEGIN@	**01.01.** des aktuellen Jahres z.B. **01.01.2015**
@YEAREND@	**31.12.** des aktuellen Jahres z.B. **31.12.2015**
@AUTONUM@	Zähler der autom. inkrementiert wird, wenn sich der Platzhalter in einem **ERPAdmin**-Aufruf befindet. Bei einem Datumswechsel wird der Zähler automatisch zurückgesetzt. Die Ausgabe ist immer 4-Stelling mit führenden Nullen. z.B. „0023“. In Kombination mit **/SetVarX** können so autom. Dateinamen bei aufeinanderfolgenden Aufrufen von **ERPAdmin** erzeugt werden.
@GUID@	Erzeugt eine GUID. Beispiel: **ce05eb22-0d59-4c2a-ab9c-46151add8601**

Hinweis:
Bitte achten Sie bei den Platzhaltern auf **Großschreibung**!

Beispiel @DATE@:
ERPAdmin -Export c:\ERPAdmin\Umsatz.BPL c:\ERPAdmin\Umsatz_@DATE@.xls
Der Export wird in der Datei **"c:\ERPAdmin\Umsatz_13.10.2015.xls**" gespeichert.

Beispiel @AUTONUM@:
Bei der Aufbereitung einer CSV-Datei durch den Befehl **CSVKonverter** sind manchmal viele Aufrufe von **ERPAdmin** notwendig und damit auch viele verschiedene Zwischenergebnisse möglich. Um Fehler leichter eingrenzen zu können ist es hilfreich, wenn die Ergebnisse eines jeden einzelnen **ERPAdmin**-Aufrufes existiert. Dazu ist es aber notwendig, dass jede Ergebnisdatei bestehen bleibt und einen eigenen Namen bekommt.

Ohne **@AUTONUM@** würden z.B. die Aufrufe von mehreren **ERPAdmin**-Befehlen wie folgt aussehen:

ERPAdmin -CSVKonverter Datei.csv Replace 61 . , /Datei:Datei_02.csv

ERPAdmin -CSVKonverter Datei_02.csv Calc 87 61 * 5 /Datei:Datei_03.csv

ERPAdmin -CSVKonverter Datei_03.csv Round 87 "xR,00" /Datei:Datei_04.csv

ERPAdmin -CSVKonverter Datei_04.csv Replace 44 N TL /Datei:Datei_05.csv

ERPAdmin -CSVKonverter Datei_05.csv Replace 44 Y TT /Datei:Datei_06.csv

Mit @AUTONUM@ (in Kombination mit **/SETVAR1:**) könnten die Aufrufe wie folgt aussehen. Der Vorteil hier ist, dass jeder Zeit die Reihenfolge der Befehle geändert und weitere Befehle hinzugefügt werden können, ohne dass die Dateinamen der **ERPAdmin**-Befehle angepasst werden müssen.

ERPAdmin -CSVKonverter Datei.csv Replace 61 . , /Datei:Datei_@AUTONUM@.csv /SETVAR1:Datei_@AUTONUM@.csv

ERPAdmin -CSVKonverter @SETVAR1@ Calc 87 61 * 5 /Datei:Datei_@AUTONUM@.csv /SETVAR1:Datei_@AUTONUM@.csv

ERPAdmin -CSVKonverter @SETVAR1@ Round 87 "xR,00" /Datei:Datei_@AUTONUM@.csv /SETVAR1:Datei_@AUTONUM@.csv

ERPAdmin -CSVKonverter @SETVAR1@ Replace 44 N TL /Datei:Datei_@AUTONUM@.csv /SETVAR1:Datei_@AUTONUM@.csv

ERPAdmin -CSVKonverter @SETVAR1@ Replace 44 Y TT /Datei:Datei_@AUTONUM@.csv /SETVAR1:Datei_@AUTONUM@.csv

Variables Datum

Soll ein Datum an **ERPAdmin** übergeben werden das in der Zukunft oder in der Vergangenheit liegt, ist ein Platzhalter wie folgt zu Verwendung: Z.B. den Platzhalter **@DATE+10@** um ein Datum 10 Tage in der Zukunft zu beschreiben. Liegt das zu berechnende Datum in der Vergangenheit, muss ein Minus verwendet werden. Leerzeichen dürfen dabei NICHT zwischen den **@-Zeichen** vorkommen.

Auf diese Weise können Werte basierend auf **@MONTH@, @YYYY@, @MM@, @DD@, @PRIMO@, @ULTIMO@**, **@KW@, @KWA@, @KWE@, @Q1A@, @Q1E@ und @h@** ermittelt werden.

Beispiel für @DATE+X@

Das Datum soll immer 10 Tage in der Zukunft liegen (X=10). Der Platzhalter heißt dann: **@DATE+10@**
Analog dazu wird das Datum 10 Tage in der Vergangenheit bestimmt: **@DATE-10@**

Auf die gleiche Weise werden Tag, Monat, Jahre, KW, Primo und Ultimo berechnet.

Beispiel für @DD-X@

Es soll der Tag gestern ermittelt werden ausgehend vom aktuellen Datum: **@DD-1@**
Es soll der Tag morgen ermittelt werden ausgehend vom aktuellen Datum: **@DD+1@**

Beispiel für @MM-X@

Es soll der Vormonat ermittelt werden ausgehend vom aktuellen Monat: **@MM-1@**
Es soll der Folgemonat ermittelt werden ausgehend vom aktuellen Monat: **@MM+1@**

Beispiel für @YYYY-X@

Es soll das Vorjahr ermittelt werden ausgehend vom aktuellen Datum: **@YYYY-1@**
Es soll das nächste Jahr ermittelt werden ausgehend vom aktuellen Datum: **@YYYY+1@**

Beispiel für @KW-X@

Es soll die vorherige Kalenderwoche ermittelt werden ausgehend vom aktuellen Datum: **@KW-1@**
Es soll die Kalenderwoche ermittelt werden ausgehend vom aktuellen Datum: **@KW+1@**

Beispiel für @KWA-X@
Es soll der Montag der vorherige KW ermittelt werden ausgehend vom aktuellen Datum: **@KWA-1@**
Es soll der Montag der folgenden KW ermittelt werden ausgehend vom aktuellen Datum: **@KWA+1@**

Beispiel für @KWE-X@
Es soll der Sonntag der vorherige KW ermittelt werden ausgehend vom aktuellen Datum: **@KWE-1@**
Es soll der Sonntag der folgenden KW ermittelt werden ausgehend vom aktuellen Datum: **@KWE+1@**

Beispiel für @MONTH+X@
Es soll der Name des Folgemonats ermittelt werden ausgehend vom aktuellen Datum: **@MONTH+1@**
Es soll der Name des vorherigen Monats ermittelt werden ausgehend vom aktuellen Datum: **@MONTH-1@**

Beispiel für @h+X@
Es soll der Wert einer Folgestunde ermittelt werden ausgehend vom aktuellen Zeitpunkt: **@h+1@**
Es soll der Wert der vorherigen Stunde ermittelt werden ausgehend vom aktuellen Zeitpunkt: **@h-1@**

Beispiel für @Q1A-X@
Es soll der Monatsanfang des 1. Quartals aus dem Vorjahr ermittelt werden ausgehend vom aktuellen Datum: **@Q1A-1@**

Analog dazu funktioniert der Platzhalten für das Ende-Datums des 1. Quartals aus dem Vorjahr: **@Q1E-1@.**

Die Platzhalter für das 2., 3. und 4. Quartal (**@Q2A-X@, @Q2E-X@,**... etc.) funktionieren auf gleiche Weise.

Der Monatserste Tag: @PRIMO@
Um den ersten Tag des Monats zu ermitteln gibt es den Platzhalten **@PRIMO@.** Für vorherige oder nachfolgende Monate kann der Primo wie folgt ermittelt werde:

Beispiel für @PRIMO-X@
Es soll der Primo des Vormonats ermittelt werden ausgehend vom aktuellen Datum: **@PRIMO-1@**
Es soll der Primo des Folgemonats ermittelt werden ausgehend vom aktuellen Datum: **@PRIMO+1@**

Das Monatsende: @ULTIMO@
Um das Monatsende zu ermitteln gibt es den Platzhalten **@ULTIMO@.** Für vorherige oder nachfolgende Monate kann der Ultimo wie folgt ermittelt werde:

Beispiel für @ULTIMO-X@
Es soll der Ultimo des Vormonats ermittelt werden ausgehend vom aktuellen Datum: **@ULTIMO-1@**
Es soll der Ultimo des Folgemonats ermittelt werden ausgehend vom aktuellen Datum: **@ULTIMO+1@**

Beispiel für @YEARBEGIN-X@
Es soll der Vorjahresanfang (01.01.) ermittelt werden ausgehend vom aktuellen Datum: **@YEARBEGIN-1@**
Es soll der Jahresanfang des Folgejahres ermittelt werden ausgehend vom aktuellen Datum: **@YEARBEGIN+1@**

Beispiel für @YEAREND-X@

Es soll der Vorjahresende (31.12.) ermittelt werden ausgehend vom aktuellen Datum: **@YEAREND-1@**

Es soll das Jahresende des Folgejahres ermittelt werden ausgehend vom aktuellen Datum:

@YEAREND+1@

16 Doubletten-Prüfung

ERPAdmin bieten bei verschiedenen Befehlen eine Doubletten-Prüfung an. Dazu muss der Name einer Doubletten-Datei als Parameter angegeben werden in der **ERPAdmin** Einträge zur Prüfung speichert. Diese Datei ist vom Typ Text-ANSI. Bitte beachten Sie dazu die jeweilige Dokumentation des **ERPAdmin** Befehls.

Wird eine Doubletten-Datei als Parameter in einem Befehl angegeben, muss immer der komplette Pfad inkl. Dateiname angegeben werden. Bei jeder Überprüfung prüft **ERPAdmin** ob ein Wert in der Doubletten-Datei vorhanden ist. Ist das der Fall handelt es sich um eine Doublette. Anstelle die Datei weiter zu verarbeiten kann diese von **ERPAdmin** in ein Verzeichnis verschoben oder die weitere Verarbeitung übersprungen werden.

Für das Verschieben in ein Verzeichnis Dieses muss in der **VAR.INI** die Variable **C_BADDIR** vorbelegt worden sein oder es wird beim Aufruf von **ERPAdmin** der Parameter **/MoveToBad:<Verzeichnis>** mit übergeben.

Im Laufe der Zeit wachsen die Doubletten-Dateien mit der Anzahl der Einträge an was zu einer Verschlechterung der Performance führen kann. Um dies zu vermeiden gibt es zwei Möglichkeiten:

- Mit der Variablen **C_DoubletFilesMax** kann ein Wert in der **VAR.INI** vorgegeben werden bei dessen Überschreitung eine Email an die unter **C_ERROR_TO** hinterlegte E-Mail versendet wird mit einem entsprechenden Hinweis. Die entsprechende Doubletten-Datei muss dann von Hand gekürzt werden.
- Mit einer zweiten Variable **C_DoubletFilesCutTo** kann eine automatische Kürzung der Doubletten-Datei durch **ERPAdmin** vorgenommen werden. Wird der Genzwert von **C_DoubletFilesMax** Einträgen überschritten wird die Doubletten-Datei auf die letzten **C_DoubletFilesCutTo** Einträge gekürzt. Eine E-Mail wird in diesem Fall nicht versendet.

Folgende **ERPAdmin** Befehle bieten eine Unterstützung bei einer Doubletten-Prüfung an:

- Import STD
- FTPGet STD
- FTPPut STD
- Import STD
- SFTPPut PRO
- SFTPGet PRO
- VorgangDruckenDatei PRO
- VorgangArchivDruckenDatei PRO
- VorgangImport PRO

Beispiele:

ERPAdmin -FTPGet *.csv *.csv .\Bestellungen\ c:\ERPAdmin\Bestellungen\ Doubletten.log
Prüft vor dem **FTP-Download** ob die zu übertragende Datei in der Doubletten Datei **Doubletten.log** enthalten ist und überträgt diese nur wenn dies nicht der Fall ist.

ERPAdmin -Import Bestellung.bpl *.csv Doubletten.log /MoveToBad:./Doubletten/
Prüft vor dem **Import** ob die zu importierende Datei in der Doubletten Datei **Doubletten.log** enthalten ist

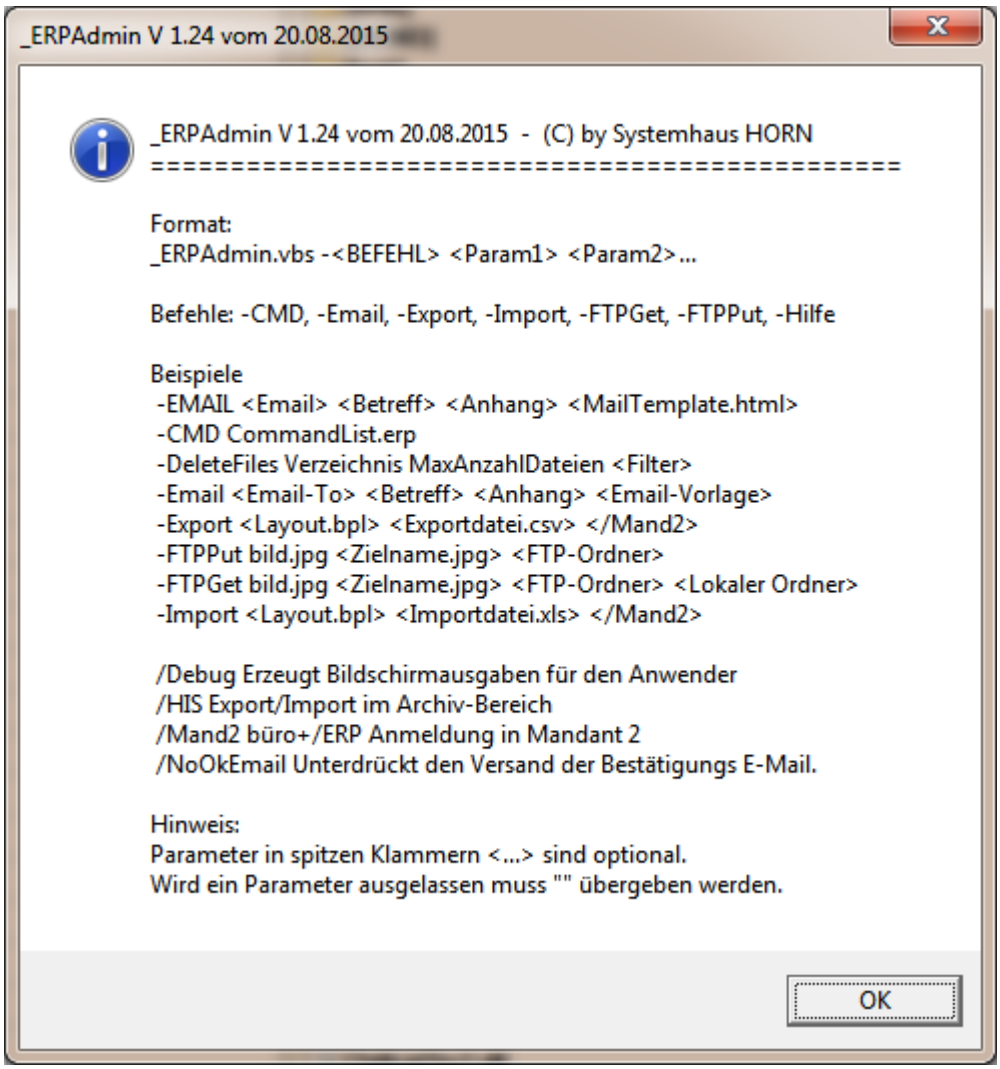
und importiert diese nur wenn dies nicht der Fall ist. Handelt es sich um eine Doublette, wird die Datei in das Verzeichnis **./Doubletten/** verschoben zur späteren Nachbearbeitung.

17 Die Option „/HELP“ BAS

ERPAdmin /Help

Der Aufruf von **ERPAdmin** ohne Angabe eines Befehls gibt die nachfolgende Hilfe-Seite aus.

Das Hilfe-Fenster enthält nicht alle Befehle und auch nicht alle Parameter, sondern nur eine Auswahl.



[Hilfe-Fenster mit einer Kurzübersicht einiger Befehle]

Zusätzlich kann zu jedem **ERPAdmin** Befehl eine individuelle Kurz-Hilfe zur Anzeige gebracht werden. Die Aufrufnotation lautet wie folgt:

ERPAdmin -<Befehl> /HELP

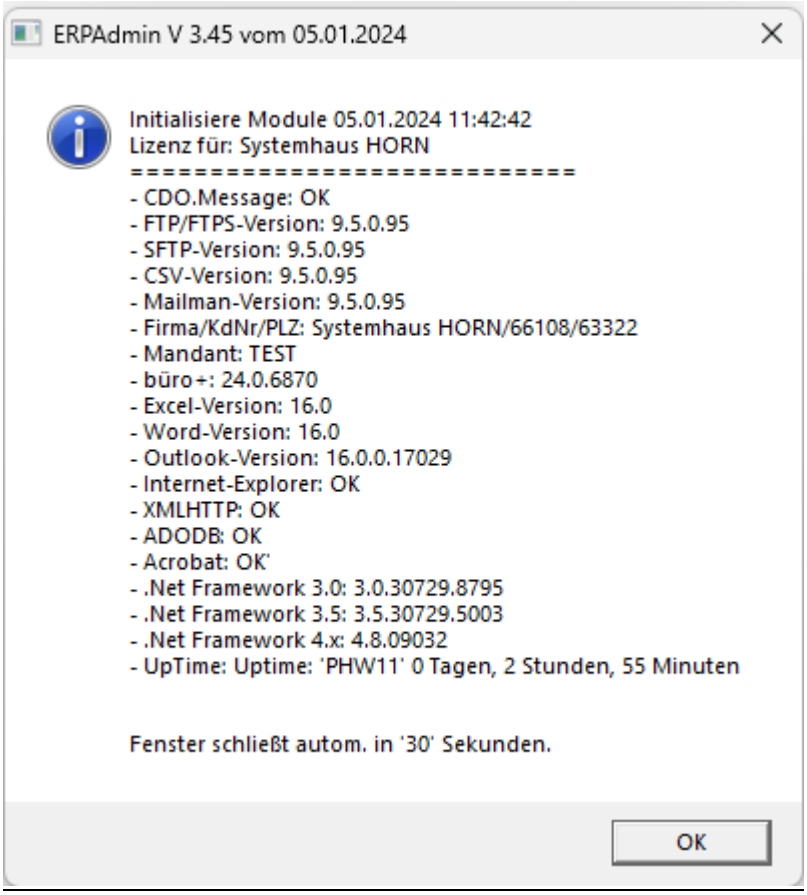
Beispiel:

ERPAdmin -DeleteData /HELP

18 Der Befehl „-Init“ BAS

ERPAdmin -Init

Um zu überprüfen, ob alle benötigten Systemkomponenten vorhanden und ansprechbar sind, kann eine Überprüfung durchgeführt werden. Als Ergebnis erhält der Anwender eine Rückmeldung, ob die entsprechenden Komponenten angesprochen werden können und liefert, falls vorhanden, die Versionsnummer zurück.



[Rückmeldung an Anwender]

Beim Aufruf werden folgende Objekte auf Vorhandensein geprüft:

- Microsoft Collaboration Data Objects (CDO)
- FTP-Komponente auf dem PC
- SFTP-Komponente auf dem PC
- CSV-Komponente auf dem PC
- MailMan-Komponente auf dem PC
- **büro+** Version
- Lizenzinformationen aus **büro+**
- Mandant in dem die Anmeldung erfolgte
- MS-Excel Version auf dem PC
- MS-Word Version auf dem PC
- MS-Outlook Version auf dem PC
- Internet-Explorer
- Microsoft XMLHTTP

- Microsoft ADOBD
- Acrobat Reader Komponente auf diesem PC
- .Net Framework 3.0
- .Net Framework 3.5
- .Net Framework 4.x
- UpTime des aktuellen PCs

Hinweis:
Befindet sich auf dem Rechner auf dem **ERPAdmin** ausgeführt werden soll kein Microsoft-Word, -Excel oder -Outlook, können die auftretenden Fehlermeldungen unterdrückt werden indem in der **VAR.INI** die folgenden Einträge hinzugefügt werden:

C_NoOutlookCheck	**= TRUE**
C_NoWordCheck	**= TRUE**
C_NoExcelCheck	**= TRUE**

19 Der Befehl „-AdressenUstIDCheck“ PRO

ERPAdmin -AdressenUstIDCheck <Selektion1> <Selektion2> /Von:AdrNr=<AdrNr> /Bis:AdrNr=<AdrNr> /Filter:<Filterbedingung> /NoLockError : /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /ReAns /ChartSet:<Zeichensatz>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Die Umsatzsteuer-ID eines Kunden muss regelmäßig auf Gültigkeit geprüft werden. Dabei unterstützt Sie **ERPAdmin.** Es können alle Adressen im Bereich **/Von:AdrNr=<AdrNr>** bis **/Bis:AdrNr=<AdrNr>** überprüft werden, bei denen eine Umsatzsteuer-ID hinterlegt ist. Es werden nur Umsatzsteuer-IDs geprüft, die NICHT mit „**DE**“ beginnen und deren Umsatzsteuerkategorie „**3 Ausland-EU**“ ist.

Wie auch bei „**ERPAdmin -VorgangUstIdCheck**“ wird auch die Richtigkeit der **Lieferanschrift** (Name 2, Straße, PLZ, Ort) überprüft. Als Ergebnis für die Umsatzsteuer-ID-Prüfung wird entweder ein „**OK**“ oder ein Fehlercode zurückgegeben und im Feld **<Selektion1>** gespeichert.

/Force **(Optional)**
ERPAdmin führt nur dann eine Ust-ID-Prüfung für eine Adresse durch, wenn das Feld **Selektion1** leer ist. Um erneut eine Abfrage auszuführen löschen Sie den Inhalt des Feldes **<Selektion1>** oder verwenden Sie die Option **/Force.** Damit wird unabhängig vom Inhalt der **Selektion 1** eine Prüfung durchgeführt.

Allgemeines
Diese Funktionalität von **ERPAdmin** setzt den Web-Server des „**Bundesamtes für Steuer**“ (BFS) voraus bei dem diese Daten online abgefragt werden. Weitere Informationen finden Sie unter https://evatr.bff-online.de/eVatR/

Ergebnis
Jede Abfrage am BFR-Server wird mit einer XML-Datei beantwortet. Diese XML-Dateien werden dann in den Dokumenten der Adressen zur Dokumentation hinterlegt und im History-Verzeichnis von **ERPAdmin** (wenn in der **VAR.INI** angegeben) für eine bestimmte Zeit archiviert

> **Hinweis:**
> Die Prüfung der Umsatzsteuer-ID ist nur für Unternehmen aus Deutschland mit deutsche Umsatzsteuer-ID möglich und nur für Kunden die sich aus deutscher Sicht im EU-Ausland befinden.

Sperren einer Adresse bei ungültiger Ust-ID
Über eine Regel kann eine Adresse nach einer Prüfung autom. gesperrt werden. Sollen bei nachfolgenden Prüfungen diese gesperrten Adressen nicht erneut geprüft werden, kann der folgende Filter übergeben werden: **"/Filter:GspGrp=0 or GspGrp=NULL"**

<Selektion1>
Name des Selektionsfeldes in der Adresse für die Rückmeldung der Prüfung. Das Feld muss **min. 50 Zeichen** groß sein und vom Typ „**String**“.

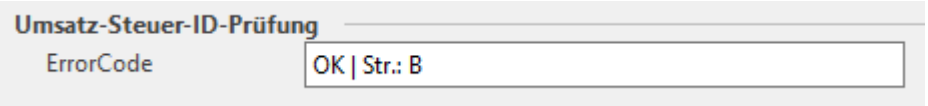
Ist das Feld **min. 400 Zeichen** groß, wird zusätzlich zum Fehlercode die komplette Fehler-Meldungen im Selektionsfeld gespeichert.

Ist die Umsatzsteuer-ID korrekt wird ein „**OK**“ zurückgegeben.

Neben der Umsatzsteuer-ID werden auch die Anschriften-Felder Firma/Name (**Li.Na2**), Straße (**Li.Str**), PLZ (**Li.PLZ**) und Ort (**LI.Ort**) geprüft. Kam es hier zu einer Abweichung, wird dieses wie folgt zurückgemeldet:

- A = stimmt überein
- B = stimmt nicht überein
- C = nicht angefragt
- D = vom EU-Mitgliedsstaat nicht mitgeteilt

Ist die Umsatzsteuer-ID korrekt aber die Straße stimmt nicht überein, sieht die Rückmeldung z.B. wie folgt aus:



Trat ein Fehler bei der Prüfung auf wird ein Fehlercode zurückgegeben**. Eine Liste aller Fehlercode finden Sie weiter unten beim Befehl ERPAdmin -VorgangUstIdCheck**.

<Selektion2>
Name des Selektionsfeldes in der Adresse für den Timestamp der Abfrage. Hier wird ein Timestamp im Format **tt.mm.jjjj hh:mm:ss** eingetragen. Die Uhrzeit des Timestamp ist Teil der Rückmeldung des BFS-Servers.

/Von:AdrNr=<AdrNr> **(optional)**
Selektiert einen Adress-Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt.
Beispiel: /Von:AdrNr=10000

/Bis:AdrNr=<AdrNr> **(optional)**
Hier wird das Ende des Adress-Bereiches vorbelegt.
Beispiel: /Bis:AdrNr=29999

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

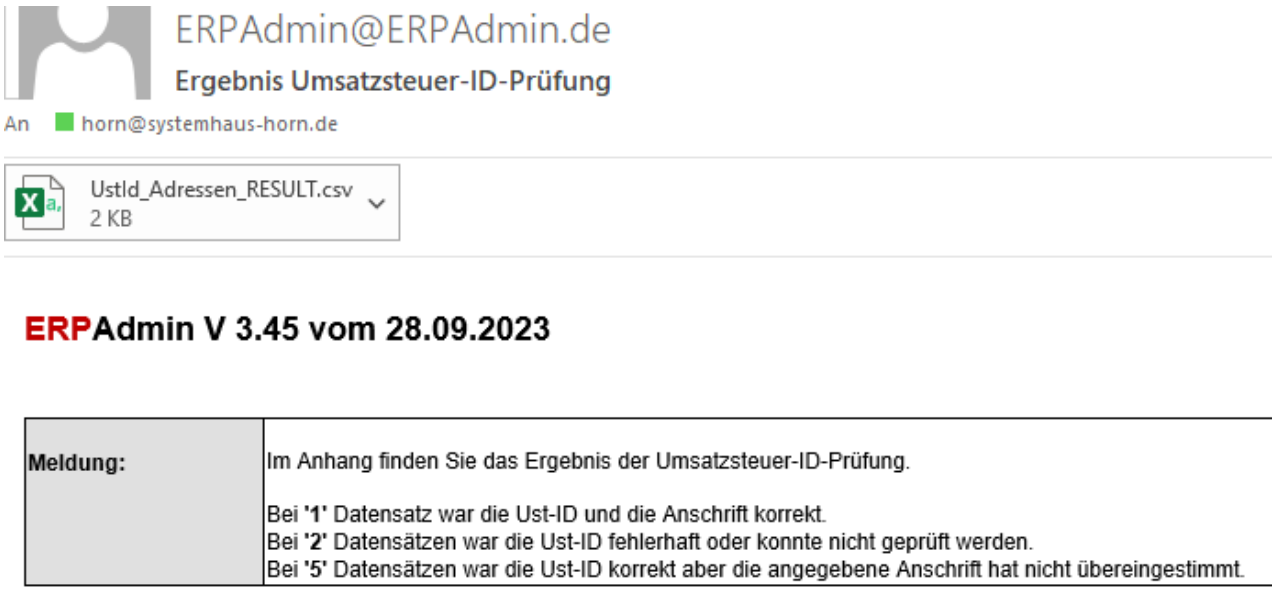
/NoLockError **(optional)**
Kann in eine Adresse nicht geschrieben werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei „**UstID_Adressen_Result.csv**“ mit allen Ergebnissen der Umsatzsteuer-ID-Prüfung sendet. Wurden keine Datei erzeugt wird keine E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „Kapitel Platzhalter und variable Parameter“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Die E-Mail kann dann wie folgt aussehen:



/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/Force **(optional)**
Eine Prüfung wird nur dann für eine Adressen ausgeführt, wenn der Inhalt des Feldes **<Selektion1>** leer ist. Durch Angabe dieser Option kann dennoch eine Prüfung erzwungen werden. Bitte beachten Sie, dass in den Dokumenten der Adresse durch wiederholten Aufruf mit dieser Option mehrere XML-Ergebnisdateien hinterlegt sein können. Diese sind durch den Timestamp unterscheidbar.

/ReAns **(optional)**
Soll anstelle der Lieferanschrift die **Rechnungsanschrift** geprüft werden muss die Option **/ReAns** übergeben werden.

/ChartSet: **<Zeichensatz>**
Wurde die Option **/Email:** verwendet, wird eine Ergebnisdatei per E-Mail verwendet. Das Format dieser CSV-Datei ist **ANSI,** damit die Datei mit Excel geöffnet werden kann. Soll diese CSV-Datei als **UTF-8** versendet werden, muss die Option **/CharSet:UTF-8** übergeben werden.

Beispiele:

ERPAdmin -AdressenUstIDCheck Sel1 Sel2 /Von:AdrNr=10000 /Bis:AdrNr=29999
Es werden alle Adressen mit den Adressnummern **10000** bis **29999** geprüft die eine Umsatzsteuer-ID enthalten <> „DE“ und wenn das Selektionsfeld **SEL1** leer ist.

ERPAdmin -AdressenUstIDCheck Sel1 Sel2 /Von:AdrNr=10000 /Bis:AdrNr=29999 /Filter:LiLand<>40
Es werden alle Adressen mit den Adressnummern **10000** bis **29999** geprüft die eine Umsatzsteuer-ID enthalten <> „DE“ und wenn das Selektionsfeld **SEL1** leer ist und wenn das Land der Lieferanschrift nicht Österreich ist.

20 Der Befehl „-ANSI2UNICODE“ STD

ERPAdmin -ANSI2UNICODE <Dateiname> /NoFileNotExistError

Dieser Befehl wandelt eine ANSI-Datei in das Dateiformat UNICODE um und speichert diese unter gleichem Dateinamen ab.

<Dateiname>
Dateiname inkl. Pfadangabe im ANSI-Format welche in ein UNICODE-Format umgespeichert werden soll

/NoFileNotExistError **(optional)**
Ist keine Datei vorhanden und soll in diesem Fall keine Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

Beispiel:
ERPAdmin -ANSI2UNICODE c:\ERPAdmin\Artikel.CSV

21 Der Befehl „-ANSI2UTF8“ STD

ERPAdmin -ANSI2UTF8 <Dateiname> /NoFileNotExistError

Dieser Befehl wandelt die ANSI-Datei **Dateiname** in das Dateiformat UTF8 um und speichert diese unter gleichem Dateinamen ab.

/NoFileNotExistError **(optional)**
Ist keine Datei vorhanden und soll in diesem Fall keine Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

Beispiel:
ERPAdmin -ANSI2UTF8 c:\ERPAdmin\Artikel.CSV

22 Der Befehl „-APPENDANSIFILE“ STD

ERPAdmin -APPENDANSIFILE <Ziel-Datei> <Quell-Datei1> <Quell-Datei2> /NoFileNotExistError

Dieser Befehl verknüpft zwei oder mehrere ANSI-Text-Dateien zu einer neuen ANSI-Datei und speichert das Ergebnis in einer neuen ANSI-Datei ab. Es werden nur Text-Dateien im ANSI-Format unterstützt.

<Ziel-Datei>
Gibt den Dateinamen inkl. Pfadangabe an, unter dem das Ergebnis der Verknüpfungen gespeichert werden soll.

<Quell-Datei1>
Gibt den Dateinamen inkl. Pfadangabe der ersten zu verknüpfenden Datei an. Als **Quell-Datei1** können auch mehrere Dateien mit einem Wildcard (*) ausgewählt werden. Es ist nur die Verwendung von einem Wildcard (*) erlaubt. Ausdrücke wie **Datei*.*** sind nicht zulässig.

<Quell-Datei2>
Gibt den Dateinamen inkl. Pfadangabe der zweiten zu verknüpfenden Datei an. Wurde in **Quell-Datei1** ein Wildcard (*) verwendet bleibt dieser Parameter leer. Die Angabe von **Quell-Datei2** ist optional.

/NoFileNotExistError (optional)
Ist keine **Quell-Datei1** oder **Quell-Datei2** vorhanden und soll in diesem Fall **keine** Fehlermeldung erzeugt werden, kann die Option **/NoFileNotExistError** verwendet werden.

Beispiel

ERPAdmin -APPENDANSIFILE c:\ERPAdmin\Auftragsliste.csv Auftrag12345.csv
Hängt die Datei **Auftrag12345.csv** an die Datei **c:\ERPAdmin\Auftragsliste.csv** an.

ERPAdmin -APPENDANSIFILE c:\ERPAdmin\Auftragsliste.csv Auftrag12345.csv Auftrag45678.csv
Hängt die Dateien **Auftrag12345.csv** und **Auftrag45678.csv** an die Datei **c:\ERPAdmin\Auftragsliste.csv** an.

ERPAdmin -APPENDANSIFILE c:\ERPAdmin\Auftragsliste.csv Auftrag*.csv
Hängt die Dateien **Auftrag*.csv** an die Datei **c:\ERPAdmin\Auftragsliste.csv** an.

ERPAdmin -APPENDANSIFILE c:\ERPAdmin\Auftragsliste.csv Auftrag*.csv Auftrag*.txt
Hängt die Dateien **Auftrag*.csv** und **Auftrag*.txt** an die Datei **c:\ERPAdmin\Auftragsliste.csv** an.

/MoveToHis, /MoveToBad, /Delete
Die unter **Quell-Datei1** und **Quell-Datei2** angegebenen Dateien können nach dem Anhängen gelöscht oder in das History-Verzeichnis verschoben werden.

Zur Verwendung der **Optionen** lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“.

23 Der Befehl „-ArtikelBildLinkCheck“ PRO

ERPAdmin -ArtikelBildLinkCheck <BildNr> /Datei:<CSV-Dateiname> /Von:<Bereich von> /Bis:<Bereich Bis> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /Header /Check /Delete

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Artikel-Bilder können in **büro+** auf drei verschiedene Arten hinterlegt werden. Als Datei direkt im Artikel, als Verlinkung auf die Bild-Datenbank oder als Verlinkung auf eine Datei die auf dem PC oder im Netzwerk liegt. Wird letztere von einem Anwender gelöscht kommt es zu einer Fehlermeldung bei Zugriff auf diese Bild-Datei innerhalb von **büro+**. Wurden mehrere dieser verlinkten Dateien gelöscht muss der Benutzer diese umständlich von Hand identifizieren und manuell Löschen um den Fehler zu bereinigen. Eine Datenkonsistenzprüfung hilft hier nicht.

ERPAdmin überprüft in der Artikeldatenbank ob die verlinkten Dateien unter dem hinterlegten Pfad noch existieren und löscht diese ggf. (Befehl /**DELETE**) aus dem Artikel falls die Datei nicht mehr vorhanden ist.

<BildNr>
Je nach Ausbaustufe können in **büro+** mehrere Bilder in einem Artikel hinterlegt werden. **BildNr** gibt dabei an welche Bild-Verlinkung überprüft werden soll. Für das erste Bild kann wahlweise eine 0 oder 1 angegeben werden. Bei allen weiteren Bildern orientziert sich **ERPAdmin** an der Nummer des Variablennamens.

/Datei:<CSV-Ergebnisdatei> **(optional)**
Werden fehlerhafte Verlinkungen gefunden, liefert **ERPAdmin** diese in Form einer CSV-Datei zurück. Hier kann ein Name übergeben werden. Wird kein Dateiname übergeben verwendet **ERPAdmin** den Dateinamen „Bild_x_LinkFehler.csv“ wobei x für die Bildnummer steht.

/Von:<Bereich von> **(optional)**
Die Option ist optional. Wird hier nichts angegeben werden alle Artikel überprüft. Soll nur ein bestimmter Teil der Artikel überprüft werden kann hier eine Einschränkung vorgenommen werden.

Beispiel: **/Von:AdrNr=10000**

/Bis:<Bereich bis> **(optional)**
Gibt das Ende der Eingrenzung an. Wird kein Wert angegeben wird der Wert **Von** verwendet.

Beispiel: **/Bis:ArtNr=19999**

/CHECK / DELETE **(optional)**
Für die Link-Prüfung gibt es zwei Befehle.

- **/CHECK**
- **/DELETE**

Mit dem Befehl /**CHECK** wird nur eine Überprüfung durchgeführt auf fehlerhafte Verlinkungen. Der Benutzer bekommt eine Rückmeldung ob Fehler vorhanden sind und um welche Artikel-Bilder es sich handelt. Es werden keine Bild-Verlinkung gelöscht.

Erst mit dem Befehl /**DELETE** werden fehlerhafte Verlinkungen aus dem Artikel gelöscht.

Wird kein **Befehl** angegeben wird /**CHECK** als Befehl verwendet.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **<CSV-Ergebnis>** mit den fehlerhaften Links sendet. Wurden kein Fehler gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „Kapitel Platzhalter und variable Parameter“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/HEADER **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit den Spaltenbezeichnungen.

Hinweis:
Führen Sie immer zuerst den Befehl **/CHECK** aus und überprüfen Sie ob die Ergebnisse korrekt sind. Führen erst dann den Befehl **/DELETE** aus.

Beispiele:

ERPAdmin -ArtikelBildLinkCheck 0 /CHECK /Email:Demon@ERPAdmin.de
Prüft für alle Artikel ob die verlinkte Bild-0-Datei existiert und sendet eine CSV-Datei mit den fehlerhaften Verlinkungen an die E-Mail. Fehlerhafte Links werden nicht entfernt (**/CHECK**).

ERPAdmin -ArtikelBildLinkCheck 0 /DELETE /Email: Demon@ERPAdmin.de
Prüft für alle Artikel ob die verlinkte Bild-0-Datei existiert und sendet eine CSV-Datei mit den fehlerhaften Verlinkungen an die E-Mail **Demon@ERPAdmin.de**. Alle fehlerhaften Links werden entfernt (**/DELETE**).

ERPAdmin -ArtikelBildLinkCheck 2 /VON:ArtNr=10000 /Bis:19999 /DELETE /Email:Demon@ERPAdmin.de
Prüft für die Artikelnummer 10000 bis 19999 ob die verlinkte Bild-2-Datei existiert und sendet die CSV-Datei mit den fehlerhaften Verlinkungen an die E-Mail **Demon@ERPAdmin.de**. Alle fehlerhaften Links werden entfernt (**/DELETE**).

ERPAdmin -ArtikelBildLinkCheck 3 Bild3Fehler.csv /HEADER /DELETE /Email:Demon@ERPAdmin.de
Prüft für alle Artikel ob die verlinkte Bild-3-Datei existiert und sendet die CSV-Datei **Bild3Fehler.csv** mit Kopfzeile (**/HEADER**) mit den fehlerhaften Verlinkungen an die E-Mail **Demon@ERPAdmin.de**. Alle fehlerhaften Links werden entfernt (**/DELETE**).

24 Der Befehl „-Artikelweglegen“ PRO

ERPAdmin -ArtikelWeglegen <BelegNrVon> <BelegNrBis> /Filter:<Filterbedingung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin bucht vorhandene Ware auf ein Kundenlager (KDxxxxx) um und reserviert damit einen vorhandenen Bestand für diese Kunden. Dabei werden der Reihe nach die Vorgangspositionen der Vorgänge **BelegNrVon** bis **BelegNrBis** abgearbeitet. Es werden nur Vorgangspositionen berücksichtigt, deren angegebene Mengen vollständig im Lager verfügbar sind. Teilmengen einer Vorgangsposition werden nicht umgebucht. Diese Funktion wird nur auf Standard-Artikel angewendet welche weder Chargen noch Seriennummern besitzen. Stücklisten-Positionen werden ebenfalls nicht umgebucht.

Unter den folgenden Bedingungen kann der Lagerbestand **nicht** in ein Kundenlager umgebucht werden:

- Bei der Vorgangsposition ist das Kennzeichen **AORGPosKz=Ja** gesetzt
- Es handelt sich um einen **Artikel mit Chargen**
- Es handelt sich um einen **Artikel mit Seriennummern**
- Das hinterlegte **AUS-Lager ist kein Stammlager**
- Es ist **keine Artikelnummer vorhanden**
- Die angegebene **Menge=0**
- Der Artikel wurde bereits durch **büro+** zurückgelegt (**ArtZurueckKz=Ja**)
- Der **verfügbare Lagerbestand ist kleiner als die benötigte Menge**
- **Lagerbestand wird mit Auftragsnummer** geführt für diese Artikel
- Ein Kundenlager kann nicht für diesen Artikel angelegt werden
- Es handelt sich um eine **Stücklisten-Position**

<BelegNrVon> <BelegNrBis>
Grenzt den Bereich der Vorgänge ein bei denen die Artikel-Bestände auf ein Kundenlager umgebucht werden sollen.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

> **Hinweis:**
> Die Umbuchung in das Kundenlager muss bei Bedarf vom Benutzer manuell wieder Rückgängig gemacht werden. Dies kann NICHT in dem Vorgang selber ausgeführt werden. Wird ein Vorgang mit umgebuchten Vorgangspositionen gelöscht, bleibt der Bestand in dem Kundenlager bestehen.

Beispiel:

ERPAdmin -ArtikelWeglegen BK1800000 BK1899999

Bucht den Bestand aller Vorgangspositionen, die die obigen Einschränkungen erfüllen, der Vorgänge **BK1800000** bis **BK1899999** in die entsprechenden Kundenlager (**KDxxxxx**) um.

> **Hinweis**:
> Dieser Befehl darf nicht für Lager verwendet werden welche das Kennzeichen „Lagerplatzverwaltung (Chaotische Lagerhaltung)“ gesetzt haben.

25 Der Befehl „-AusgabeverzeichnisSafeToFile“ PRO

ERPAdmin -AusgabeverzeichnisSafeToFile <Verzeichnis> <Dateiname> /Von:<Bereich von> /Bis:<Bereich Bis> /Filter:<Filterbedingung> /His /Delete

oder

ERPAdmin -AVSafeToFile <Verzeichnis> <Dateiname> /Von:<Bereich von> /Bis: /Filter:<Filterbedingung> /HIS /DELETE

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin kennt für diesen Befehl zwei Schreibweisen: Eine lange Schreibweise **„AusgabeverzeichnisSafeToFile“** und eine kurze Schreibweise **„AVSafeToFile“.** Wir verwenden zur Vereinfachung nur die kurze Schreibweise.

Mit diesem Befehl können Druck-, Fax- und E-Mail-Daten aus dem Ausgabeverzeichnis oder dem Ausgabeverzeichnis-Archiv in Dateien gespeichert werden. E-Mails werden dabei im Dateiformat „Microsoft Compound Document“ ausgegeben was sich leider nicht zur Anzeige eignet.

<Verzeichnis>
Angabe des Verzeichnisses in dem die Dateien abgelegt werden sollen.

<Dateinamen>
Gibt den Aufbau der Dateinamen unter dem die Dateien gespeichert werden sollen an. Der Dateiname wird hierbei aus Platzhaltern zusammengestellt. Folgende Platzhalter stehen zur Verfügung:

- **@ID@** Eindeutige Nummer aus des Datenbank Ausgabeverzeichnis oder Archiv
- **@AdrNr@** Adressnummer des Datensatzes soweit vorhanden
- **@BelegNr@** Belegnummer des Datensatzes soweit vorhanden
- **@BelegArt@** Belegart-Bezeichnung soweit vorhanden
- **@Bez@** Bezeichnung des Datensatzes soweit vorhanden
- **@Text@** Beschreibungstext des Datensatzes soweit vorhanden
- **@Date@** Datum des Datensatzes soweit vorhanden
- **@DateTime@** Datum und Uhrzeit des Datensatzes soweit vorhanden
- **@DIRAdrNr@** Erzeugt pro Adressnummer ein Unterverzeichnis. Dieser Platzhalter darf nur als erster Eintrag verwendet werden da er sonst zu einem Fehler führt.

Wird kein Dateiname angegeben verwendet **ERPAdmin** den Aufbau „**@ID@ - @BEZ@**“.

Hinweis:
Es sollte immer die **ID** im Dateinamen mit angegeben werden damit bei Namens-Doppelungen eine Datei nicht überschieben wird!

Die Endung (**.ll**) und verweist auf das Datei-Format List&Label. Zur Ansicht dieser Dateien wird ein List&Label Viewer benötigt welches kostenfrei von der Webseite https://www.combit.net bei Combit geladen werden kann.

/Von:<Bereich von> und **/Bis:<Bereich bis>**
Über **/Von:** kann eine Einschränkung auf die Tabelle gelegt werden wie z.B. **/Von:Dat=01.01.2016**. Soll ein Bereich selektiert werden muss auch ein Wert **/Bis:** übergeben werden z.B. **/Bis:Dat=31.12.2016**

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/His **(optional)**
Wird diese Option übergeben wird auf das Archiv des Ausgabeverzeichnisses zugegriffen.

/Delete **(optional)**
Löscht nach dem Export den Datensatz aus dem Ausgabeverzeichnis oder dem Archiv.
Achtung: Es wird keine Überprüfung durchgeführt ob der Export der Datensatze erfolgreich war. Deshalb sollte zuerst ein Aufruf ohne **/DELETE** erfolgen und das Ergebnis genau geprüft werden.

Beispiele:

ERPAdmin -AusgabeVerzeichnisSafeToFile "c:\ERPAdmin\Ausgabe\" "@ID@ - @BEZ@" /Von:DAT=01.01.2010 /Bis:DAT=31.12.2010 /HIS
Schreibt alle Dateien aus dem Ausgabeverzeichnis-Archiv (**/HIS**) aus dem Zeitraum 01.01.2010 bis 31.12.2010 in das Verzeichnis **c:\ERPAdmin\Ausgabe\.** Die Dokumente bekommen als Dateinamen die ID und die Bezeichnung.

ERPAdmin -AusgabeVerzeichnisSafeToFile "c:\ERPAdmin\Ausgabe\" "@DirAdrNr@@ID@ - @BEZ@" /Von:DAT=01.01.2010 /Bis:DAT=31.12.2010 /HIS /DELETE
Schreibt alle Dateien aus dem Ausgabeverzeichnis-Archiv (**/HIS**) aus dem Zeitraum 01.01.2010 bis 31.12.2010 in das Verzeichnis **c:\ERPAdmin\Ausgabe\.** Pro Adressnummer die erkannt wird, erzeugt **ERPAdmin** ein Unterverzeichnis in dem die dazugehörigen Dokumente abgelegt werden. Die Dokumente bekommen als Dateinamen die **ID** und die **Bezeichnung**. Nach dem Export werden die Datensätze gelöscht unabhängig ob der Export erfolgreich war.

26 Der Befehl „-BackupRestoreVorgangArten“ BAS

ERPAdmin -BackupRestoreVorgangArten <Befehl> /Datei:<Dateiname>

Dieser Befehl sichert die fortlaufenden Belegnummern aus den Vorgängen eines Mandanten und stellt diese auch wieder her. Das ist hilfreich, wenn man in einem Mandanten Änderungen im Bereich **Vorgänge und Zwischenbelege->Arten** durchgeführt hat und diese Änderungen mit dem Paket-Manager in einen anderen Mandanten übernehmen möchte ohne, dass dabei in dem Ziel-Mandanten aktuelle Belegnummern verloren gehen. Erstellen Sie deshalb vorher eine Sicherung, spielen Sie dann mit dem Paketmanager die Daten ein und sichern Sie anschließend die zuvor gesicherten Belegnummern zurück.

<Befehl>
Mit dem Befehl „**Backup**“ werden die fortlaufenden Belegnummern eines Mandanten in eine CSV-Datei gesichert.

Mit dem Befehl „**Restore**“ werden die fortlaufenden Belegnummern aus einer CSV-Datei in einen Mandanten zurückgespielt. Sollen nicht alle fortlaufende Belegnummern aus der CSV-Datei zurückgesichert werden, editieren Sie die Ergebnisdatei nach dem ausführen des Befehls „**Backup**“ mit einem Text-Editor und löschen Sie die nicht benötigten Zeilen.

/Datei:<Dateiname> **(optional)**
Gibt den Dateinamen der CSV-Datei an auf die mit den Befehlen „**Backup**“ und „**Restore**“ zugegriffen wird. Wird kein Dateiname angegeben, wird der Dateiname „**BackupRestore-VorgangArten.csv**“ verwendet.“

> **Hinweis**:
> Erstellen Sie unbedingt vor jeder Ausführung dieses Befehls eine Datensicherung Ihres **büro+**-Mandanten.

Beispiele:

ERPAdmin -BackupRestoreVorgangArten Backup
Erzeugt eine ANSI-CSV-Datei mit den Fortlaufenden Belegnummern des Mandanten. Das Ergebnis könnte wie folgt aussehen:

Nr	BelegLfdNr
1	1
10	723
15	115
20	456
30	0
40	12
50	224
60	0
70	219
71	0
80	0
90	0

Die erste Spalte „**Nr**“ gibt die **ArtNr** eines Vorgangs an. Diese findet man in den **büro+**-**Parametern** unter „**Vorgänge und Zwischenbelege->Arten**“

	15		Angebot
	20		Auftragsbestätigung
	30		Bestellung vom Kunden
	40		Bestellung an Lieferant
	41		Eingangsrechnung
	50		Lieferschein
	70		Rechnung
	90		Gutschrift

ERPAdmin -BackupRestoreVorgangArten Restore

Sichert die zur mit dem Befehl „**Backup**“ gesicherten fortlaufenden Belegnummern in den Mandanten zurück.

27 Der Befehl „-BelegNrCheck“ BAS

ERPAdmin -BelegNrCheck <VogArtNr> <DatumVon> <DatumBis> <BelegNrDef> /Datei:<CSV-Dateiname> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin überprüft im Vorgangs-Archiv die Vorgänge im Bereich **DatumVon** bis **DatumBis** ob diese gebucht und fortlaufend nummeriert sind. Werden Unstimmigkeiten gefunden werden diese in der CSV-Datei „**BelegNr-Check_<VorArtNr>_<DatumVon>-<DatumBis>_Fehler.csv**“ gespeichert.

Zur Überprüfung wird eine Liste aller gefundenen Belegnummern in der Datei „**BelegNr-Check_<VorArtNr>_<DatumVon>-<DatumBis>_Liste.csv**“ gespeichert.

> **Hinweis**:
> Wenn ein abweichendes Geschäftsjahr verwendet wird oder der laufende Nummernkreis der Vorgangsart zum Jahresende zurückgesetzt wird, darf der Zeitraum **DatumVon - DatumBis** nicht Vorgänge über diese zeitlichen Grenzen hinweg enthalten.

< VogArtNr >
Gibt mit der Vorgangs-Art-Nummer (**VogArtNr)** an, welche Vorgänge geprüft werden sollen. Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.**

15	Angebot	
20	Auftragsbestätigung	
30	Bestellung vom Kunden	
40	Bestellung an Lieferant	
41	Eingangsrechnung	
50	Lieferschein	
70	Rechnung	
90	Gutschrift	

<DatumVon>
Gib das Anfangsdatum an ab dem Die Belege geprüft werden sollen. Die Prüfung beginnt mit der ersten gefundenen Belegnummer ab Datum **DatumVon.**

<DatumBis>
Gibt das Ende-Datum an bis zu den Belegen geprüft werden sollen. Die Prüfung endet mit der letzter gefundenen Belegnummer **DatumBis**.

<BelegNrDef>
Hier wird **ERPAdmin** mitgeteilt an welcher Stelle in der Belegnummer die Nummerierung fortlaufend sein muss. Diese Stellen sind mit einem **@**-Zeichen zu versehen.

Beispiel: Die **rot** markierten Zahlen beschreiben die Stellen der fortlaufenden Nummerierung.
Belegnummern: RE2215675 RE22@@@@@
Belegnummern: 2215675RE 22@@@@@RE

/Datei<CSV-Dateiname> **(optional)**
Hier kann ein alternativer Dateiname für die Liste mit Unstimmigkeiten übergeben werden.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** das Ergebnis der Belegnummern-Prüfung sendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „Kapitel Platzhalter und variable Parameter“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Beispiel:

ERPAdmin -BelegNrCheck 70 01.01.2022 31.12.2022 RE22@@@@@
Prüft für das Jahr 2022 alle Belegnummer der Vorgang-Art-Nr=**70** (Rechnung) auf Korrektheit.

Das Ergebnis (Fehlerliste) könnte wie folgt aussehen:

	A	B	C	D	E
1	**BelegNr**	**AdrNr**	**Dat**	**GebuchtKz**	**Bemerkung**
2	RE2137825				Belegnummer fehlt
3	RE2238946	12473	03.01.2022	Ja	Belegnummer wurde im Vorgang 'Abschlagsrechnung' gefunden
4	RE2239058	12473	31.01.2022	Ja	Belegnummer wurde im Vorgang 'Abschlagsrechnung' gefunden
5	RE2239066	12473	15.02.2022	Ja	Belegnummer wurde im Vorgang 'Schlussrechnung' gefunden
6	RE2239177				Belegnummer fehlt
7	RE2239178	12462	16.03.2022	Nein	Vorgang NICHT gebucht
8	RE2239495	12462	09.06.2022	Ja	Belegnummer wurde im Vorgang 'Abschlagsrechnung' gefunden
9	RE2239650	12462	19.07.2022	Ja	Belegnummer wurde im Vorgang 'Schlussrechnung' gefunden

28 Der Befehl „-CalcMinArtikelEK“ PRO

ERPAdmin -CalcMinArtikelEK <ArtNrVon> <ArtNrBis> <Selektion> <ArtikelArt> /Datei:<Ergebnis-Datei> /Header

Für die ausgewählten Artikel im Bereich **ArtNrVon** bis **ArtNrBis** wird der Best-Preis-EK berechnet über alle hinterlegten Lieferanten. Dies gilt sowohl für Standard-Artikel als auch für Stücklisten (einstufig). Das Ergebnis wird in der **Selektion** gespeichert.

Bei mehrstufigen Stücklisten muss die Berechnung für die Stücklisten pro zusätzliche Stufe jeweils einmal zusätzlich aufgerufen werden.

<ArtNrVon>…. <ArtNrBis>
Gibt den Bereich der Artikelnummern an für den der Best-Preis-EK berechnet werden soll.

<Selektion>
Gibt den Namen eines Selektionsfeldes aus dem Bereich Artikel vom Typ Float (Betrag) an, in dem das Ergebnis gespeichert werden soll. Bereits vorhandene Werte werden überschrieben.

<ArtikelArt> **(optional)**
Soll der Best-Preis-EK nur für Standard-Artikel berechnet werden muss der Wert „0“ übergeben werden. Soll der Best-Preis-EK nur für Stücklisten berechnet werden muss der Wert „1“ übergeben werden. Wird kein wert übergeben wird der Best-Preis-EK für alle Standard-Artikel und Stücklisten im vorgegebenen Artikelnummern Bereich berechnet.

/Datei:<Ergebnis-Datei> **(optional)**
Das Ergebnis der Berechnungen wird zusätzlich in einer CSV-Datei gespeichert. Wird kein Dateiname angegeben wird die Datei „**Best-Preis.EK.csv**“ erzeugt.

/Header **(optional)**
Soll die Ergebnis-Datei eine Kopfzeile enthalten muss die Option **/Header** angegeben werden.

Beispiel:

ERPAdmin -CalcMinArtikelEK 10000 19999 Sel7 /Header
Es wird der Best-Preis-EK für alle Standard-Artikel und Stücklisten im Bereich **10000** bis **19999** berechnet und im Selektionsfeld **Sel7** abgespeichert. Die Ergebnisdatei enthält eine Kopfzeile

ERPAdmin -CalcMinArtikelEK 10000 19999 Sel7 1
Es wird der Best-Preis-EK nur für Stücklisten im Bereich **10000** bis **19999** berechnet und im Selektionsfeld **Sel7** abgespeichert.

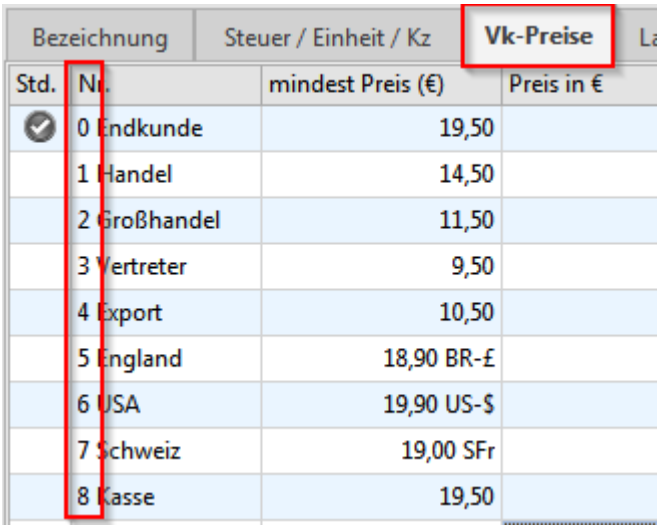
29 Der Befehl „-CalcStkLstMinVK“ STD

ERPAdmin -CalcStkLstMinVk <ArtNrVon> <ArtNrBis> <VKNr>

Dieser Befehl berechnet den Mindest-Verkaufspreis einer einstufigen Stückliste auf Basis der Summe der Mindestverkaufspreise der einzelnen Stücklistenpositionen.

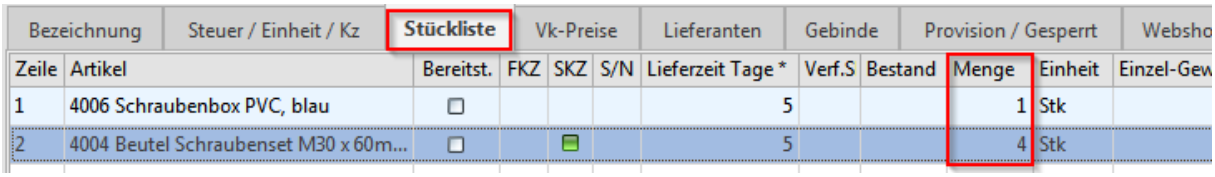
<ArtNrVon> und <ArtnrBis>
Mit den beiden Parametern **<ArtNrVon>** und **<ArtNrBis>** wird der Bereich der Stücklisten über die Artikelnummer eingeschränkt.

<VKNr>
Gibt die Nummer der Verkaufspreisliste an. Es ist in der Regel ein Wert von 0..9.

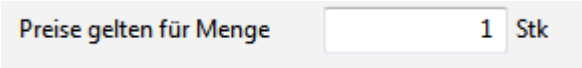


Ermittelt werden dazu die folgenden Werte:

Menge (Anzahl) der Stücklisten-Positionen

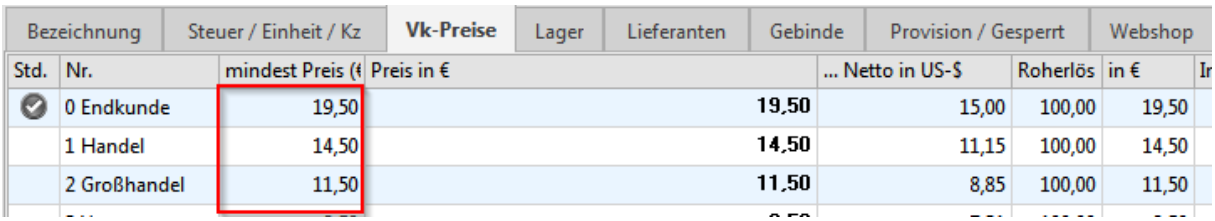


Preise gelten für Menge in den Stammdaten der Stücklistenposition



Hinweis:
Ist hier kein Wert eingetragen wird der Wert „**1**“ angenommen.

<u>Mindestverkaufspreis aus den Stammdaten der Stücklistenposition</u> (In Abhängigkeit der VKNr)



Bezeichnung	Steuer / Einheit / Kz	Vk-Preise	Lager	Lieferanten	Gebinde	Provision / Gesperrt	Webshop

Std.	Nr.	mindest Preis (€	Preis in €	... Netto in US-$	Roherlös	in €	Ir
✓	0 Endkunde	19,50	19,50	15,00	100,00	19,50	
	1 Handel	14,50	14,50	11,15	100,00	14,50	
	2 Großhandel	11,50	11,50	8,85	100,00	11,50	

Die Berechnung wird dann mit diesen Werten wie folgt durchgeführt:

<u>Menge * MinVKPreis</u>
Preise gelten für Menge

Das Ergebnis wird in den Mindestverkaufspreis des Stücklistenartikels zurück geschrieben in Abhängigkeit der **VKNr**.

<u>Beispiel:</u>

ERPAdmin - CalcStkLstMinVk 10000 99999 1
Berechnet der Mindestverkaufspreis für alle Stücklisten mit den Artikelnummern 10000 bis 99999 in der Verkaufspreisgruppe 0 (VK0).

30 Der Befehl „-CopyFromFile“ PRO

ERPAdmin -CopyFromFile <CSV-Datei> <Quellspalte> <Zielspalte> /Header /Delete

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin kopiert Dateien anhand der Spalteninhalte der beiden Spalten **Quellspalte** und **Zielspalte** aus der **CSV-Datei**. In der **Quellspalte** steht der Dateiname inkl. Pfadangabe der zu kopierenden Datei. In der **Zielspalte** steht der Dateiname inkl. Pfadangabe wohin dies Datei kopiert werden soll. Existiert die Zieldatei, wird diese überschrieben. Existiert das Zielverzeichnis nicht wird es vor dem Kopieren erzeugt.

/Header **(optional)**
Besitzt die CSV-Datei eine Kopfzeile kann diese durch Angabe von **/Header** bei der Verarbeitung übersprungen werden.

/Delete **(optional)**
Sollen die Quelldateien nach dem kopieren gelöscht werden, muss die Option **/Delete** angegeben werden.

Beispiel:

ERPAdmin -CopyFromFile Bildliste.csv 3 4 /Delete

	A	B	C	D
1	1	3267.jpg	C:\ERPAdmin\Bilder-Quelle\3267.jpg	C:\ERPAdmin\Bilder-Zie\1\3267.jpg
2	2	3268.jpg	C:\ERPAdmin\Bilder-Quelle\3268.jpg	C:\ERPAdmin\Bilder-Zie\2\3268.jpg
3	3	3269.jpg	C:\ERPAdmin\Bilder-Quelle\3269.jpg	C:\ERPAdmin\Bilder-Zie\3\3269.jpg
4	4	3270.jpg	C:\ERPAdmin\Bilder-Quelle\3270.jpg	C:\ERPAdmin\Bilder-Zie\4\3270.jpg
5	5	3271.jpg	C:\ERPAdmin\Bilder-Quelle\3271.jpg	C:\ERPAdmin\Bilder-Zie\5\3271.jpg

ERPAdmin kopiert fünf Dateien aus der Spalte **3 (C)** in die Verzeichnisse der Spalte **4 (D)**. Die Unterverzeichnisse „1“, „2“, „3“, „4“ und „5“ werden dabei angelegt. Nach dem kopieren werden die Quelldateien gelöscht (**/Delete**).

31 Der Befehl „-CopyLinesToFile“ PRO

ERPAdmin -CopyLinesToFile <CSV-Datei> <Zeile-Nr.> <Anzahl> /Datei:<Ziel-Datei>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin kopiert **Anzahl** von Zeilen, beginnend mit **Zeile-Nr.** aus Datei **CSV-Datei** und speichert das Ergebnis unter gleichem Namen ab.

<CSV-Datei>
ANSI-**CSV-Datei** aus der Zeilen kopiert werden sollen. Wir keine Ergebnisdatei **/Datei:<Zieldatei>** übergeben wird das Ergebnis in der **CSV-Datei** gespeichert. Der vorherige Inhalt geht damit verloren.

<Zeile-Nr>
Zeile-Nr aus **CSV-Datei** ab der Inhalte kopiert werden sollen.

<Anzahl>
Gibt die Anzahl der Zeilen an die kopiert werden sollen. Sollen alle Zeilen bis zum Ende der Datei kopiert werden kann **@Z** übergeben werden.

/Datei:<Ziel-Datei> (optional)
Wir eine **Ziel-Datei** angegeben, wird das Ergebnis unter diesem Namen gespeichert. Die Original-Datei bleibt damit erhalten.

Beispiel:

Die ersten beiden Zeilen der Datei **DATEV.csv** sollen in der Datei **Header.csv** gespeichert werden.

ERPAdmin -CopyLinesToFile DATEV.csv 1 2 /Datei:Header.csv

```
"EXTF";700;21;Buchungsstapel;9;20220623132414000;;BP;MB;;1596;60;20210101;4;20210101;20211231;
"Umsatz (ohne Soll/Haben-Kz)";Soll/Haben-Kennzeichen;"WKZ Umsatz";Kurs;Basis-Umsatz;"WKZ Basis
34,98;"H";"EUR";1,000000;;"";70499;4380;"";101;"";"";;"";;"";;;;"";"";"";"";"";"";"";"";"";"DsG
648,65;"H";"EUR";1,000000;;"";70208;4520;"";101;"";"";;"";;"";;;;"";"";"";"";"";"";"";"";"";"Ds
54,00;"H";"EUR";1,000000;;"";12034;4380;"";101;"";"";;"";;"";;;;"";"";"";"";"";"";"";"";"";"DsG
700,00;"H";"EUR";1,000000;;"";70544;4210;"";101;"2021/01";"";;"Miete für Büro 2021/01";;"";;;;"
```

Das Ergebnis ist der rot umrandete Bereich der Datei.

32 Der Befehl „-ConCatFiles“ STD

ERPAdmin -ConCatFiles <CSV-Datei-1> < CSV-Datei-2> <Ergebnis-Datei> /DELETE

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl können die Inhalte von zwei ANSI-Text-Dateien zusammengeführt werden. Dabei werden die Inhalte in der Reihenfolge „**CSV-Datei 1**“ „**CSV-Datei 2**“ aneinandergehängt und als „**Ergebnis-Datei**“ abgespeichert.

Ein Typisches Anwendungsbeispiel wären CSV-Dateien in denen die Kopfzeilen mit den Feldbezeichnern fehlen.

<CSV-Datei 1>
Dateiname inkl. Pfadangabe der ersten Datei z.B. Datei mit Kopfzeileninformationen

<CSV-Datei 2>
Dateiname inkl. Pfadangabe die mit der Datei CSV-Datei 1 verknüpft werden soll.

Sind mehrere Dateien zu bearbeiten kann ein Wildcard (*) verwendet werden. Es wird dann pro „CSV-Datei 2“ auch eine Ergebnis-Datei erzeugt.

<Ergebnis-Datei>
Dateiname inkl. Pfadangabe in der das Ergebnis abgespeichert werden soll.

Wurde in „**CSV-Datei 2**“ ein Wildcard (*) verwendet, muss auch hier an der entsprechenden Stelle ein Wildcard (*) angegeben werden.

/Delete **(optional)**
Sollen die Dateien „**CSV-Datei 2**“ nach erfolgreicher Verarbeitung aus dem Verzeichnis gelöscht werden, erfolgt dies mit der Option **/Delete**.

Beispiel:

ERPAdmin -ConCatFiles Header.csv Daten.Csv Eregbnis.csv
Verketten den Inhalt der Datei **Header.csv** und **Daten.csv** und speichert das Ergebnis in **Ergebnis.csv** ab.

ERPAdmin -ConCatFiles Header.csv Daten_*.csv Eregbnis_*.csv /Delete
Verketten den Inhalt der Datei **Header.csv** mit den jeweiligen Inhalten der Dateien **Daten_*.csv** und speichert das Ergebnis in den Dateien **Ergebnis_*.csv** ab. Anschließend werden die Dateien „**Daten:*.csv**“ gelöscht.

Beispiel-Daten-Dateien:	**Beispiel-Ergebnis-Dateien**
Header.csv	
Daten_1234567.csv	Ergebnis_123456.csv
Daten_11223344.csv	Ergebnis_11223344.csv
Daten_15.10.2019_123305.csv	Ergebnis_15.10.2019_123305.csv

Geben Sie in **der VAR.INI** ein History-Verzeichnis (**C_HistoryDir**=“….“ an dann werden alle Daten dort historisiert.

33 Der Befehl „-CreateSN“ PRO

ERPAdmin -CreateSN <Format> <Anzahl> <Ergebnis-Datei> <Präfix> <Postfix> /INIT:< Erste-Seriennummer >

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin erzeugt Seriennummern im vorgegebenen Format. Diese werden in einer ANSI-CSV-Datei gespeichert und können damit direkt in für **büro+** importiert werden. Verwenden Sie hierzu z.B. die Befehle „**ERPAdmin -VorgangImport**“ oder „**ERPAdmin -Import“.**

<Format>
Beschreibt den Aufbau der Seriennummer. Es muss der Platzhalter **@@** enthalten welcher von **ERPAdmin** durch die eigentliche Seriennummer ersetzt wird. Beispiel: **ABC.100.33.@@**.

Beim ersten Aufruf wird die fortlaufende S/N-Nummer mit **10000** vorbelegt. Soll mit einer anderen Seriennummer begonnen werden muss zuvor einmalig ein Aufruf **/INIT:<Erste-Seriennummer>** übergeben werden. Dieser Wert wird in der Datei **SN.INI** gespeichert.

Bei allen weiteren Aufrufen wird jeweils die letzte fortlaufende Seriennummer in der Datei **SN.INI** gespeichert. Diese kann auch mit einem Texteditor bearbeitet werden um manuell einen anderen Startwert für den nächsten Aufruf zu hinterlegen.

<Anzahl>
Gibt die Anzahl der Seriennummern an die erzeugt werden sollen. Die nächste zu verwendende Seriennummer steht in der Datei **SN.INI**

<Ergebnis-Datei>
Alle Seriennummern werden in einer ANSI-CSV-Datei unter dem Namen **Ergebnis-Datei** gespeichert.

<Präfix> **(optional)**
Es kann eine Zeichenkette übergeben werden welche vor die fortlaufenden Seriennummern in die CSV-Datei geschrieben wird.

<Postfix> **(optional)**
Es kann eine Zeichenkette übergeben werden welche nach der fortlaufenden Seriennummern in die CSV-Datei geschrieben wird.

/INIT:< Erste-Serialnummer> **(optional)**
Vor dem erzeugen der Seriennummern kann ein Anfangswert, die erste Seriennummer, für **ERPAdmin** gesetzt werden. Wurde die Option **/INIT:<…>** angegeben werden mit diesem Aufruf **KEINE** Seriennummern erzeugt, sondern nur der Startwert in der Datei SN.INI geschrieben.

ERPAdmin -CreateSN /Init:<Erste-Seriennummer>

Beispiele:

ERPAdmin -CreateSN /Init:10000
ERPAdmin -CreateSN @@ 150 sn.csv
Erzeugt die CSV-Datei **sn.csv** mit den Seriennummern **10000** bis **10149**.

ERPAdmin -CreateSN /Init:10000
ERPAdmin -CreateSN ABC.100.@@ 100 sn.csv "71050;4130;1;" ";Stk"
Erzeugt die CSV-Datei **sn.csv** mit vier Spalten. In der letzten Spalte stehen die Seriennummern **ABC.100.10000** bis **ABC.100.10099**. Die Datei enthält 100 Datenzeilen.

Die **Ergebnis-Datei** sieht dann wie folgt aus:

	A	B	C	D	E
1	71050	4130	1	ABC.100.10000	Stk
2	71050	4130	1	ABC.100.10001	Stk
3	71050	4130	1	ABC.100.10002	Stk
4	71050	4130	1	ABC.100.10003	Stk
5	71050	4130	1	ABC.100.10004	Stk

ERPAdmin -CreateSN /Init:10016
ERPAdmin -CreateSN ABC.100.@@ 4 sn.csv "70002;4000;1;"
ERPAdmin -VorgangImport sn.csv 1 VorgangImport_Layout.csv 41 70000 Stk Vog.AdrNr /Buchen
Erzeugt **4** Seriennummern im Format **ABC.100.xxxxx** und Erzeugt anschließend einen Eingangslieferschein (**41**) für den Lieferanten **70002** mit **4** Positionen des Artikels **4000** und dessen Seriennummern.

In der Datei **VorgangImport_Layout.csv** sind die Platzhalter für den Vorgangs-Import definiert:
Vog.AdrNr;VPo.ArtNr;VPo.Mge;VPo.BSNR

Der Vorgang wird nach Erzeugung sofort gebucht (**/Buchen**).

Das Ergebnis in **büro+** sieht dann wie folgt aus:

Adresse | **Positionen** | Infoblatt | Vor- / Nachtext | Adr.-Kennzeichen | Vorgaben | Selektionen | Pa

Pos	SW	Artikelnummer	Bezeichnung	Bestell-/Serien-Nr.	Charge	Menge
1.	☐	4000	Ausweis	ABC.100.10016		1
2.	☐	4000	Ausweis	ABC.100.10017		1
3.	☐	4000	Ausweis	ABC.100.10018		1
4.	☐	4000	Ausweis	ABC.100.10019		1

34 Der Befehl „-CryptPwd“ BAS

ERPAdmin -CryptPwd <VAR-INI-Variable>

ERPAdmin verschlüsselt den Wert einer Passwort-Variablen in der Datei **VAR.INI**. Wird keine Variable angegeben erscheint ein Dialog und es kann ein beliebiger Text verschlüsselt werden.

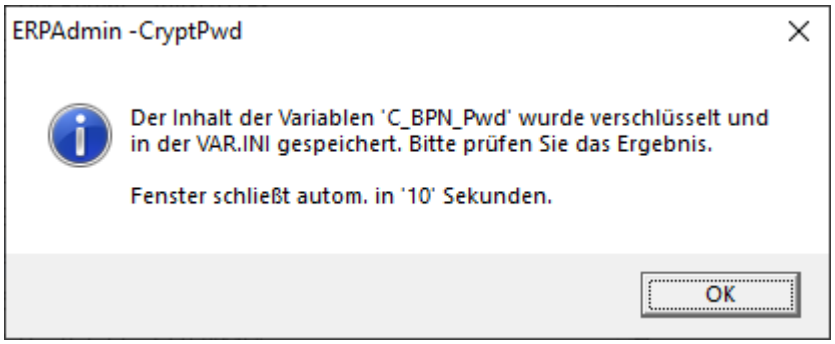
<VAR-INI-Variable> STD **(optional)**
Übergibt den Variablen-Namen aus der **VAR.INI** dessen Inhalt nachträglich verschlüsselt werden soll. Zulässige Variablen sind:

- **C_SMTPPWD**
- **C_POPPWD**
- **C_POPConnectPwd**
- **C_BPN_PWD**
- **C_BPN_PWD2**
- **C_BPN_PWD3**
- **C_BPN_PWD4**
- **C_BPN_PWD5**
- **C_FTPPWD**
- **C_FTPPWD2**
- **C_FTPPWD3**
- **C_FTPPWD4**
- **C_FTPPWD5**
- **C_SFTPPWD**
- **C_SFTPPWD2**
- **C_SFTPPWD3**
- **C_SFTPPWD4**
- **C_SFTPPWD5**

Ist ein unverschlüsselter Wert in der **VAR.INI** in der angegebenen Variablen hinterlegt, wird dieser Wert ohne weitere Rückfrage verschlüsselt und wieder in der **VAR.INI** abgespeichert.

Beispiel:

ERPAdmin - CryptPwd C_BPN_PWD
Prüft ob die Variable **C_BPN_PWD** in der **VAR.INI** existiert und bereits ein unverschlüsseltes Kennwort enthält. Ist das der Fall wird dieses Kennwort verschlüsselt und es erscheint die folgende Meldung:

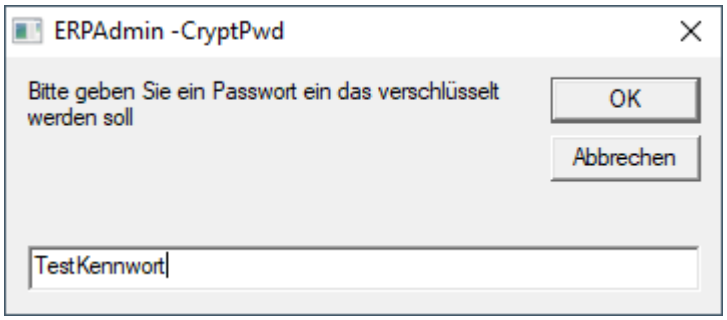


Ist noch **kein** Wert bei der Variablen **C_BPN_PWD** in der **VAR.INI** hinterlegt oder existiert die Variable nicht in der **VAR.INI,** fragt ein Benutzerdialog das Kennwort ab, verschlüsselt das Kennwort und speichert es dann wieder in der **VAR.INI** ab.

Wird beim Aufruf keine **VAR-INI-Variable** übergeben erscheint ein Dialog der vom Benutzer den zu verschlüsselnden Wert abfragt und das Ergebnis anschließend am Bildschirm zur Weiterverarbeitung anzeigt.

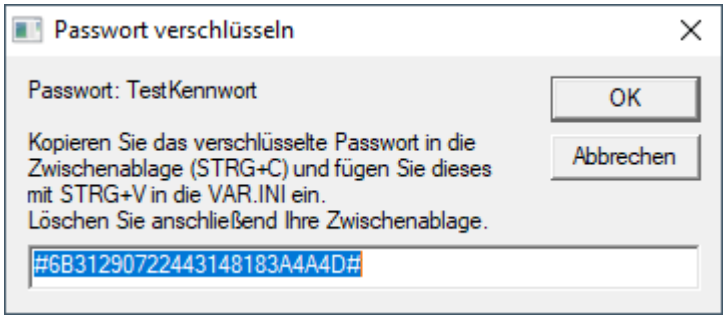
Beispiele:

ERPAdmin - CryptPwd
Öffnen den Dialog zur Verschlüsselung eines beliebigen Kennwortes



[Beispiel für die Eingabe des Kennwortes „TestKennwort“]

Nach Betätigung des OK-Buttons erscheint der folgende Dialog:



35 Der Befehl „-CSV2HTML“ STD

ERPAdmin -CSV2HTML <CSV-Datei> <HTML-Datei>

Für die Verwendung dieses Befehls benötigen Sie die Standard Version von **ERPAdmin**!

ERPAdmin wandelt die **CSV-Datei** in eine **HTML-Datei** um.

<CSV-Datei>
Dateiname inkl. Pfadangabe für Datei im ANSI-CSV-Format

<HTML-Datei>
Dateiname inkl. Endung und Pfadangabe für die Ergebnisdatei im HTML-Format.

Beispiel:

ERPAdmin -csv2html Order.csv Order.html
Erzeugt aus der CSV-Datei „**Order.csv**“ die HTML-Datei „**Order.html**“

36 Der Befehl „-CSV2XLS“ PRO

ERPAdmin -CSV2XLS <CSV-Datei> <XLS-Datei> <Trennzeichen> <Zellen-Format> /NoFileNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl konvertiert **ERPAdmin** unter Verwendung von MS-Excel (**muss auf dem System installiert sein**) die **CSV-Datei** in eine **XLS-Datei** (Excel). Existiert die XLS-Datei bereits wird diese überschrieben.

<CSV-Datei>
Dateiname inkl. Pfadangabe für Datei im ANSI-CSV-Format

<XLS-Datei>
Dateiname inkl. Endung und Pfadangabe für die Ergebnisdatei im Excel-Format.

<Trennzeichen> **(optional)**
Mit dem Parameter **Trennzeichen** kann ein Trennzeichen übergeben werden welches in der CSV-Datei verwendet wird. Wird kein Trennzeichen übergeben wird das Semikolon (**;**) verwendet.

<Zellen-Format> **(optional)**
Mit dem Parameter **Zellen-Format** kann ein Zellen-Format erzwungen werden. Als Werte sind „**@**“ für TEXT und „**0**“ für Zahlen zulässig. Wird kein **Format** übergeben, ermittelt Excel selbständig das Format der Zellen.

/NoFileNotExistError **(optional)**
Ist keine **CSV-Datei** vorhanden und soll in diesem Fall **keine** Fehlermeldung erzeugt werden, kann die Option **/NoFileNotExistError** verwendet werden.

> **Hinweis**
> Wird als Endung bei der **Excel-Datei** **.xlsx** angegeben, wird eine Excel-Datei im Excel-2010 Format erzeugt.

Beispiele

ERPAdmin -CSV2XLS C:\ERPAdmin\Adressen.csv
Konvertiert die CSV Datei in eine Excel-Datei (.xls). Da kein Dateiname für die Excel-Datei angegeben wurde, wird der gleiche Name verwendet wie bei der CSV-Datei nur mit der Endung.XLS. Als Trennzeichen wird ein Semikolon verwendet. Das Format der Zellen bestimmt Excel selbstständig.

ERPAdmin -CSV2XLS C:\ERPAdmin\Adressen.csv c:\ERPAdmin\Adressen.xls
Konvertiert die CSV Datei in eine Excel-Datei (.xls). Als Trennzeichen wird ein Semikolon verwendet. Das Format der Zellen bestimmt Excel selbstständig.

ERPAdmin -CSV2XLS C:\ERPAdmin\Adressen.csv c:\ERPAdmin\Adressen.xlss
Konvertiert die CSV Datei in eine Excel-Datei (.xlss). Als Trennzeichen wird ein Semikolon verwendet. Das Format der Zellen bestimmt Excel selbstständig.

ERPAdmin -CSV2XLS C:\ERPAdmin\Adressen.csv c:\ERPAdmin\Adressen.xls "," "@“
Konvertiert die CSV Datei in eine Excel-Datei (.xls). Als Trennzeichen wird ein Komma verwendet. Das Format der Zellen ist Text. Dies hat den „möglichen“ Vorteil, dass Felder mit führenden Nullen auch mit diesen importiert und dargestellt werden.

37 Der Befehl „-CSV2XML“ STD

ERPAdmin -CSV2XML <CSV-Datei> <XML-Datei> /Header

Mit diesem Befehl konvertiert **ERPAdmin** die **CSV-Datei** in eine **XML-Datei** (Excel). Das Zeichen-Format der XML-Datei ist UFT-8 und kann mit einem Text-Editor per Doppelklick direkt geöffnet werden. Alle Zelleninhalte werden als Text formatiert um fehlerhafte Darstellungen von Zahlen mit führenden Nullen zu vermeiden.

/Header **(optional)**
Mit dieser Option wird **ERPAdmin** mitgeteilt, dass es sich bei der ersten Zeile um eine Kopfzeile handelt.

Beispiel

ERPAdmin -CSV2XML C:\ERPAdmin\Adressencsv C:\ERPAdmin\Adressen.xml

Inhalt der Datei Adressen.csv :

Mandant	ArtikelID	Bezeichnung	Status	Menge	Datum	Bewegungstyp	Lieferant	SeqNr	ExportNr	PosNr
1010	11551	Edding 3000	OK	2	15.01.2018	A5	70000	A100		1
1010	12456	Verschlusskappe	OK	1	15.01.2018	A5	70000	A200		2
1010	10023	Edding SET gelb	OK	4	15.01.2018	A5	70000	A300		3
1010	11023	Porto und Verpackung	OK	1	20.01.2018	A5	70000	A300		3

Auszug aus Ergebnisdatei Adressen.xml


```
<?mso-application progid="Excel.Sheet"?>
▼<Workbook xmlns="urn:schemas-microsoft-com:office:spreadsheet" xmlns:o="urn:schemas-microsoft-
com:office:office" xmlns:x="urn:schemas-microsoft-com:office:excel" xmlns:ss="urn:schemas-microsoft-
com:office:spreadsheet" xmlns:html="http://www.w3.org/TR/REC-html40">
 ▼<DocumentProperties xmlns="urn:schemas-microsoft-com:office:office">
    <Author>ERPAdmin</Author>
    <LastAuthor>ERPAdmin</LastAuthor>
    <Created>2021-03-09T08:40:53Z</Created>
    <Company>Systemhaus HORN</Company>
    <Version>11.9999</Version>
  </DocumentProperties>
 ▼<ExcelWorkbook xmlns="urn:schemas-microsoft-com:office:excel">
    <WindowHeight>13230</WindowHeight>
    <WindowWidth>19380</WindowWidth>
    <WindowTopX>120</WindowTopX>
    <WindowTopY>135</WindowTopY>
    <ProtectStructure>False</ProtectStructure>
    <ProtectWindows>False</ProtectWindows>
  </ExcelWorkbook>
 ▼<Styles>
  ▼<Style ss:ID="Default" ss:Name="Normal">
     <Alignment ss:Vertical="Bottom"/>
     <Borders/>
     <Font/>
     <Interior/>
     <NumberFormat/>
     <Protection/>
    </Style>
  </Styles>
 ▼<Worksheet ss:Name="Tabelle1">
  ▼<Table ss:ExpandedColumnCount="11" ss:ExpandedRowCount="5" x:FullColumns="1" x:FullRows="1"
   ss:DefaultColumnWidth="70" ss:DefaultRowHeight="13.241379310344827">
   ▼<Row>
    ▼<Cell>
       <Data ss:Type="String">Mandant</Data>
     </Cell>
    ▼<Cell>
       <Data ss:Type="String">ArtikelID</Data>
     </Cell>
    ▼<Cell>
       <Data ss:Type="String">Bezeichnung</Data>
     </Cell>
```

```
▼<Cell>
    <Data ss:Type="String">Status</Data>
  </Cell>
▼<Cell>
    <Data ss:Type="String">Menge</Data>
  </Cell>
▼<Cell>
    <Data ss:Type="String">Datum</Data>
  </Cell>
▼<Cell>
    <Data ss:Type="String">Bewegungstyp</Data>
  </Cell>
▼<Cell>
    <Data ss:Type="String">Lieferant</Data>
  </Cell>
▼<Cell>
    <Data ss:Type="String">SeqNr</Data>
  </Cell>
▼<Cell>
    <Data ss:Type="String">ExportNr</Data>
  </Cell>
▼<Cell>
    <Data ss:Type="String">PosNr</Data>
  </Cell>
</Row>
▼<Row>
  ▼<Cell>
    <Data ss:Type="String">1010</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">0011551</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">Edding 3000</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">OK</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">2</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">15.01.2018</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">A5</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">70000</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">A100</Data>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String"/>
  </Cell>
  ▼<Cell>
    <Data ss:Type="String">1</Data>
  </Cell>
</Row>
```

38 Der Befehl „-CSVKonverter“ PRO

ERPAdmin -CSVKonverter <CSV-Datei> <Befehl> <SpalteNr> <Param1> <Param2> <Param3> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /NoFileNotExistError /Header /Path:<Ziel-Verzeichnis> /Datei:<Neuer Dateiname> /Delimiter:<Trennzeichen> /MoveToHis /MoveToBad

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl können verschiedene Manipulationen an einer oder mehrerer **ANSI**-CSV-Dateien mit **ERPAdmin** vorgenommen werden. Dabei wird im Parameter **SpalteNr** die Spalte übergeben, welche bearbeitet werden soll. Die Angabe der Spalte erfolgt als Zahl.

Die Übergabe einer Spaltenbezeichnung ist nur erlaubt in Kombination mit der Option **/Header**. **ERPAdmin** bearbeitet alle Zeilen der **CSV-Datei**. Sollte die **CSV-Datei** eine Kopfzeile enthalten, wird auch diese bearbeitet, wenn nicht die Option **/Header** übergeben wird.

Trennzeichen ist ein Semikolon (**;**) weshalb dieses Zeichen innerhalb der Daten nicht erlaubt ist.

> **Hinweis**
> Doppelte Anführungszeichen (**"**) sind in den CSV-Dateien nicht erlaubt. Führen Sie vorher eine Ersetzung durch mit **ERPAdmin -FileReplace Datei.csv "\"" '**

Wurde in der **VAR.INI** ein History-Verzeichnis angegeben (**C_HistoryDIR**), wird dort eine Kopie der Original-Dateien abgelegt bevor mit der Verarbeitung begonnen wird. So kann jeder zeit auf den vorherigen Zustand der Datei zurück gegriffen werden.

<CSV-Datei>
Dateiname inkl. Pfadangabe für eine Datei im **ANSI-CSV-Format**. Sollen mehrere Dateien verarbeitet werden kann ein Wildcard (*) verwendet werden z.B. *.csv .

Das Trennzeichen zwischen den Datenfeldern in der CSV-Datei ist ein Semikolon (**;**). Das Zeilenende und damit Ende eines Datensatzes ist ein **Carriage Return** (**CRLF**). Es werden keine Begrenzungszeichen wie Anführungszeichen etc. verwendet. Sollte der Zeilenumbruch ein **LF** sein (Datei kommt von einem MAC oder einem Linux-Rechner), muss die Datei vorher mit dem Befehl **ERPAdmin -CSVKonveter 0 ReplaceLF** konvertiert werden.

<SpalteNr>
Spaltennummer in der **CSV-Datei** die bearbeitet werden soll. Die Spaltennummerierung starten mit 1.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **CSV-Datei** sendet. Wurden keine Datei gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe

„**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/NoFileNotExistError **(optional)**
Ist keine Datei vorhanden und soll in diesem Fall **keine** Fehlermeldung erzeugt werden, kann die Option **/NoFileNotExistError** verwendet werden.

/Header **(optional)**
Mit dieser Option wird die erste Zeile (Kopfzeile) ignoriert und nur die Datenzeilen der Datei bearbeitet. Datenzeilen sind dann alle Zeilen ab der Zeile 2 und Folgende.

/Path:<Ziel-Verzeichnis> **(optional)**
Soll das Ergebnis in einem anderen Verzeichnis oder auf einem anderen Laufwerk gespeichert werden kann hier ein alternativer Pfad angegeben werden. Die Ursprungsdatei bleibt dabei unverändert erhalten.

/Datei:<Neuer Dateiname> **(optional)**
Soll die CSV-Datei unverändert bestehen bleiben und das Ergebnis unter einem anderen Namen abgespeichert werden, kann ein abweichender Dateiname mit der Option **/Datei:** übergeben werden.

Wurde in einem vorherigen Aufruf ein abweichender Dateiname mit der Option **/Datei:<Neuer Dateiname>** übergeben, kann diese in einem Folgeraufruf über den Platzhalter **@LASTFILENAME@** verwendet werden. Es wird immer der zu Letzt übergebene Dateiname verwendet!

/MoveToHis **(optional)**
Verschiebt die **CSV-Datei** nach erfolgreicher Verarbeitung in das History-Verzeichnis (**C_HistoryDir**).
Diese Option macht nur Sinn in Kombination mit der Option **/Datei:<Neuer Dateiname>.**

/MoveToBad **(optional)**
Verschiebt die **CSV-Datei** nach fehlerhafter Verarbeitung in das Bad-Verzeichnis (**C_BADDir**).
Diese Option macht nur Sinn in Kombination mit der Option **/Datei:<Neuer Dateiname>.**

Beispiel:
ERPAdmin -CSVKonverter Datei.csv SetValue 6 1 "Ja" /Datei:Datei_Neu.csv
ERPAdmin -CSVKonverter @LASTFILENAME@ Sort 3 /Datei:Datei_Neu2.csv
Beim 2. **ERPAdmin**-Aufruf wird der Dateiname „**Datei_Neu.csv**“ verwendet. Bei einem weiteren Aufruf von **ERPAdmin** würde bei der Verwendung des Platzhalters **@LASTFILENAME@** der Wert „**Datei_Neu2.csv**“ verwendet werden.

/Delimiter:<Trennzeichen> **(optional)**
Soll **ERPAdmin** eine CSV-Datei verarbeiten bei der nicht das Semikolon das Trennzeichen zwischen den einzelnen Spalten darstellt, so kann hier das entsprechende Trennzeichen übergeben werden.

Beispiel: Es soll ein Komma als Trennzeichen verwendet werden: **/Delimiter:,**

Wird kein Trennzeichen übergeben wird immer ein Semikolon (**;**) als Trennzeichen verwendet.

<Befehl>
Es stehen die folgenden Befehle zur Bearbeitung von CSV-Dateien zur Verfügung. Eine genaue Beschreibung der einzelnen Befehle inkl. Beispielen finden Sie nachfolgend beschrieben.

- AddCol
- AddGuid
- AddHeader
- AddString
- AutoNum
- Calc
- CalcDate
- ConCat
- Copy
- CopyMid
- CrossCopy
- CutLeft
- CutLeftTo
- CutMid
- CutRight
- CutRightTo
- DBLookUp
- Delete
- DeleteCol
- DeleteColByName
- DeleteDoubleSpaces
- DeleteDubRow
- DeleteHeader
- DeleteRow
- FileMapping
- FormatDate
- GetFilename
- GetPath
- ISOLand2UstKat
- ISOLandNr2UstKat
- Move
- Replace
- ReplaceCRLF
- ReplaceLF
- ReplaceTAB
- ReplaceISOLand
- Round
- SearchAndReplace
- SetBoolean
- SetupNr
- SetValue
- Sort
- Split
- SumCol

- SumRow
- Trim

AddCol

ERPAdmin -CSVKonverter <CSV-Datei> ADDCOL < ErgebnisSpalte > <SpalteWert> <SpalteName>

Fügt eine neue Spalte an Position **ErgebnisSpalte** ein und befüllt diese mit dem Inhalt **SpalteWert**. Wurde ein **SpalteName** mit übergeben (ist bei Datei mit Kopfzeile Pflicht **/Header**) wird dieser Wert als Kopfzeilen-Bezeichnung eingetragen. Soll die **ErgebnisSpalte** als letzte Spalte hinzugefügt werden kann der Wert **@Z** übergeben werden.

Die Angaben **SpaltenWert** und **SpaltenName** sind **optional (**in Abhängigkeit ob die Option **/Header** verwendet wurde).

Wird eine **CSV-Datei** mit Kopfzeile bearbeitet (**/Header**) muss der **Spaltenname** eindeutig sein und darf nicht existieren. Existiert ein Spaltenname bei einer CSV-Datei mit Kopfzeile bereits, wird eine Fehlermeldung ausgegeben und die Spalte **nicht** hinzugefügt.

Sollen mehrere Spalten mit einem Aufruf hinzugefügt werden, sind die jeweiligen **SpaltenWerte** und **SpaltenNamen** durch ein Semikolon zu trennen. In diesem Fall müssen genauso viele **SpaltenWerte** wie **SpaltenNamen** angegeben werden.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv ADDCOL 3 TEST
Fügt als **3**. Spalte eine neue Spalte hinzu und belegt die Spalten-Werte mit „**TEST**“.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D	E
1	111	22222	TEST	12345678	44444444
2	111	22222	TEST	12345678	44444444
3	111	22222	TEST	12345678	44444444

ERPAdmin -CSVKonverter Liste.csv ADDCOL 5 555 Test /Header
Fügt als 5. Spalte eine neue Spalte mit der Überschrift „**Test**“ und mit dem Spalten-Wert „**555**“ ein.

Datei vorher:

	A	B	C	D	E
1	Spalte1	Spalte2	Spalte3	Spalte4	
2	111	22222	12345678	4444444444	
3	111	22222	12345678	4444444444	
4	111	22222	12345678	4444444444	

Datei nachher

	A	B	C	D	E
1	Spalte1	Spalte2	Spalte3	Spalte4	Test
2	111	22222	12345678	4444444444	555
3	111	22222	12345678	4444444444	555
4	111	22222	12345678	4444444444	555

ERPAdmin -CSVKonverter Liste.csv ADDCOL @Z 555; Neu1;Neu2 /Header
Fügt als 5. und 6. Spalte neue Spalten mit den Überschriften „**Neu1**“ und „**Neu2**“ hinzu und belegt die Spalte „**Neu1**“ mit dem Wert „**555**“ und die Spalte „**Neu2**“ ohne Wert.

Datei vorher:

	A	B	C	D	E
1	Spalte1	Spalte2	Spalte3	Spalte4	
2	111	22222	12345678	4444444444	
3	111	22222	12345678	4444444444	
4	111	22222	12345678	4444444444	

Datei nachher

	A	B	C	D	E	F
1	Spalte1	Spalte2	Spalte3	Spalte4	NEU1	Neu2
2	111	22222	12345678	4444444444	555	
3	111	22222	12345678	4444444444	555	
4	111	22222	12345678	4444444444	555	

AddGuid

ERPAdmin -CSVKonverter <CSV-Datei> ADDGUID SpalteNr

Fügt in der **CSV-Datei** in **SpaltrNr** eine **GUID** ein. Bereits vorhandene Werte werden dabei überschrieben.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv ADDGUID 3
Fügt in der Datei **Liste.csv** in der Spalte **3** eine **GUID** ein.

Datei vorher:

	A	B	C	D
1	111	2222	12345678	44444444
2	111	2222	12345678	44444444
3	111	2222	12345678	44444444
4				

Datei nachher

	A	B	C	D
1	111	22222	56cd0de1-d1bf-4947-a9e9-9dbf2e0c17d0	44444444
2	111	22222	6ce3cb32-b259-40f2-a0ba-0fa1a680609f	44444444
3	111	22222	8065e4eb-3bb3-40de-a816-367e863005dc	44444444
4				

AddHeader

ERPAdmin -CSVKonverter <CSV-Datei> ADDHEADER 0 <CSV-Header>

Fügt in der **CSV-Datei** einen **CSV-Header** hinzu. Die einzelnen Spaltenbezeichner müssen mit einem Semikolon (**;**) getrennt sein. Dabei ist darauf zu achten, dass die Spaltenanzahl der **CSV-Datei** und die des **CSV-Header** identisch sind.

Achtung: Die Zahl „**0**“ nach dem Befehl **ADDHEADER** muss als Festwert immer mit übergeben werden.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv ADDHEADER 0 Spalte1;Spalte2;Spalte3;Spalte4
Erzeugt eine Kopfzeile mit den Werten „**Spalte1;Spalte2;Spalte3;Spalte4**“ in der Datei **Liste.csv**.

Datei vorher:

	A	B	C	D
1	111	2222	12345678	44444444
2	111	2222	12345678	44444444
3	111	2222	12345678	44444444
4				

Datei nachher

	A	B	C	D
1	Spalte1	Spalte2	Spalte3	Spalte4
2	111	2222	12345678	44444444
3	111	2222	12345678	44444444
4	111	2222	12345678	44444444

> **Tipp:**
> Wurde die Option **/Header** übergeben wird die alte Header-Zeile aus der CSV-Datei gelöscht und durch den neuen Header ersetzt.

AddString

ERPAdmin -CSVKonverter <CSV-Datei> AddString <SpalteNr> <SpalteWert> <Position> <Vergleichswert> <Vergleichsspalte> /Cond:<Bedingung>

Bearbeitet die Datensätze der Spalte **SpalteNr** und fügt an der **Position** den String **SpaltenWert** ein. Wird keine **Position** angegeben (**optional**), wird **SpaltenWert** an das Feld Ende angehängt. Um einen **SpaltenWert** am Anfang der **SpalteNr** einzufügen muss als **Position** **"1"** übergeben werden.

/Cond:<Bedingung>
Es kann eine Bedingung an **ERPAdmin** übergeben werden, wenn nicht in jedem Fall der **SpaltenWert** hinzugefügt werden soll. Folgende Bedingungen sind möglich:

- **EQ** Fügt **SpaltenWert** nur hinzu, wenn **Vergleichswert** = **Vergleichsspaltenwert**
- **NotEQ** Fügt **SpaltenWert** nur hinzu, wenn **Vergleichswert** <> **Vergleichsspaltenwert**
- **LeftEQ** Fügt **SpaltenWert** nur hinzu, wenn **die Linken x-Zeichen aus Vergleichsspalte** = **Vergleichsspaltenwert**

Wird keine Bedingung übergeben, wird der **SpaltenWert** immer hinzugefügt.

Beispiel:
Aus den Werten in der **Spalte 2** soll ein korrektes Datum erstellt werden. Aktuell sind nur Tag und Monat als Zahlen und ohne Trennzeichen hinterlegt. Dazu werden zwei **ERPAdmin** Aufrufe benötigt.

1. Aufruf

ERPAdmin -CSVKonverter Liste.csv AddString 2 ". " 3
Fügt in der Spalte **2** an der **3**. Stelle einen **Punkt** ein.

Datei vorher:

	A	B	C
1	23,50	1001	RE2010123
2	17,95	1501	RE2010168
3	21,00	2301	RE2010199

Datei nachher

	A	B	C
1	23,50	10.01	RE2010123
2	17,95	15.01	RE2010168
3	21,00	13.01	RE2010199

2. Aufruf

ERPAdmin -CSVKonverter Liste.csv AddString 2 ".2020"
Hängt in Spalte **2** die Zeichenfolge **.2020** an.

Datei vorher:

	A	B	C
1	23,50	10.01	RE2010123
2	17,95	15.01	RE2010168
3	21,00	13.01	RE2010199

Datei nachher

	A	B	C
1	23,50	10.01.2020	RE2010123
2	17,95	15.01.2020	RE2010168
3	21,00	23.01.2020	RE2010199

AutoNum

ERPAdmin -CSVKonverter <CSV-Datei> AutoNum <SpalteNr> <Startwert> <Vergleichswert> <Vergleichsspalte> /Cond:<Bedingung>

Nummeriert die Datensätze in der **CSV-Datei** in **SpalteNr** fortlaufend beginnend mit dem **Startwert**. Wenn SpalteNr die nächste Spalte nach dem letzten Datensatz bezeichnet wird diese Spalte automatisch hinzugefügt.

Optional können ein **Vergleichswert**, eine **Vergleichsspalte** und eine **Bedingung** übergeben werden. Ist dies der Fall, werden nur die Datensätze nummeriert, deren Inhalt der **Vergleichsspalte** die **Bedingung** mit **Vergleichswert** erfüllt.

Folgende **Bedingungen** sind zulässig:

- **EQ** Nummeriere **SpalteNr**, wenn **Vergleichswert=**Inhalt aus **Vergleichsspalte**
- **NotEQ** Nummeriere **SpalteNr**, wenn **Vergleichswert<>**Inhalt aus **Vergleichsspalte**
- **Instr** Nummeriere **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte vorkommt**.
- **NotInstr** Nummeriere **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte NICHT vorkommt**
- **LT** Nummeriere **SpalteNr**, wenn Wert aus **Vergleichswert** < **Vergleichsspalte**
- **LTEQ** Nummeriere **SpalteNr**, wenn Wert aus **Vergleichswert** <=**Vergleichsspalte**
- **GT** Nummeriere **SpalteNr**, wenn Wert aus **Vergleichswert**>**Vergleichsspalte**
- **GTEQ** Nummeriere **SpalteNr**, wenn Wert aus **Vergleichswert** >=**Vergleichsspalte**

Beispiel:
Adressen aus einem Alt-System sollen in büro+ überspielt werden. Dabei sollen alle Lücken in den Adressnummern aufgefüllt werden so dass ein zusammenhängender Block von Adressen entsteht. Diese sollen ab Adress-Nummer 20000 beginnen.

ERPAdmin -CSVKonverter Adressen.csv AutoNum 4 20000 /Header

Datei vorher:

	A	B	C	D
1	AdrNr-Alt	Name	PLZ/Ort	AdrNr-Neu
2	10304	Müller GmbH	63069 Offenbach	
3	17354	Maier OHG	60310 Frankfurt	
4	19234	Schulze Gbr	21030 Hamburg	
5	19765	Becker + Partner	80070 München	

Datei nachher

	A	B	C	D
1	AdrNr-Alt	Name	PLZ/Ort	AdrNr-Neu
2	10304	Müller GmbH	63069 Offenbach	20000
3	17354	Maier OHG	60310 Frankfurt	20001
4	19234	Schulze Gbr	21030 Hamburg	20002
5	19765	Becker + Partner	80070 München	20003

Calc

ERPAdmin -CSVKonverter <CSV-Datei> Calc <ErgebnisSpalte> <Spalte1> <Rechenoperation> <Spalte2>

Führt die **Rechenoperation** mit dem Feldinhalten von **Spalte1** und **Spalte2** aus und speichert das Resultat in der **ErgebnisSpalte**. Folgende Rechenoperationen werden unterstützt:

- **+** **Addition**
- **-** **Subtraktion**
- ***** **Multiplikation**
- **/** **Division**

Wird ein Feldinhalt nicht als Zahl erkannt wird der Inhalt als **0** interpretiert. Bei der Division ist darauf zu achten, dass der Wert in **Spalte2** nicht **0** sein darf.

Rechnen mit Zahlen ($xxx)

Soll eine Berechnung mit einem vorgegebenen Zahlenwert durchgeführt werden, so kann anstelle einer Spaltennummer (**Spalte1** oder **Spalte2**) auch ein Zahlenwert mit einem Vorangestelltem **$-Zeichen** übergeben werden.

Beispiele:

ERPAdmin -CSVKonverter Liste.csv Calc 1 2 + 3 /Header

Addiert die Inhalte von Spalte **2** und **Spalte3** und speichert das Ergebnis in Spalte **1.** Durch die Angabe der Option **/Header** wird die Kopfzeile bei der Verarbeitung ignoriert.

Datei vorher:

	A	B	C
1	Ergebhnis	Wert1	Wert2
2	1	10	15
3	2	25	23
4	3	147	33

Datei nachher

	A	B	C
1	Ergebhnis	Wert1	Wert2
2	25	10	15
3	48	25	23
4	180	147	33

ERPAdmin -CSVKonverter Liste.csv Calc 1 2 + $10 /Header
Addiert zu dem Wert von Spalte **2** den Wert **10** und speichert das Ergebnis in Spalte **1**

Datei vorher:

	A	B	C
1	Ergebhnis	Wert1	Wert2
2	1	10	15
3	2	25	23
4	3	147	33

Datei nachher

	A	B	C
1	Ergebhnis	Wert1	Wert2
2	20	10	15
3	35	25	23
4	157	147	33

CalcDate

ERPAdmin -CSVKonverter <CSV-Datei> CalcDate <ErgebnisSpalte> <Spalte Datum> <Rechenoperation> <Spalte Anzahl Tage> /Header

Ermittelt ein Datum durch Addition oder Subtraktion von Tagen in einer Spalte einer CSV-Datei und speichert das Resultat in der **ErgebnisSpalte**. Dabei gibt „**Spalte Datum**“ die Nummer der Spalte an in der sich das Basis-Datum befindet. In „**Spalte Anzahl Tage**“ steht ein Ganz Wert mit dem das neue Datum berechnet werden soll. **Rechenoperation** gibt an ob die Anzahl Tage a**ddiert** oder **subtrahiert** werden.

Soll die Berechnung mit einer festen Anzahl an Tagen erfolgen, kann ein Zahlenwert mit einen **$-Zeichen** übergeben werden.

Besitzt die **CSV-Datei** einen Header wird diese mit der Option **/HEADER** bei der Berechnung ignoriert.

Zulässige Rechenoperationen sind:

- \+ Addition
- \- Subtraktion

Beispiele:

ERPAdmin -CSVKonverter Mahndaten.csv CalcDate 3 3 + 4 /Header
Addiert auf die Datumswerte in der Spalte **3** die Anzahl Tage aus Spalte **4** und speichert das Ergebnis wieder in Spalte **3**.

Datei vorher:

	A	B	C	D
1	BelegNr	MahnStufe	Mahndatum	Tage
2	RE201234	0	15.08.2020	10
3	RE201235	0	20.08.2020	20
4	RE201236	0	26.08.2020	30

Datei nachher

	A	B	C	D
1	BelegNr	MahnStufe	Mahndatum	Tage
2	RE201234	0	25.08.2020	10
3	RE201235	0	30.08.2020	20
4	RE201236	0	25.09.2020	30

ERPAdmin -CSVKonverter Mahndaten.csv CalcDate 3 3 + $10 /Header
Addiert auf die Datumswerte in der Spalte **3** den Wert von **10** Tagen und speichert das Ergebnis wieder in Spalte 3 ab.

Datei vorher:

	A	B	C
1	BelegNr	MahnStufe	Mahndatum
2	RE201234	0	15.08.2020
3	RE201235	0	20.08.2020
4	RE201236	0	26.08.2020

Datei nachher

	A	B	C
1	BelegNr	MahnStufe	Mahndatum
2	RE201234	0	25.08.2020
3	RE201235	0	04.09.2020
4	RE201236	0	05.09.2020

ConCat

ERPAdmin -CSVKonverter <CSV-Datei> ConCat <ErgebnisSpalte> <Spalte-Nr-1> <SpalteNr-2> <Trennzeichen> <Prefix Spalte-Nr-1> <Sufix Spalte-Nr-2>
Verknüpft den Inhalt aus **Spalte1** und **Spalte2.**

<Trennzeichen> (Optional)
Es kann ein **Trennzeichen** übergeben werden welches zwischen den Werten **Spalte-Nr-1** und **Spalte-Nr-2** eingefügt wird.

<Prefix Spalte-Nr.1> (Optional)
Es kann eine Zeichenkette **Prefix Spalte-Nr-1** vor dem Wert **Spalte-Nr-1** eingefügt werden**.**

<Suffix Spalte-Nr 2> (Optional)
Es kann eine Zeichenkette **Suffix Spalte-Nr-2** nach dem Wert **Spalte-Nr-2** eingefügt werden.

Ist einer der beiden Werte aus **Spalte-Nr-1** oder **Spalte-Nr-2** leer dann wird das **Trennzeichen**, das **Prefix** und das **Sufix** weggelassen. Sind beide Werte leer wird ein Leerstring in **ErgebnisSpalte** geschrieben ohne **Trennzeichen**.

Sollen mehrere Spalten miteinander verknüpft werden können diese in den Parametern **Spalte-Nr-1** und **Spalte-Nr-2** mit einem Semikolon getrennt übergeben werden. Die Spaltenwerte aus **Spalte-Nr-1** werden direkt aneinander gehängt ohne Trennzeichen. Die Werte aus **Spalte-Nr-2** werden jeweils mit dem **Trennzeichen** verbunden sofern ein **Trennzeichen** übergeben wurde.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv ConCat 3 1 2

Verknüpft den Inhalt der Spalte 1 und 2 und speichert das Ergebnis in Spalte 3 ab.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	11122222	44444444
2	111	22222	11122222	44444444
3	111	22222	11122222	44444444

ERPAdmin -CSVKonverter Liste.csv ConCat 3 1 2 * # @

Verknüpft den Inhalt von Spalte **1** und **2** und fügt zwischen den Spalten-Werten * ein sowie davor **#** und danach **@** ein und speichert das Ergebnis in Spalte **3**.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher:

	A	B	C	D
1	111	22222	#111*22222@	4444444444
2	111	22222	#111*22222@	4444444444
3	111	22222	#111*22222@	4444444444

ERPAdmin -CSVKonverter Liste.csv ConCat 2 2 1 " "

Verknüpft den Inhalt der Spalten **2** und **1** mit einem **Leerzeichen** getrennt und speichert das Ergebnis wieder in Spalte **2** ab.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222 111	12345678	44444444
2	111	22222 111	12345678	44444444
3	111	22222 111	12345678	44444444

ERPAdmin -CSVKonverter Liste.csv ConCat 1 "2;3" "4;5;6" "/"

Verknüpft den Inhalt der Spalten **2** und **3** ohne Trennzeichen mit den Werten aus den Spalten **4, 5** und **6** welche jeweils mit einem **/** getrennt sind. Das Ergebnis wird in Spalte **1** gespeichert.

Datei vorher:

	A	B	C	D	E	F
1	111	222	333	444	555	666
2	111	222	333	444	555	666
3	111	222	333	444	555	666

Datei nachher

	A	B	C	D	E	F
1	111	222333/444/555/666	333	444	555	666
2	111	222333/444/555/666	333	444	555	666
3	111	222333/444/555/666	333	444	555	666

Copy

ERPAdmin -CSVKonverter <CSV-Datei> COPY <QuellSpalte> <ZielSpalte><Vergleichswert> <VergleichssSpalte> /Cond:<Bedingung>

Kopiert den Inhalt der Spalte **Quellspalte** nach **Zielspalte.** Die Inhalte in der **Zielspalte** werden dabei überschrieben. Sollen die Inhalte einer Zielspalte erhalten bleiben, fügen Sie zuerst eine zusätzliche Spalte mit dem Befehl **AddCol** hinzu in welche der Wert kopiert wird.

Optional kann eine Bedingung **/Cond:<Bedingung>**, eine **Vergleichsspalte** und ein **Vergleichswert** angegeben werden. Ist dies der Fall, wird der Wert aus der **Vergleichsspalte** mit dem **Vergleichswert** verglichen und nur wenn die **Bedingung** erfüllt ist, wird der Wert aus der **QuellSpalte** in die **ZielSpalte** kopiert. Wird keine **Bedingung** angegeben, wird der Kopier-Befehl immer ausgeführt.

Folgende **Bedingungen** sind für das Kopieren eines Feldinhaltes zulässig:

- **EQ** Kopiert Inhalt **SpalteNr**, wenn **Vergleichswert=**Inhalt aus **Vergleichsspalte**
- **NotEQ** Kopiert Inhalt **SpalteNr**, wenn **Vergleichswert<>**Inhalt aus **Vergleichsspalte**
- **Instr** Kopiert Inhalt **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte vorkommt**.
- **NotInstr** Kopiert Inhalt **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte NICHT vorkommt**
- **LT** Kopiert Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert <Vergleichsspalte**
- **LTEQ** Kopiert Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert <=Vergleichsspalte**
- **GT** Kopiert Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert >Vergleichsspalte**
- **GTEQ** Kopiert Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert >=Vergleichsspalte**

Beispiel:
ERPAdmin -CSVKonverter Liste.csv COPY 1 4
Kopiert den Inhalt aus Spalte **1** in die Spalte **4**.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	12345678	111
2	111	22222	12345678	111
3	111	22222	12345678	111

ERPAdmin -CSVKonverter Liste.csv COPY 1 2 Kunde 3 /Cond:EQ
Kopiert den Inhalt aus Spalte **1** in die Spalte **2**, wenn der Wert aus Spalte **3** gleich (**EQ**) „**Kunde**“ ist.

Datei vorher:

	A	B	C	D
1	111	22222	Kunde	4444444
2	111	22222	Lieferant	4444444
3	111	22222	Kunde	4444444

Datei nachher

	A	B	C	D
1	111	111	Kunde	4444444
2	111	22222	Lieferant	4444444
3	111	111	Kunde	4444444

CopyMid

ERPAdmin -CSVKonverter <CSV-Datei> COPYMID <QuellSpalte> <ZielSpalte> <Position> <Anzahl>

Kopiert die **Anzahl** an Zeichen ab **Position** aus der **Quellspalte** nach **Zielspalte.** Die Inhalte in der **<Zielspalte>** werden dabei überschrieben.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv CopyMID 3 4 2 4
Kopieret den Inhalt aus Spalte **3** in Spalte **4**. Dabei werden nur **4**-Zeichen ab Position **2** übernommen.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	12345678
2	111	22222	12345678	12345678
3	111	22222	12345678	12345678

Datei nachher

	A	B	C	D
1	111	22222	12345678	2345
2	111	22222	12345678	2345
3	111	22222	12345678	2345

CrossCopy

ERPAdmin -CSVKonverter <CSV-Datei> CrossCopy <VonSpalte> <BisSpalte> <AnzahlSpalten>

Stehen mehrere Daten in einer Zeile obwohl sich diese jeweils in einem eigenen Datensatz befinden sollten, können diese umformatiert werden.

Beispiel:
Von einem Kunden bekommen Sie eine CSV-Datei mit Bestelldaten für die kommenden drei Monate (Januar bis März). Pro Artikel befinden sich aber die Bestellmengen und das Lieferdatum in einer Zeile.

ArtNr	Jan-Mge	Jan-LiefDat	Feb-Mge	Feb-LiefDat	Mar-Mge	Mar-LiefDat
1157A	7	05.01.2024	8	05.02.2024	9	07.03.2024
1499C	3	05.01.2024	4	05.02.2024	5	07.03.2024
2316T	4	05.01.2024	5	05.02.2024	6	07.03.2024

Benötigt werden die Daten ab im folgenden Format:

ArtNr	Menge	LiefDat
1157A	7	05.01.2024
1499C	3	05.01.2024
2316T	4	05.01.2024
1157A	8	05.02.2024
1499C	4	05.02.2024
2316T	5	05.02.2024
1157A	9	07.03.2024
1499C	5	07.03.2024
2316T	6	07.03.2024

Mit dem Befehl **CrossCopy** kann dies wie folgt hergestellt werden. Von der **Spalte 2** bis zur **Spalte 7** sollen die Daten jeweils als **2**'er Paar umkopiert werden.

ERPAdmin -CSVKonverter Bestellung.csv CrossCopy 2 7 2

Eine Umbenennung der Kopfzeile muss in einem separaten Befehl erfolgen.

CutLeft

ERPAdmin -CSVKonverter <CSV-Datei> CutLeft <SpalteNr> <Anzahl Zeichen>

Entfernt aus **SpalteNr** die linken **Anzahl Zeichen**.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv CutLeft 3 3

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	45678	444444444
2	111	22222	45678	444444444
3	111	22222	45678	444444444

CutLeftTo

ERPAdmin -CSVKonverter <CSV-Datei> CutLeftTo <SpalteNr> <ZielSpalte> <Zeichenkette>

Sucht in **SpalteNr** von **links** das erste Vorkommen der **Zeichenkette** und löscht alle Zeichen bis zum Ende dieser **Zeichenkette**. Das Ergebnis wird in **Zielspalte** abgespeichert.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv CutLeftTo 1 1 "/retailsuite/"
Datei vorher:

	A	B	C
1	https://www.Systemhaus-horn.de/retailsuite/1000-01.jpg	22222	3333333
2	https://www.Systemhaus-horn.de/retailsuite/1000-02.jpg	22222	3333333
3	https://www.Systemhaus-horn.de/retailsuite/1000-03.jpg	22222	3333333

Datei nachher

	A	B	C
1	1000-01.jpg	22222	3333333
2	1000-02.jpg	22222	3333333
3	1000-03.jpg	22222	3333333

CutMid

ERPAdmin -CSVKonverter <CSV-Datei> CUTMID <SpalteNr> <Start Position> <Anzahl Zeichen>

Entfernt aus **SpalteNr** die **Anzahl Zeichen** beginnen ab Zeichen **Start Position**.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv CUTMID 3 5 2

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	123478	44444444
2	111	22222	123478	44444444
3	111	22222	123478	44444444

CutRight

ERPAdmin -CSVKonverter <CSV-Datei> CutRight <SpalteNr> <Position>

Entfernt aus **SpalteNr** alle Zeichen die rechts stehen von **Position**.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv CUTRIGHT 3 5

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	12345	444444444
2	111	22222	12345	444444444
3	111	22222	12345	444444444

CutRightTo

ERPAdmin -CSVKonverter <CSV-Datei> CutRightTo <SpalteNr> <Zielspalte> <Zeichenkette>

Sucht in **SpalteNr** von **rechts** das erste Vorkommen der **Zeichenkette** und löscht alle Zeichen Rechts von der **Zeichenkette** inkl. **Zeichenkette**. Das Ergebnis wird in **Zielspalte** abgespeichert.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv CutRightTo 1 1 "/"
Entfernt in Spalte **1** alle Zeichen rechts vom ersten Forslash (**/**) und speichert das Ergebnis wieder in Spalte **1**.

Datei vorher:

	A	B	C
1	www.Systemhaus-horn.de/1000-01.jpg	2222	333333
2	www.Systemhaus-horn.de/1000-02.jpg	2222	333333
3	www.Systemhaus-horn.de/1000-03.jpg	2222	333333

Datei nachher

	A	B	C
1	www.Systemhaus-horn.de	2222	333333
2	www.Systemhaus-horn.de	2222	333333
3	www.Systemhaus-horn.de	2222	333333

DBLookUp

ERPAdmin -CSVKonverter <CSV-Datei> DBLookUp <ErgebnisSpalte> <Tabelle> <SuchIndex> <SuchIwert-SpalteNr> <büro+-Feldname>

Liest Daten aus der büro+ **Tabelle** aus dem Feld **büro+-Feldname** und schreibt den Wert in **ErgebnisSpalte**. Dazu wird der **SuchIndex** der Tabelle verwendet. Als Suchbegriff wird der Wert aus der Spalte **Suchwert-SpaltenNr** verwendet.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv DBLookUp 2 Artikel ArtNr 1 Sel1
Der Wert **Artikel-Selektion1 (Farbe)** soll in Spalte **2 (B)** geschrieben werden. Als Suchindex für die Artikel wird der Suchindex **ArtNr** verwendet. Die Artikelnummern werden aus **Spalte 1 (A)** gelesen.

Datei vorher:

	A	B	C	D
1	A1000	11	22222	3333333
2	B1000	11	22222	3333333
3	C1000	11	22222	3333333

Datei nachher:

	A	B	C	D
1	A1000	rot	22222	3333333
2	B1000	blau	22222	3333333
3	C1000	gelb	22222	3333333

Delete

ERPAdmin -CSVKonverter <CSV-Datei> Delete <SpalteNr> <Vergleichswert> <Vergleichsspalte> /Cond:<Bedingung>

Lösche den Inhalt aus den Feldern von **SpalteNr** in der **CSV-Datei**.

Optional kann eine Bedingung **/Cond:<Bedingung>**, eine **Vergleichsspalte** und ein **Vergleichswert** angegeben werden. Ist dies der Fall, wird der Wert aus der **Vergleichsspalte** mit dem **Vergleichswert** verglichen und nur wenn die **Bedingung** erfüllt ist, wird der Wert aus der **SpalteNr** gelöscht. Wird keine **Bedingung** angegeben, werden alle Werte gelöscht.

Folgende **Bedingungen** sind für das Löschen eines Feldinhaltes zulässig:

- **EQ** Löscht Inhalt **SpalteNr**, wenn **Vergleichswert =** Inhalt aus **Vergleichsspalte**
- **NotEQ** Löscht Inhalt **SpalteNr**, wenn **Vergleichswert<>**Inhalt aus **Vergleichsspalte**
- **LeftEQ** Löscht Inhalt **SpalteNr**, wenn die linken x-Zeichen (Länge **Vergleichswert**) mit dem Inhalt aus **Vergleichsspalte übereinstimmen**
- **NotLeftEQ** Löscht Inhalt **SpalteNr**, wenn die linken x-Zeichen (Länge **Vergleichswert**) mit dem Inhalt aus **Vergleichsspalte** mit **NICHT übereinstimmen**
- **RightEQ** Löscht Inhalt **SpalteNr**, wenn die rechten x-Zeichen (Länge **Vergleichswert**) mit dem Inhalt aus **Vergleichsspalte übereinstimmen**
- **NotRightEQ** Löscht Inhalt **SpalteNr**, wenn die rechten x-Zeichen (Länge **Vergleichswert**) mit dem Inhalt aus **Vergleichsspalte NICHT übereinstimmen**
- **Instr** Löscht Inhalt **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte vorkommt**.
- **NotInstr** Löscht Inhalt **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte NICHT vorkommt.**
- **IsDate*** Löscht Inhalt **SpalteNr**, wenn der Wert in **Vergleichsspalte ein Datum ist**

- **NotIsDate*** Löscht Inhalt **SpalteNr**, wenn der Wert in **Vergleichsspalte kein Datum ist**
- **IsEmail*** Löscht Inhalt **SpalteNr**, wenn der Wert in **Vergleichsspalte eine Email-Adresse ist**
- **NotIsEmail*** Löscht Inhalt **SpalteNr**, wenn der Wert in **Vergleichsspalte keine Email-Adresse ist**
- **LengthEQ*** Löscht Inhalt **SpalteNr**, wenn die Länge des **Vergleichswertes gleich** dem Inhalt in **Vergleichsspalte** ist-
- **NotLengthEQ*** Löscht Inhalt **SpalteNr**, wenn die Länge des **Vergleichswertes NICHT gleich** dem Inhalt in **Vergleichsspalte** ist
- **IsNumeric*** Löscht Inhalt **SpalteNr**, wenn die der Inhalt aus **Vergleichsspalte eine Zahl ist**
- **NotIsNumeric*** Löscht Inhalt **SpalteNr**, wenn die der Inhalt aus **Vergleichsspalte keine Zahl ist**

Achtung:
* Bei diesen Bedingungen wird ein übergebener **Vergleichswert** ignoriert. Soll die **Vergleichsspalte** abweichend von **SpalteNr** sein, muss ein fiktiver **Vergleichswert** übergeben werden.

Beispiele:

ERPAdmin -CSVKonverter Liste.csv Delete 2 ABC 2

Datei vorher:

	A	B	C	D
1	111	ABC	ABCDEFG	01.09.2021
2	222	DEF	ABCHIKLM	09. Jan 21
3	333	GHI	LMNOPQR	2021.09.01

Datei nachher

	A	B	C	D
1	111		ABCDEFG	01.09.2021
2	222	DEF	ABCHIKLM	09. Jan 21
3	333	GHI	LMNOPQR	2021.09.01

ERPAdmin -CSVKonverter Liste.csv Delete 3 ABC 3 /Cond:LeftEQ

Datei vorher:

	A	B	C	D
1	111	ABC	ABCDEFG	01.09.2021
2	222	DEF	ABCHIKLM	09. Jan 21
3	333	GHI	LMNOPQR	2021.09.01

Datei nachher

	A	B	C	D
1	111	ABC		01.09.2021
2	222	DEF		09. Jan 21
3	333	GHI	LMNOPQR	2021.09.01

ERPAdmin -CSVKonverter Liste.csv Delete 4 /Cond:NotIsDate

Datei vorher:

	A	B	C	D
1	111	ABC	ABCDEFG	01.09.2021
2	222	DEF	ABCHIKLM	09. Jan 21
3	333	GHI	LMNOPQR	2021.09.01

Datei nachher

	A	B	C	D
1	111	ABC	ABCDEFG	01.09.2021
2	222	DEF	ABCHIKLM	
3	333	GHI	LMNOPQR	

DeleteCol

ERPAdmin -CSVKonverter <CSV-Datei> DeleteCol <SpalteNr>

Lösche **SpalteNr** aus der **CSV-Datei.** Soll die letzte Spalte der Daten gelöscht werden kann auch der Wert **@Z** übergeben werden.

Sollen mehrere Spalten gelöscht werden, müssen die einzelnen **SpaltenNr** in aufsteigender Reihenfolge und mit einem Semikolon getrennt übergeben werden.

Besitzt die **CSV-Datei** eine Kopfzeile, werden die entsprechenden Spalten-Köpfe automatisch mit gelöscht.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv DeleteCol 3
Löscht Spalte 3 aus der Datei.

Datei vorher:

	A	B	C	D
1	111	22222	3333333	444444444
2	111	22222	3333333	444444444
3	111	22222	3333333	444444444

Datei nachher

	A	B	C
1	111	22222	444444444
2	111	22222	444444444
3	111	22222	444444444

Beispiel:
ERPAdmin -CSVKonverter Liste.csv DeleteCol 2;4
Löscht Spalte 2 und 4 aus der Datei.

Datei vorher:

	A	B	C	D
1	111	22222	3333333	444444444
2	111	22222	3333333	444444444
3	111	22222	3333333	444444444

Datei nachher

	A	B
1	111	3333333
2	111	3333333
3	111	3333333

DeleteColByName

ERPAdmin -CSVKonverter <CSV-Datei> DeleteColbyName 0 <Spaltenname> /Header

Löscht in einer **CSV-Datei** mit Kopfzeile alle Spalten deren Kopfzeile **Spaltenname** heißt. Mehrere Spaltennamen müssen mit Semikolon getrennt werden.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv DeleteColbyName 0 DUMMY;EkoNr /Header
Löscht alle Spalten mit der Kopfzeile „**DUMMY**“ und „**EkoNr**“ aus **Liste.csv**.

Datei vorher:

	A	B	C	D	E
1	DUMMY	ArtNr	DUMMY	EkoNr	WgrNr
2	111	10000	0	AB123	Hardware
3	111	12500	17	ECC123	Software
4	111	12645	12	ADD	Software

Datei nachher

	A	B
1	ArtNr	WgrNr
2	10000	Hardware
3	12500	Software
4	12645	Software

DeleteDoubleSpaces

ERPAdmin -CSVKonverter <CSV-Datei> DeleteDoubleSpaces <SpalteNr>

ERPAdmin entfernt aus **SpalteNr** alle mehrfach hintereinander vorkommenden Leerzeichen. Es bleibt somit nicht mehr als ein Leerzeichen in Folge stehen. Sollen mehrere Spalten in einem Aufruf verarbeitet werden, müssen die einzelnen Spalten mit einem Semikolon getrennt übergeben werden. Zusätzlich werden Leerzeichen am Feld-Anfang und am -Ende entfernt.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv > DeleteDoubleSpaces 2;3>
Aus der Spalte 2 und 3 sollen alle mehrfach hintereinander stehen Leerzeichen entfernt werden so dass nicht mehrere Leerzeichen direkt hintereinanderstehen.

Datei vorher:

	A	B	C	D
1	111	22 22 22	33 33	4444444
2	111	22 22 22	33 33	4444444
3	111	22 22 22	33 33	4444444

Datei nachher

	A	B	C	D
1	111	22 22 22	33 33	4444444
2	111	22 22 22	33 33	4444444
3	111	22 22 22	33 33	4444444

DeleteDubRow

ERPAdmin -CSVKonverter <CSV-Datei> DeleteDubRow <SpalteNr>
ERPAdmin entfernt doppelte Datensätze zeilenweise aus der **CSV-Datei**. Dabei werden die Werte in **SpalteNr** überprüft und doppelte Datensätze gelöscht. Es bleibt immer nur der zuerst gefundene Datensatz in der **CSV-Datei** erhalten.

Sortierung nach SpalteNr
Die Reihenfolge/Sortierung der **CSV-Datei** kann mit dem folgenden **ERPAdmin -**Befehl verändert werden und somit Einfluss darauf genommen werden welche Datensätze erhalten bleiben:

ERPAdmin -CSVKonverter Sort ….

Beispiel:
ERPAdmin -CSVKonverter Liste.csv DeleteDubRow 2
Löscht alle Datensätze/Zeilen aus der Datei **Liste.csv** wenn in Spalte **2** Werte mehrfach vorkommen. Es wird immer nur die Zeile mit dem ersten Auftreten eines Wertes behalten.

Datei vorher:

	A	B	C
1	111	11111	AAA
2	222	22222	BBB
3	333	11111	CCC
4	444	22222	DDD
5	555	33333	EEE

Datei nachher

	A	B	C
1	111	11111	AAA
2	222	22222	BBB
3	555	33333	EEE

Hinweis:
Enthält ein Feld keinen Wert (Leerstring) bleibt auch ein Datensatz mit diesem Leerstring erhalten. Wenn das nicht gewünscht ist, müssen zuvor alle Datensätze die in der Spalte einen Leerwert haben gelöscht werden. (**ERPAdmin -CSVKonverter <Datei> DeleteRow…)**

DeleteHeader

ERPAdmin -CSVKonverter <CSV-Datei> DeleteHeader <X-Zeilen>

ERPAdmin entfernt die erste Zeile (Kopfzeile) aus der **CSV-Datei**.

Optional können die ersten **X-Zeilen** der CSV-Date gelöscht werden. Wird kein Wert für **X-Zeilen** angegeben wird immer nur die 1. Zeile gelöscht.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv DeleteHeader
Löscht die erste Zeile der Datei **Liste.csv.**

Datei vorher:

	A	B	C
1	**AdrNr**	**Name1**	**Plz**
2	10000	Schulz GmbH	10000
3	20000	Maier OHG	63069
4	30000	Müller AG	74500

Datei nachher

	A	B	C
1	10000	Schulz GmbH	10000
2	20000	Maier OHG	63069
3	30000	Müller AG	74500

DeleteRow

ERPAdmin -CSVKonverter <CSV-Datei> DELETEROW <SpalteNr> <Vergleichswert> /Cond:<Bedingung> /Header

Entfernt alle Datenzeilen aus der **CSV-Datei** bei denen in der **SpalteNr** der **Vergleichswert** steht.

Optional kann eine **Bedingung** angegeben werden welche beim Löschen beachtet wird. Wird keine Bedingung angegeben wird immer die Bedingung **/Cond:EQ** (**Gleichheit**) verwendet (Default Bedingung).

Folgende **Bedingungen** für das Löschen eines Datensatzes sind zulässig:

- **EQ** Löscht, wenn der Wert in **SpalteNr =Vergleichswert**
- **NotEQ** Löscht, wenn der Wert in **SpalteNr <>Vergleichswert**
- **LeftEQ** Löscht, wenn die linken x-Zeichen (Länge **Vergleichswert**) aus **SpalteNr** mit dem **Vergleichswert übereinstimmen**
- **NotLeftEQ** Löscht, wenn die linken x-Zeichen (Länge **Vergleichswert**) aus **SpalteNr** mit dem **Vergleichswert nicht übereinstimmen**
- **RightEQ** Löscht, wenn die rechten x-Zeichen (Länge **Vergleichswert**) aus **SpalteNr** mit dem **Vergleichswert übereinstimmen**
- **NotRightEQ** Löscht, wenn die rechten x-Zeichen (Länge **Vergleichswert**) aus **SpalteNr** mit dem **Vergleichswert nicht übereinstimmen**
- **Instr** Löscht, wenn der Wert in **SpalteNr** im **Vergleichswert enthalten ist**
- **NotInstr** Löscht, wenn der Wert in **SpalteNr** im **Vergleichswert nicht enthalten ist**
- **IsDate** Löscht, wenn der Wert in **SpalteNr** ein Datum im Format **tt.mm.jjjj** ist
- **NotIsDate** Löscht, wenn ob der Wert in **SpalteNr** kein Datum im Format **tt.mm.jjjj** ist

- **IsEmail** Löscht, wenn der Wert in **SpalteNr** dem Format einer **E-Mail entspricht**
- **NotIsEmail** Löscht, wenn ob der Wert in **SpalteNr** dem Format einer **E-Mail nicht entspricht**
- **LengthEQ** Löscht, wenn die Anzahl der Zeichen in **SpalteNr** dem **Vergleichswert entspricht**
- **NotLengthEQ** Löscht, wenn die Anzahl der Zeichen in **SpalteNr** dem **Vergleichswert nicht entspricht**
- **IsNumeric** Löscht, wenn der Wert in **SpalteNr eine Zahl ist**
- **NotIsNumeric** Löscht, wenn der Wert in **SpalteNr keine Zahl ist**
- **IsEmpty** Löscht, wenn in **SpalteNr keine Daten** enthalten sind (leer)
- **NotIsEmpty** Löscht, wenn in **SpalteNr Daten enthalten sind** (nicht leer)
- **GT** Löscht, wenn der Wert in **SpalteNr** > **Vergleichswert**
- **GTEQ** Löscht, wenn der Wert in **SpalteNr** >= **Vergleichswert**
- **LT** Löscht, wenn der Wert in **SpalteNr** < **Vergleichswert**
- **LTEQ** Löscht, wenn der Wert in **SpalteNr** <= **Vergleichswert**
- **IsPhone** Löscht, wenn der Wert in **SpalteNr** einer **Telefonnummer entspricht**
- **NotIsPhone** Löscht, wenn der Wert in **SpalteNr** einer **Telefonnummer nicht entspricht**

> **Hinweis:**
> Bei allen Vergleichen wird **keine** Unterscheidung nach Groß-/Kleinschreibung vorgenommen.

Vergleichsbedingungen für Datums-Werte:

- **LTDate** Löscht, wenn das Datum in **SpalteNr** < Datum **Vergleichswert**
- **LTEQDate** Löscht, wenn das Datum in **SpalteNr** <= Datum **Vergleichswert**
- **GTDate** Löscht, wenn das Datum in **SpalteNr** > Datum **Vergleichswert**
- **GTEQDate** Löscht, wenn das Datum in **SpalteNr** >= Datum **Vergleichswert**

Sonderfall /Cond:NotEQ

Bei der Bedingung **/Cond:NotEQ** können mehrere Vergleichswerte durch Semikolon getrennt übergeben werden. Bei allen anderen Bedingungen ist das nicht möglich.

Sonderfall „@Z“

Soll der letzte Datensatz aus der **CSV-Datei** gelöscht werden kann dies durch Übergabe des Wertes **„@Z“** erfolgen. In diesem Fall werden alle weiteren Parameter ignoriert.

Beispiel: **ERPAdmin -CSVKonverter <CSV-Datei> DELETEROW @Z**

Vergleich mit einem Wert aus einer anderen Spalte

Soll der Wert aus **SpalteNr** mit einem Wert aus einer anderen Spalte der **CSV-Datei** verglichen werden, muss als **Vergleichswert** die Spalten-Nummer mit einen führenden **$-Zeichen** übergeben werden.

Soll z.B. der Vergleich mit dem Wert aus **Spalte 4** erfolgen, muss als Vergleichswert **$4** übergeben werden.

Beispiel: **ERPAdmin -CSVKonverter <CSV-Datei> DELETEROW 2 $4**

Weitere Beispiele:

ERPAdmin -CSVKonverter Liste.csv DELETEROW 2 22222

Löscht alle Zeilen die in Spalte **2** den Wert **22222** haben.

Datei vorher:

	A	B	C
1	111	11111	11111111
2	222	22222	2222222
3	333	33333	3333333

Datei nachher

	A	B	C
1	111	11111	11111111
2	333	33333	3333333

Hinweis:
Wird kein **Vergleichswert** angegeben werden alle Zeilen/Datensätze gelöscht die in **SpalteNr** keine Daten enthalten.

ERPAdmin -CSVKonverter Liste.csv DELETEROW 2 22222 /Cond:NotEQ

Löscht alle Zeilen welche in Spalte **2** **nicht** den Wert **22222** stehen haben.

Datei vorher:

	A	B	C
1	111	11111	11111111
2	222	22222	2222222
3	333	33333	3333333

Datei nachher

	A	B	C
1	222	22222	2222222

ERPAdmin -CSVKonverter Liste.csv DELETEROW 2 11 /Cond:LeftEQ

Lösche alle Zeilen die in Spalte **2** mit dem Wert **11** beginnen.

Datei vorher:

	A	B	C
1	111	11111	11111111
2	222	22222	2222222
3	333	33333	3333333

Datei nachher

	A	B	C
1	222	22222	2222222
2	333	33333	3333333

FileMapping

ERPAdmin -CSVKonverter <CSV-Datei> FileMapping <SpalteNr> <Mappingdatei> <Suchspalte> <Ersetzungsspalte>

Führt in der Spalte **SpalteNr** der **CSV-Datei** eine Ersetzung durch mit dem Inhalt aus der **Ersetzungsspalte** der **Mappingdatei**. Bei der **Mappingdatei** muss es sich um eine ANSI-CSV-Datei handelt mit mindestens zwei Spalten.

<Suchspalte>, <Ersetzungsspalte> **(optional)**
Suchspalte und **Ersetzungsspalte** sind Spaltennummern aus der **Mappingdatei** nach denen gesucht bzw. dessen Werte zur Ersetzung benutzt werden sollen. Werden keine Werte überbergeben werden die Werte **Suchspalte=1** und **Ersetzungsspalte=2** verwendet.

Beim Mapping werden die Werte der **SpalteNr** mit dem Inhalt der **Suchspalte** der **Mappingdatei** verglichen. Bei Übereinstimmung wird der Inhalt aus **Ersetzungsspalte** in die **Zielspalte** der **CSV-Datei** übernommen.

Damit ist es sehr einfach möglich, mehrere Suchen und Ersetzten Aufrufe durch einen Aufruf und eine passende **Mappingdatei** zu ersetzten. Die Funktionalität entspricht in abgewandelter Form dem Excel-Befehl „**SVerweis**“

Beispiel:

ERPAdmin -CSVKonverter Liste.csv FileMapping 1 Mappingdatei.csv

Datei vorher:

	A	B	C
1	111	11111	11111111
2	222	22222	2222222
3	333	33333	3333333

Mappingdatei

	A	B
1	111	AAA
2	222	BBB
3	333	CCC

Datei nachher

	A	B	C
1	AAA	11111	11111111
2	BBB	22222	2222222
3	CCC	33333	3333333

FormatDate

ERPAdmin -CSVKonverter <CSV-Datei> FormatDate <SpalteNr> <Quellformat> <Zielformat>

Formatiert die Spalte **SpalteNr** der **CSV-Datei** als Datum im **Zielformat**.

<Quellformat>
Quellformat gibt den Aufbau des Feldes in der **CSV-Datei** wieder, wobei **DD**=Tag, **MM**=Monat und **YYYY**=Jahr deren Positionen wiedergeben. Im **Quellaufbau** dürfen die Zeichenfolgen **YYYY**, **MM** und **DD** nur einmal vorkommen. Als Trennzeichen dürfen beliebige Zeichen verwendet werden.

<Zielformat> **(optional)**
Gibt den Aufbau des Zielformates an. Hier können die folgenden drei Platzhalter **DD, MM, YYYY** verwendet werden oder **KW** für die Umwandlung des Datums in eine **Kalenderwoche**. Wird kein Zielformat übergeben wird immer der Aufbau **DD.MM.YYYY** verwendet.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv FormatDate 2 YYYY-MM-DD

Formatiert die Werte in Spalte **2** als Datum. Die Werte liegen im Format **YYYY-MM-DD** vor.

Datei vorher:

	A	B	C
1	111	2019-05-23T02:23.00	33333
2	111	2019-02-11T15:30.00	33333
3	111	2019-01-04T22:19.12	33333

Datei nachher

	A	B	C
1	111	23.05.2019	33333
2	111	11.02.2019	33333
3	111	04.01.2019	33333

GetFilename

ERPAdmin -CSVKonverter <CSV-Datei> GetFilename <SpalteNr> <ZielSpalte>

Sucht in **CSV-Datei** in **SpalteNr** nach dem ersten „****“-Zeichen von **rechts** und separiert alle Zeichen rechts davon als Dateinamen. Das Ergebnis wird in **ZielSpalte** gespeichert.

ERPAdmin -CSVKonverter Liste.csv GetFilename 2 4

Datei vorher:

	A	B	C	D
1	111	c:\Bilder\10234.jpg	333333	
2	111	c:\Bilder\Shop\14235667.01.jpg	333333	
3	111	c:\Bilder\Shop\DE\13244.jpg	333333	

Datei nachher

	A	B	C	D
1	111	c:\Bilder\10234.jpg	333333	10234.jpg
2	111	c:\Bilder\Shop\14235667.01.jpg	333333	14235667.01.jpg
3	111	c:\Bilder\Shop\DE\13244.jpg	333333	13244.jpg

GetPath

ERPAdmin -CSVKonverter <CSV-Datei> GetPath <SpalteNr> <ZielSpalte>

Sucht in **CSV-Datei** in **SpalteNr** nach dem ersten „****“-Zeichen von **rechts** und schneidet alle Zeichen rechts davon weg. Das Ergebnis wird in **ZielSpalte** gespeichert.

ERPAdmin -CSVKonverter Liste.csv GetPath 2 4

Datei vorher:

	A	B	C	D
1	111	c:\Bilder\10234.jpg	333333	
2	111	c:\Bilder\Shop\14235667.01.jpg	333333	
3	111	c:\Bilder\Shop\DE\13244.jpg	333333	

Datei nachher

	A	B	C	D
1	111	c:\Bilder\10234.jpg	333333	c:\Bilder\
2	111	c:\Bilder\Shop\14235667.01.jpg	333333	c:\Bilder\Shop\
3	111	c:\Bilder\Shop\DE\13244.jpg	333333	c:\Bilder\Shop\DE\

ISOLand2UstKat

ERPAdmin -CSVKonverter <CSV-Datei> ISOLand2UstKat <SpalteNr> <ISO-Spalte-Nr>
Entscheidet anhand des Länder-ISO-Code aus Spalte **ISO-Spalte-NR,** ob es sich bei dem Datensatz um die büro+-Umsatzsteuerkategorie „**1 Inland**“, „**2 Ausland**“ oder „**3 Ausland EU**“ handelt und speichert die entsprechenden Werte in **SpalteNr**. Wird kein Wert gefunden bricht **ERPAdmin** mit einem Fehler ab. Entspricht ein Wert aus der **ISO-Spalte-Nr.** keinem ISO-Länder-Code aus der EU wird immer „**2**“ (Ausland) zurückgegeben.

Es entspricht DE=Deutschland, AT=Österreich und CH=Schweiz.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv ISOLand2UstKat 2 2
Wandelt die ISO-Länderkennzeichen in Spalte **2** in eine büro+-Umsatzsteuerkategorie um.

Datei vorher:

	A	B	C	D
1	111	DE	33333	4444444
2	111	AT	33333	4444444
3	111	CH	33333	4444444

Datei nachher

	A	B	C	D
1	111	1	33333	4444444
2	111	3	33333	4444444
3	111	2	33333	4444444

ISOLandNr2UstKat

ERPAdmin -CSVKonverter <CSV-Datei> ISOLandNr2UstKat <SpalteNr> <ISO-Spalte-Nr>
Entscheidet anhand der Länder-ISO-Code-Nr. aus **ISO-Spalte-NR,** ob es sich bei dem Datensatz um die büro+-Umsatzsteuerkategorie „**1 Inland**“, „**2 Ausland**“ oder „**3 Ausland EU**“ handelt und speichert die entsprechenden Werte in **SpalteNr** ab. Wird kein Wert gefunden bricht **ERPAdmin** mit einem Fehler ab. Entspricht ein Wert aus der **ISO-Spalte-Nr.** keinem ISO-Länder-Code aus der EU wird immer „**2**“ (Ausland) zurückgegeben.

Es entspricht 276=Deutschland, 40=Österreich und 756=Schweiz

Beispiel:

ERPAdmin -CSVKonverter Liste.csv ISOLandNr2UstKat 2 2
Wandelt die ISO-Länder-Nummer in Spalte **2** in eine büro+-Umsatzsteuerkategorie um.

Datei vorher:

	A	B	C	D
1	111	276	33333	4444444
2	111	40	33333	4444444
3	111	756	33333	4444444

Datei nachher

	A	B	C	D
1	111	1	33333	4444444
2	111	3	33333	4444444
3	111	2	33333	4444444

Move

ERPAdmin -CSVKonverter <CSV-Datei> Move <QuellSpalte> <ZielSpalte> <Vergleichswert> <Vergleichsspalte> /Cond:<Bedingung>

Verschiebt den Inhalt der Spalte **Quellspalte** nach **Zielspalte.** Die Inhalte in der **Zielspalte** werden dabei überschrieben.

Optional kann eine Bedingung **/Cond:<Bedingung>**, eine **Vergleichsspalte** und ein **Vergleichswert** angegeben werden. Ist dies der Fall, wird der Wert aus der **Vergleichsspalte** mit dem **Vergleichswert** verglichen und nur wenn die **Bedingung** erfüllt ist, wird der Wert aus der **QuellSpalte** in die **ZielSpalte** verschoben. Wird keine **Bedingung** angegeben, wird der Verschieben-Befehl immer ausgeführt.

Folgende **Bedingungen** sind für das Verschieben eines Feldinhaltes zulässig:

- **EQ** Verschiebt Inhalt **SpalteNr**, wenn **Vergleichswert=**Inhalt aus **Vergleichsspalte**
- **NotEQ** Verschiebt Inhalt **SpalteNr**, wenn **Vergleichswert<>**Inhalt aus **Vergleichsspalte**
- **Instr** Verschiebt Inhalt **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte vorkommt**.
- **NotInstr** Verschiebt Inhalt **SpalteNr**, wenn **Vergleichswert** in **Vergleichsspalte NICHT vorkommt**
- **LT** Verschiebt Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert < Vergleichsspalte**
- **LTEQ** Verschiebt Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert<=Vergleichsspalte**
- **GT** Verschiebt Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert>Vergleichsspalte**
- **GTEQ** Verschiebt Inhalt **SpalteNr**, wenn Wert aus **Vergleichswert>=Vergleichsspalte**

Beispiel:
ERPAdmin -CSVKonverter Liste.csv Move 1 2
Verschiebt den Inhalt aus Spalte **1** in die Spalte **2**.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1		111	12345678	444444444
2		111	12345678	444444444
3		111	12345678	444444444

ERPAdmin -CSVKonverter Liste.csv MoveCol 1 2 Kunde 3 /Cond:EQ
Verschiebt den Inhalt aus Spalte **1** in die Spalte **2**, wenn der Wert in Spalte **3** gleich „**Kunde**“ ist.

Datei vorher:

	A	B	C	D
1	111	22222	Kunde	4444444
2	111	22222	Lieferant	4444444
3	111	22222	Kunde	4444444

Datei nachher

	A	B	C	D
1		111	Kunde	4444444
2		22222	Lieferant	4444444
3		111	Kunde	4444444

Replace

ERPAdmin -CSVKonverter <CSV-Datei> REPLACE <SpalteNr> <Suchen> <Ersetzen>

Ersetzt innerhalb der Felder in **SpalteNr** die Zeichenkette **Suchen** durch die Zeichenkette **Ersetzen**.

Im Gegensatz zum Befehl „**SearchAndReplace**“ wird nicht das ganze Feld auf Gleichheit geprüft, sondern es können auch Zeichenketten innerhalb eines Feldes ersetz werden.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv REPLACE 3 "3456" "000"
Ersetzt in Spalte **3** die Zeichenfolge **3456** durch **000.**

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	1200078	444444444
2	111	22222	1200078	444444444
3	111	22222	1200078	444444444

Achtung:
Maskierte Zeichenfolgen wie z.B. \" können nicht ersetzt werden.

ReplaceCRLF

ERPAdmin -CSVKonverter <CSV-Datei> ReplaceCRLF 0 <Ersetzung>
Führt eine Ersetzung aller Zeilenumbrüche (**CRLF** -CarriageReturn LineFeed) durch **<Ersetzung>** aus. Es wird die komplette CSV-Datei überarbeitet.

Die Angabe der „**0**“ ist hier zwingend notwendig und darf nicht verändert oder weggelassen werden.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv ReplaceCRLF 0 |
Führt eine Ersetzung in der Datei **Liste.csv** durch und ersetzt Umbrüche **CRLF** durch das Pipe-Zeichen (**|**)

Datei vorher:

```
111;22222;3333333CRLF
111;22222;3333333CRLF
111;22222;3333333CRLF
```

Datei nachher

```
111;22222;3333333|111;22222;3333333|111;22222;3333333|
```

Hinweis:
Zeilenumbrüche in Form eins einfachen **LF** werden nicht ersetzt.

Um unsichtbare Zeichen wie **LF** oder **CRLF** in einer Textdatei sichtbar zu machen verwenden Sie das kostenfrei Tool Notepad+.

ReplaceLF

ERPAdmin -CSVKonverter <CSV-Datei> ReplaceLF 0 <Ersetzung>

Dateien von Linux-Systemen oder MAC enthalten oft ein **LF** (LineFeed) als Zeilenumbruch anstelle eines **CRLF** (CarriageReturn LineFeed). Windows Programme können diese Form des Zeilenumbruchs nicht korrekt interpretieren. **ERPAdmin** ersetzt alle **LF** Umbrüche durch **<Ersetzung>.** Wird keine **<Ersetzung>** angegeben wird automatisch **CRLF** als **<Ersetzung>** verwendet. Es wird die komplette CSV-Datei überarbeitet.

Die Angabe der „**0**“ ist hier zwingend notwendig und darf nicht verändert oder weggelassen werden.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv ReplaceLF

Datei vorher:

```
111;22222;3333333;444444444LF
111;22222;3333333;444444444CRLF
111;22222;3333333;444444444LF
```

Datei nachher

```
111;22222;3333333;444444444CRLF
111;22222;3333333;444444444CRLF
111;22222;3333333;444444444CRLF
```

Hinweis:
Um unsichtbare Zeichen wie **LF** oder **CRLF** in einer Textdatei sichtbar zu machen verwenden Sie das kostenfrei Tool **Notepad+**.

ReplaceTAB

ERPAdmin -CSVKonverter <CSV-Datei> ReplaceTAB 0 <Ersetzung>

Einige CSV-Dateien z.B. aus Linux-Datenbanken enthalten oft ein **TAB** (Tabulator) als Trennzeichen anstelle eines **Semikolons (;)**. Wenn **ERPAdmin** eine CSV-Dateien erwartet sind immer Dateien gemeint welche als Trennzeichen ein **Semikolon** (**;**)verwenden. Mit diesem Befehl ersetzt **ERPAdmin** alle **TAB-Zeichen** durch das Zeichen **<Ersetzung>.** Es wird die komplette CSV-Datei überarbeitet.

Die Angabe der „**0**“ ist hier zwingend notwendig und darf nicht verändert oder weggelassen werden.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv ReplaceTAB 0 ";"

Führt in der Datei **Liste.csv** eine Ersetzung alles **Tabulatoren** durch ein **Semikolon** durch.

Datei vorher:

```
111→222→333→444CRLF
111→222→333→444CRLF
111→222→333→444CRLF
```

Datei nachher

```
111;222;333;444CRLF
111;222;333;444CRLF
111;222;333;444CRLF
```

> **Hinweis:**
> Um unsichtbare Zeichen wie **TAB** in einer Textdatei sichtbar zu machen verwenden Sie das kostenfrei Tool **Notepad+**.

ReplaceISOLand

ERPAdmin -CSVKonverter <CSV-Datei> ReplaceISOLand <SpalteNr>

Ersetzt in der **CSV-Datei** in der **SpalteNr** die ISO-Länderkennzeichen oder die Ländernamen durch die ISO-Ländernummer aus **büro+**. Soll die Ersetzung in mehreren Spalten durchgeführt werden, müssen die Spalten durch ein Semikolon getrennt übergeben werden.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv ReplaceISOLand 2

Ersetzt in der Spalte **2** der Datei **Liste.csv** alle ISO-Ländercodes durch die ISO-Ländernummer aus büro+.

Datei vorher:

	A	B	C
1	111	DE	111111
2	222	Deutschland	222222
3	333	AT	333333

Datei nachher:

	A	B	C
1	111	276	1111111
2	222	276	2222222
3	333	40	3333333

Round

ERPAdmin -CSVKonverter <CSV-Datei> Round <SpalteNr> <Rundungsschema>

Runde die Zahlenwerte in **SpalteNr** nach dem **Rundungsschema**.

<Rundungsschema>

Folgende Rundungsschemen werden aktuell unterstützt:

- x, Schneider die Nachkommastellen ab ohne zu runden
- xR,00 Rundet auf den vollen Betrag ohne Nachkommastellen
- x,R0 Rundet auf die erste Nachkommastelle
- x,R5 Rundet die zweite Nachkommastelle auf 0 oder 5

Beispiel:
ERPAdmin -CSVKonverter Liste.csv Round 2 xR,00
Rundet in der Datei **Liste.csv** in Spalte **2** alle Werte auf ganze Beträge ohne Nachkommastelle.

Datei vorher:

	A	B	C
1	111	14,56	11111
2	222	123,29	22222
3	333	99,95	33333

Datei nachher

	A	B	C
1	111	15	11111
2	222	123	22222
3	333	100	33333

SearchAndReplace

ERPAdmin -CSVKonverter <CSV-Datei> SearchAndReplace <SpalteNr> <Suchen> <Ersetzen> <Vergleichsspalte> /Cond:<Bedingung)

Ersetzt alle Feldinhalte **Suchen** durch die Zeichenfolge **Ersetzen** in **SpalteNr**. Es wird immer der komplette Spalten-Inhalt verglichen. **Es findet keine Teilstring-Ersetzung innerhalb eines Feldes statt.**

Soll innerhalb einer Spalte eine Zeichenketten oder ein Teil davon ersetzt werden, verwenden Sie den **ERPAdmin -CSVKonveter <CSV-Datei> Replace**.

Optional kann eine Bedingung **/Cond:<Bedingung>** und eine **Vergleichsspalte** angegeben werden. Ist dies der Fall, wird der Wert aus der **Vergleichsspalte** mit dem Wert aus **SpalteNr** verglichen und nur wenn die **Bedingung** erfüllt ist, wird die Ersetzung ausgeführt.

Wird als **SpalteNr** der Wert „**0**“ übergeben, werden alle Spalten der **CSV-Datei** bearbeitet.

Sonderzeichen wie **LineFeed**, **CarriageReturn-LineFeed** und **Tabulatoren** können durch Platzhalter übergeben werden. Verwenden Sie dazu als als-Platzhalter "**<LF> "** für **LineFeed**, "**<CRLF>"** für **CarriageReturn-LineFeed** und "**<TAB>"** für **Tabulator**.

> **Hinweis:**
> Platzhalter **<LF>**, **<CRLF>** und **<TAB>** müssen immer in doppelten Anführungszeichen übergeben werden.
>
> Beispiel: **"<TAB>"** oder **"<CRLF>"**

$xx
Soll bei der Ersetzung kein fester Wert verwendet werden sondern der Wert aus einer anderen Spalte, so kann diese in der Form **$xx** als Ersetzen-Parameter übergeben werden wobei **xx** für die Spalte steht dessen Wert als Ersetzung verwendet werden soll.
Beispiel: **ERPAdmin -CSVKonverter <CSV-Datei> SearchAndReplace 1 Ja $5 3**
Schreibt der Wert aus Spalte 5 (**$5**) in Spalte **1**, wenn in Spalte **3** ein „**Ja**“ steht.

/Cond:<Bedingung> (optional)

Wird eine Spalte ausgewählt (Spalte<>0) wird die Ersetzung nur dann ausgeführt, wenn die angegebene Bedingung erfüllt ist. Wird keine Beding angegeben wird **EQ** verwendet.

Bei der Angabe Spalte="**0**" werden alle Bedingung ignoriert und beim Vergleich wird auf Groß-/Kleinschreibung geachtet.

Folgende **Bedingungen** für sind zulässig:

- **EQ** Prüft Spaltenwert und **Suchen** auf Gleichheit (Groß-/Kleinschreibung wird beachtet)
- **NotEQ** Prüft Spaltenwert und **Suchen** auf Ungleichheit
- (Groß-/Kleinschreibung wird beachtet)
- **GT** Prüft Spaltenwert und **Suchen** auf > (Groß-/Kleinschreibung wird beachtet)
- **GTEQ** Prüft Spaltenwert und **Suchen** auf >= (Groß-/Kleinschreibung wird beachtet)
- **LT** Prüft Spaltenwert und **Suchen** auf < (Groß-/Kleinschreibung wird beachtet)
- **LTEQ** Prüft Spaltenwert und **Suchen** auf <= (Groß-/Kleinschreibung wird beachtet)
- **LeftEQ** Prüft ob die linken x-Zeichen (Länge **Suchen**) mit **Suchen** übereinstimmen
- (Groß-/Kleinschreibung wird beachtet)
- **Instr** Prüft ob die Zeichenfolge **Suchen** enthalten ist
- (Groß-/Kleinschreibung wird beachtet)
- **NotInstr** Prüft ob die Zeichenfolge **Suchen** nicht enthalten ist
- (Groß-/Kleinschreibung wird beachtet)
- **EQ-NoCS** Prüft Spaltenwert und **Suchen** auf Gleichheit (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **NotEQ-NoCS** Prüft Spaltenwert und **Suchen** auf Ungleichheit (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **GT-NoCS** Prüft Spaltenwert und **Suchen** auf > (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **GTEQ-NoCS** Prüft Spaltenwert und **Suchen** auf >= (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **LT-NoCS** Prüft Spaltenwert und **Suchen** auf < (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **LTEQ-NoCS** Prüft Spaltenwert und **Suchen** auf <= (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **LeftEQ-NoCS** Prüft ob die linken x-Zeichen (Länge **Suchen**) mit Suchen übereinstimmen (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **Instr-NoCs** Prüft ob die Zeichenfolge **Suchen** enthalten ist
- (Groß-/Kleinschreibung wird **NICHT** beachtet)
- **NotInst-NoCS** Prüft ob die Zeichenfolge **Suchen** nicht enthalten ist
- (Groß-/Kleinschreibung wird **NICHT** beachtet)

Beispiel:
ERPAdmin -CSVKonverter *.csv SearchAndReplace 0 "<CRLF>"
Es werden alle Zeilenumbrüche innerhalb der Datensätze (Zeilen) entfernt. Die Zeilenumbrüche für das Datensatzende bleiben erhalten.

ERPAdmin -CSVKonverter *.csv SearchAndReplace 3 Wahr True
Ersetzt in der Spalte **3** alle Werte „**Wahr**“ durch „**True**“. Werte wie „**WAHR**“ oder „**wahr**“ werden ignoriert.

ERPAdmin -CSVKonverter *.csv SearchAndReplace 3 Wahr True /Cond:EQ-NoCS
Ersetzt in der Spalte **3** alle Werte „**Wahr**“, „**WAHR**“ und „**wahr**“ (und auch alle anderen Schreibweisen) durch „**True**“.

ERPAdmin -CSVKonverter *.csv SearchAndReplace 3 70000 Lieferant 8 /Cond:GTEQ
Schreibt den Wert **Lieferant** in der Spalte **3** wenn der Wert in Spalte **8** größer-gleich (**/Cond:GTEQ**) ist als **70000**.

SetBoolean

ERPAdmin -CSVKonverter <CSV-Datei> SETBOOLEAN <SpalteNr> <TRUE-Wert> <FALSE-Wert>

Ersetzt in **SpalteNr** die Zeichenkette **TRUE-Wert** durch **Ja** und die Zeichenkette **FALSE-Wert** durch **Nein**.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv SETBOOLEAN 3 Wahr Falsch

Datei vorher:

	A	B	C	D
1	111	22222	WAHR	444444444
2	111	22222	FALSCH	444444444
3	111	22222	WAHR	444444444

Datei nachher

	A	B	C	D
1	111	22222	Ja	444444444
2	111	22222	Nein	444444444
3	111	22222	Ja	444444444

SetupNr

ERPAdmin -CSVKonverter <CSV-Datei> SetupNr <SpalteNr> <Tabelle> <Startwert> /Header

Ermittelt in der **Tabelle** von büro+ die nächsten freien Nummern ab **Startwert** und speichert die Ergebnisse in der **SpalteNr** ab. Alle bereits vorhandenen Daten in der Spalte **SpalteNr** werden dabei überschrieben. Bei vielen Datensätzen leider die Performance, weil pro Datensatz ein Zugriff auf die büro+-Datenbank erfolgen muss.

Beispiel:
ERPAdmin -CSVKonverter Neue_Adressen.csv SetupNr 1 Adressen 10500
Fügt in der Datei **Neue_Adressen.csv** in Spalte **1** die nächsten freien Adress-Nummern ein die in büro+ im Bereich **Adressen** ab Nummer **10500** unbelegt sind.

Datei vorher:

	A	B	C
1	**AdrNr**	**Name1**	**Straße**
2		Müller & Sohn	Am Buchrain 11
3		Steffis Tintenfass	Odenwaldstr. 23
4		Müller & Milch HmbH	Goetheweg 7a

Datei nachher

	A	B	C
1	**AdrNr**	**Name1**	**Straße**
2	10507	Müller & Sohn	Am Buchrain 11
3	15011	Steffis Tintenfass	Odenwaldstr. 23
4	10524	Müller & Milch HmbH	Goetheweg 7a

SetValue

ERPAdmin -CSVKonverter <CSV-Datei> SETVALUE <SpalteNr> <ZeileNr> <Feldwert>

Schreibt **Feldwert** in das Feld (**SpalteNr**, **ZeileNr**) der **CSV-Datei**.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv SETVALUE 3 2 Test
Schreibt den Wert „**Test**“ in Spalte **3**, Zeile **2**.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	12345678	44444444
2	111	22222	Test	44444444
3	111	22222	12345678	44444444

Soll ein **Feldwert** für eine ganze **SpalteNr** gesetzt werden, muss **ZeileNr** leer bleiben. Damit bekommen alle Felder der **SpalteNr** den **Feldwert** zugewiesen. Sollen mehrere **Feldwerte** eines Datensatzes mit demselben **Feldwert** überschrieben werden, sind die Werte der Spalten in **SpalteNr** mit einem Semikolon getrennt anzugeben.

Beispiel: **ERPAdmin -CSVKonverter Liste.csv SETVALUE 3 "" Test**
In Spalte **3** wird der Wert **Test** in jedes Feld geschrieben.

Datei vorher:

	A	B	C	D
1	111	22222	12345678	444444444
2	111	22222	12345678	444444444
3	111	22222	12345678	444444444

Datei nachher

	A	B	C	D
1	111	22222	Test	4444444444
2	111	22222	Test	4444444444
3	111	22222	Test	4444444444

Soll der Inhalt eines Feldes mit dem Wert eines anderen Feldes aus der **CSV-Datei** gesetzt werden, kann dieser **Feldwert** wie folgt ausgewählt werden. Die Notation sieht wie folgt aus: **xy**
Dabei steht **$x** für die Spalte-Nr und **$x** für die Zeilen-Nr von der Zelle mit dem **Feldwert**.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv SETVALUE 4 2 $1$2
Schreibt den Wert „**222**“ aus Spalte-Nr. **1**, Zeile-Nr. **2** in das Feld Spalte-Nr. **4** Zeile-Nr. **1**

Datei vorher:

	A	B	C	D
1	111	11111	1111111	1111111111
2	222	22222	2222222	2222222222
3	333	33333	3333333	3333333333

Datei nachher

	A	B	C	D
1	111	11111	1111111	222
2	222	22222	2222222	2222222222
3	333	33333	3333333	3333333333

Auf gleiche Weise kann auch eine ganze Spalte mit einem **Feldwert** belegt werden.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv SETVALUE 4 "" $1$2

Schreibt den Wert „**222**“ aus Spalte-Nr. **1**, Zeile-Nr. **2** in alle Felder der **SpalteNr 4**

Datei vorher:

	A	B	C	D
1	111	11111	1111111	1111111111
2	222	22222	2222222	2222222222
3	333	33333	3333333	3333333333

Datei nachher

	A	B	C	D
1	111	11111	1111111	222
2	222	22222	2222222	222
3	333	33333	3333333	222

Sort

ERPAdmin -CSVKonverter <CSV-Datei> Sort <SortierSpalteNr> /DESC /Cond:<Format>

Sortiert die **CSV-Datei alphabetisch** nach **SortierSpalteNr** aufsteigend, wenn keine andere Sortierrichtung **(/Desc)** angegeben wurde.

Soll die Datei nach mehreren Spalten sortiert werden, müssen die Spaltennummern im Parameter **SortierSpalteNr** durch Semikolon getrennt übergeben werden. Es dürfen keine Leerzeichen enthalten sein. Die Sortierung nach mehreren Spalten dauert deutlich länger.

/Desc **(Optional)**

Soll die **CSV-Datei** absteigend sortiert werden muss die Option **/DESC** als Sortierrichtung mit übergeben werden.

/Cond:<Format> **(optional)**

Die Standard Sortierung erfolgt nach Alphabet.

Optional kann auch nach Datum oder nach Zahlen (nummerisch) sortiert werden.

Enthält die **SpalteNr** Datums-Werte im Format **TT.MM.JJJJ** (10-Stellig) und soll nach diesem Datum sortiert werden, muss die Option **/Cond:Date** übergeben werden.

Soll eine Spalte als Zahl für die Sortierung interpretiert werden, muss als Option **/Cond:Number** übergeben werden. Der Feldwerte dürfen max. 15 Vor- und 15 Nachkommastellen haben.

Wenn nach Datum oder Zahl sortiert werden soll, darf nur ein Spalten-Index in SpalteNr übergeben werden.

Beispiele:

ERPAdmin -CSVKonverter Liste.csv Sort 3;2

Die Date **Liste.csv** wird nach erst nach der Spalte **3** und dann nach der Spalte **2** sortiert.
Weil die Option **/Header** nicht angegeben wurde werden alle Zeilen sortiert.

Datei vorher:

	A	B	C
1	111	4444	3333333
2	222	3333	1111111
3	333	2222	4444444
4	444	1111	1111111

Datei nachher

	A	B	C
1	444	1111	1111111
2	222	3333	1111111
3	111	4444	3333333
4	333	2222	4444444

ERPAdmin -CSVKonverter Liste.csv Sort 3 /Header

Die Date **Liste.csv** wird nach der Spalte **3** sortiert. Dabei wird der erste Datensatz bei der Sortierung ignoriert (**/Header**) und erst mit dem **2**. Datensatz begonnen.

Datei vorher:

	A	B	C
1	333	33333	3333333
2	222	22222	2222222
3	111	11111	1111111

Datei nachher

	A	B	C
1	333	33333	3333333
2	111	11111	1111111
3	222	22222	2222222

Split

ERPAdmin -CSVKonverter <CSV-Datei> Split <SpalteNr> <Zielspalte1> <Zielspalte2> <Trennzeichen>

Teilt den Inhalt von **SpalteNr** anhand des ersten Vorkommens von **Trennzeichen** auf. Das Ergebnis wird in **Zielspalte1** (linker Teil vor den Trennzeichen) und **Zielspalte2** (rechter Teil nach dem Trennzeichen) gespeichert.

Sind mehrere **Trennzeichen** in **SpalteNr** vorhanden und sollen alle Trennungen vorgenommen werden, muss der Befehlt entsprechend oft auf der jeweiligen **Zielspalte2** wiederholt werden.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv Split 1 2 4 "|"

Für einen Artikel-Import mit Varianten sollen **Farbe** und **Größe** mit den Ausprägungen (Farbe=schwarz, grün, spark, Größe=S, M, L) anhand des Trennzeichens „**|**“ (Pipe) in die Spalten **2** und **4** getrennt werden.

Datei vorher:

	A	B	C	D	E
1	Farbe: schwarz | Größe: S				
2	Farbe: grün | Größe: S				
3	Farbe: spark | Größe: S				

Datei nachher

	A	B	C	D
1	Farbe: schwarz | Größe: S	Farbe: schwarz		Größe: S
2	Farbe: grün | Größe: S	Farbe: grün		Größe: S
3	Farbe: spark | Größe: S	Farbe: spark		Größe: S

ERPAdmin -CSVKonverter Liste.csv Split 2 2 3 :
Anschließend soll die Ausprägung Farbe getrennt werden anhand des Trennzeichens „**:**“ (Doppelpunkt). Das Ergebnis der Trennung **Farbe** aus Spalte **2** soll wieder in Spalte **2** (wird überschrieben) und **3** gespeichert werden.

Datei nachher

	A	B	C	D	E
1	Farbe: schwarz | Größe: S	Farbe	schwarz	Größe: S	
2	Farbe: grün | Größe: S	Farbe	grün	Größe: S	
3	Farbe: spark | Größe: S	Farbe	spark	Größe: S	

ERPAdmin -CSVKonverter Liste.csv Split 4 4 5 :
Zuletzt sollen die Ausprägungen **Größe** getrennt werden anhand des Trennzeichens „:“ (Doppelpunkt). Die Ergebnisse der Trennung **Größe** aus Spalte **4** soll wieder in Spalte **4** (wird überschrieben) und in Spalte **5** gespeichert werden.

Datei nachher

	A	B	C	D	E
1	Farbe: schwarz | Größe: S	Farbe	schwarz	Größe	S
2	Farbe: grün | Größe: S	Farbe	grün	Größe	S
3	Farbe: spark | Größe: S	Farbe	spark	Größe	S

Hinweise:
Bestimmte Trennzeichen müssen immer in Anführungszeichen gesetzt werden z.B. Pipe.

Soll als Trennzeichenkette **//** verwendet werden muss als Trennzeichen "**/@/**" übergeben werden.

SumCol

ERPAdmin -CSVKonverter <CSV-Datei> SumCol <SpalteNr> <Anfangs-Spalte> <Ende-Spalte>

ERPAdmin Bildet eine Summe über alle Werte der Spalten **Anfangs-Spalte** bis **Ende-Spalte** bis sich der Wert in **SpalteNr** ändert. Das Ergebnis ist dann eine Liste mit den Werten aus **SpalteNr** und den Summen der Spalten. Soll nur eine einzige Spalte aussummiert werden kann der Wert **Ende-Spalte** weggelassen werden.

> **Hinweis:**
> Die Datei muss nach **SpalteNr** sortiert sein. Dazu kann der Befehl **ERPAdmin -CSVKonverter -Sort** verwendet werden.

Ist in den Feldern die summiert werden ein Wert enthalten der **keiner** Zahl entspricht, wird dieser Wert bei der Summierung übersprungen. In der Ergebnis-E-Mail bekommt der Benutzer dann einen Hinweis auf diesen Datensatz ausgegeben.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv SumCol 1 2 5
Summiert die Inhalte aus den Spalten **2, 3 , 4** und **5** auf und Speichert pro Wert aus **SpalteNr** die Ergebnisse ab. Sobald sich der Wert in **SpalteNr** ändert wird eine neue Summe gebildet.

Datei vorher:

	A	B	C	D	E	F	G
1	111	2	3	4	5	6	7
2	222	3	4	5	6	7	8
3	222	4	5	6	7	8	9
4	222	5	6	7	8	9	10
5	333	6	7	8	9	10	11
6	333	7	8	9	10	11	12
7	444	8	9	10	11	12	13
8	555	9	10	11	12	13	14
9	666	10	11	12	13	14	15

Datei nachher

	A	B	C	D	E
1	111	2	3	4	5
2	222	12	15	18	21
3	333	13	15	17	19
4	444	8	9	10	11
5	555	9	10	11	12
6	666	10	11	12	13

SumRow

ERPAdmin -CSVKonverter <CSV-Datei> SumRow <SpalteNr> <Anfangs-Spalte> <Ende-Spalte>

ERPAdmin Bildet pro Zeile der **CSV-Datei** eine Summe der Feldhalte aus **Anfangs-Spalte** bis **Ende-Spalte** und speichert das Ergebnis im Feld **SpalteNr**.

Ist in den Feldern die summiert werden ein Wert enthalten der keiner Zahl entspricht, wird dieser Wert bei der Summierung übersprungen. In der Ergebnis-E-Mail bekommt der Benutzer dann einen Hinweis ausgegeben.

Beispiel:
ERPAdmin -CSVKonverter Liste.csv SumRow 1 2 5
Summiert Zeilenweise die Inhalte aus den Spalten **2, 3 , 4** und **5** auf und speichert das Ergebnis in Spalte **1**

Datei vorher:

	A	B	C	D	E	F
1	10	20	30	40	50	60
2	11	21	31	41	51	61
3	12	22	32	42	52	62

Datei nachher

	A	B	C	D	E	F
1	140	20	30	40	50	60
2	144	21	31	41	51	61
3	148	22	32	42	52	62

Trim

ERPAdmin -CSVKonverter <CSV-Datei> Trim <SpalteNr>

Entfernt vorhandene Leerzeichen am Anfang und am Ender von **SpalteNr**. Sollen mehrere Spalten in einem Aufruf verarbeitet werden, müssen die **SpaltenNr** mit einem Semikolon getrennt werden.

Beispiel:

ERPAdmin -CSVKonverter Liste.csv Trim 2

Die Leerzeichen am Ende von Spalte 2 sollen entfernt werden.

Datei vorher:

```
DEU;AB2100351          ;2021-03-10;Z
DEU;AB2001616          ;2021-02-10;Z
DEU;AB2100153          ;2021-01-28;Z
DEU;LI21000322         ;2021-01-29;Z
```

Datei nachher

```
DEU;AB2100351;2021-03-10;Z
DEU;AB2001616;2021-02-10;Z
DEU;AB2100153;2021-01-28;Z
DEU;LI21000322;2021-01-29;
```

39 Der Befehl „-DateDiff“ STD

ERPAdmin -DateDiff <Datenbank> <Feld1> <Feld2> <Zielfeld> <Bedingungs-Wert> /Cond:<Bedingung> /Von:<Bereich von> /Bis: /Filter:<Filterbedingung> /NoLockError

Mit dem Befehl **DateDiff** ermittelt **ERPAdmin** in der **Datenbank** die Datumsdifferenz zwischen den beiden Feldern **Feld1** und **Feld2** in Tagen und schreibt das Ergebnis (Differenz in Tagen) in das **Zielfeld**.

Die Berechnung wird dabei wie folgt ausgeführt:

Inhalt **Feld1**:	20.02.2021
Inhalt **Feld2**:	15.02.2021
Ergebnis **Zielfeld**:	5

Inhalt **Feld1**:	10.02.2021
Inhalt **Feld2**:	15.02.2021
Ergebnis **Zielfeld**:	-5

<Datenbank>
büro+-Tabelle in der sich die Felder **Feld1**, **Feld2** und **Zielfeld** befinden.

<Feld1>
Erstes Feld für den Datumsvergleich in Tagen. **Feld1** muss vom Typ **Date** sein.

<Feld2>
Erstes Feld für den Datumsvergleich in Tagen. **Feld2** muss vom Typ **Date** sein.

<Zielfeld>
In dieses Feld wird das Ergebnis der Berechnung gespeichert. Das Feld muss vom Typ **UnicodeString** oder **Integer** sein.

<Bedingungs-Wert> **(optional)**
Soll das Ergebnis im **Zielfeld** nur in Abhängigkeit einer weiteren Bedingung gespeichert werden, kann hier ein Zahlenwert angegeben werden. Der **Zielfeld-Wert** wird dann mit diesem Wert verglichen. Der Wert wird nur dann beachtet, wenn auch eine Bedingung mit **/Cond:<Bedingung>** angegeben und erfüllt wurde.

/Cond:<Bedingung> **(optional)**
Vergleichsbedingung zwischen dem **Zielfeld-Wert** und dem **Bedingungs-Wert**.

Folgende **Bedingungen** sind zulässig:

- **EQ** Zielfeld-Wert **=** Bedingungs-Wert
- **NotEQ** Zielfeld-Wert **<>** Bedingungs-Wert
- **GT** Zielfeld-Wert **>** Bedingungs-Wert
- **GTEQ** Zielfeld-Wert **>=** Bedingungs-Wert
- **LT** Zielfeld-Wert **<** Bedingungs-Wert
- **LTEQ** Zielfeld-Wert **<=** Bedingungs-Wert

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Die zu berechnen Datensätze können über die Werte **/Von:** bis **/Bis:** eingeschränkt werden. Wird kein Bereich angegeben, werden alle Datensätze der gewählten Tabelle verarbeitet.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/NoLockError **(optional)**
Kann ein Datensatz nicht gespeichert werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

Beispiel:

ERPAdmin -DateDiff Vorgang Dat LiefDat NettoTg 5 /Cond:GTEQ /Von:BelegNr=LI2100000 /Bis:BelegNr=LI2999999
Läuft über alle Vorgänge **LI2100000** bis **LI2999999** (Lieferscheine) und ermittelt die Datumsdifferenz zwischen dem Vorgangsdatum (**Dat**) und dem Lieferdatum (**LiefDat**). Wenn die Differenz >= 5 (**5 /Cond:GTEQ**) ist wird der Differenz-Wert in das Feld Nettotage (**NettoTg**) geschrieben.

40 Der Befehl „-DeleteData“ STD

ERPAdmin -DeleteData <Datenbank> /Von:<Bereich von> /Bis: /Filter:<Filterbedingung>

Mit dem Befehl **DeleteData** kann **ERPAdmin** einzelne Datensätze oder auch ganze Datenbanken aus **büro+** löschen.

<Datenbank>
Gibt die **büro+**-Tabelle an, aus der Daten gelöscht werden sollen z.B. „**Artikel**“, „**Adressen**“ oder „**Vorgang**“.

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Über **/Von:** kann eine Einschränkung auf die Tabelle gelegt werden wie z.B. **/Von:Status=Kunde** oder **/Von:AdrNr=10000**. Soll ein Bereich selektiert werden muss auch ein Wert **/Bis:** übergeben werden z.B. **/Bis:AdrNr=19999**

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Beispiele:

Lösche alle Adressen mit Adressnummern im Bereich 10000 - 19999
ERPAdmin -DeleteData Adressen /Von:AdrNr=10000 /Bis:AdrNr=19999

Lösche alle Angebote des Vertreters 13
ERPAdmin -DeleteData Vorgang /Von:Art=15 /Bis:Art=15 /Filter:VtrNr='13'

Lösche alle Adressen des Vertreters 13
ERPAdmin -DeleteData Adressen VtrNr='13'

Lösche alle Adressen im Bereich 10000-19999 von Vertreter 4 mit dem Status „Alt“
ERPAdmin -DeleteData Adressen /Von:AdrNr=10000 /Bis:AdrNr=19999 "/Filter:VtrNr= '4' and Status='Alt '"

Lösche alle Versanddatensätze. Achtung: Das Löschen einer ganzen Datenbank kann einige Zeit in Anspruch nehmen.
ERPAdmin -DeleteData Versand

Vor einem Import müssen bestimmte Vorgangspositionen (BK0050000 bis BK0059999) aus der Datenbank gelöscht werden. Mit Boardmitteln der **büro+** ist das nicht machbar.
ERPAdmin -DeleteData VorgangPosition /Von:BelegNr=0050000 /Bis:BelegNr=0059999

Im Lagerbuch sollen alle versehentlich erzeugten Einträge deren Belegnummer mit RE17xxxxx beginnen gelöscht werden. Mit Boardmitteln der **büro+** ist das nicht machbar.

ERPAdmin -DeleteData Lagerbuch /Von:BelegNr=RE1700000 /Bis:BelegNr=RE1799999

41 Der Befehl „-DeleteDataFromFile“ PRO

ERPAdmin -DeleteDataFromFile <Datenbank> <SpaltenNr> <ERP-Feldbezeichner> /Datei:<CSV-Datei> /NoFileNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit dem Befehl **DeleteDataFromFile** kann **ERPAdmin** bestimmte Datensätze aus den Datenbanken von **büro+** löschen. Die zu löschenden Datensätze werden in einer CSV-Datei mit Spaltenüberschriften übergeben. Die erste Zeile der CSV-Datei wird deshalb ignoriert!

<Datenbank>
Name der **büro+** -Datenbank z.B. **Artikel** oder **Adressen**

<SpaltenNr>
Spalten-Nummer der CSV-Datei in der die Schlüsselwerte der zu löschenden Datensätze stehen. Die Angabe eines Spaltennamens ist trotz vorhandener Überschriften nicht zulässig.

<ERP-Feldbezeichner>
Feldbezeichner aus der Datenbank mit dem der zu löschende Datensatz eindeutig referenziert wird. Es müssen immer Feldnamen verwendet werden über den ein Datensatz eindeutig referenziert wird z.B. Adressnummer oder Artikelnummer.

/Datei:<CSV-Datei>
CSV-Datei mit Spaltenüberschriften und einer Liste von Feldwerten die aus **büro+** gelöscht werden sollen.

/NoFileNotExistError **(optional)**
Wurde keine **CSV-Datei** gefunden wird **keine** Fehlermeldung erzeugt.

Beispiel

ERPAdmin -DeleteDataFromFile Artikel 1 ArtNr /Datei:c:\ERPAdmin\Artikel_loeschliste.csv
In der 1. Spalte der Datei **c:\ERPAdmin\Artikel_loeschliste.csv** stehen die Artikelnummern der Artikel die aus der Datenbank „**Artikel**“ gelöscht werden sollen.

Die Datei **c:\ERPAdmin\Artikel_loeschliste.csv** könnte dabei wie folgt aussehen:

	A	B	C	D
1	**Art.ArtNr**	**Art.SuchBeg**	**Art.KuBez1**	**Art.WgrNr**
2	AS-C-A	AS-C-A	Aktualitats-Service fur buro+ commercial (Warenwirtschaft) Business Edition	330
3	AS-C-A-SBE	AS-C-A-SBE	Aktualitats-Service fur buro+ commercial (Warenwirtschaft) Small Business Edition	330
4	AS-C-AF	AS-C-AF	Aktualitats-Service fur buro+ commercial (Warenwirtschaft, FiBu) Business Edition	330
5	AS-C-AF-SBE	AS-C-AF-SBE	Aktualitats-Service fur buro+ commercial (Warenwirtschaft, FiBu) Small Business Edition	330
6	AS-C-AFL	AS-C-AFL	Aktualitats-Service fur buro+ commercial (Warenwirtschaft, FiBu, Lohn) Business Edition	330
7	AS-C-AFL-SBE	AS-C-AFL-SBE	Aktualitats-Service fur buro+ commercial (Warenwirtschaft, FiBu, Lohn) Small Business Edition	330
8	AS-C-AFL-SCHUL	AS-C-AFL-SCHUL	Aktualitats-Service fur buro+ commercial (Warenwirtschaft, FiBu, Lohn) als Schulversion	330
9	AS-C-AL	AS-C-AL	Aktualitats-Service fur buro+ commercial (Warenwirtschaft, Lohn) Business Edition	330

42 Der Befehl „-DeleteFiles“ BAS

ERPAdmin -DeleteFiles <Verzeichnis> <MaxAnzahlDateien> <Datei-Filter>

Mit diesem Befehl kann die maximale Anzahl der Dateien in einem Verzeichnis beschränkt werden. Dabei werden die überschüssigen Dateien nach Datum gelöscht, die ältesten zuerst.

<Verzeichnis>
Gibt den Verzeichnisnamen inkl. Laufwerk an, dass auf eine maximale Anzahl von Dateien beschränkt werden soll.

> **Hinweis:**
> Eine Verzeichnisangabe in Anführungszeichen darf **nicht** mit einem „\“ (Backslash) enden.

<MaxAnzahlDateien>
Gibt die Maximale Anzahl von Dateien in dem angegebenen Verzeichnis verbleiben sollen. Es werden immer die vom Datum her neusten Dateien behalten. Die ältesten werden zuerst gelöscht.

<Datei-Filter> **(optional)**
Sollen nicht alle Dateien in die Prüfung mit einbezogen werden, sondern z. B. nur Dateien mit einer bestimmten Endung, kann dies über den Filter eingeschränkt werden. Wird kein Filter angegeben, werden alle Dateien in dem Verzeichnis beachtet.

Beispiele:

ERPAdmin -DeleteFiles "c:\Backup" 7
Im Verzeichnis **„c:\Backup\“** werden alle Dateien bis auf die **7** neuesten gelöscht.

ERPAdmin -DeleteFiles "c:\Backup" 7 "*.TIB"
Im Verzeichnis **„c:\Backup\“** werden alle Dateien mit der Endung **„.tib“** bis auf die **7** neuesten gelöscht.

43 Der Befehl „-DeleteFilesByDate“ STD

ERPAdmin -DeleteFilesByDate <Verzeichnis> <Datum älteste Datei> <Datei-Filter>

Bei diesem Befehl handelt es sich um eine Erweiterung des Befehls **DeleteFiles**. Möchte man die Menge der Dateien in einem Verzeichnis nicht über die Anzahl begrenzen, sondern über das Datum kann dies wie folgt geschehen:

<Verzeichnis>
Gibt den Verzeichnisnamen inkl. Laufwerk an, dass auf eine maximale Anzahl von Dateien (nach Datum) beschränkt werden soll.

> **Hinweis:**
> Eine Verzeichnisangabe in Anführungszeichen darf nicht mit einem „\“ (Backslash) enden.
>
> Beispiel: **"C:\Mein Verzeichnis"** statt **"C:\Mein Verzeichnis\"**

<Datum älteste Datei>
Gibt das Datum der ältesten Datei in diesem Verzeichnis an. Alle älteren Dateien werden gelöscht.

<Datei-Filter> (optional)
Sollen nicht alle Dateien in die Prüfung mit einbezogen werden, sondern z. B. nur Dateien mit einer bestimmten Endung, kann dies über den Filter eingeschränkt werden. Wird kein Filter angegeben, werden alle Dateien in dem Verzeichnis beachtet.

Beispiel:

ERPAdmin -DeleteFilesByDate "c:\Backup" 13.10.2017
Im Verzeichnis **„c:\Backup“** werden alle Dateien gelöscht die älter sind als der **13.10.2017**

Variables Datum
Sollen in einem Verzeichnis immer die Dateien der letzten 30 Tage zur Verfügung stehen, kann der Platzhalter **@LASTxxDAYS@** anstelle eines konkreten Datums eingefügt werden. **xx** steht dabei für die Anzahl an Tagen.

Beispiele:

ERPAdmin -DeleteFilesByDate "c:\Backup" @LAST30DAYS@
Im Verzeichnis **„c:\Backup\“** werden alle Dateien gelöscht die älter sind als 30 Tage.

ERPAdmin -DeleteFilesByDate "c:\Backup" @LAST45DAYS@ "*.TIB"
Im Verzeichnis **„c:\Backup“** werden alle Dateien mit der Endung **„.tib“** gelöscht die älter sind als 45 Tage.

44 Der Befehl „-DeleteFoldersByDate“ STD

ERPAdmin -DeleteFoldersByDate <Verzeichnis> <Datum ältestes Verzeichnis>

Mit diesem Befehl können Sie mit **ERPAdmin** Unterverzeichnisse löschen die älter sind als ein bestimmtes Datum. Dabei wird das letzte Änderungsdatum des Unterverzeichnisses zur Prüfung verwendet. Die betroffenen Unterverzeichnisse werden ohne Prüfung auf evtl. vorhandene weitere Unterverzeichnisse oder Daten gelöscht.

<Verzeichnis>
Gibt das Verzeichnis inkl. Laufwerk an, in dem die Anzahl von Unterverzeichnissen (nach Datum) beschränkt werden soll. Die Prüfung erfolgt nur in dem genannten Verzeichnis. Eine rekursive Überprüfung darunter liegenden Verzeichnisse wird nicht durchgeführt.

> **Hinweis:**
> Eine Verzeichnisangabe in Anführungszeichen darf **nicht** mit einem „\“ (Backslash) enden.
>
> Beispiel: **"C:\Mein Verzeichnis"** statt **"C:\Mein Verzeichnis\"**

<Datum ältestes Verzeichnis>
Gibt das Datum des ältesten Unterverzeichnisses in diesem Verzeichnis an. Alle älteren Unterverzeichnisse werden gelöscht.

Beispiel:

ERPAdmin -DeleteFoldersByDate "c:\ERPAdmin_History" 01.10.2017
Im Verzeichnis **„c:\ERPAdmin\-History“** werden alle Unterverzeichnisse gelöscht die ein älteres Datum (DateLastModified) besitzen als den **01.10.2017**

Variables Datum
Sollen in einem Verzeichnis immer die Unterverzeichnisse der letzten 30 Tage zur Verfügung stehen, kann der Platzhalter **@LASTxxDAYS@** anstelle eines konkreten Datums eingefügt werden. **xx** steht dabei für die Anzahl an Tagen.

Beispiele:

ERPAdmin -DeleteFoldersByDate "c:\ERPAdmin_History" @LAST30DAYS@
Im Verzeichnis **„c:\ERPAdmin_History“** werden alle Dateien gelöscht die älter sind als 30 Tage.

45 Der Befehl „-DeleteLF“ PRO

ERPAdmin -DeleteLF<CSV-Datei>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Dateien von Linux-Systemen können **LF** (LineFeed) als Zeilenumbruch innerhalb von Zeichenketten enthalten. Diese sind manchmal unerwünscht. **ERPAdmin** löscht einzelne **LF** Umbrüche. **CRLF** bleiben erhalten.

> **Hinweis:**
> Möchten Sie die **LF**-Zeichen durch ein anderes Zeichen ersetzen verwenden Sie den **ERPAdmin-**Befehl **ERPAdmin -CSVKonverter ReplaceLF**

Beispiel:
ERPAdmin -DeleteLF Liste.csv
Lösche alle alleinstehenden **LF** Zeichen in der Datei **Liste.csv. CRLF** bleiben erhalten.

> **Hinweis:**
> Um unsichtbare Zeichen in einer Textdatei sichtbar zu machen können Sie das kostenfrei Tool Notepad+ verwenden.

46 Der Befehl „-Dir2CSV“ PRO

ERPAdmin -Dir2CSV <Verzeichnis> <Datei-Filter> <Typ-Filter> /Datei:<CSV-Datei> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC /EmailBCC /Header

ERPAdmin erzeugt eine CSV-Datei mit dem Inhalt des angegebenen **Verzeichnisses**.

<Verzeichnis>
Name des Verzeichnisses dessen Inhalt in eine CSV-Datei eingelesen werden soll. Wurde kein Verzeichnis übergeben wird das aktuelle Verzeichnis verwendet.

<Datei-Filter>
Sollen nicht alle Dateien das Verzeichnisses eingelesen werden kann ein Filter in der Form *.pdf verwendet werden. Hier ist nur ein Wildcard (*) erlaubt.

<Typ-Filter>
Es kann selektiert werden ob nur Dateien, nur Verzeichnisse oder beides ausgegeben werden sollen.
0=Dateien und Unterverzeichnisse
1=Nur Dateien
2=Nur Unterverzeichnisse

Wird keine Angabe gemacht werden Dateien und Unterverzeichnisse ausgegeben.

/Datei:<CSV-Datei> **(optional)**
Übergibt an **ERPAdmin** den Dateinamen in dem das Ergebnis gespeichert werden soll. Wird kein Dateiname angegeben wird das Ergebnis in der Datei „**Verzeichnis-Inhalt.csv**“ gespeichert.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „Kapitel Platzhalter und variable Parameter“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

Beispiel:
ERPAdmin -Dir2CSVc:\ERPAdmin\ *.pdf /Datei:PDF-Dateien.csv /Email:Demo@ERPAdmin.de /HEADER

Erzeuge die CSV-Datei „**PDF-Dateien.csv**“ und gibt darin alle PDF-Dateien aus dem Verzeichnis **C:\ERPAdmin** aus. Die Datei besitzt eine Kopfzeile (/HEADER). Die Datei wird dann an die E-Mail-Adresse Demo@ERPAdmin.de gesendet.

Die Ergebnisdatei könnte wie folgt aussehen:

	A	B	C	D	E	F	G	H	I	J	K
1	**Pos**	**Name**	**Path**	**Fullname**	**Type**	**Size**	**Attributes**	**Created**	**LastModified**	**Ext**	**NameWithoutExt**
2	0	_Logfiles	C:\ERPAdmin\Beispiele\Dir2CSV_Logfiles\		<DIR>			06.04.2020 14:28	06.04.2020 15:27		
3	1	ERPAdmin PHW10.log	C:\ERPAdmin\Beispiele\Dir2CSV\	C:\ERPAdmin\Beispiele\Dir2CSV\ERPAdmin PHW10.log	Textdokument	6985	32	06.04.2020 14:28	06.04.2020 15:28	log	ERPAdmin PHW10
4	2	ERPAdmin.exe	C:\ERPAdmin\Beispiele\Dir2CSV\	C:\ERPAdmin\Beispiele\Dir2CSV\ERPAdmin.exe	Anwendung	1062912	32	06.04.2020 14:26	06.04.2020 15:28	exe	ERPAdmin
5	3	VAR.ini	C:\ERPAdmin\Beispiele\Dir2CSV\	C:\ERPAdmin\Beispiele\Dir2CSV\VAR.ini	Konfigurationseinstellungen	565	32	06.04.2020 14:27	09.04.2019 15:23	ini	VAR
6	4	_runme.cmd	C:\ERPAdmin\Beispiele\Dir2CSV\	C:\ERPAdmin\Beispiele\Dir2CSV_runme.cmd	Windows-Befehlsskript	229	32	06.04.2020 14:27	06.04.2020 15:05	cmd	_runme
7	5	__RUNNING_PHW10__	C:\ERPAdmin\Beispiele\Dir2CSV\	C:\ERPAdmin\Beispiele\Dir2CSV__RUNNING_PHW10__	Datei	80	32	06.04.2020 15:28	06.04.2020 15:28		__RUNNING_PHW10__

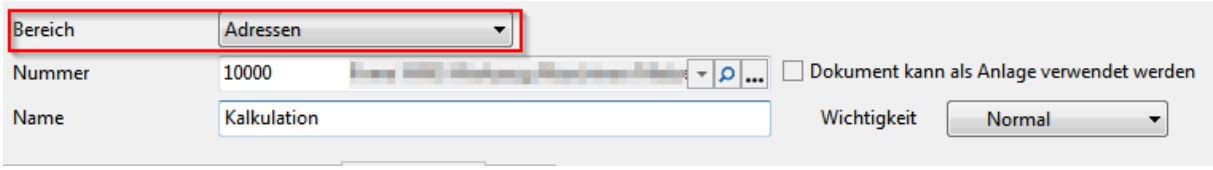
47 Der Befehl „-DokumentDoublettenCheck“ PRO

ERPAdmin -DokumentDoublettenCheck <Bereich> /Von:<Bereich von> /Bis: /Datei: /Check /Delete /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /Header

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit dem Befehl **DokumentDoublettenCheck** kann **ERPAdmin** die Dokumentendatenbank auf Doubletten untersuchen und diese löschen. Bei der Prüfung wird auf Gleichheit der Dokumenten-Nummer, des Dokumenten-Namens und der Dateierweiterung geprüft. Sind Dokumente mit gleicher Nummer, dem gleichem Dokumenten-Namen (nicht Dateiname) und der gleichen Dateierweiterung vorhanden werden diese als Doublette eingestuft. Der Benutzer entscheidet über den Befehl ob er nur eine Prüfliste erstellen oder ob die Löschung dieser Dokumente durchführen möchte. Das Ergebnis wird in der Datei **CSV-Ergebnis** gespeichert.

<Bereich>
Gibt an, welcher Bereich der Dokumentendatenbank auf Doubletten hin geprüft werden soll.
Beispiel: **Adressen**



Unterstützt werden aktuell die folgenden Bereiche:

- Adressen
- Artikel
- Projekte
- Warengruppen
- Vorgang

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen, nach denen auch sortiert werden kann.

Beispiel: Bereich=Adressen, **/Von:Nr=10000** => entspricht Adressnummer=10000

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

Beispiel: Bereich=Adressen, **/Bis:Nr=19999** => entspricht Adressnummer=19999

/Datei:<Dateiname> **(optional)**
Die Rückmeldung an den Benutzer erfolgt im **ERPAdmin** Verzeichnis in der Datei **Doubletten_Bereich.csv**. Dabei steht **Bereich** für „Adressen“ oder „Artikel“, etc. Über die Option **/Datei:** kann ein beliebiger Dateiname übergeben werden für die Speicherung.

/Check und **/Delete** **(optional)**
Für die Doubletten-Prüfung gibt es zwei Befehle.

- **/CHECK**
- **/DELETE**

Mit dem Befehl /**CHECK** wird nur eine Überprüfung durchgeführt auf vorhandene Doubletten. Der Benutzer bekommt eine Rückmeldung ob Doubletten vorhanden sind und um welche Dokumente es sich handelt. Es werden keine Dokumente gelöscht.

Erst mit dem Befehl /**DELETE** werden gefundene Doubletten aus den Dokumenten gelöscht.

Wird kein **Befehl** angegeben wird /**CHECK** als Befehl verwendet.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **/Datei:** mit den Doubletten sendet. Wurden keine Doubletten gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „Kapitel Platzhalter und variable Parameter“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/HEADER **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit den Spaltenbezeichnungen.

Beispiele:

ERPAdmin -DokumentDoublettenCheck Adressen /Von:Nr=10000
Führt einen Doubletten-Prüfung im Bereich „**Adressen**“ durch für die Adressen **10000**. Das Ergebnis wird in der Datei **.\Doubletten_Adressen.csv**“ gespeichert.

ERPAdmin -DokumentDoublettenCheck Adressen /Von:Nr=10000 /Bis:Nr=19999
Führt einen Doubletten-Prüfung im Bereich „**Adressen**“ durch für die Adressen **10000** bis **19999**. Das Ergebnis wird in der Datei **.\Doubletten_Adressen.csv** gespeichert.

ERPAdmin -DokumentDoublettenCheck Adressen C:\ERPAdmin\Doubletten-Liste.txt /Von:Nr=10000 /Bis:Nr=19999 /DELETE
Lösche alle gefundenen Doubletten im Bereich „**Adressen**“ durch für die Adressen **10000** bis **19999**. Das Ergebnis wird in der Datei **C:\ERPAdmin\Doubletten-Liste.txt** gespeichert.

ERPAdmin -DokumentDoublettenCheck Artikel /Von:Nr=A2000 /Bis:Nr=Z9999 /DELETE
Lösche alle gefundenen Dokumenten-Doubletten im Bereich „**Artikel**“ durch für die Artikel **A2000** bis **Z9999**. Das Ergebnis wird in der Datei **.\Doubletten_Artikel.csv** gespeichert.

Hinweis:
Sind in dem Dokumentennamen aus der **büro+** Datenbank Semikolons (;) enthalten werden diese für die CSV-Datei raus gefiltert.

Achtung:
Bei der Prüfung auf eine Doublette wird nicht der tatsächliche Dateiname des Dokumentes beachtet, sondern nur der zugeordnete Name in **büro+.**

Empfehlung:
Führen Sie auf jeden Fall immer erst einen /**CHECK** Befehl aus und prüfen Sie dann ob die Ergebnisliste auch dem erwarteten entspricht. Erst wenn Sie sicher sind das alle Doubletten korrekt erkannt wurde und es keine fehlerhafte Erkennung gab führen Sie den Aufruf mit dem Befehl /**DELETE** aus.

48 Der Befehl „-DokumentLinkCheck“ PRO

ERPAdmin -DokumentLinkCheck /Datei: /Von:<Bereich von> /Bis: /Filter:<Filterbedingung> /<Befehl> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /HEADER

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin führt eine überprüft aller verlinkten Dokumente durch ob die verlinkte Datei noch existiert und erzeugt eine CSV-Datei mit allen fehlerhaften Verlinkungen. Die Ausführung von **ERPAdmin** muss auf dem Server erfolgen auf dem auch der Datenbank-Server installieret ist, weil die Pfadangaben aus Sicht dieses Servers hinterlegt wurden.

/Datei:<CSV-Ergebnis>
Die Rückmeldung an den Benutzer erfolgt im **ERPAdmin** Verzeichnis in der Datei **CSV-Ergebnis**. Hier wird der Name der CSV-Datei mit Fehlerhaften Verknüpfungen aus dem Datenbankbereich der Dokumente angegeben.

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen, nach denen auch sortiert werden kann.

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/<Befehl> **(optional)**
Für den **DokumentLinckCheck** gibt es zwei Befehle:

- **/CHECK**
- **/DELETE**

/CHECK überprüft alle Dokumenten-Verlinkungen und erzeugt eine CSV-Datei mit allen gefundenen Fehlern.
/DELETE überprüft alle Dokumenten-Verlinkungen und löschet alle fehlerhaften Verlinkungen und erzeugt eine CSV-Datei mit den gelöschten Dokumenten.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **<CSV-Ergebnis>** mit den fehlerhaften Dokumenten-Links sendet. Wurden keine fehlerhaften Verlinkungen gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „Kapitel Platzhalter und variable Parameter“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/HEADER **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit den Spaltenbezeichnungen.

> **Hinweis:**
> Führen Sie immer zuerst den Befehl **/CHECK** aus und überprüfen Sie ob die Ergebnisse korrekt sind. Führen erst dann den Befehl **/DELETE** aus.

Beispiele:

ERPAdmin –DokumentLinkCheck/Datei: Linkliste.csv
oder
ERPAdmin –DokumentLinkCheck /Datei:Linkliste.csv /CHECK
Führt eine Überprüfung (**/CHECK**) aller Dokumente aus und speichert das Ergebnis in der CSV-Datei **Linkliste.csv**

ERPAdmin –DokumentLinkCheck /Datei:Linkliste.csv /Von:Rechnung1800000 /Bis:Rechnung1899999 /CHECK
Führt eine Überprüfung aller Dokumente mit dem Namen „**Rechnung1800000**“ bis „**Rechnung1899999**“ aus und speichert das Ergebnis in der CSV-Datei **Linkliste.csv**

ERPAdmin –DokumentLinkCheck /Datei:Linkliste.csv /DELETE
Führt eine Überprüfung aller Dokumente aus und löscht alle fehlerhaften Links. speichert das Ergebnis in der CSV-Datei **Linkliste.csv**

ERPAdmin –DokumentLinkCheck /Datei:FehlerhafteLinks.csv /DELETE /Email:Kunde@Firma.de
Führt eine Überprüfung aller Dokumente aus und löscht (**/DELETE**) alle fehlerhaften Links. speichert das Ergebnis in der CSV-Datei **FehlerhafteLinks.csv** und sendet diese an die E-Mail-Adresse **Kunde@Firma.de.**

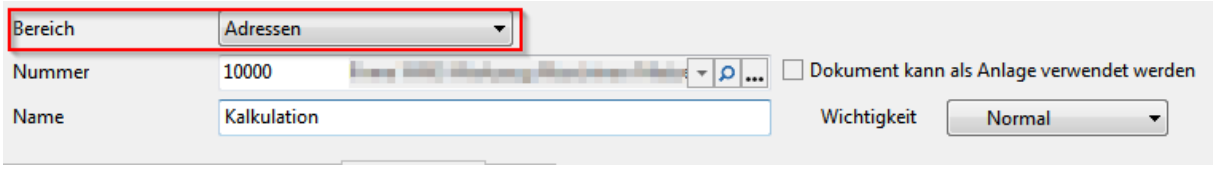
49 Der Befehl „-DokumentSearchAndReplace“ PRO

ERPAdmin -DokumentSearchAndReplace <Bereich> <Suchen> <Ersetzen> /Von:<Bereich von> /Bis:<Bereich bis> /Datei: /Check /Replace /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /Header /NoLockError /Cond:<Bedingung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin führt ein **„Suchen und Ersetzen**“ auf dem Dokumenten-Feld „**Name**“ aller ausgewählten Dokumente durch und erzeugt eine CSV-Datei als Ergebnis.

<Bereich>
Gibt an, welcher Bereich der Dokumentendatenbank beim „Suchen und Ersetzen“ bearbeitet werden soll.
Beispiel: **Adressen**



Unterstützt werden aktuell die folgenden Bereiche:

- Adressen
- Artikel
- Projekte
- Warengruppen
- Vorgang

<Suchen>
Übergibt eine Zeichenkette die im Feld „Name“ ersetzt werden soll. Soll der komplette Inhalt des Feldes „Name“ ersetzt werden kann der Platzhalter **@NAME@** verwendet werden.

<Ersetzen>
Übergibt die Zeichenkette durch die „**Suchen“** ersetzt werden soll. Folgende Platzhalter können dabei verwendet werden:

- **@ERSTDAT@** Wird durch den Feldinhalt „**Erstelldatum**“ ersetzt z.B. **22.03.2019 15:23:17**
- **@ERSTDAT.DATE@** Wird durch das Datum aus „**Erstelldatum**“ ersetzt z.B. **22.03.2019**
- **@ERSTDAT.TIME@** Wird durch die Uhrzeit aus „**Erstelldatum**“ ersetzt z.B. **15:23:17**
- **@ERSTDAT.YYYY@** Wird durch die Jahreszahl aus „**Erstelldatum**“ ersetzt z.B. **2019**
- **@ERSTDAT.MM@** Wird durch den Monat aus „**Erstelldatum**“ ersetzt z.B. **03**
- **@ERSTDAT.DD@** Wird durch den Tag aus „**Erstelldatum**“ ersetzt z.B. **15**

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen, nach denen auch sortiert werden kann.

Beispiel: Bereich=Adressen, **/Von:Nr=10000** => entspricht Adressnummer=10000

/Bis:<Bereich bis>(optional)
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

Beispiel: Bereich=Adressen, **/Bis:Nr=19999** => entspricht Adressnummer=19999

/Datei:<Dateiname> **(optional)**
Die Rückmeldung an den Benutzer erfolgt im **ERPAdmin** Verzeichnis in der Datei **DokumenteS&R_Bereich.csv**. Dabei steht **Bereich** für „Adressen“ oder „Artikel“. Über die Option **/Datei:** kann ein beliebiger anderer Dateiname übergeben werden für die Speicherung.

/Check und **/Replace** **(optional)**
Für die „Suchen und Ersetzen“-Funktion gibt es zwei Befehle.

- **/CHECK**
- **/REPLACE**

Mit dem Befehl /**CHECK** wird nur eine Überprüfung durchgeführt welche Dokumentennamen betroffen sind. Der Benutzer bekommt eine Rückmeldung um welche Dokumente es sich handelt. Es werden keine Dokumentennamen verändert.

Erst mit dem Befehl /**REPLACE** werden gefundene Dokumentennamen auch abgeändert.

Wird kein **Befehl** angegeben wird /**CHECK** als Befehl verwendet.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **/Datei:** mit den Dokumentennamen sendet. Wurden keine Dokumente gefunden, wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/HEADER **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit den Spaltenbezeichnungen.

/NoLockError **(optional)**
Kann ein Datensatz nicht geändert werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

/Cond:<Bedingung> **(optional)**
Wird als Bedingung „**EQ**“ (Equal) angegeben muss der Dokumenten-Namen und der Suchstring vollständig übereinstimmen. Damit wird eine Teilersetzung im Dokumenten-Namen unterbunden.

Tipp:
Führen Sie immer zuerst den Befehl **/CHECK** aus und überprüfen Sie ob die Ergebnisse korrekt sind. Führen erst dann den Befehl **/REPLACE** aus.

Beispiele:

ERPAdmin -DokumentSearchAndReplace Adressen 2018_ 2018. /Von:Nr=10000 /Bis:Nr=19999 /Check /Email:ERPAdmin@Systemhaus-HORN.de /Header
Durchsuche alle Dokumentennamen im Bereich **Adressen** im Nummernkreis **10000** bis **19999** und überprüft bei welche Dokumentennamen die Zeichenkette **"2018_"** durch **"2018."** Ersetzt würde. Die CSV-Ergebnisdatei wird mit einem Header erstellt (**/Header**) und wird per E-Mail an die Adresse ERPAdmin@Systemhaus-HORN.de gesendet. Es wird keine Veränderung an den Daten vorgenommen.

ERPAdmin -DokumentSearchAndReplace Adressen "2018_" "2018." /Von:Nr=10000 /Bis:Nr=19999 /Replace /Email:ERPAdmin@Systemhaus-HORN.de /Header
Durchsuche alle Dokumentennamen im Bereich **Adressen** im Nummernkreis **10000** bis **19999** und ersetzt im Dokumentennamen die Zeichenkette **"2018_"** durch **"2018."**. Die CSV-Ergebnisdatei wird mit einem Header erstellt (**/Header**) und wird per E-Mail an die Adresse ERPAdmin@Systemhaus-HORN.de gesendet.

ERPAdmin -DokumentSearchAndReplace Adressen "@NAME@" "@ERSTDAT-YYYY@.@ERSTDAT-MM@ @NAME@" /Von:Nr=10000 /Bis:Nr=19999 /Replace /Email:ERPAdmin@Systemhaus-HORN.de /Header
Durchsuche alle Dokumentennamen im Bereich **Adressen** im Nummernkreis **10000** bis **19999** und stellt den Dokumentennamen die **Jahreszahl.Monat** voran. Diese Daten werden aus dem Erstelldatum des Dokumentes ausgelesen. Damit erscheinen die Dokumente dann sortiert nach dem Erstellungsdatum in der Detail-Ansicht. Die CSV-Ergebnisdatei wird mit einem Header erstellt (**/Header**) und wird per E-Mail an die Adresse ERPAdmin@Systemhaus-HORN.de gesendet.

Aus dem Dokumentennamen „**Preisliste Aktualitätsservice**“ wird dann “**2019.03 Preisliste Aktualitätsservice**“.

50 Der Befehl „-Email“ BAS

ERPAdmin -Email <Email-To> <Email-Betreff> < Anhang> <Email-Template> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /NoFileNotExistError

Der Aufruf **„ERPAdmin -Email“** versendet eine Test-E-Mail an die in der **VAR.INI** hinterlegte Adresse **C_OK_TO**.

Soll eine „normale“ E-Mail versenden, kann dies im Text- oder HTML-Format erfolgen. Der E-Mail-Versand erfordert das vorhanden sein des „**Microsoft Collaboration Data Objects**“ (CDO). Auf den meisten Windows System ist dieses bereits installiert.

<Email-To>
Gibt die Empfänger E-Mail-Adresse an. Soll die E-Mail an mehrere Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

<Email-Betreff> **(optional)**
Gibt den Betreff (Subject) der E-Mail an.

<Anhang> **(optional)**
Übergibt eine Datei inkl. Pfadangaben die per E-Mail versendet werden soll. Mehrere Dateien sind mit einem Semikolon getrennt zur übergeben.

<Email-Template> **(optional)**
Übergibt eine Datei inkl. Pfadangabe an **ERPAdmin** die als E-Mail-Text versendet werden soll. Handelt es sich um eine E-Mail im text-Format ist die Datei-Endung **.txt**. Soll der Versand im HTML-Format erfolgen muss die Datei die Endung **.html** haben.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/NoFileNotExistError **(optional)**
Ist eine Anhang-Datei nicht vorhanden und soll in diesem Fall keine Fehlermeldung erzeugt werden, kann die Option **/NoFileNotExistError** verwendet werden.

Folgende Einstellungen sind in der **VAR.INI** vorzunehmen:

C_SMTPSERVER	SMTP-Server/IP-Adresse
C_SMTPSERVERPORT	SMTP-Port (25/465/587)
C_SMTPUSER	SMTP-Benutzername
C_SMTPPWD	SMTP-Kennwort
C_SMTPSSL	SMTP-Versand mit SSL-Verschlüsselung (TRUE/FALSE)
C_SMTPTLS	SMTP-Versand mit TLS-Verschlüsselung (TRUE/FALSE)
C_EMAIL_CC	E-Mail-Adresse für CC (Kopie)

C_EMAIL_BCC E-Mail-Adresse für BCC (Blinde-Kopie)

Beispiele:

ERPAdmin -Email Demo@ERPAdmin.de
ERPAdmin versendet eine Test-E-Mail an die Adresse **Demo@ERPAdmin.de**

ERPAdmin -Email Demo@ERPAdmin.de "Neue E-Mail" "c:\ERPAdmin\ ERPAdmin.log"
ERPAdmin versendet eine Test-E-Mail an **Demo@ERPAdmin.de** mit dem Betreff „**Neue Email“** und hängt die Datei **c:\ERPAdmin\ERPAdmin.log** an.

ERPAdmin -Email Demo@ERPAdmin.de "E-Mail Betreff" "c:\ERPAdmin\ERPAdmin.log" "c:\ERPAdmin\Email-Inhalt.html"
ERPAdmin versendet eine **HTML-E-Mail** an **Demo@ERPAdmin.de** mit dem Betreff „**E-Mail Betreff**“ und hängt die Datei „**c:\ERPAdmin\ ERPAdmin.log**“ an. Der Inhalt der E-Mail kommt aus der **HTML-Datei „c:\ERPAdmin\Email-Inhalt.html**“. An der Endung **.HTML** erkennt das Programm, ob es sich um eine HTML-formatierte oder bei „**.txt**“ um eine Text-basierte E-Mail handelt.

> **Hinweis**:
> Soll bei dem Aufruf z. B. kein Betreff übergegeben werden, muss ein Leerstring an dessen Stelle übergeben werden. Verwenden Sie dazu zwei Anführungszeichen. Als Betreff wird in diesem Fall der Betreff aus der Datei **VAR.INI** „**C_OK_SUBJECT“** verwendet.

ERPAdmin -Email Demo@ERPAdmin.de " " " " "c:\ERPAdmin\Email-Inhalt.html"
ERPAdmin versendet eine HTML-E-Mail an **Demo@ERPAdmin.de** ohne Anhang mit dem Betreff aus der **VAR.INI**. Der Inhalt der E-Mail kommt aus der Datei „**c:\ERPAdmin\Email-Inhalt.html**“ und wird im HTML-Format versendet.

Kombiniert man den Export-Befehl mit dem E-Mail-Versand, kann ein Export an einen Benutzer per E-Mail gesendet werden. Dazu müssen die beiden Befehle nacheinander ausgeführt werden.

ERPAdmin -Export "Artikel.bpl" "C:\ERPAdmin\Artikel_Export.csv"
ERPAdmin -Email "Kunde@test.de" "Neue Artikeldaten" " " "C:\ERPAdmin\Artikel_Export.csv" /NoFIleNotExistError

51 Der Befehl „-EmailFromFile“ PRO

ERPAdmin -EmailFromFile <Datei> <Template-Datei> /NoFileNotExistError /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /Subject:<Betreff>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Die Voraussetzungen für den E-Mail-Versand sind dieselben wie beim Befehl „**ERPAdmin -Email**“. Die Funktionsweise ist aber eine abweichende.

Mit diesem **ERPAdmin**-Befehl können E-Mails anhand einer der Daten aus einer CSV-Datei versendet werden. In der CSV-Datei können folgende Angaben übergeben werden welche über die Kopfzeile zugeordnet werden. Die reigenfolge der Felder spielt keine Rolle.

- **Email-To**: E-Mail-Adresse Empfänger
- **Email-Subject**: E-Mail-Betreff
- **Email-Attachment**: List mit Dateien die an E-Mail angehängt werden sollen.
  Mehrere Dateien sind durch ein Komma (,) zu trennen.
- **Email-CC**: E-Mail-Adresse der Empfänger im CC
  Mehrere E-Mail-Adressen sind durch ein Komma (,) zu trennen.
- **Email-BCC**: E-Mail-Adresse der Empfänger im BCC
  Mehrere E-Mail-Adressen sind durch ein Komma (,) zu trennen.

Die Versendung dieser E-Mails erfolgt mit dem Absender **C_EMAIL_FROM** aus der **VAR.INI** Datei.

<Datei>
Dateiname inkl. Pfadangabe. Hier kann entweder eine **ANSI-CSV-Datei mit Kopfzeile** übergeben werden oder ein **büro+-Export-Layout** aus dem Bereich „**Offene-Posten**“ mit der Endung **.bpl**.

> **Hinweis**:
> Wird ein büro+ Export-Layout übergeben, muss zusätzlich noch eine Angabe gemacht werden aus welchem Programm-Bereich (Auftrag oder Fibu) der Export der Offenen-Posten erfolgen soll. Hierfür ist die Angabe **/GL:AU** oder **/GL:FI** notwendig.

Wird eine CSV-Datei übergeben, müssen min. die folgenden Spalten enthalten sein:

- **Email-To**
- **Email-Subject**

Enthalten mehrere aufeinander folgende Datensätze dieselbe E-Mail-Empfänger-Adresse, werden die Dateianhänge dieser Datensätze gesammelt und der Versand erfolgt in einer einzigen E-Mail mit den Daten des letzten Datensatzes dieses Empfängers. Die **CSV-Datei** sollte deshalb nach der Spalte **Email-To** sortiert sein wenn nur eine E-Mail pro E-Mail-Adresse versendet werden soll.

<Template-Datei>
Die **Template-Datei** enthält ist eine Text-Datei im ANSI-Format welche den Inhalt der E-Mail enthält im HTML-Format. Hier können alle Platzhalter von **ERPAdmin** (Siehe Kapitel Platzhalter und variable

Parameter) verwendet werden. Weiterhin können alle Daten aus der **CSV-Datei** über Platzhalter zugeordnet werden. Dazu muss lediglich die Spaltenüberschrift mit einen **@**-Zeichen vorne und hinten eingefügt werden.
Beispiel:
Der Platzhalter **@Anrede@** wird mit dem Inhalt der **CSV-Datei** aus der Spalte „**Anrede**“ befüllt.

/NoFileNotExistError **(optional)**
Wird eine Datei aus der Spalte **Email-Attachment** nicht gefunden erfolgt der Versand an den Empfänger ohne Anhang. Dies gilt auch, wenn nur einer von mehreren Anhängen nicht gefunden wurde. Soll der Versand mit den existierenden Anhängen erfolgen und die fehlerhaften Anhänge weggelassen werden, kann dieser Option angegeben übergeben werden.

/EmailCC:<E-Mail-Adresse> **(optional)**
Wird kein Wert „**Email_CC**“ in der **Datei** übergeben, kann hier ein Standard-Wert übergeben werden. Dieser gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Wird kein Wert „**Email_BCC**“ in der **Datei** übergeben, kann hier ein Standard-Wert übergeben werden. Dieser gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/Subject:<Betreff> **(optional)**
Es kann ein Betreff übergeben werden der für den Versand aller E-Mails verwendet werden soll. In diesem Fall wird keine Spalte „**Email-Subject**“ in der Datei benötigt.

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

Beispiel:

Am Ende eines Monats sollen alle Kunden im EU-Ausland eine E-Mail bekommen mit den angehängten Rechnungen als PDF um hierfür eine Gelangensbestätigung anzufordern.

Zuerst muss hierfür ein Export aus dem Vorgangs-Archiv erfolgen für den Zeitraum „Letzter Monat“
ERPAdmin -VorgangExport Vorgangliste.csv /Von:Dat=@PRIMO-1@ /Bis:Dat=@ULTIMO-1@ "/Filter:Art=70 and UstKat=3" /His

Die Ergebnis-Datei **Vorgangliste.csv** könnte wie folgt aussehen:

	A	B	C	D	E	F	G
1	EMail-To	Email-Subject	Email-Attachment	BelegNr	Dat	PreisNt	Anrede
2	[redacted]	Rechnungen aus Monat @VORMONATNAME@ für Gelangungsbestätigung	Rechnung_RE2137754.pdf	RE2137754	14.01.2021	90	Sehr geehrter [redacted]
3	[redacted]	Rechnungen aus Monat @VORMONATNAME@ für Gelangungsbestätigung	Rechnung_RE2138164.pdf	RE2138164	17.05.2021	149	Sehr geehrter [redacted]
4	[redacted]	Rechnungen aus Monat @VORMONATNAME@ für Gelangungsbestätigung	Rechnung_RE2138572.pdf	RE2138572	13.09.2021	2.412,00	Sehr geehrter [redacted]
5	[redacted]	Rechnungen aus Monat @VORMONATNAME@ für Gelangungsbestätigung	Rechnung_RE2138576.pdf	RE2138576	13.09.2021	3.444,00	Sehr geehrter [redacted]

Mit dem folgenden **ERPAdmin**-Befehl werden die Rechnungen aus der Spalte **BelegNr** als PDF ausgeben und unter dem Dateinamen aus der Spalte **Email-Attachment** gespeichert.
ERPAdmin -VorgangArchivDruckenDatei 70 Rechnung-PDF BelegNr BelegNr Vorgangsliste.csv /Datei:Email-Attachment

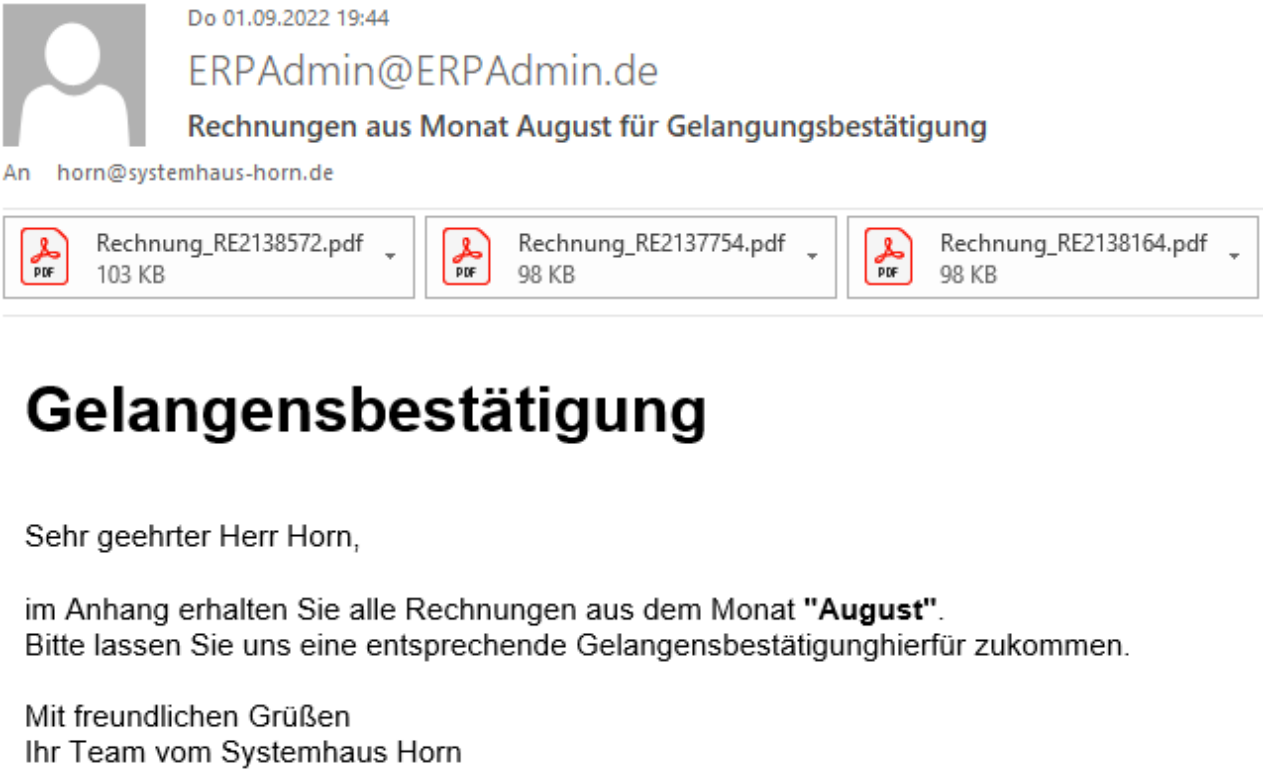
Die Datei **Vorgangliste.csv** muss noch nach der Spalte „**Email-To**“ sortiert werden damit pro Empfänger nur eine Email mit allen Anhängen erzeugt werden kann.
ERPAdmin -CSVKonverter Vorgangliste.csv Sort 1 /Header

Jetzt erfolgt der eigentliche E-Mail-Versand. Da die Datei **Vorgangliste.csv** nach der Spalte **Email-To** sortiert ist, werden alle Dokumente die an die gleiche E-Mail-Adresse gehen in einer E-Mail zusammengefasst.
ERPAdmin -EmailFromFile Gelangensbestaetigung.html Vorgangliste.csv

Die verwendete Templatedatei „**Gelangensbestaetigung.html**“ sieht hier folgt aus:

```
<P>
<H1> Gelangensbestätigung </H1>
<BR>
@Anrede@<BR>
<BR>
im Anhang erhalten Sie alle Rechnungen aus dem Monat <B>"@VORMONATNAME@"</B>.<BR>
Bitte lassen Sie uns eine entsprechende Gelangensbestätigung hierfür zukommen.<BR>
<BR>
Mit freundlichen Grüßen<BR>
Ihr Team vom Systemhaus Horn
<BR>
<BR>
```

Die E-Mail kommt dann so beim Kunden an:



52 Der Befehl „-EmailNotify“ PRO

ERPAdmin -MailNotify <CSV-Datei> <E-Mail-Adresse> <Betreff> <E-Mail-Vorlage> /NoFileNotExistError /Cond:<Wechselbedingung> /Datei:<Attachment> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Mit diesem Befehl werden Anwender per E-Mail über bestimmt Zustände oder Sachlager in **büro+** informiert. Dies könnten z.B. sein:

- Anstehenden Termine
- Lagermindestbestand unterschritten
- Liefertermin für Kunden überschritten
- Liefertermin der Lieferanten überschritten
- Überfällige Offene-Posten vorhanden
- SEPA-Einzüge müssen an Bank übergeben werden für Fristgerechten Einzug
- Liste aller Belege aus dem vergangenen Monat an alle Kunden senden

Die sind nur einige Beispiele für eine mögliche Anwendung.

Basis ist immer eine **CSV-Datei** mit Informationen welche per Export oder **ReadData** aus **büro+** erzeugt wurden oder welche auch aus einem anderen Programm kommen kann.

<CSV-Datei>
Mit diesem Parameter wird eine Datei im Format ANSI-CSV übergeben. **Die CSV-Datei muss eine Kopfzeile enthalten** damit die einzelnen Spalten angesprochen werden können. Alle Spalten aus dieser Datei können in der E-Mail als Tabelle ausgegeben werden. Sofern es sich um Daten aus **büro+** handelt ist auch ein

direkter Sprung per Doppelklick aus der E-Mail direkt in den Datensatz von **büro+** möglich sofern sich ein **büro+** Client auf dem PC befindet auf dem die E-Mail gelesen wird.

<E-Mail-Adresse>
E-Mail-Adresse des Empfängers an den die Nachricht versendet werden soll. Mehrere E-Mail-Adressen sind durch ein Semikolon (**;**) zu trennen.

Versendung mehrere Emails an verschiedene Empfänger
Der Versand an mehrere verschiedenen Empfänger ist aber auch möglich. Dazu müssen die E-Mail-Adresse der Empfänger bereits in der **CSV-Datei** in einer Spalte enthalten sein. Die **CSV-Datei** sollte auch nach dieser Spalte sortiert sein. Anstelle einer E-Mail-Adresse wird jetzt der Spalten-Bezeichner (Kopfzeile) dieser Spalte übergeben werden.

Wichtig: In jedem Datensatz der E-Mail-Spalte muss eine gültige E-Mail-Adresse vorhanden sein.

ERPAdmin erzeugt jetzt für alle aufeinanderfolgenden gleichen E-Mail-Adressen jeweils eine E-Mail und versendet diese an die Empfänger.

<Betreff>
Betreff der E-Mail Hier können auch Platzhalter verwendet werden Lesen Sie dazu das Kapitel „“

<E-Mail-Vorlage>
Gib eine Vorlagendatei mit dem Inhalt der E-Mail-Nachricht an die versendet werden soll. Die Zuordnung zu den Daten aus der CSV-Datei erfolgt über Platzhalter. Als Platzhalter werden die Spaltenbezeichnungen aus der CSV-Datei verwendet denen ein @-Zeichen voran und nachgestellt werden müssen.

Beispiel: Die Daten aus der Spalte „**Vog.BelegNr**“ soll ausgegeben werden. Der Platzhalter heißt dann: **@Vog.BelegNr@**

/NoFileNotExistError **(optional)**
Existiert keine **CSV-Datei** zur Verarbeitung und soll in diesem Fall keine Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** angegeben werden.

/Cond:<Wechselbedingung> **(optional)**
Gibt den Spaltennamen eines Datenfeldes an nach dem die Ausgabe sortiert ist. Ändert sich der Wert, können diese Daten z.B. mit einer anderen Hintergrundfarbe ausgegeben werden. Damit kann eine optische besser lesbare Darstellung erfolgen.

/Datei:<Attachment> **(optional)**
Hier kann ein Dateianhang übergeben werden der mit versendet werden soll.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Beispiel:
Einmal am Tag soll der die Einkaufsabteilung per E-Mail darüber informiert werden, bei welchen Artikel der Lager-Mindestbestand unterschritten wurde damit ggf. Nachbestellungen ausgelöst werden können.

Die Bereitstellung der Daten könnte wie folgt aussehen:

ERPAdmin -ReadData Lager ArtNr;ArtNrInfo;LagNr;Mge;MinMge /Datei:Lagerbestand.csv /Von:LagNr=HRLAGER /Bis:LagNr=HRLAGER "/Filter:Mge<MinMge" /Header

Mit obigem Befehl wird die CSV-Datei „**Lagerbestand.csv**“ mit Kopfzeile (**/Header**) erzeugt welche alle Artikel aus dem **Lager** „HRLAGER“ (**/Von:LagNr=HRLAGER /Bis:LagNr=HRLAGER**) enthält, bei denen der Mindestbestand **"/Filter:Mge<MinMge"** unterschritten wurde.

Hinweis:
Anstelle des Befehls **ERPAdmin -ReadData** können auch mit dem Befehl **ERPAdmin -Export** die gewünschten Daten aus **büro+** als Export erzeugt werden.

Das Ergebnis ist eine CSV-Datei die wie folgt aussieht:

	A	B	C	D	E
1	ArtNr	ArtNrInfo	LagNr	Mge	MinMge
2	3267	3267 Edding	HRLAGER	50	75
3	3270	3270 Edding	HRLAGER	0	50
4	3273	3273 Edding	HRLAGER	0	20
5	3275	3275 Edding	HRLAGER	0	15
6	3277	3277 Spitze f	HRLAGER	0	10
7	3282	3282 Edding	HRLAGER	0	25
8	3283	3283 Edding	HRLAGER	100	250
9	3284	3284 Edding	HRLAGER	0	50
10	3429	3429 Edding	HRLAGER	0	25

Die Kopfzeilen dieser Datei können nun als Platzhalter in unserer E-Mail verwendet werden. Den Spaltenüberschriften wird dazu ein **@** Voran- und nachgestellt. Die Platzhalter heißen dann wie folgt:

- Spalte **ArtNr:** **@ArtNr@**
- Spalte **ArtNrInfo** **@ArztNrInfo@**
- Spalte **LagNr** **@LagNr@**
- Spalte **Mge** **@Mge@**
- Spalte **MinMge** **@MinMge@**

Bei der E-Mail-Vorlage handelt es sich um eine Datei im HTML-Format. Ein Beispiel dafür könne wie folgt aussehen und steht auf unseren Server zum Download zur Verfügung:

```
<P>
<H1>Der Mindestbestand der nachfolgenden Artikel wurde am @DATE@ unterschritten</H1>
<P>
<BR>
<U><B>Bitte folgende Lagerbestände prüfen:</B></U><BR>
<BR>
<TABLE border=1>
<TR><TH>Artikelnr. / Bezeichnung</TH><TH>LagerNr.</TH><TH>Bestand</TH><TH>min. Bestand</TH></TR>
@DETAILS_BEGIN@
<TR>
        <TD> @ArtNrInfo@ </TD>
        <TD> @LagNr@ </TD>
        <TD><B> @Mge@ </B></TD>
        <TD><FONT color=red><B> @MinMge@ </B></FONT></TD>
</TR>
@DETAILS_END@
</TABLE>
```

Die E-Mail-Vorlagen enthält eine Tabelle. Innerhalb dieser Tabelle wird der Anfang des Bereiches in dem die Daten ausgegeben werden sollen mit **@DETAILS_BEGIN@** markiert und das Ende mit **@DETAILS_END@**. Dazwischen stehen die Platzhalter und markieren die Position an der die entsprechenden Daten **@ArtNrInfo@**, **@LagNr@**, **@Mge@** und **@MinMge@** der CSV-Datei eingefügt werden sollen.

Der Aufruf für die Erzeugung der E-Mail hat dann folgende Gestalt:

ERPAdmin -EmailNotify Lagerbestand.csv Demo@ERPAdmin.de "Mindestbestand unterschritten" Email_Vorlage.html

Die E-Mail die versendet wird sieht wie folgt aus:

Fr 23.07.2021 10:43

ERPAdmin@ERPAdmin.de

Mindesbestand unterschritten

An Demo@ERPAdmin.de

Der Mindestbestand der nachfolgenden Artikel wurde am 23.07.2021 unterschritten

Bitte folgende Lagerbestände prüfen:

Artikelnr. / Bezeichnung	LagerNr.	Bestand	min. Bestand
3267 Edding 3000, schwarz	HRLAGER	50	75
3270 Edding 3000, blau	HRLAGER	0	50
3273 Edding Tinte, schwarz	HRLAGER	0	20
3275 Edding 3000 gelb, 5ml	HRLAGER	0	15
3277 Spitze für Edding 3000, gelb	HRLAGER	0	10
3282 Edding Etikett für 25 ml Flasche	HRLAGER	0	25
3283 Edding 3000 SET verpackt	HRLAGER	100	250
3284 Edding 3000 Papdeckel	HRLAGER	0	50
3429 Edding 3000 SET3	HRLAGER	0	25

Diese Email wurde automatisch am 23.07.2021 10.42.55 mit ERPAdmin erzeugt.

53 Der Befehl „-ExcelCMD“ PRO

ERPAdmin -ExcelCMD <Befehlsdatei> /NoFileNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Der Befehl **ExcelCMD** (Excel-Command) ist eine umfangreiche Funktion von **ERPAdmin** mit vielen Parametern. Hier können ganze Befehlsketten im VB-Script-Dialekt (Visual-Basic-Script) programmiert werden. Die Befehlsdateien werden mit einem beliebigen Texteditor z. B. Notepad erzeugt (ANSI-Format) und erhalten die Endung **„.erp“**.

> **Hinweis:**
> Werden innerhalb der Zuweisungen Excel-Befehle verwendet werden, muss diese in englischer Sprache erfolgen.
>
> Beispiel: Anstelle **"=Verketten(A2;B2)"** verwenden Sie **"=Concatenate(A2;B2)"**
>
> Zur Übersetzung der Befehle empfehlen wir die Webseite http://de.excel-translator.de/

<Befehlsdatei>
Datei welche Anweisungen im VB-Script Dialekt enthält. Wurde **ERPAdmin** mit dem Parameter **„-Install“** einmal installiert, können die Dateien mit der Endung **„.erp“** auch per Doppelklick ausgeführt werden.

/NoFileNotExistError **(optional)**
Wird mit dem Excel-Befehl **XLS_Open(..)** versucht eine Datei zu öffnen welche nicht existiert, wird eine Fehlermeldung mit dieser Option unterdrückt.

Beispiel:
ERPAdmin -ExcelCMD C:\ERPAdmin\Datenkonvertierung.erp

Automatische Konvertierung von Excel-Dateien
Mit den nachfolgenden Befehlen können Excel-Dateien nach einem vorgegebenen Ablaufplan voll automatisch konvertiert und aufbereitet werden, um die Daten anschließend in **büro+** zu importieren. So können wiederkehrende Importe z.B.: Händlerpreislisten ohne händische Anpassungen konvertiert und anschließend direkt importiert werden.

> **Hinweis:**
> Während **ERPAdmin** läuft darf **keine** weitere Excel-Instanz auf dem PC geöffnet werden oder geöffnet sein, da es ansonsten zu Fehlern kommt.

Achtung: Bei der Automatisierung der Excel-Konvertierung über den Taskplaner (Scheduler) kann es zu Fehlern kommen, wenn die folgenden Verzeichnisse nicht auf dem Server existieren:

- **C:\Windows\SysWOW64\config\Desktop** (Nur bei 64 Bit Systemen)
- **C:\Windows\SysWOW64\config\systemprofile\Desktop** (Nur bei 64 Bit Systemen)

- **C:\Windows\System32\config\systemprofile\Desktop**

Die Excel-Befehle in alphabetischer Reihenfolge:

- XLS_AddColSum(xlsCol)
- XLS_AddComment(xlsCell, Value)
- XLS_AddTable(TableName)
- XLS_ClearCol(xlsColumn)
- XLS_ClearRow(xlsRow)
- XLS_ClearRowWithEmptyCol(xlsCell)
- XLS_ClearRowWithColValue(xlsColumn, Value)
- XLS_ColIndex(xlsCell)
- XLS_CopyCol(xlsColFrom, xlsColTo)
- XLS_CopyRangeToTable "xlsRangeFrom", "xlsTabelleFrom", "xlsRangeTo","xlsTabelleTo"
- XLS_DeleteCol(xlsCol)
- XLS_DeleteRowWithEmptyCol(xlsColumn)
- XLS_DeleteTable(xlsTableName)
- XLS_DeleteTablesButNot(xlsTableName)
- XLS_GetValue(xlsCell)
- XLS_GetMaxCol()
- XLS_GetMaxRow()
- XLS_ImportCSVFile(csvFilename, Delimiter, NumberFormat)
- XLS_InsertCol(xlsColumn)
- XLS_InsertRow(xlsRow)
- XLS_New()
- XLS_Open(xlsFilename)
- XLS_RemoveFormelFromCol(xlsColumn)
- XLS_ReplaceCol(xlsColumn, xlsSuchen, xlsErsetzen, Partial)
- XLS_ReplaceISOLand xlsCol
- XLS_RowIndex(xlsCell)
- XLS_Save()
- XLS_SaveAs(xlsFilename, Overwrite)
- XLS_SetRangeFormat(xlsRange, xlsCellFormat)
- XLS_SetRangeValue(xlsRange, Value)
- XLS_SelectTable(xlsTableName)
- XLS_SetName(xlsCell, Name)
- XLS_SetValue(xlsCell, Value)
- XLS_Sort(xlsCell)
- XLS_SortDesc(xlsCell)
- XLS_Quit()

Die Parameter in alphabetischer Reihenfolge:

DescSort	**TRUE**, wenn die Sortierreihenfolge umgekehrt werden soll, ansonsten **FALSE**
Overwrite	**TRUE**, wenn bereits vorhandene Daten überschrieben werden dürfen, ansonsten **FALSE**
Partial	**FALSE**, wenn der Zelleninhalt genau übereinstimmen muss, **TRUE** wenn auch ein Teilstring innerhalb der Zelle ersetzt werden sollen
Value	Zeichenfolge / Feldinhalt
xlsCell	beliebige Zelle innerhalb einer Excel-Tabelle z. B. „**D17**“ oder „**AG23**“
xlsColumn	beliebige Spalte innerhalb der Excel-Tabelle z. B. „**D**“.
xlsErsetzen	Zeichenfolge durch die **xlsSuchen** ersetzt werden soll
xlsFilename	Dateiname einer Excel-Datei z. B.: „**c:\ERPAdmin\Beispiel.xlsx**“
xlsRange	zusammenhängender Zellen-Bereich z. B.: „**B2:B100**“
xlsRow	beliebige Zeile der geöffneten Excel-Tabelle z. B. „**17**“.
xlsSuchen	Zeichenfolge nach der gesucht werden soll
xlsTableName	Name eines Tabellenblattes innerhalb der Excel-Datei z. B.: „**Tabelle1**“

Beispiele der Befehle:

XLS_AddColSum "D"
Fügt am Ende der Spalte **D** die Spaltensumme ein und addiert ab der zweiten Zeile alle Spalten-Werte auf.

XLS_AddComment ("F2", "Nur nummerische Wert zulässig")
Erzeugt einen Kommentar für das Feld **„F2“** mit dem Inhalt **„Nur nummerische Wert zulässig“**.

XLS_AddTable "Daten"
Erzeugt ein neues Arbeitsblatt mit dem Namen „**Daten**“. Wird kein Name angegeben wird ein Name von Excel vergeben z.B. „**Tabelle 4**“

XLS_ClearCol "D"
Löscht den Inhalt der Spalte **„D“**.

XLS_ClearRow "12"
Löscht den Inhalt der Zeile **„12“**.

XLS_ClearRowWithEmptyCol "D"
Löscht den Inhalt aller Zellen in einer Zeile, wenn das jeweilige Feld in der Spalte **„D“** leer ist.

XLS_ClearRowWithColValue "D", „Interessent“
Ist der Inhalt im jeweiligen Feld der Spalte **„D“ = „Interessent“**, wird der Inhalt der kompletten Daten-Zeile gelöscht. Hinweis: Es wird nur den Zelleninhalt gelöscht. Die Zeile bleibt erhalten. Zurück bleiben leere Zellen.

XLS_CopyCol "A", "H"
Kopiert den Inhalt der Spalte “**A**” in die Spalte “**H**”

XLS_CopyRange "A1:A20", "H1:H20"
Kopiert den Inhalt des Bereiches "A1:A20" in den Bereich "H1:H20"

XLS_CopyRangeToTable "A1:A99", "Tabelle1", "H1:H99","Tabelle3"
Kopiert den Inhalt des Bereiches "**A1:A99**" aus "**Tabelle1**" in den Bereich "**H1:H99**" in Tabelle "**Tabelle3**". Beide Tabellen müssen existieren.

XLS_DeleteCol „C“
Löscht die Spalte „**C**“ und verschiebt alle anderen Spalten eins nach links

XLS_DeleteRows "5", "1"
Löscht Zeile „**5**“.

XLS_DeleteRows "5", "10"
Löscht **„10“** Zeilen beginnend mit Zeile **„5“**.

XLS_DeleteRowWithEmptyCol "D"
Löscht eine Datenzeile aus der Tabelle, wenn das jeweilige Feld in der Spalte **„D“** leer ist.

XLS_DeleteRowWithColValue "B", “Interessent”
Lösche alle Zeilen aus der Tabelle, die in der Spalte **“B”** den Wert **“Interessent”** stehen haben.

XLS_DeleteRowWithOutColValue "B", "Interessent", "2"
Lösche alle Zeilen aus der Tabelle, die in der Spalte **“B”** **nicht** den Wert **“Interessent”** stehen haben. Der letzte Parameter **„2“** gibt an, dass erst ab der 2. Zeile mit der Abarbeitung begonnen werden soll damit die Kopfzeile nicht versehentlich gelöscht wird.

XLS_DeleteTable "Tabelle3"
Löscht das Tabellenblatt „**Tabelle3**“ der geöffneten Excel-Datei.

XLS_DeleteTablesButNot "Tabelle1"
Entfernt alle Tabellenblätter aus der geöffneten Excel-Datei **außer** dem Tabellenblatt „**Tabelle1**“.

XLS_GetValue "D12"
Liefert als Ergebnis den Inhalt der Zelle **„D12“**.

XLS_GetMaxCol
Ermittelt die letzte Spalte der aktuellen Tabelle, z. B. **„DC“.**

XLS_GetMaxRow
Ermittelt die maximale Anzahl an belegten Zeilen in der Excel-Tabelle z. B. „**2423**“.

XLS_ImportCSVFile "c:\ERPAdmin\Adressen.csv", ";", "@"
Importiert die CSV-Datei „**c:\ERPAdmin\Adressen.csv**“ in eine Excel-Tabelle und überschreibt alle vorhandenen Werte beginnen mit der Zelle A1. Als Trennzeichen wird ein „;“ verwendet und die Zellen werden als Text (**@**) Formatiert.

XLS_InsertCol "D"
Fügt an Stelle der Spalte **„D“** eine leere Spalte ein. Alle vorhandenen Spalten werden nachts rechts verschoben.

XLS_InsertRow "5"
Fügt an Stelle der Zeile **„5“** eine leere Zeile ein. Alle vorhandenen Spalten werden nachts unten verschoben.

XLS_New
Erzeugt eine leere Excel-Tabelle.

XLS_Open "c:\ERPAdmin\Adressen.xlsx"
Öffnet die Excel-Datei **„c:\ERPAdmin\Adressen.xlsx“** zur Bearbeitung durch **ERPAdmin**. Nach Beendigung muss die Datei mit **XLS_Save** oder **XLS_SaveAs** gespeichert und mit **XLS_Quit** geschlossen werden.

XLS_RemoveFormelFromCol "D"
Entfernt alle Formeln aus der Spalte **„D“**, so dass nur noch die eigentlichen Werte in den Zellen stehen bleiben.

XLS_ReplaceCol "Spalte", "Suchen", "Ersetzen", Teilstringersetzung
Mit dieser Funktion wird eine **„Suchen-und-Ersetzen“** Funktion auf der Excel-Spalte **„Spalte“** ausgeführt. „**Suchen**“ gibt die Zeichenkette an, nach der gesucht werden soll. „**Ersetzen**“ gibt die Zeichenkette an, durch die die unter „**Suchen**“ gefundene Zeichenkette ersetzt werden soll. „**Teilstringersetzung**“: **TRUE**, wenn auch mehrfach vorkommende Zeichenketten innerhalb einer Zelle ersetzt werden sollen. **FALSE**, wenn der Inhalt der Zelle 1:1 mit dem gesuchten Begriff übereinstimmen muss.

XLS_ReplaceCol "J", "0", "Falsch", FALSE
Ersetzt in der Spalte „**J**“ alle **„0“**-Werte durch das Wort **„Falsch“.** „**FALSE**“ besagt dabei, dass der komplette Inhalt der Zelle mit dem Wert **„0“** übereinstimmen muss. Es werden also nur Felder mit dem Inhalt **„0“** ersetzt.

XLS_ReplaceCol "J", "!", "_", TRUE
Ersetzt in der Spalte „**J**“ alle vorkommenden von „**!**“ durch einen Unterstrich. **„TRUE“** besagt dabei, dass ggf. alle **„!“** innerhalb der Zelle ersetzt werden. Aus **„!!!!!“** wird damit **„_____“.**

XLS_ReplaceISOLand xlsCol
Ersetzt in der Spalte **xlsCol** alle ISO-Länderkürzel z.B. DE und alle Länderbezeichnungen z.B. Deutschland durch die ISO-Land-Nummer z.B. 276 aus **büro+**. Der Wert hierfür werden aus der **büro+** Datenbank „Länder“ ermittelt.

XLS_Save
Speichert alle Änderungen der geöffneten Excel-Tabelle.

XLS_SaveAs "c:\ERPAdmin\Adressen_neu.xlsx", TRUE
Speichert den Inhalt der geöffneten Excel-Datei unter dem Namen **c:\ERPAdmin\Adressen_neu.xlsx“** ab. Die Endung muss dabei dem gewünschten Format der Datei entsprechen. Folgende Formate werden unterstützt: .**xls, xlsx, csv, txt, html**. Existiert die Datei **„c:\ERPAdmin\Adressen_neu.xlsx“** bereits, kann sie durch Angabe des Parameters **TRUE** gelöscht und anschließend überschrieben werden. Bei Angabe von **FALSE** wird eine bereits vorhandene Datei nicht gelöscht und somit auch nicht gespeichert und überschieben.

Die Verwendung von Platzhalter (Siehe Kapitel „Platzhalter und variable Parameter“) wird unterstützt.

XLS_SelectTable "Tabelle3"
Aktiviert das Tabellenblatt **„Tabelle3“** als aktive Tabelle zur weiteren Bearbeitung.

XLS_SetRangeFormat "A2: A1000", "Number"
Formatiert den Bereich A2-A1000 als Zahl (Number). Folgende Formate sind zulässig: "Text", "General", "Percentage" und "Number".

XLS_SetRangeValue "A2: A1000", "Interessent"
Füllt den Bereich A2-A1000 mit dem Wert „Interessent“.

XLS_SetRangeValue "A2:A@9", „Interessent“
Füllt den Bereich A2-Axx mit dem Wert „Interessent“. xx wird zuvor von **ERPAdmin** mit dem Wert der letzten Zeile in der Tabelle befüllt, z. B. 1000.

Kürzer geht es mit den Platzhaltern **@9** oder **@Z**, welche bei der Ausführung des Befehls automatisch von **ERPAdmin** mit der letzten Zeilen-Nummer bzw. der letzten Spalten-Nummer der Tabelle ersetzt werden. Siehe dazu folgendes Beispiel:

XLS_SetRangeValue "A2:A@9", "Interessent"
Befüllt den Bereich „**A2-A@9**“ mit dem Wert „**Interessent**“. Dabei wird „**@9**“ durch die letzte Zeilen-Nummer der Tabelle ersetzt z. B. **1000**.
Damit wird dann der folgende Befehl ausgeführt: **XLS_SetRangeValue "A2:A1000", "Interessent"**

XLS_SetRangeValue "A2:@Z2", "Interessent"
Befüllt den Bereich „**A2-@Z2**“ mit dem Wert „**Interessent**“, wobei „**@Z**“ von **ERPAdmin** durch die letzte Spalten-Nummer der Tabelle ersetzt wird, die mit Daten belegt ist z.B.: „**DC**“.
Damit wird dann der folgende Befehl ausgeführt: **XLS_SetRangeValue "A2:DC2", "Interessent"**

XLS_SetName "A1", "Status"
Bennent die Zelle **„A1“** um in „**Status**“.

XLS_SetValue "B1", "Status"
Füllt die Zelle **„B1“** mit dem Wert **„Status“**.

XLS_SetValue "B@9", "Status"
Ersetzt **@9** durch die letzte Zeile der Excel-Tabelle und Schreibt in das Feld den Wert Status.

XLS_SetValue "B@9-1", "Status"
Ersetzt **@9-1** durch die vorletzte Zeile der Excel-Tabelle und Schreibt in das Feld den Wert Status.

XLS_SetValue "B@9+1", "Status"
Ersetzt **@9+1** durch die letzte Zeile+1 der Excel-Tabelle und Schreibt in das Feld den Wert Status.

XLS_SetValue "C@9+1", "=SUM(C2:C@9)"
Bildet die Spalten-Summe für Spalte **C** und summiert ab der 2. Zeile alle Werte auf.

XLS_Sort "D"
sortiert den Inhalt der Excel-Tabelle **aufsteigend** nach dem Inhalt der Spalte **„D“.**

XLS_SortDesc "D"
Sortiert den Inhalt der Excel-Tabelle **absteigend** nach dem Inhalt der Spalte **„D“.**

XLS_Quit
Schließt die aktuell geöffnete Excel-Instanz. Dies muss immer der letzte Befehl bei der Bearbeitung einer Excel-Tabelle sein. Wurden Änderungen an der Excel-Datei vorgenommen, sollten diese zuvor mit dem Befehl **XLS_Save** oder **XLS_SaveAs** gespeichert werden.

> **Hinweis**:
> Wird der Befehl **XLS_Quit** nicht ausgeführt, bleibt die Excel-Instanz weiterhin geöffnet. Diese müssen dann im Taskmanager von Hand beendet werden. Dies ist auch dann der Fall, wenn es bei der Abarbeitung zu einem Fehler kommt und der Befehlt **XLS_Quit** nicht mehr ausgeführt werden kann.

Zusatzfunktionen und Zusatzvariablen

G_XLSError
Tritt bei der Ausführung der XLS-Funktionen ein Fehler auf der kontrolliert abgefangen werden kann, wird in der Variable **G_XLSError** der Wert **TRUE** zurück geliefert. Eine Fehlermeldung kann mit der Funktion **OutputMsgboxError** ausgegeben werden.

G_XLSErrorMessage
Eine Liste der aufgetreten Fehler wird in der Variable „**G_XLSErrorMessage“** gespeichert.

Die Ausgabe von Fehlermeldungen kann dann am Ende der CMD Datei ausgegeben werden.
Beispiel: **IF G_XLSError THEN OutputMsgboxError(G_XLSErrorMessage)**

G_XLSFilename
Der Dateiname der aktuell geöffneten Excel-Datei befindet sich in der Variable **G_XLSFilename**.

G_XLSTableCount
Die Anzahl der Arbeitsblätter (WorkSheet) in einer Excel-Datei sind in der Variablen **G_XLSTableCount** zu finden.

54 Der Befehl „-Export“ STD

ERPAdmin -Export <Export-Layout> <Export-Dateiname> /Von:<Bereich von> /Bis: /Filter:<Filterbedingung> /His /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>/LtzAend

Für die Verwendung der Optionen **/Von**, **/Bis**, **/Filter** und **/LtzAend** benötigen Sie die Professional-Version von **ERPAdmin**!

Für die Exporte wird ein Export-Layout aus **büro+** benötigt. Dieses wir mit **büro+** erzeugt und per Drag&Drop in das Verzeichnis von **ERPAdmin** gezogen. Dateien mit der Endung **.BPL** sollten im gleichen Verzeichnis liegen wie **ERPAdmin.**

Der Aufruf führt einen Export aus **büro+** aus. Folgende Einstellungen sind dazu in der **VAR.INI** vorzunehmen:

C_BPN_Firma	Name der Firma, auf die **büro+** registriert ist
C_BPN_User	Benutzername für Anmeldung im Mandanten
C_BPN_Pwd	Passwort für Benutzeranmeldung im Mandanten
C_BPN_MandNr	Mandanten-Nummer

<Export-Layout>
büro+ Export-Layout (**Datei mit Endung .bpl**) welches per Drag&Drop in ein Verzeichnis gezogen wurde.

<Export Dateiname> **(optional)**
Dateiname inkl. Pfadangabe unter dem das Ergebnis des Exportes gespeichert werden soll. Die Angabe ist optional. Wird kein Name angegeben wird der Export unter dem Pfad- und Dateinamen abgespeichert der im Export-Layout in **büro+** hinterlegt wurde.

Im Dateinamen könne zwei Platzhalter verwendet werden: **@VON@** und **@BIS@**. Diese werden mit den übergebenen Werten aus **/Von:** bzw. **/Bis:** ersetzt. Wird einer der Platzhalter **@VON@** oder **@BIS@** verwendet muss der Daten-Bereich (Sortierung) des Exportes auch mit **/Von:** und **/Bis:** eingeschränkt werden! In den übergebenen Werten **/Von:** und **/Bis:** dürfen sich keine Semikolons befinden.

Beispiel: **ERPAdmin -Export OffenePosten.BPL OP-Liste_@VON@.csv /Von:AdrNr=20018**
Der Export erzeugt dann die Datei OP-Liste_20018.csv

/Von:<Bereich von> PRO **(optional)**
Wie bei den Exporten aus **büro+** üblich kann dieser auf bestimmte Daten eingeschränkt werden. Hier wird der Anfangsbereich „Sortierung von…“ vorbelegt.

/Bis:<Bereich bis> PRO **(optional)**
Hier wird das Ende des Bereiches „Sortierung bis…“ vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

/Filter:<Filterbedingung> PRO **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

> **Hinweis:**
> Wird eine oder mehrere der Optionen **/Von: /Bis**: oder **/Filter:** verwendet, werden alle Sortierungen und Filter im Exportlayout überschrieben.

/His **(optional)**
Für den Datenexport aus bestimmten Archiv-Bereichen muss die Option **/His** zusätzlich übergeben werden. Verwenden Sie diese Option, wenn Sie Daten aus den folgenden Datenbanken Exportieren wollen:

- **VorgangArchiv**
- **KontaktArchiv**

/Email:<E-Mail-Adresse> PRO **(optional)**
Sendet das Export-Ergebnis an die angegebene E-Mail sofern der Export eine Ergebnisdatei erzeugt hat. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> PRO **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> PRO **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> PRO **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/LtzAend PRO **(optional)**
Für einen Export unter Berücksichtigung des Feldes „**Letzte Datensatzänderung**“ durch. Diese Option muss zunächst in **DBManager** für die entsprechende Datenbank aktiviert werden und es muss eine entsprechende Sortierung vorhanden sein. Beim ersten Aufruf werden alle Datensätze ab dem 30.12.1899 aus der Datenbank exportiert. **ERPAdmin** speichert nach dem Export in der Datei **<Exportlayout>.INI** den aktuellen Timestamp und verwenden diesen beim nächsten Aufruf als Startdatum. Damit werden bei

> **Hinweis**:
> Wenn Sie **ERPAdmin** nicht auf dem BPServer ausführen und die Option **/LtzAend** verwenden, müssen Sie dafür sorgen, dass die Systemzeiten beider Rechner synchron sind.
>
> **Wichtig:**
> **Mit der Verwendung der Option /LtzAend werden alle im Export hinterlegten Filter und Sortierungen ignoriert. Sie müssen ggf. die benötigten Filter oder die Bereichsauswahl beim ERPAdmin -Aufruf mit übergeben**

allen Folgeaufrufen nur noch die Datensätze exportiert, bei denen sich seit dem letzten Export eine Änderung ergeben hat oder die neu dazugekommen sind.

Beispiele:

ERPAdmin -Export
Wird der Befehl ohne weitere Parameter aufgerufen werden alle Export- -Layouts im gleichen Verzeichnis abgearbeitet. Der Export erfolgt unter dem im Layout angegebenen Datei- und Pfadnamen. Befinden sich auch Import-Layouts im gleichen Verzeichnis werden diese Layouts übersprungen.

ERPAdmin -Export "c:\ERPAdmin\Adressen.bpl"
Soll nur ein bestimmter Export ausgeführt werden muss der Dateiname des Layouts angegeben werden. Dann wird der Export nur für dieses Layout durchgeführt. Die Export-Datei wird dann unter dem im Layout angegebenen Datei- und Pfadnamen gespeichert.

ERPAdmin -Export "c:\ERPAdmin\Adressen.bpl" "c:\ERPAdmin\Kunden.xls"
Wird als 3. Parameter ein Dateiname angegeben, erfolgt der Export in die angegebene Zieldatei (z. B. c:\ERPAdmin\Kunden.xls).

ERPAdmin -Export c:\ERPAdmin\Adressen.bpl c:\ERPAdmin\Kunden.xls /Von:AdrNr=10000 /Bis:AdrNr=19999
Es wird ein Adressen-Export aus dem Bereich **AdrNr=10000** bis **19999** erzeugt und unter dem Namen **Kunden.**xls abgespeichert**.**

ERPAdmin -Export c:\ERPAdmin\Adressen.bpl c:\ERPAdmin\Kunden.xls /Von:AdrNr=10000 /Bis:AdrNr=19999 /Filter:Status=‘Kunde‘
Es werden nur die Adressen aus dem Bereich **10000-19999** mit dem Status „**Kunde**“ exportiert.

ERPAdmin -Export c:\ERPAdmin\Adressen.bpl c:\ERPAdmin\Kunden.xls "/Von:LtzAend=@DATE@ 00:00:00" "/Bis:LtzAend=@DATE@ 23:59:59"
Es werden nur die Adressen exportiert die am aktuellen Tag geändert wurden.

ERPAdmin -Export c:\ERPAdmin\Adressen.bpl c:\ERPAdmin\Kunden.xls /Email:Demo@ERPAdmin.de "/Subject:Adressliste '@FILENAME@' mit Kundendaten"
Exportiert Adressen mit dem Export-Layout **c:\ERPAdmin\Adressen.bpl** und speichert das Ergebnis unter dem Dateinamen **c:\ERPAdmin\Kunden.xls** ab. Anschließend wird eine E-Mail an die Adresse Demo@ERPADmin.de versenden mit der Datei **Kunden.xls** als Anhang und mit dem Betreff „**Adressliste 'Kunden.xls' mit Kundendaten**“

ERPAdmin -Export c:\ERPAdmin\Adressen.bpl c:\ERPAdmin\Kunden.xls /LtzAend
Es werden nur die Adressen exportiert die sich seit dem letzten Export geändert haben oder neu dazu gekommen sind. Der neue Timestamp wird in der Datei **Adressen.INI** gespeichert.

ERPAdmin -Export c:\ERPAdmin*.bpl /LtzAend

Es werden alle Export-Layouts in dem aktuellen Verzeichnis nacheinander ausgeführt und nur die geänderten und neuen Datensätze exportiert.

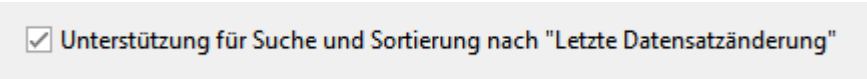
Hinweis:
Das Export-Format einer Datei (Excel, Text, DB, etc.) kann nicht durch Angabe einer anderen Endung geändert werden. Wurde als Dateiformat im Layout „**Text ANSI**“ angegeben, kann der Export keine Excel-Datei erzeugen.

55 Der Befehl „-ExportLtzAend“ PRO

ERPAdmin -ExportLtzAend <Datenbank> <Timestamp/Filename> <CSV-Datei> <ERP-Feldliste> /Init /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Der Aufruf führt einen Daten-Export im Bereich **Datenbank** aus. Dabei werden nur Datensätze exportiert, bei denen der Feldwert „**Letzte Datensatz-Änderung**“ neuen ist als der unter **TimeStamp/Filename** übergebene Wert. Um diese Funktion nutzen zu können muss im **DBManager** diese Funktion aktiviert werden.



<Datenbank>
Gibt die Datenbank aus **büro+** an, aus der die Daten exportiert werden sollen.

<Timestamp/Filename>
Hier muss ein **Timestamp** im Format „**DD.MM.YYYY HH:MM:SS**“ übergeben werden. Es werden dann nur Datensätze exportiert. Deren letztes Änderungsdatum größer-gleich diesem **Timestamp** ist.

Sollen in Intervallen immer die letzten Änderungen exportiert werden kann an Stelle des **Timestamp** auch ein ANSI-Datei angegeben werden in der ein **Timestamp** hinterlegt ist. Nach erfolgtem Export wird der **Timestamp** in der Datei aktualisiert so dass beim nächsten Aufruf nur die Änderungen die seit dem letzten Aufruf exportiert werden.

Der Timestamp wird in der Datei im folgenden Format gespeichert: **06.11.2017 09:21:04**

> **Hinweis:**
> Dieser Befehl sollte nur direkt auf dem ERP-Server ausgeführt werden, weil es sonst zu Differenzen beim Timestamp zwischen Server und PC kommen kann.

<CSV-Datei>
Name der Datei inkl. Pfadangabe in die die Daten im CSV-Format exportiert werden sollen. Existiert die Datei bereits wird diese gelöscht auch wenn keine neuen Daten zum Erzeugen einer neuen Datei vorhanden sind.

<ERP-Feldliste>
Liste von Felder aus der **Datenbank.** Eine Liste aller Feldbezeichner kann mit dem Befehl **ERPAdmin -ListDatenbankfelder** erzeugt werden. Die Feldliste ist mit einem Semikolon zu trennen. Leerzeichen sind nicht erlaubt.

/Init **(optional)**
Wird die Option **/Init** bei einem Aufruf verwendet, wird nur die **Timestamp-Datei** mit einem aktuellen

Timestamp erzeugt. Das ist dann sinnvoll, wenn man mit dem ersten Aufruf nicht den Export der kompletten Datenbank erzeugen möchte.

/Header **(optional)**
Mit dieser Option wird die Ausgabe der Spaltenbezeichner in der **CSV-Datei** aktiviert. Dies soll die Zuordnung der Spalten für eine Überprüfung vereinfachen. Ein direkter Import dieser Datei in **büro+** ist mit Kopfzeilen nicht möglich. Die Kopfzeile muss für einen Import zuvor entfernt werden.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei CSV-**Datei** versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Beispiele

ERPAdmin -ExportLtzAend Artikel "26.06.2017 10:23:34" "c:\ERPAdmin\Artikel.csv" "ArtNr;KuBez1;Suchbeg"
Exportiert alle Artikel die ein Letztes Änderungsdatum neuer oder gleich **26.06.2017 10:23:34** haben. Das Ergebnis wird als CSV-Export in der Datei **c:\ERPAdmin\Artikel.csv** gespeichert. Es werden die Felder **ArtNr** (Artikelnummer), **KuBez1** (Kurz-Bezeichnung 1) und **Suchbeg** (Suchbegriff) exportiert.

ERPAdmin -ExportLtzAend Artikel "c:\ERPAdmin\Artikel_LtzAend.log" "c:\ERPAdmin\Arftikel.csv" "ArtNr;KuBez1;Suchbeg"
Exportiert alle Artikel die ein letztes Änderungsdatum neuer oder gleich dem Eintrag in der Datei **c:\ERPAdmin\Artikel_LtzAend.log** haben. Das Ergebnis wird als CSV-Export in der Datei **c:\ERPAdmin\Artikel.csv** gespeichert. Es werden die Felder **ArtNr** (Artikelnummer), **KuBez1** (Kurz-Bezeichnung 1) und **Suchbeg** (Suchbegriff) exportiert.

Die Datei **c:\ERPAdmin\Artikel_LtzAend.log** hat vor dem ersten Aufruf folgenden Inhalt:
26.06.2017 10:23:34

Nach dem Aufruf wurde der **Timestamp** in der Datei mit Datum/Uhrzeit des letzten Exportes aktualisiert. Beim nächsten Aufruf werden somit wieder nur die Änderungen exportiert die seit dem letzten Export dazu gekommen sind.

ERPAdmin -ExportLtzAend Artikel "c:\ERPAdmin\Artikel_LtzAend.log" "c:\ERPAdmin\Arftikel.csv" "ArtNr;KuBez1;Suchbeg" /INIT
Erzeugt nur die Datei **c:\ERPAdmin\Artikel_LtzAend.log** mit dem aktuellen Timestamp.

> **Hinweis**:
> Wird ein Dateiname übergeben, muss diese Datei existieren und es muss auch ein gültiger Timestamp enthalten sein. Die Datei kann mit der Option **/INIT** beim Erstaufruf erzeugt werden.

56 Der Befehl „-FilenameSearchAndReplace“ STD

ERPAdmin -FilenameSearchAndReplace <Dateiname> <Suchen> <Ersetzen> /NoFileNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl führt **ERPAdmin** ein „**Suchen und Ersetzen**“ in den **Dateinamen** durch. Es werden alle gefundenen Zeichenkette **Suchen** durch die Zeichenkette **Ersetzen** ersetzt.

Soll die Ersetzung in mehreren Dateien erfolgen kann ein Wildcard (*) verwendet werden.

<Datename>
Selektiert einen oder mehrere Dateien in denen die Zeichenkette **Suchen** durch die Zeichenkette **Ersetzen** ersetzt wird.

<Suchen>
Zeichenkette innerhalb der gefundenen Dateinamen.

<Ersetzen>
Zeichenkette durch die die gefundene Zeichenkette **Suchen** im Dateinamen ersetzt werden soll.

/NoFileNotExistError **(optional)**
Existiert keine Datei mit dem Namen **Dateinamen** und soll in diesem Fall **keine** Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

Beispiel

ERPAdmin -FilenameSearchAndReplace *.csv " " "_"
Ersetze alle **Leerzeichen** in den Dateinamen ***.csv** durch einen **Unterstrich**

ERPAdmin -FilenameSearchAndReplace *.csv " " ""
Entfernt alle Leerzeichen in den Dateinamen ***.csv**

57 Der Befehl „-FileReplace“ STD

ERPAdmin -FileReplace <Ansi-Datei> <Suchen> <Ersetzen> /SubDir: /NoFileNotExistError /Datei:<Ergebnis-Datei>

Mit diesem Befehl führt **ERPAdmin** ein „**Suchen und Ersetzen**“ in der **Ansi-Datei** durch. Es werden alle gefundenen Zeichenketten **Suchen** durch die Zeichenkette **Ersetzen** ersetzt.

Soll die Ersetzung in mehreren Dateien erfolgen kann ein Wildcard (*) verwendet werden.

Es wird nur das ANSI-Datei-Format unterstützt

<Suchen>
Gibt die Zeichenfolge an nach der gesucht werden soll.

> **Hinweis:**
> Sollen Anführungszeichen ersetzt werden muss als Suchzeichenfolge **"\""** verwendet werden.

<Ersetzen>
Gibt die Zeichenfolge an durch die gefundene Zeichen ersetzt werden sollen. Wird kein Wert angegeben, werden die gefundenen Zeichen gelöscht.

/SubDir:<Verzeichnis> PRO **(optional)**
Wird ein Verzeichnis als Startpunkt übergeben wir dieses Verzeichnis und alle darunter befindlichen Verzeichnisse rekursiv nach **Ansi-Datei** durchsucht und ggf. eine Ersetzung durchgeführt. Diese Option darf nicht zusammen mit **/Datei:<…>** verwendet werden.

/NoFileNotExistError **(optional)**
Ist keine Datei vorhanden und soll in diesem Fall keine Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

> **Hinweis:**
> Angaben wie ***.*** sind **nicht** zulässig. Es darf immer nur ein Wildcard verwendet werden.

/Datei:<Ergebnis-Datei> **(optional)**
Soll die **Ansi-Datei** in der ursprünglichen Form erhalten bleiben kann mit der Option /Datei: ein Dateiname übergeben werden. Wurde in **Ansi-Datei** ein Wildcard verwendet, muss auch in **Ergebnis-Datei** ein Wildcard verwendet werden.

Beispiel
ERPAdmin -FileReplace "c:\ERPAdmin\ShopAdressen.csv" "," ";"
In der Datei **c:\ERPAdmin\ShopAdressen.csv** werde alle Kommas durch Semikolons ersetzt.

ERPAdmin -FileReplace "c:\ERPAdmin\Bestellung*.csv" "," ";"
Ersetze in allen Dateien **c:\ERPAdmin\Bestellung*.csv** das Zeichen Komma durch ein Semikolon.

ERPAdmin -FileReplace "c:\ERPAdmin\Bestellung*.csv" "\""
Entfernt in allen Dateien **c:\ERPAdmin\Bestellung*.csv** die Anführungszeichen.

ERPAdmin -FileReplace VAR.INI "C_EMAIL_SENDATONCE=FALSE" "C_EMAIL_SENDATONCE=TRUE" /SubDir:c:\ERPAdmin
Führt in allen Dateien **VAR.INI** im Verzeichnis **c:\ERPAdmin** und allen darunter befindlichen Verzeichnisses eine Ersetzung der Zeichenfolge **"C_EMAIL_SENDATONCE=FALSE"** durch **"C_EMAIL_SENDATONCE=TRUE"**

58 Der Befehl „-FileReplaceCol“ PRO

ERPAdmin -FileReplaceCol <CSV-Datei> <Spaltenname> <Suchen> <Ersetzen> /NoFileNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl führt **ERPAdmin** ein „**Suchen und Ersetzen**“ in der Datei **CSV-Datei** mit Spaltenbezeichnern durch.

<CSV-Dateiname>
Bei der **CSV-Datei** muss es sich um eine ANSI-Datei im CSV-Format handeln, also mit Semikolon getrennten Feldern. Die erste Zeile der Datei enthält die Spaltenbezeichner. Eine Datei ohne Spaltenbezeichner ist **nicht** zulässig.

<Spaltenname>
Es werden alle gefundenen Zeichenketten **Suchen** durch die Zeichenkette **Ersetzen** in der Spalte **Spaltenname** ersetzt.

<Suchen>
Zeichenkette aus der Spalte **Spaltennamen** der mit dem Inhalt von Ersetzen ersetzt werden soll

<Ersetzen>
Zeichenkette durch die der Inhalt von *Suchen* ersetzt werden soll

/NoFileNotExistError (optional)
Ist keine Datei vorhanden und soll in diesem Fall keine Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

Beispiele

ERPAdmin -FileReplaceCol c:\ERPAdmin\ShopAdressen.csv Datum "/" "."
In der CSV-Datei **c:\ERPAdmin\Bestellungen.csv** werden in der Spalte **Datum** alle / durch einen Punkt ersetzt.

ERPAdmin -FileReplaceCol c:\ERPAdmin\Bestellungen*.csv Datum "/" "."
Ersetze in allen CSV-Dateien **c:\ERPAdmin\Bestellungen*.csv** in der Spalte **Datum** alle Zeichen **/** durch einen **Punkt** ersetzt.

59 Der Befehl „-FTPDelete“ PRO

ERPAdmin -FTPDelete <FTP-Datei> <FTP-Verzeichnis>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin löscht die Datei(en) die durch den Filter **FTP-Datei** selektiert werden aus dem **FTP-Verzeichnis**

<FTP-Datei>
Gibt den Namen der Datei an die auf dem FTP-Server gelöscht werden soll. Sollen mehrere Dateien gelöscht werden kann auch ein Wildcard (*****) verwendet werden.

<FTP-Verzeichnis> **(optional)**
Verzeichnis auf dem FTP-Server. Wurde kein **FTP-Verzeichnis** angegeben wird das aktuelle Verzeichnis **./** verwendet.

Beispiele

ERPAdmin -FTPDelete Bild1.jpg
Löscht die Datei **Bild1.jpg** aus dem Root-Verzeichnis des FTP-Servers.

ERPAdmin -FTPDelete Bild*.jpg ./Bilder/
Löscht alle Dateien **Bild*.jpg** aus dem FTP-Verzeichnis **./Bilder/**.

60 Der Befehl „-FTPDir“ STD

ERPAdmin -FTPDir **<FTP-Verzeichnis> /Datei:<CSV-Dateiname>**

ERPAdmin liest den Inhalt eines Verzeichnisses vom FTP-Server am Bildschirm. Dieser Befehl ist nicht für die Skript gesteuerte Ausführung geeignet.

<FTP-Verzeichnis> **(optional)**
Gibt ein Verzeichnis auf dem FTP-Server an. Wurde kein **FTP-Verzeichnis** angegeben wird das aktuelle Verzeichnis **./** verwendet. Wurde in der **VAR.INI** die Variable **C_FTPDefaultDownloadDir** mit einem Verzeichnis vorbelegt, wird der Inhalt dieses Verzeichnisses ausgegeben. Das Ergebnis wird am Bildschirm ausgegeben (die ersten 20 Datensätze) und zusätzlich in einer Datei abgespeichert

/Datei:<CSV-Dateiname> **(optional)**
Übergabe eines Dateinamens unter dem das Ergebnis als CSV-Datei gespeichert wird. Wird kein Dateiname übergeben speichert **ERPAdmin** das Ergebnis unter dem Dateinamen **FTP-Verzeichnis.csv** ab.

Beispiele

ERPAdmin -FTPDir
Listet den Inhalt des FTP-Verzeichnisses am Bildschirm auf. Es werden nur die ersten 20 Einträge angezeigt. Zusätzlich wird der komplette Inhalt als CSV-Datei gespeichert.

ERPAdmin -FTPDir ./Bilder/ /Datei:Bilder.csv
Listet den Inhalt des FTP-Login-Verzeichnisses am Bildschirm auf. Es werden nur die ersten 20 Einträge angezeigt. Der vollständige Inhalt wird in der Datei **Bilder.csv** abgespeichert.

61 Der Befehl „-FTPPut“ STD

ERPAdmin -FTPPut <QuellDatei> <FTP-ZielDatei> <FTP-Zielverzeichnis> <DoubletFile> /MoveToHis /MoveToBad /Delete /NoFileNotExistError

Für die Verwendung dieses Befehls ist Registrierung der DLL „**chilkatax-9.5.0-x64.msi**“ (&4 Bit Version) oder der „**chilkatax-9.5.0-win32.msi**“ (32 Bit Version) notwendig. Mit dem Aufruf „**ERPAdmin -Init**“ können Sie prüfen, ob die DLL korrekt installiert wurde.

Das Programm stellt eine Verbindung zu dem in der **VAR.INI** angegebenen FTP-Server her und überträgt dann Daten an diesen Server. Dazu werden die ebenfalls in der **VAR.INI** hinterlegten Zugangsdaten verwendet.

Folgende Einstellungen können in der **VAR.INI** vorgenommen werden:

C_FTPSERVER	FTP-Server oder IP-Adresse
C_FTPUSER	FTP-Benutzername
C_FTPPWD	FTP-Passwort
C_FTPPASSIV	"0"=AKTIV Mode, "1"=PASSIV Mode
C_FTPTLS	"0"=Standard FTP, "1"=TLS explicit (AUTH SSL)
C_FTPSERVER2	FTP-Server oder IP-Adresse 2
C_FTPUSER2	FTP-Benutzername 2
C_FTPPWD2	FTP-Passwort 2
C_FTPPASSIV2	"0"=AKTIV Mode, "1"=PASSIV Mode
C_FTPTLS2	"0"=Standard FTP, "1"=TLS explicit (AUTH SSL)
C_FTPSERVER3	FTP-Server oder IP-Adresse 3
C_FTPUSER3	FTP-Benutzername 3
C_FTPPWD3	FTP-Passwort 3
C_FTPPASSIV3	"0"=AKTIV Mode, "1"=PASSIV Mode
C_FTPTLS3	"0"=Standard FTP, "1"=TLS explicate (AUTH SSL)
C_FTPSERVER4	FTP-Server oder IP-Adresse 4
C_FTPUSER4	FTP-Benutzername 4
C_FTPPWD4	FTP-Passwort 4
C_FTPPASSIV4	"0"=AKTIV Mode, "1"=PASSIV Mode
C_FTPTLS4	"0"=Standard FTP, "1"=TLS explicit (AUTH SSL)
C_FTPSERVER5	FTP-Server oder IP-Adresse 5
C_FTPUSER5	FTP-Benutzername 5
C_FTPPWD5	FTP-Passwort 5
C_FTPPASSIV5	"0"=AKTIV Mode, "1"=PASSIV Mode
C_FTPTLS5	"0"=Standard FTP, "1"=TLS explicit (AUTH SSL)

<Quelldatei>
Gibt die zu übertragende Datei mit Pfadangabe an.
Beispiel: **c:\ERPAdmin\Bild.jpg**

Für den Upload mehrerer Dateien kann ein Wildcard (*) angegeben werden. Bitte beachten Sie, dass der 2. Parameter FTP-**ZielDatei** bei der Verwendung von Wildcards **immer leer** sein muss.
Beispiel: **c:\ERPAdmin*.jpg**

<FTP-ZielDatei>
Gibt den Ziel-Dateinamen ohne Pfadangabe auf dem FTP-Server an. Wird kein Dateiname angegeben, wird der gleiche Dateiname verwendet der mit **Quelldatei** übergeben wurde.
Beispiel: **Bild123.jpg**

<FTP-Zielverzeichnis>
Verzeichnisnamen auf dem FTP-Server in dem der Upload erfolgen soll.
Beispiel: **/Images/**

<DoubletFile > **(optional)**
Dateiname inkl. Pfadangabe in dem alle übertragenen Dateien protokolliert werden. Vor jeder Übertragung prüft **ERPAdmin** ob die zu übertragende Datei in **DoubletFile** vorhanden ist und verschiebt in diesem Fall die Datei nach **C_BADDIR** wenn die Option **/MoveToBad** übergeben wurde. Weitere Informationen zur Doubletten-Prüfung entnehmen Sie bitte dem Kapitel „**Doubletten-Prüfung**“
Beispiel: **C:\ERPAdmin\Doubletten_FTPGet.log**

/Delete **(optional)**
Lösche die lokale Datei nach erfolgreicher FTP-Übertragung.

/NoFileNotExistError **(optional)**
Existiert die **Quelldatei** nicht auf dem Server und soll in diesem Fall **keine** Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

Beispiele:

ERPAdmin -FTPPut "c:\ERPAdmin\Bild.jpg"
Die Datei „**c:\ERPAdmin\bild.jpg**“ wird auf den FTP-Server transferiert und im Root-Verzeichnis des Benutzers unter gleichem Namen abgelegt.

ERPAdmin -FTPPut "c:\ERPAdmin\Bild.jpg" "Artikel12345.jpg" "./Images/"
Die Datei „**c:\ERPAdmin\Bild.jpg**“ wird per FTP transferiert und im Verzeichnis „**./Images/**“ unter dem Namen „**Artikel12345.jpg**“ abgespeichert.

Zugriff auf mehrere FTP-Server

Soll **ERPAdmin** auf verschiedene FTP-Server zugreifen, können die Zugangsdaten für zwei zusätzliche FTP-Server in der **VAR.INI** zu hinterlegt werden. Der Aufruf von **ERPAdmin** muss dann um die Option **/FTP2** oder **/FTP3** erweitert werden.

ERPAdmin -FTPPut "c:\ERPAdmin\Bild.jpg" /FTP2

Historisierung der FTP-Uploads

Ist die Variable **C_HISTORYDIR** in der Datei **VAR.INI** mit einem gültigen Verzeichnis belegt, erzeugt das Programm in diesem Verzeichnis ein weiteres Unterverzeichnis mit Zeitstempel (Datum und Uhrzeit) als Namen (Beispiel: 2013-03-11 10.38.55). Darin wird dann eine Sicherungskopie der übertragenen Datei gespeichert.

Beispiel

ERPAdmin -FTPPut "c:\ERPAdmin*.jpg" "" "./Images/"
Die Dateien „**c:\ERPAdmin*.jpg**“ werden per FTP transferiert und im Verzeichnis **„./Images/**“ abgespeichert.

Variable Dateinamen
Sollen in regelmäßigen Abständen Dateien per FTP übertragen werden und die Daten der vorherigen Übertragung dabei nicht überschrieben werden, so muss der Dateiname automatisch angepasst werden. Dazu können beim FTP-Transfer die Dateinamen mit Platzhalter versehen werden. Weitere Details dazu finden Sie im Kapitel „**Platzhalter und variable Parameter**“.

Beispiel mit variablen Platzhaltern

ERPAdmin -FTPPut "TagesUmsatz.xls" "TagesUmsatz_@YYYY@@MM@@DD@.xls"
Der FTP-Transfer wird mit dem Dateinamen **"TagesUmsatz_2016.08.08.xls**" übertragen.

Hinweis:
Bitte unbedingt auf Groß-/Kleinschreibung bei den Platzhaltern achten!

Beispiel mit Historisierung

ERPAdmin -FTPPut "c:\ERPAdmin*.jpg" " " "/Images/" /MoveToHis /MoveToBad
Es werden alle Bild-Dateien mit der Endung **.jpg** per FTP übertragen.

Nach der Übertragung werden die erfolgreich übertragen Dateien in das Verzeichnis **C_HISTORYDIR\TIMESTAMP** verschoben.

Alle nicht übertragenen Dateien werden in das Verzeichnis **C_BADDIR\TIMESTAMP** verschoben.

Die Optionen **/MoveToHis** und **/MoveToBad** können unabhängig voneinander verwendet werden. Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

62 Der Befehl „-FTPGet“ STD

ERPAdmin -FTPGet <FTP-QuellDatei> <Ziel-Datei> <FTP-Quellverzeichnis> <Lokales-Verzeichnis> <DoubletFile> /DELETE /NoFileNotExist /Ignore0ByteFiles

Für die Verwendung dieses Befehls ist Registrierung der DLL „**chilkatax-9.5.0-x64.msi**“ (&4 Bit Version) oder der „**chilkatax-9.5.0-win32.msi**“ (32 Bit Version) notwendig. Mit dem Aufruf „**ERPAdmin -Init**“ können Sie prüfen, ob die DLL korrekt installiert wurde.

Das Programm stellt eine Verbindung zu dem in der **VAR.INI** angegebenen FTP-Server her und überträgt dann Daten von diesem FTP-Server. Dazu werden die ebenfalls in der **VAR.INI** hinterlegten Zugangsdaten verwendet.

Folgende Einstellungen sind in der **VAR.INI** vorzunehmen:

C_FTPSERVER	FTP-Server oder IP-Adresse
C_FTPUSER	FTP-Benutzername
C_FTPPWD	FTP-Passwort
C_FTPPASSIV	"0" = AKTIV Mode, "1" = PASSIV Mode
C_FTPTLS	"0"=Standard FTP, "1"=TLS explicit (AUTH SSL)

<FTP-Quelldatei>
Dateiname ohne Verzeichnisangabe der zu übertragenden.
Beispiel: **Umsatzliste.xls**

Für den Download mehrerer Dateien kann ein Wildcard (*) angegeben werden.
Beispiel: ***.jpg**

<Ziel-Datei>
Dateiname unter dem der Download auf der Festplatte abgelegt werden soll. Wird kein Ziel-Dateiname angegeben wird die Datei unter dem gleichen Namen gespeichert wie unter FTP-Quelldatei angegeben.
Beispiel: **Umsatzliste2017.xls**

> **Hinweis:**
> Bitte beachten Sie, dass dieser Parameter bei der Verwendung von einem Wildcard (*) im Parameter <FTP-Quelldatei> **ignoriert wird**.

<FTP-Quellverzeichnis>
Verzeichnis auf dem FTP-Server aus dem die Datei geladen werden soll.
Beispiel: **/Images/**

<Lokales-Verzeichnis>
Verzeichnis auf dem PC in dem der Download abgelegt werden soll.
Beispiel: **C:\ERPAdmin\FTP-Bilder**

<DoubletFile > **(optional)**
Dateiname inkl. Pfadangabe in dem alle übertragenen Dateien protokolliert werden. Vor jeder Übertragung prüft **ERPAdmin** ob die zu übertragende Datei in **DoubletFile** vorhanden ist und überspringt diese ggf. Weitere Informationen zur Doubletten-Prüfung entnehmen Sie bitte dem Kapitel „**Doubletten-Prüfung**“
Beispiel: **C:\ERPAdmin\Doubletten_FTPGet.log**

/DELETE **(optional)**
Zusätzlich kann die Option **/DELETE** angegeben werden, welcher nach dem Download die Datei auf dem FTP-Server löscht. Wird die Option **/DELETE** Verwendet, sollte die Historisierung aktiviert sein, damit alle per FTP geladenen Dateien als Sicherungskopie im Historie-Verzeichnis abgelegt werden. Das hilft bei einer späteren Fehlersuche.

/NoFileNotExistError **(optional)**
Existiert die **FTP-Quelldatei** nicht auf dem Server und soll in diesem Fall **keine** Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

/Ignore0ByteFiles **(optional)**
Vor dem Download wird geprüft ob die zu übertragende Datei eine Größe von **0-Byte** hat. Ist dies der Fall wird die Datei nicht übertragen und eine Fehlermeldung erzeugt.

Beispiele

ERPAdmin -FTPGet "Umsatzliste.xls" "Umsatzliste2017.xls" ".\Dokumente\" "c:\ERPAdmin\Download"
Transferiert die Datei **„Umsatzliste.xls“** aus dem FTP-Verzeichnis „/Dokumente“ in das lokale Verzeichnis **„c:\ERPAdmin\Download“** unter dem Namen **„Umsatzliste2017.xls“**.

ERPAdmin -FTPGet "*.jpg" "" "/Images/" "c:\ERPAdmin\FTP-Bilder\"
Dieser Aufruf transferiert alle Dateien aus dem FTP-Verzeichnis **„/Images/“** mit der Endung **„.jpg“** in das lokale Verzeichnis **„c:\ERPAdmin\FTP-Bilder“**.

ERPAdmin -FTPGet "*.jpg" "" "/Images/" "c:\ERPAdmin\FTP-Bilder\" /DELETE
Dieser Aufruf transferiert alle Dateien wie im vorherigen Beispiel. Nach der FTP-Übertragung werden die Dateien dann auf dem FTP-Server gelöscht.

ERPAdmin -FTPGet "*.jpg" "" "/Images/" "c:\ERPAdmin\FTP-Bilder\" /DELETE /FTP3
Gleicher Aufruf wie zuvor nur das die Zugangsdaten für **FTP3** aus der **VAR.INI** verwendet werden.

ERPAdmin -FTPGet "Bestellungen*.csv" "" "/Orders/" "c:\ERPAdmin\Bestellungen\" "c:\ERPAdmin\Doubletten.log" /DELETE /FTP3
Lädt Dateien **Bestellung*.csv** mit Bestellungen vom FTP-Server (**/FTP3**) und legt diese im Verzeichnis **c:\ERPAdmin\Bestellungen** ab. In der Datei **c:\ERPAdmin\Doubletten.log** wird eine Liste aller bereits per FTP übertragenen Dateinamen geführt. Wurde eine Datei bereits per FTP übertrage (gleicher

Hinweis:
Angaben wie ***.*** sind nicht zulässig. Verwenden Sie stattdessen nur ***

Dateiname) wird diese nach dem Download in das Verzeichnis **C_BADDIR** verschoben welches in der **VAR.INI** definiert wurde.

63 Der Befehl „-HTTPDownload“ PRO

ERPAdmin -HTTPDownload <URL> /Datei:<Dateiname>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Lädt die angegebene Datei **URL** vom Webserver und speichert diese auf dem lokalen Rechner unter **Dateiname** ab. **/Datei:<Dateiname>** ist bei diesem Befehl ein Pflichtfeld.

Beispiel

ERPAdmin -HTTPDownload http://www.systemhaus-horn.de/erpadmin/erpadmin.zip /Datei: ERPAdmin.zip

Lädt die Datei **ERPAdmin.zip** vom Webserver **http://www.systemhaus-horn.de/erpadmin/** und speichert diese im aktuellen Verzeichnis unter dem Namen **ERPAdmin.ZIP** ab.

64 Der Befehl „-HTTPDownloadFromFile“ PRO

ERPAdmin -HTTPDownloadFromFile <CSV-Datei> <URL-SpalteNr> <Datei-SpalteNr> /Prefix:<Prefix>

Dieser befahl Läd die in der **CSV-Datei** angegebenen Dateien (Bilder, Dokumente etc.) unter der in **URL-SpalteNr** angebenen URL aus dem Internet und speichert die Datei unter dem in **Datei-SpalteNr** angegebenen Namen ab.

/Prefix **(optional)**
Jedem Dateinamen aus **Datei-SpalteNr** kann ein **Prefix** vorangestellt werden.

Beispiel:

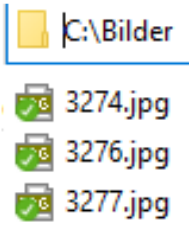
ERPAdmin -HTTPDownloadFromFile Shop-Bilder.csv> 2 3 "/Prefix:c:\Bilder\ "

In der Datei **Shop-Bilder.csv** sind in Spalte **2** die URL aller Webshop-Bilder hinterlegt. Diese sollen aus dem Internet geladen werden und unter dem Dateinamen in Spalte **3** im Verzeichnis **c:\Bilder** gespeichert werden.

Shol-Bilder.csv

	A	B	C
1	10000	https://www.Systemhaus-horn.de/ERPAdmin/HTTPDownloadBilder/3274.jpg	3274.jpg
2	10001	https://www.Systemhaus-horn.de/ERPAdmin/HTTPDownloadBilder/3276.jpg	3276.jpg
3	10002	https://www.Systemhaus-horn.de/ERPAdmin/HTTPDownloadBilder/3277.jpg	3277.jpg
4	10003	https://www.Systemhaus-horn.de/ERPAdmin/HTTPDownloadBilder/3278.jpg	3278.jpg

Nach dem Aufruf von **ERPAdmin** befinden sich alle Bilder im Verzeichnis **C:\Bilder**



Hinweis:
Endet die Angabe von **/Prefix** mit einen **Backslash** z.B. **"/Prefix:c:\Bilder\ "** und wurde die Option in Anführungszeichen gesetzt, muss nach dem Backslash vor dem Anführungszeichen ein zusätzliches Leerzeichen angeben werden!

65 Der Befehl „-Import“ STD

ERPAdmin -Import <büro+ Import-Layout> <Import-Dateiname> <DoubletFile> /NoFileNotExistError /MoveToHis, /MoveToBad, /Delete /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Importe wird ein Import-Layout aus **büro+** benötigt. Dieses wir mit **büro+** erzeugt und per Drag&Drop in ein Verzeichnis gezogen. Diese Dateien mit der Endung **.BPL** sollten dann im gleichen Verzeichnis liegen wie **ERPAdmin.**

Der Import funktioniert analog zum Export. Die Parameter haben die gleiche Bedeutung nur auf den Import bezogen.

<büro+ Import-Layout>
Gibt den Namen inkl. Pfad einer **büro+** Layout-Definition an.

<Import-Dateiname> **(optional)**
Gibt den Namen inkl. Pfad der Datei an deren Daten importiert werden sollen. Wir empfehlen immer einen Dateinamen mit zu übergeben, da **ERPAdmin** dann prüfen kann ob diese Datei existiert und den Import ggf. nicht startet.

Wird kein **Import-Dateiname** übergeben, wird der Dateiname aus dem **büro+ Import-Layout** verwendet und es kann zu Verklemmungen kommen, wenn diese Datei nicht existiert.

<DoubletFile > **(optional)**
Dateiname inkl. Pfadangabe in dem alle importierten Dateien protokolliert werden. Vor jedem Import prüft **ERPAdmin** ob die Importdatei in **DoublettenFile** vorhanden ist und verschiebt in diesem Fall die Datei nach **C_BADDIR**. Weitere Informationen zur Doubletten-Prüfung entnehmen Sie bitte dem Kapitel „**Doubletten-Prüfung**“.

Beispiel: **C:\ERPAdmin\Doubletten_Import.log**

/NoFileNotExistError **(optional)**
Wenn keine Import-Datei existiert ist erzeugt **büro+** eine Fehlermeldung am Bildschirm und der Import wird nicht fortgesetzt bis jemand diese Meldung weg Klickt.

Damit dieses Problem vermieden wird muss ein **Import-Dateiname** beim Aufruf von ERPAdmin mit übergeben werden. **ERPAdmin** prüft dann vor dem eigentlichen Import ob die Datei vorhanden ist und gibt eine Fehlermeldung an den Benutzer zurück und überspringt den Import.

Soll in diesem Fall keine Fehlermeldung an den Benutzer zurückgegeben werden, kann die Option **/NoFileNotExistError** verwendet werden. Dies ist dann z.B. sinnvoll, wenn dem Import ein Daten-Export vorausgegangen ist und dieser nicht immer ein Ergebnis liefert.

> **Achtung:**
> Bei zusammengesetzten Importen kann die Option **/NoFileNotExistError** nur dann verwendet werden, wenn ein Dateiname als Parameter **<Import Dateiname>** übergeben wurde.

/Email:<E-Mail-Adresse> PRO **(optional)**
Sendet das Import-Ergebnis an die angegebene E-Mail. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> PRO **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Importes einfügt.

/EmailCC:<E-Mail-Adresse> PRO **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> PRO **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

Beispiele:

ERPAdmin -Import
Führt einen Import für alle Import-Layouts durch die sich im gleichen Verzeichnis befinden. Export-Layouts werden übersprungen.

ERPAdmin -Import "c:\ERPAdmin\Artikel.bpl" " c:\ERPAdmin\Artikel.xls"
Es wird die Excel-Datei „**c:\ERPAdmin\Artikel.xls“** unter Verwendung des Layouts „**c:\ERPAdmin\Artikel.bpl**“ importiert.

Import mit Wildcard (*)

Für den zeitgleichen Import mehrerer Dateien z.B. mehrerer WebShop-Bestellungen kann ein Wildcard versendet werden. **Achtung: Es darf nur ein Wildcard (*) verwendet werden.**

ERPAdmin -Import "c:\ERPAdmin\Bestellungen.bpl" " c:\ERPAdmin\Order*.txt"
Es werden alle Text-Dateien „**c:\ERPAdmin\Order*.txt“** unter Verwendung des Import-Layouts „**c:\ERPAdmin\Bestellungen.bpl**“ importiert.

66 Der Befehl „-InsertDataFromFile“ PRO

ERPAdmin -InsertDataFromFile <Datenbank> <SuchIndexListe> /Datei:<CSV-Filename> /MoveToHis /MoveToBad /Delete

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin fügt neue Datensätze in eine Datenbank von **büro+** ein ohne dass ein Importlayout benötigt. Damit können auch Datenbanken mit Werten befüllt werden für die keine Importe in **büro+** zur Verfügung stehen wie z.B. das Lagerbuch.

Es werden nur neue Datensätze in einer Datenbank hinzugefügt. Existiert bereits ein Eintrag zu einem **SuchIndex** wird der Datensatz übersprungen. Für das aktualisieren von Datensätzen verwenden Sie den Befehl: **ERPAdmin -WriteDataFromFile**

<Datenbank>
büro+ Datenbank in die die neuen Daten geschrieben werden sollen.

<SuchIndexListe>
Übergibt eine durch Semikolon getrennte Liste mit den **SuchIndex**-Feldern. Eine Liste der erlaubten **SuchIndex**-Felder kann mit der Funktion **ERPAdmin -ListDatenbankFelder /Index** erzeugt werden

/Datei:<CSV-Filename>
Datei in ANSI-CSV-Format deren Daten in die **Datenbank** hinzugefügt werden sollen. Wildcards (*) sind nicht erlaubt. Die Spalten-Überschriften müssen den Feldbezeichnern aus **büro+** entsprechen jedoch ohne den Datenbanknamen. Spalten die nicht eingelesen werden sollen beginnen mit dem Namen **DUMMY**.

Beispiel CSV-Beispiel-Datei für die Datenbank Warengruppen:

```
WgrNr;Bez;WgSumVDrKz;DUMMY
101;Dienstleistungen;Ja;2,5
102;Hardware;Nein;3,7
103;Software;Ja;1,8
```

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung dieser Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

> **Hinweis:**
> Wenn z.B. Adressen eingefügt werden, ist darauf zu achten, dass Anschriften NICHT direkt beim Erzeugen einer Adresse mit hinzugefügt werden können, da es sich bei den Anschriften um eine eigenständige Datenbank handelt. Gleiches gilt z.B. auch für die Ansprechpartner, etc.

Beispiele:

Für das Erzeugen von Wagengruppen kann ein Aufruf wie folgt aussehen:
ERPAdmin -InsertDataFromFile Warengruppen WgrNr /Datei:C:\ERPAdmin\Warengruppen.csv

Für das Hinzufügen von Ansprechpartnern sieht der Aufruf z.B. wie folgt aus:
ERPAdmin -InsertDataFromFile Ansprechpartner "AdrNr;AnsNr;AspNr" /Datei:C:\ERPAdmin\Ansprechpartner.csv

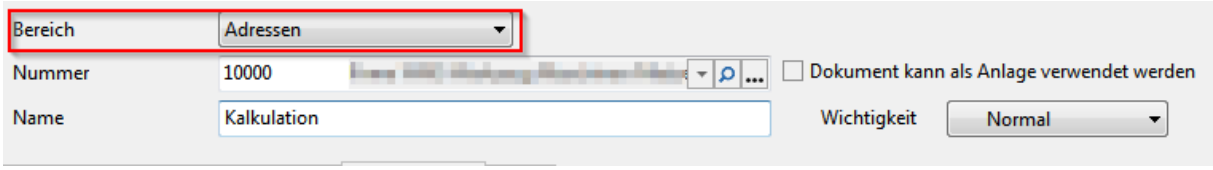
67 Der Befehl „-InsertDocument“ PRO

ERPAdmin -InsertDocument <Bereich> <Index> <Dateiname> <Beschreibung> <LinkType> <Priorität> /Cond:<Bedingung> /AnlageKz

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin fügt eine oder mehrere Dateien **Dateiname** in die Dokumenten-Datenbank von **büro+** ein.

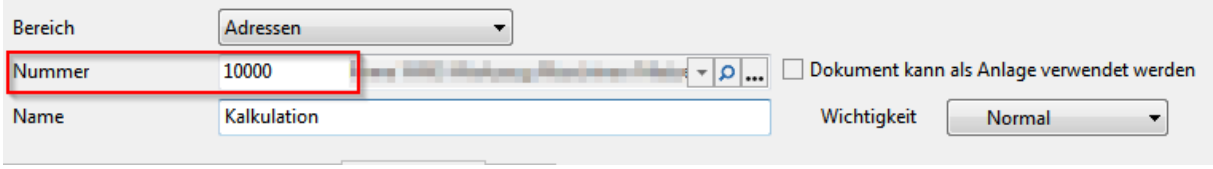
<Bereich>
Gibt an in welchem Bereich der Dokumentendatenbank die Daten hinzugefügt werden sollen. Beispiel: **Adressen**



Unterstützt werden aktuell die folgenden Bereiche:

- **Adressen** Zuordnung über die Adressnummer
- **Anlagen** Zuordnung über die Anlagennummer
- **Artikel** Zuordnung über die Artikelnummer
- **Kostenstellen** Zuordnung über die Kostenstellennummer
- **Projekte** Zuordnung über die Projektnummer
- **SEPAMandate** Zuordnung über die Mandats-Referenz
- **Vorgang** Zuordnung über die Auftragsnummer
- **VorgngArchiv** Zuordnung über die Auftragsnummer

<Index>
Schlüsselwert dem die Dateien innerhalb der Datenbank zugeordnet werden sollen.
Beispiel: 10000 für Adressnummer 10000



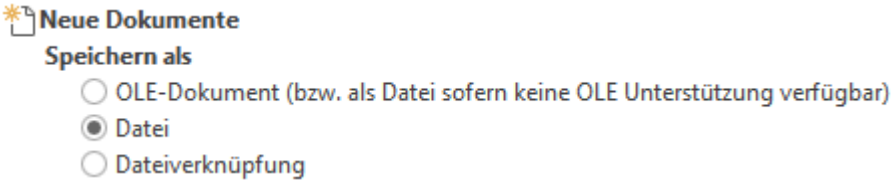
<Dateiname>
Name der Datei inkl. Pfadangabe die hinzugefügt werden sollen. Wenn mehrere Dateien hinzugefügt werden sollen darf auch ein Wildcard (*) verwendet werden.
Beispiel: Kalkulation.xlsx oder Kalkulation*.xlsx

<Beschreibung>
Beschreibungstext unter dem die Datei in den Dokumenten angezeigt werden soll. Werden mehrere Dateien durch ein Wilcard (*) eingelesen entspricht die Beschreibung dem Dateinamen ohne Endung.

<LinkType> **(optional)**
Gibt an wie die Datei eingefügt werden soll. In der **VAR.INI** kann ein Vorgabewert **C_DocumentDefaultLinkType** hinterlegt werden. Wird kein Wert vorbelegt oder beim Aufruf übergeben wird der Wert **0 (**Datei) verwendet. Folgende Werte stehen zur Verfügung:

- **0** = Datei
- **1** = OLE-Dokument
- **2** = Dateiverknüpfung

Die Einstellung entspricht den Parametern im Bereich Dokumente:



<Priorität> **(optional)**
Gibt die Wichtigkeit an unter der die Datei geführt werden soll. In der **VAR.INI** kann ein Vorgabewert **C_DocumentDefaultPriority** hinterlegt werden. Wird kein Wert vorbelegt oder beim Aufruf übergeben wird der Wert **1 (**Normal) verwendet. Zulässige Werte sind:

- **0** = Niedrig
- **1** = Normal
- **2** = Hoch

/Cond:<Bedingung> **(optional)**
Soll ein Dokument einem Vorgang/Archiv-Vorgang nicht über die Auftragsnummer sondern über die Belegnummer zugeordnet werden, kann diese mit der Option **/Cond:BelegNr** erfolgen.

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

/AnlageKz **(optional)**
Mit dieser Option wird das Kennzeichen „**Dokument kann als Anlage verwendet werden**“ in den Dokumenten gesetzt.

Beispiele:

ERPAdmin -InsertDocument Artikel 11500 c:\ERPAdmin\Datenblatt.pdf
Fügt die Datei **c:\ERPAdmin\Datenblatt.pdf** mit der Bezeichnung **Datenblatt** im Bereich **Artikel** bei Artikel-Nr. **11500** in die Dokumente als **Datei** ein. Die **Priorität** ist Normal

ERPAdmin -InsertDocument Artikel 11500 "c:\ERPAdmin\Datenblatt.pdf" Herstellerdaten 0 2
Fügt die Datei **c:\ERPAdmin\Datenblatt.pdf** mit der Bezeichnung **Herstellerdaten** im Bereich **Artikel** bei Artikel-Nr. **11500** in die Dokumente als **Datei** ein. Die **Priorität** ist Hoch

ERPAdmin -InsertDocument Artikel 11500 "c:\ERPAdmin\Datenblatt*.pdf"
Fügt die Dateien **c:\ERPAdmin\Datenblatt*.pdf** im Bereich **Artikel** bei Artikel-Nr. **11500** in die Dokumente als **Datei** ein. Die **Priorität** ist normal. Als Beschreibung werden die Dateinamen ohne Endung verwendet.

ERPAdmin -InsertDocument Adressen 23456 "c:\ERPAdmin\Ausschreibung.pdf" "" 2 1
Fügt die Datei **c:\ERPAdmin\Ausschreibung.pdf** mit der Bezeichnung **Ausschreibung** im Bereich **Adressen** bei Adress-Nr.**23456** in die Dokumente als **Verknüpfung** ein. Die **Priorität** ist Normal

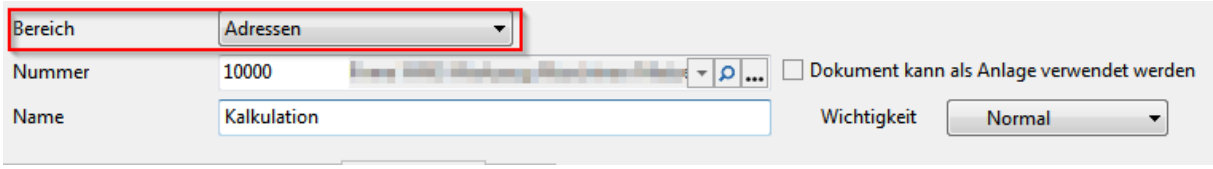
68 Der Befehl „-InsertDocumentFromDir“ PRO

ERPAdmin -InsertDocumentFromDir <Bereich> <IndexMaske> <Dateinamen> <LinkType> <Priorität> /NoAdrNrNotExistError /MoveToHis /MoveToBad /Delete /Cond:<Bedingung> /AnlageKz

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin fügt eine oder mehrere Dateien aus einem Verzeichnis in die Dokumenten-Datenbank von **büro+** ein. Dabei erfolgt die Zuordnung automatisch anhand des **Dateinamens**. Als Dokumentenbeschreibung wird der Dateiname verwendet ohne die ermittelten **Indexmaske**.

<Bereich>
Gibt an in welchem Bereich der Dokumentendatenbank die Daten hinzugefügt werden sollen. Beispiel: **Adressen**



Unterstützt werden aktuell die folgenden Bereiche:

- **Adressen** Zuordnung über die Adressnummer
- **Artikel** Zuordnung über die Artikelnummer
- **Projekte** Zuordnung über die Projektnummer
- **SEPAMandate** Zuordnung über die Mandats-Referenz
- **Vorgang** Zuordnung über die Auftragsnummer
- **VorgangArchiv** Zuordnung über die Auftragsnummer

<IndexMaske>
Zur Ermittlung eines Indexwertes muss **ERPAdmin** mitgeteilt werden, wo im Dateinamen (Linken x-Zeichen oder rechten X-Zeichen) sich ein Indexwert befindet. Folgende Angaben sind zulässig:

- **L4:** Interpretiert die linken 4 Zeichen des Dateinamens als Indexwert
- **L5:** Interpretiert die linken 5 Zeichen des Dateinamens als Indexwert
- …
- **L9:** Interpretiert die linken 9 Zeichen des Dateinamens als Indexwert
- **L_:** Interpretiert alle Zeichen links vom ersten Unterstrich im Dateinamen als Indexwert
- **L-:** Interpretiert alle Zeichen links vom ersten Bindestrich im Dateinamen als Indexwert
-
- **R4:** Interpretiert die rechten 4 Zeichen des Dateinamens als Indexwert
- **R5:** Interpretiert die rechten 5 Zeichen des Dateinamens als Indexwert
- **…**
- **R9:** Interpretiert die rechten 9 Zeichen des Dateinamens als Indexwert
- **R_:** Interpretiert alle Zeichen rechts vom ersten Unterstrich im Dateinamen als Indexwert
- **R-:** Interpretiert alle Zeichen rechts vom ersten Bindestrich im Dateinamen als Indexwert

L5 interpretiert vom Dateinamen „**11500**Datenblatt.xls“ **11500** als Indexwert.
R7 interpretiert vom Dateinamen „Datenblatt**1234567**.xls“ **1234567** als Indexwert.

Zusätzlich kann die Bezeichnung mit dem die Datei in den Dokumenten abgelegt werden soll angepasst werden. Es kann ein freier Text gewählt und alle Platzhalter aus dem Kapitel „**Platzhalter und variable Parameter**“ verwendet werden. Um den aktuellen Dateinamen an einer bestimmten Stelle einzufügen (ohne die Indexnummer) verwenden Sie den Platzhalter **@NAME@** an der entsprechenden Stelle. Um den Index einzufügen (Adress-Nummer, Artikel-Nummer, Auftragsnummer, etc.) verwenden Sie den Platzhalter **@INDEX@**.

Beispiel:	**Dateiname:**	**11500Datenblatt.xls**
	IndexMaske:	**"L5@DATE@ @NAME@-Technik (@INDEX@)"**

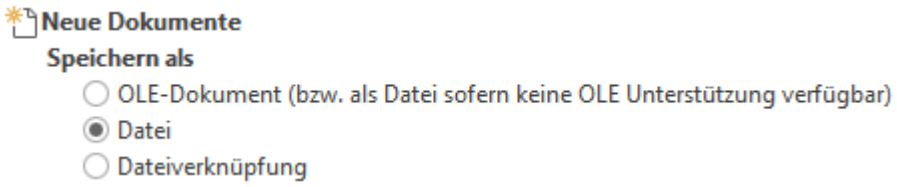
Interpretiert die linken 5-Zeichen (**L5**) als Indexnummer. Der Dokumentenname besteht dann aus dem aktuellen Tagesdatum (**@DATE@**), den Dateinamen ohne Endung (**@NAME@=Datenblatt**) und dem Index (**@INDEX@=11500**) z.B. **15.01.2021 Datenblatt-Technik (11500)**

<Dateinamen>
Pfadangabe zu einem Verzeichnis mit Dokumenten. Wenn mehrere Dateien hinzugefügt werden sollen darf auch ein Wildcard (*****) verwendet werden. Sollen alle Dateien hinzugefügt werden geben Sie als Dateifilter nur den Stern (*****) an.
Beispiel: **c:\ERPAdmin\Dok*** oder **c:\ERPAdmin\Dok*.pdf**

<LinkType> **(optional)**
Gibt an wie die Datei eingefügt werden soll. In der **VAR.INI** kann ein Vorgabewert **C_DocumentDefaultLinkType** hinterlegt werden. Wird kein Wert vorbelegt oder beim Aufruf übergeben wird der Wert **0 (**Datei) verwendet. Folgende Werte stehen zur Verfügung:

- **0** = Datei
- **1** = OLE-Dokument
- **2** = Dateiverknüpfung

Die Einstellung entspricht den Parametern im Bereich Dokumente:



<Priorität> **(optional)**
Gibt die Wichtigkeit an, unter der das Dokument in büro+ geführt werden soll. In der **VAR.INI** kann ein Vorgabewert **C_DocumentDefaultPriority** hinterlegt werden. Wird kein Wert vorbelegt oder beim Aufruf übergeben wird der Wert **1 (**Normal) verwendet. Zulässige Werte sind:

- 0 = Niedrig
- 1 = Normal
- 2 = Hoch

/NoAdrNrNotExistError **(optional)**
Existiert die im Dateinamen übergebene Indexnummer (**AdrNr, ArtNr** oder **AuftrNr**) nicht in der **büro+** Datenbank, kann über diesen Parameter gesteuert werden, dass in diesem Fall keine Fehlermeldung sondern nur eine Warnung ausgegeben wird. Wird diese Option übergeben, bekommt der Anwender nur eine Warnmeldung.

Unabhängig vom gewählten **Bereich** (Adressen, Artikel, Vorgang, etc.) ist diese Option für alle Bereiche gleich!

/Cond:<Bedingung> **(optional)**
Soll ein Dokument einem Vorgang/Archiv-Vorgang nicht über die Auftragsnummer sondern über die Belegnummer zugeordnet werden, kann diese mit der Option **/Cond:BelegNr** erfolgen.

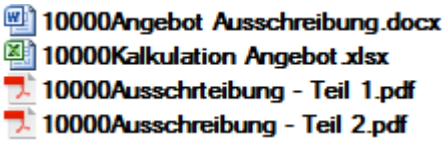
/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

/AnlageKz **(optional)**
Mit dieser Option wird das Kennzeichen „**Dokument kann als Anlage verwendet werden**“ in den Dokumenten gesetzt.

Beispiele:

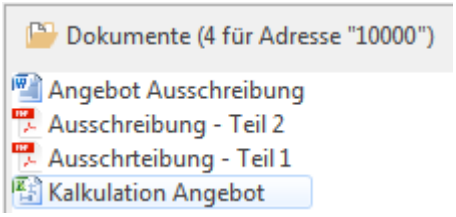
ERPAdmin -InsertDocumentFromDir Adressen L5 c:\ERPAdmin\Dok*
Fügt die alle Dateien aus dem Verzeichnis **c:\ERPAdmin** im Bereich **Adressen** ein. Als Indexwert werden die Linken 5-Zeichen interpretiert. Als Dokumentennamen werden die verbleibenden Zeichen interpretiert. Die **Priorität** ist Normal

In dem Verzeichnis **c:\ERPAdmin\Dok** befinden sich die folgenden Dateien:



[Verzeichnis C:\ERPAdmin\Dok\]

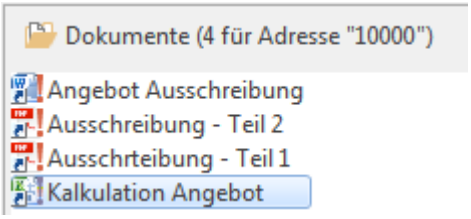
Nach dem Aufruf erscheinen diese Dateien in den Dokumenten der Adresse **10000** wie folgt:



[Adressen->Details Dokumente]

ERPAdmin -InsertDocumentFromDir Adressen L5 c:\ERPAdmin\Dok* 2 2
Fügt die alle Dateien aus dem Verzeichnis **c:\ERPAdmin** im Bereich **Adressen** als **Dateiverknüpfung** ein. Als Indexwert werden die Linken 5-Zeichen interpretiert. Als Dokumentennamen werden die verbleibenden Zeichen interpretiert. Die **Priorität** ist **Hoch**

Dass Ergebnis in **büro+** sieht dann wie folgt aus:



[Adressen->Details Dokumente]

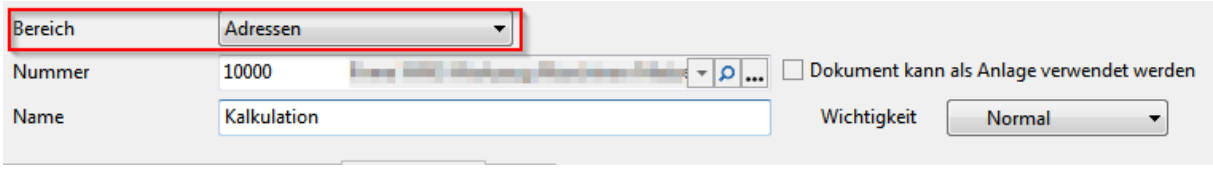
69 Der Befehl „-InsertDocumentFromFile“ PRO

ERPAdmin -InsertDocumentFromFile <Bereich> <CSV-Datei> <SpalteNr-Dateiname> <SpalteNr-Index> <LinkType> <Priorität> /NoAdrNrNotExistError /MoveToHis, /MoveToBad, /Delete /Header /AnlageKz

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin fügt eine oder mehrere Dateien aus einem Verzeichnis in die Dokumenten-Datenbank von **büro+** ein. Dabei erfolgt die Zuordnung automatisch über eine CSV-Datei. Als Dokumentenbeschreibung wird der Dateiname verwendet ohne Endung.

<Bereich>
Gibt an in welchem Bereich der Dokumentendatenbank die Daten hinzugefügt werden sollen. Beispiel: **Adressen**



Unterstützt werden aktuell die folgenden Bereiche:

- **Adressen** Zuordnung über die Adressnummer
- **Artikel** Zuordnung über die Artikelnummer
- **Projekte** Zuordnung über die Projektnummer
- **SEPAMandate** Zuordnung über die Mandats-Referenz
- **Vorgang** Zuordnung über die Auftragsnummer
- **VorgangArchiv** Zuordnung über die Auftragsnummer

<CSV-Datei>
CSV-Datei im ANSI-Format welche in zwei Spalten zum einen dem Dateinamen des Dokumentes und zum anderen mit dem Suchindex dem das Dokument zugeordnet werden soll. Soll das Dokument z.B. einer Adresse zugeordnet werden, muss hier die entsprechende Adressnummer übergeben werden. Bei einem Artikel muss die Artikelnummer übergeben werden. Sollen mehrere CSV-Dateien verarbeitet werden darf ein Wildcard (*****) verwendet werden.

Enthält die **CSV-Datei** eine Kopfzeile muss die Option **/HEADER** angegeben werden.

<SpalteNr-Dateiname>
Gibt die Spalte innerhalb der **CSV-Datei** an in der die Dateinamen der Dokumente zu finden sind. Liegen diese nicht im gleichen Verzeichnis, muss der komplette Pfad inkl. Laufwerk mit angegeben werden.

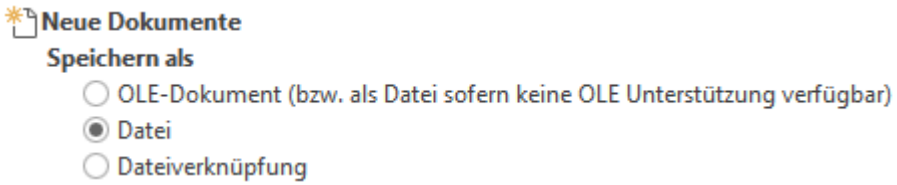
<SpalteNr-Index>
Gibt die Spalte innerhalb der **CSV-Datei** an mit dem des Dokument einem Datensatz aus dem **Bereich** eindeutig zugeordnet werden kann. Beispiel: Bereich=Adressen -> Index=Adressnummer.

<LinkType> **(optional)**
Gibt an wie die Datei eingefügt werden soll. In der **VAR.INI** kann ein Vorgabewert

C_DocumentDefaultLinkType hinterlegt werden. Wird kein Wert vorbelegt oder beim Aufruf übergeben wird der Wert **0 (**Datei) verwendet. Folgende Werte stehen zur Verfügung:

- **0** = Datei
- **1** = OLE-Dokument
- **2** = Dateiverknüpfung

Die Einstellung entspricht den Parametern im Bereich Dokumente:



<Priorität> **(optional)**
Gibt die Wichtigkeit an unter der die Datei geführt werden soll. In der **VAR.INI** kann ein Vorgabewert **C_DocumentDefaultPriority** hinterlegt werden. Wird kein Wert vorbelegt oder beim Aufruf übergeben, wird der Wert **1 (**Normal) verwendet. Zulässige Werte sind:

- 0 = Niedrig
- 1 = Normal
- 2 = Hoch

/NoFileNotExistError **(optional)**
Existiert die CSV-Datei nicht wird keine Fehlermeldung ausgegeben. Das gleiche gilt für nicht vorhandene Dokumente die in der CSV-Datei angegeben wurden.

/Header **(optional)**
Enthält die **CSV-Datei** eine Kopfzeile muss die Option **/Header** angegeben werden.

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

/AnlageKz **(optional)**
Mit dieser Option wird das Kennzeichen „**Dokument kann als Anlage verwendet werden**“ in den Dokumenten gesetzt.

Beispiel:

ERPAdmin -InsertDocumentFromFile Artikel Liste.csv 1 2
Liest die Datei **Liste.csv** ein und Importiert die angegebenen Dateien aus Spalte **1** (A) in die Dokumente im Bereich **Artikel** und ordnet diese anhand der Artikel-Nummer in Spalte **2** (B) zu.

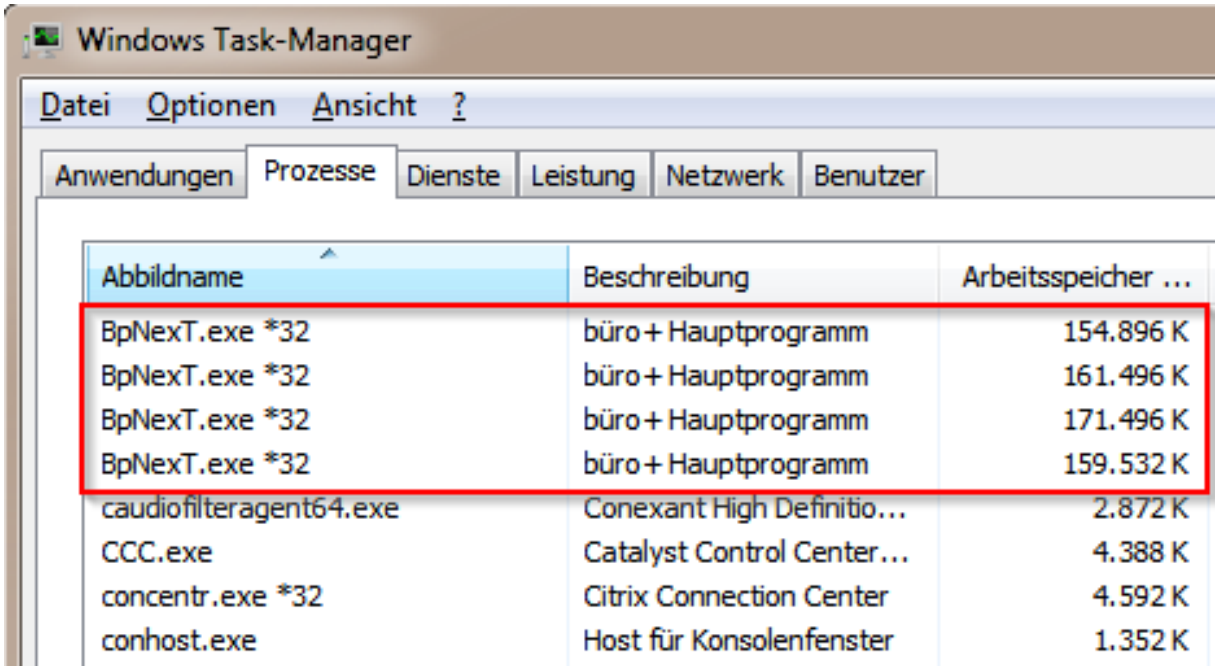
Beispiel für Datei „**Liste.csv**“

	A	B	C
1	039673464.PDF	10500	33333
2	134364455.PDF	11453	33333
3	649658474.PDF	17934	33333

70 Der Befehl „-KillProcess“ BAS

ERPAdmin -KillProcess <ProzessName> <Sekunden>

Beim täglichen Arbeiten kann es immer wieder passieren, dass COM- oder andere Anwendungen nicht korrekt beendet werden und dadurch das System blockieren. Diese Programme (Prozesse) können dann nur über den Task-Manager beendet werden oder automatisiert mit **ERPAdmin.**



In obigem Beispiel wurde die Anwendung **BpNexT.exe** (büro+) vier Mal geöffnet und nicht korrekt beendet obwohl auf dem Desktop keine Anwendung mehr sichtbar ist.

<ProzessName>
Name der Anwendung/des Prozesses aus dem Taskmanager welche beendet werden soll.

<Sekunden>
Anzahl an Sekunden die mind. vergangen sein müssen seit dem Start der Anwendung bevor diese beendet wird.

Beispiele:

Es werden alle noch laufenden Instanzen von **büro+** (BpNexT.exe) beendet welche seit mehr als 4 Stunden (=14400 Sekunden) laufen.
ERPAdmin -KillProcess BpNexT.exe 14400

Beendet alle noch laufenden Instanzen von **büro+** (BpNexT.exe) im Speicher
ERPAdmin -KillProcess BpNexT.exe 1

Beende alle noch laufenden Excel-Instanzen
ERPAdmin -KillProcess excel.exe 1

71 Der Befehl „-KontaktSenden“ PRO

ERPAdmin -KontaktSenden <Datenbank> <LayoutGruppe> <LayoutName> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

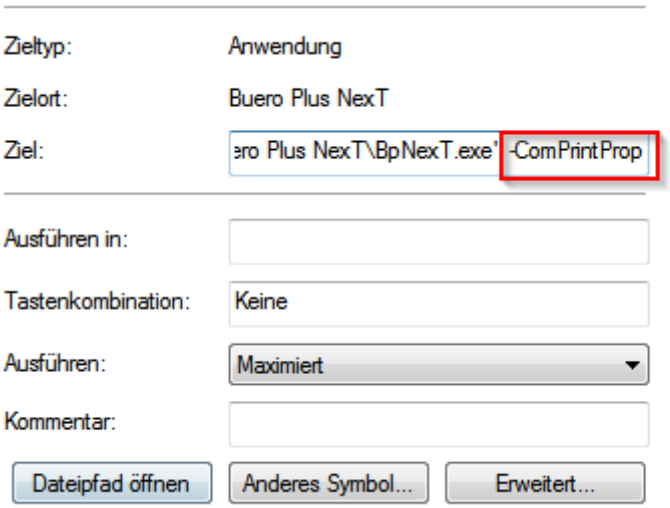
Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin versendet einen Kontakt per Email. Dabei kann es sich sowohl um einen einzelnen Kontakt als auch um einen Serienkontakt handeln.

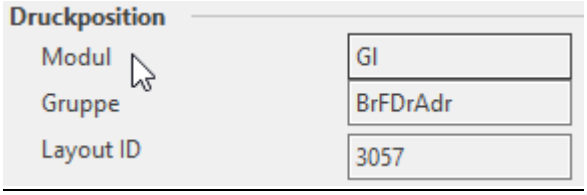
<Datenbank>
Di Datenbank gibt de Bereich an aus dem der Versand ausgeführt werden soll. Zulässige Datenbanken sind:

- **Adressen**
- **Kontakte**
- **KontakteArchiv**
- **Vorgang**
- **VorgangArchiv**

<LayoutGruppe>
Hier muss das Modul- und die Gruppe aus **büro+** angegeben werden. Damit diese Werte angezeigt werden, muss **büro+** mit dem Parameter **„-ComPrintProp**“ (ohne die Anführungszeichen) gestartet werden.



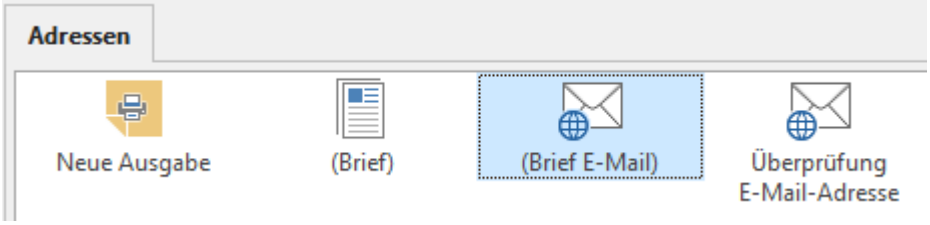
Die Angaben stehen in den Eigenschaften jedes Kontaktlayouts auf dem Register „**Parameter**“ in der Gruppe „**Druckoptionen**“.



[Beispiel für ein Kontaktlayout aus den Adressen]

Die **Layoutgruppe** setzt sich aus dem Modul „**Gl**“ und der Gruppe „**BrFDrAdr**“ zusammen. In obigem Beispiel würde die **Layoutgruppe** dann **GlBrFDrAdr** heißen.

<LayoutName>
Name eines Kontaktlayouts aus **büro+** z.B. „(Brief E-Mail)“. Bitte darauf achten, dass wenn Leerzeichen im **LayoutNamen** enthalten sind der ganze Ausdruck in Doppelte Anführungszeichen gesetzt werden muss.



[Liste mit Namen der Kontakt-Layouts aus dem Bereich Stammdaten->Adressen]

In obigem Beispiel würde der Parameter **"(Brief E-Mail)"** heißen.

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen, nach denen auch sortiert werden kann. Bei der Versendung von Serien-Kontakten muss auf die Verwendung verzichtet werden.

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet. Bei der Versendung von Serien-Kontakten muss auf die Verwendung verzichtet werden.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Beispiele:

Sende die Kontakt-E-Mail „**Aktuelle Preisliste**“ (Layoutgruppe: **GlBrFDrAdr)** aus dem Kontakte-Bereich **Adressen** an alle Kunden mit den Adressnummern **10000** bis **19999**
ERPAdmin -KontaktSenden Adressen GlBrFDrAdr "Aktuelle Preisliste" /Von:AdrNr=10000 /Bis:AdrNr=19999

Sende die Kontakt-Serien-E-Mail „**Serie-Aktuelle Preisliste**“ (Layoutgruppe: **GlSBrFDrAdr)** aus dem Serien-Kontakte-Bereich **Adressen**. Dabei werden alle Verteiler und sonstigen Einstellungen beachtet.
ERPAdmin -KontaktSenden Adressen GlSBrDrAdr "Serie-Aktuelle Preisliste"

72 Der Befehl „-LagerAusbuchen“ PRO

ERPAdmin -LagerAusbuchen <LagerVon> <LagerBis> <ArtikelVon> <ArtikelBis> /Filter:<Filterbedingung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl können Sie vorhandene Lagerbestände automatisch ausbuchen lassen. Dabei werden Seriennummern ebenso berücksichtigt wir hinterlegte Chargen. Sofern die Lagerbuchführung aktiviert ist, werden auch die entsprechenden Einträge im Lagerbuch erzeugt.

Hinweis 1:
Lagerbestände mit Auftragsnummer werden beim ausbuchen im Lagerbuch ohne Auftragsnummer geführt.

Hinweis 2:
Bei Lagern mit **Stellplatzverwaltung** werden die Lagerplatzbewegungen nicht automatisch mitgeführt. Diese müssen Sie ggf. innerhalb von **büro+** nachbuchen.

<LagerVon> und <LagerBis> **(optional)**
Mit den beiden Parametern **LagerVon** und **LagerBis** können Sie angeben, welche Lager beachtet werden sollen. Wird kein Lager ausgewählt werden die Bestände aller Lager ausgebucht.

<ArtikelVon> und **<ArtikelBis>** **(optional)**
Mit diesen beiden Parametern **ArtikelVon** und **ArtikelBis** können Sie den Bereich auf Artikelnummern-Ebene weiter einschränken. Wird kein Artikel ausgewählt, werden alle Artikel ausgebucht.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich Der Lager-Datenbank weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Um den Prozess zu beschleunigen wird der Filter mit folgendem Wert von **ERPAdmin** vorbelegt: **(Mge<>0 and Mge<>NULL**). Wird vom Benutzer ein Filterwert eingegeben, wird der folgenden Filter an **büro+** übergeben: **(Mge<>0 and Mge<>NULL**) **and (<Filter>)**

Beispiele:

Alle vorhandenen Lagerbestände sollen ausgebucht werden.
ERPAdmin -LagerAusbuchen

Alle Lagerbestände aus dem Lager HAULAG ausbuchen.
ERPAdmin -LagerAusbuchen HAULAG

Alle Lagerbestände der Artikel 10000 bis 20000 aus dem Lager HAULAG ausbuchen
ERPAdmin -LagerAusbuchen HAULAG HAULAG 10000 20000

Buche alle Lagerbestände aus Kundenlagern aus
ERPAdmin -LagerAusbuchen KD10000 KD69999

Buche alle Lagerbestände aus STAMMLAGER aus die negativ sind.
ERPAdmin -LagerAusbuchen STAMMLAGER STAMMLAGER "" "" "/Filter:Mge<0"

Dieser Befehl darf nicht für Lager verwendet werden welche das Kennzeichen „Lagerplatzverwaltung (Chaotische Lagerhaltung)“ gesetzt haben.

73 Der Befehl „-LagerbestandUmbuchen“ PRO

ERPAdmin -LagerbestandUmbuchen <AusLager> <NachLager> <ArtikelVon> <ArtikelBis> /Filter:<Filterbedingung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl können Sie vorhandene Lagerbestände automatisch auf ein anderes Lager umbuchen. Dabei werden Seriennummern ebenso berücksichtigt wir hinterlegte Chargen. Sofern die Lagerbuchführung aktiviert ist, werden auch die entsprechenden Einträge im Lagerbuch erzeugt.

> **Hinweis:**
> Bei Lagern mit **Stellplatzverwaltung** werden die Lagerplatzbewegungen nicht automatisch mitgeführt. Diese müssen Sie ggf. innerhalb von **büro+** nachbuchen.

<AusLager>
Gibt an aus welchem Lager Bestände umgebucht werden sollen

<NachLager>
Gibt an in welches Lager die Bestände umgebucht werden sollen.

<ArtikelVon> und **<ArtikelBis>** **(optional)**
Mit diesen beiden Parametern **ArtikelVon** und **ArtikelBis** können Sie den Bereich auf Artikelnummern-Ebene weiter einschränken. Wird kein Artikel ausgewählt, werden alle Artikel umgebucht die im Lager vorhanden sind.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich Der Lager-Datenbank weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Um den Prozess zu beschleunigen wird der Filter mit folgendem Wert von **ERPAdmin** vorbelegt: **(Mge<>0 and Mge<>NULL**). Wird vom Benutzer ein Filterwert übergeben, wird der Filter-Ausdruck wie folgt an **büro+** übergeben: **(Mge<>0 and Mge<>NULL**) **and (<Filter>)**

> **Hinweis:**
> Beim Umbuchen von Beständen mit Charge ist darauf zu achten, dass beide Lager die gleichen Einstellungen in den Parametern besitzen bzgl. Charge und Verfalldatum. Das gleiche gilt die die Umbuchung von Artikeln mit Seriennummer.

Beispiele:

ERPAdmin - LagerbestandUmbuchen STAMMLAGER HAULAG
Alle vorhandenen Lagerbestände sollen aus dem STAMMLAGER in das HAULAG umgebucht werden.

ERPAdmin - LagerbestandUmbuchen HAULAG STAMMLAGER 10000 20000

Alle Lagerbestände der Artikel 10000 bis 20000 aus dem Lager HAULAG in das STAMMLAGER umbuchen.

Hinweis:
Dieser Befehl darf nicht für Lager verwendet werden welche das Kennzeichen „Lagerplatzverwaltung (Chaotische Lagerhaltung)“ gesetzt haben.

74 Der Befehl „-ListAdressVerteiler“ BAS

ERPAdmin -ListAdressVerteiler /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller Adress-Verteiler aus **büro+** und speichert das Ergebnis in einer **CSV-Datei**.

/Datei:<CSV-Datei> **(optional)**
Speichert eine Liste aller vorhandenen Adress-Verteiler im CSV-Format unter **CSV-Datei** ab. Wird kein Dateiname übergeben wird das Ergebnis unter „**Liste-Adress-Verteiler.csv**“ gespeichert.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListAdressVerteiler /Datei:Liste-Verteiler.csv /Header /Email:Demo@ERPAdmin.de "/Subject:Liste Adress-Verteiler"
Erzeugt eine CSV-Datei mit allen Adress-Verteilern und speichert diese in der CSV-Datei „**Liste-Verteiler.csv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste Adress-Verteiler“**.

75 Der Befehl „-ListAll“ BAS

ERPAdmin -ListAll /NoDefaults /Header /Email:<E-Mail-Adresse> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin erzeugt CSV-Exporte mit den jeweils folgenden Informationen:

- Abteilungen
- Adress-Verteiler
- Bankverbindungen des Mandanten
- Benutzer
- Build-Stände auf diesem Rechner aller installierten Updates
- Drucker auf dem lokalen PC
- Einheiten
- Erfassungsvorlagen
- Exporte (aus allen Bereichen)
- Filter
- Importe (aus allen Bereichen)
- Landeszuweisung für Umsatzsteuerkategorie
- Layouts (aus allen Bereichen)
- Liste aller Datenbankfelder
- Liste der Mandantenberechtigungen
- Rohstoffkurse
- Reguläre Ausdrücke
- Regeln (aus allen Bereichen)
- Selektionsfelder (Aller Datenbanken)
- Stammlager
- Textbausteine
- Umsatzsteuer-Schlüssel
- Umsatzsteuerkategorien
- Vorgangs-Arten (mit Belegnummern, Lagern etc.)
- Wechselkurse

/Header **(optional)**
Erzeugt eine Kopfzeile in den CSV-Dateien mit den Feldbezeichnern.

/NoDefaults **(optional)**
Sollen nur die veränderten Layouts ausgegeben werden und nicht die Standard **büro+**-Layouts (Layouts die in Klammern stehen) kann diese Option verwendet werden.

/Email:<E-Mail-Adresse> **(optional)**
Versendet die CSV-Dateien an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

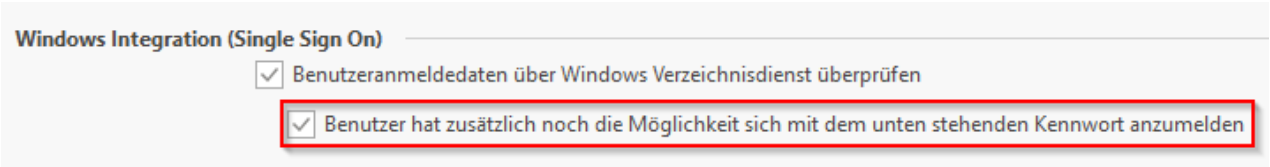
Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Aufruf ohne VAR.INI – Anmeldung an büro+ über Dialog

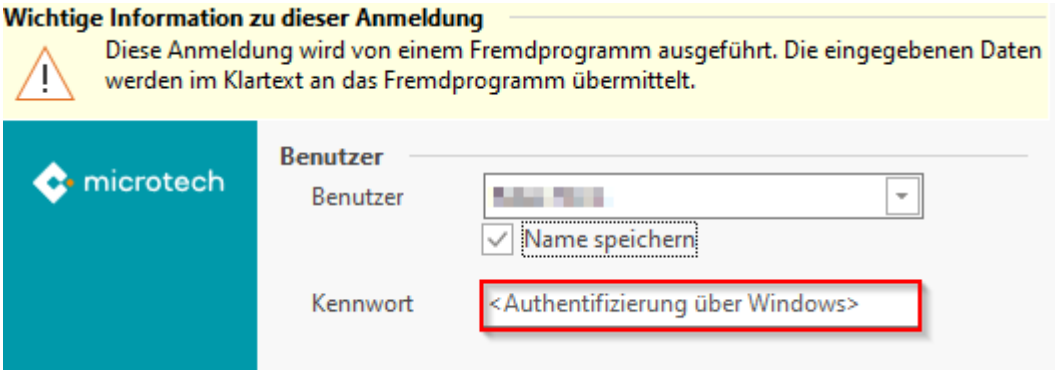
Bei einem Aufruf von **ERPAdmin** ohne eine **VAR.INI** öffnet sich der Anmelde-Dialog von büro+ um die Anmeldedaten und damit die Berechtigung zu prüfen.

Sollte bei einem Benutzer „**Windows Integration (Single Sign On)**“ aktiviert sein, …

muss ein Kennwort eingegeben werden da **ERPAdmin** sonst nicht in der Lage ist sich bei büro+ anzumelden. Die Option zur Eingabe eines Passwortes muss dazu beim Benutzer ebenfalls aktiviert sein.



Beim erscheinen des nachfolgenden Anmeldedialoges muss in dem Feld „**Kennwort**“ dann das Kennwort eingegeben werden anstelle von „**<Authentifizierung über Windows>**“.



76 Der Befehl „-ListBuilds“ BAS

ERPAdmin -ListBuilds /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller bisher auf diesem PC installierten **büro+** Build-Stände inkl. Installations-Datum aus und speichert das Ergebnis in einer CSV-Datei.

/Datei:<CSV-Datei> **(optional)**
Speichert eine Liste aller vorhandenen **büro+** Filter im CSV-Format unter **CSV-Datei** ab. Wird kein Dateiname übergeben wird das Ergebnis unter „**Liste buero+-builds.csv**“ gespeichert.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListBuilds /Datei:Liste_Builds.csv /Header /Email:Demo@ERPAdmin.de "/Subject:Liste mit büro+ Build-Installationen"
Erzeugt eine CSV-Datei mit allen vorhandenen **büro+** BUILD-Installationen und speichert diese in der CSV-Datei „**Liste_Builds.csv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste mit büro+ Build-Installationen“**.

Das Ergebnis könnte wie folgt aussehen:

	A	B
1	BUILD	Datum Installation
2	19.0.6004.0	04.09.2018 12:00
3	22.15.6548.0	17.03.2022 11:52
4	22.15.6552.0	17.03.2022 12:01
5	22.15.6553.0	22.03.2022 11:41
6	22.25.6568.0	11.05.2022 08:44
7	22.25.6580.0	24.06.2022 15:30
8	22.35.6588.0	26.07.2022 09:05
9	22.35.6601.0	07.09.2022 19:13
10	22.35.6702.0	20.09.2022 18:51

77 Der Befehl „-ListDatenbankFelder“ BAS

ERPAdmin -ListDatenbankFelder <Dateiname> /<Auswahl> /CSV /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Beim Arbeiten mit **büro+** oder **ERPAdmin** benötigt man regelmäßig eine Liste von Feldbezeichner oder von den Such-Indizes die **büro+** verwendet.

Eine Liste inkl. aller Zusatzfelder (Selektionen) und Sortierungen (SuchIndex) können mit dem Befehl **ListDatenbankFelder** erzeugt werden.

<Dateiname> **(optional)**
Gibt dabei den Namen der Datei an unter dem diese Liste gespeichert werden soll. Das Ergebnis ist eine Text-Datei, wenn nicht die Option **/CSV** verwendet wird. Wird kein Dateiname angegeben, belegt **ERPAdmin** den Namen mit „**ERP-Fieldlist.txt**“ vor.

/<Auswahl> **(optional)**
Die Ausgabe kann eingeschränkt werden mit folgenden Vorgaben:
- **/ALLE** Es werden alle Datenfelder und alle Such-Indizes ausgegeben
- **/FELDER** Es werden nur die Datenfelder ausgegeben
- **/INDEX** Es werden nur die Such-Indizes ausgegeben
- **/SELEKTION** Es werden nur die Selektionsfelder ausgegeben zur Dokumentation

Durch die Einschränkung über die Auswahl wird die Ergebnisdatei schneller erzeugt und die Dateilänge insgesamt verkürzt.

/CSV **(optional)**
ERPAdmin Erzeugt eine Ergebnis-Datei im CSV-Format. Dazu ist die Option **/CSV** anzugeben.

Wird die Option **/CSV** verwendet wird als Vorbelegung „**ERP-Fieldlist.csv**“ verwendet. Die Vorbelegung kann in der **VAR.INI** ebenfalls in einer Variable **C_Filename_ERP_FieldList_CSV** geändert werden.

Das Ergebnis kann wie folgt aussehen:

```
DataSet: Postleitzahlen - Postleitzahlen
  Field: ID - ID (AutoInc, 4)  +
  Field: Land - Land (Integer, 4)  +
  Field: PLZ - Postleitzahl (UnicodeString, 15)  +
  Field: *PLZInfo - Land & PLZ (UnicodeString)  +
  Field: *LandKennz - Länderkennzeichen (UnicodeString)  +
  Field: Ort - Ort (UnicodeString, 50)  +
  Field: Vw - Vorwahl (UnicodeString, 15)  +
  Field: BreiteDezGrad - Breitengrad (als Dezimalgrad) (Double, 8)  +
```

Dataset: Gibt den Namen der Tabelle an z.B. „**Postleitzahlen**“
Field: Gibt den Feldnamen z.B. „**PLZ**“, die Bezeichnung z.B. „**Postleitzahl**“ und den Feldtypen z.B. „**UnicodeString**“ an. Dahinter wird die Feldgrößer ausgegeben.
Der Stern (*****) vor einem Feld gibt an, das es sich um eine berechnetes Feld handelt welches nicht importiert werden kann.
Das Plus (**+**) am Ende der Zeile sagt aus, dass auf das Feld zugegriffen werden kann. Ein Minus (**-**) sagt aus, dass kein Zugriff auf das Feld möglich ist.

```
Index: Land - Land
  IndexField:Land - Land
  IndexField:PLZ - Postleitzahl
  IndexField:Ort - Ort
Index: PLZ - Postleitzahl
  IndexField:PLZ - Postleitzahl
  IndexField:Ort - Ort
```

Index: Namen des Such-Index z.B. „**Land**“ und die Bezeichnung des Suchindex.

IndexField: Indexfeld (z.B. „**PLZ**“) und die Bezeichnung des Indexfeldes (z.B. „**Postleitzahl**“). die befüllt werden könne. Das erste Indexfeld muss immer befüllt sein.
Sind mehrere Indexfelder vorhanden müssen diese immer in der Reihenfolge befüllt werden wie Sie in der Liste stehen. Beim Befüllen der Indexfelder dürfen keine Felder leer bleiben außer die nachfolgenden. Beim Index „**Land**“ muss das Indexfeld „**Land**“ immer befüllt sein wohingegen die beiden Felder „**PLZ**“ und „**Ort**“ leer bleiben können.

/Email:<E-Mail-Adresse> **(optional)**

Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **Dateiname** versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**

Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**

Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**

Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

> **Hinweis:**
> Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiele:

ERPAdmin -ListDatenbankFelder

Erzeugt eine Feldliste inkl. aller Such-Indizes und speichert das Ergebnis im aktuellen Verzeichnis unter dem Namen „**ERP-Fieldlist.txt**“ ab.

ERPAdmin -ListDatenbankFelder "c:\ERPAdmin\Feldliste.txt" /ALLE

Erzeugt eine Feldliste inkl. aller Such-Indizes und speichert das Ergebnis unter C:\ERPAdmin\Feldliste.txt ab.

ERPAdmin -ListDatenbankFelder "c:\ERPAdmin\Feldliste.csv" /ALLE /CSV
Erzeugt eine Feldliste inkl. aller Such-Indizes im CSV-Format und speichert diese unter C:\ERPAdmin\Feldliste.csv ab.

ERPAdmin -ListDatenbankFelder "c:\ERPAdmin\Suchindex.txt" /INDEX
Erzeugt eine Liste aller Such-Indizes und speichert diese unter C:\ERPAdmin\Suchindex.txt ab.

ERPAdmin -ListDatenbankFelder "c:\ERPAdmin\Selektionen.txt" /SELEKTION
Erzeugt eine Liste aller Selektionen und speichert diese unter C:\ERPAdmin\Selektionen.txt ab.

Tipp:
Dieser Befehl steht **auch ohne** eine **ERPAdmin** -Lizenz zur Verfügung und kann **ohne VAR.INI** aufgerufen werden. Es erscheint dann der Anmeldedialog zu **büro+** um die Benutzerdaten abzufragen.

78 Der Befehl „-ListEinheiten“ BAS

ERPAdmin -ListEinheiten /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller **büro+** Einheiten aus und speichert diese in einer **CSV-Datei** ab. Wird kein **Dateiname** angegeben wird in dem aktuellen **ERPAdmin**-Verzeichnis die Datei **Layouts_<Rechnername>.csv** erzeugt.

/Datei:<CSV-Datei>
Speichert eine Liste aller vorhandenen **büro+** Filter im CSV-Format unter **CSV-Datei** ab.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListEinheiten /Datei:Liste_Einheiten.csv /Header /Email:Demo@ERPAdmin.de "/Subject: Liste mit büro+ Einheiten"
Erzeugt eine CSV-Datei mit allen vorhandenen **büro+** Einheiten und speichert diese in der CSV-Datei „**Liste_Einheiten.csv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste mit büro+ Einheiten**“.

79 Der Befehl „-ListExports“ BAS

ERPAdmin -ListExports /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller **büro+** Export-Layouts aus und speichert diese in einer **CSV-Datei** ab. Wird kein **Dateiname** angegeben wird in dem aktuellen **ERPAdmin**-Verzeichnis die Datei **Layouts_<Rechnername>.csv** erzeugt.

/Datei:<CSV-Datei>
Speichert eine Liste aller vorhandenen **büro+** Export-Layouts im CSV-Format unter **CSV-Datei** ab.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

> **Hinweis:**
> Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListExports /Datei:Liste_Import-Layouts.csv /Header /Email:Demo@ERPAdmin.de "/Subject: Liste mit büro+ Export-Layouts"
Erzeugt eine CSV-Datei mit allen vorhandenen **büro+** Layouts und speichert diese in der CSV-Datei „**Liste_Import-Layouts.cscv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste mit büro+ Export- Layouts“**.

Tipp:
Dieser Befehl steht **auch ohne** eine **ERPAdmin** -Lizenz zur Verfügung und kann **ohne VAR.INI** aufgerufen werden. Es erscheint dann der Anmeldedialog zu **büro+** um die Benutzerdaten abzufragen.

80 Der Befehl „-ListFilter“ BAS

ERPAdmin -ListFilter /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller **büro+** Filter aus und speichert diese in einer **CSV-Datei** ab. Wird kein **Dateiname** angegeben wird in dem aktuellen **ERPAdmin**-Verzeichnis die Datei **Layouts_<Rechnername>.csv** erzeugt.

/Datei:<CSV-Datei>
Speichert eine Liste aller vorhandenen **büro+** Filter im CSV-Format unter **CSV-Datei** ab.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListFilter /Datei:Liste_Filter.csv /Header /Email:Demo@ERPAdmin.de "/Subject: Liste mit büro+ Filtern"
Erzeugt eine CSV-Datei mit allen vorhandenen **büro+** Filtern und speichert diese in der CSV-Datei „**Liste_Filter.csv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste mit büro+ Filtern“**.

81 Der Befehl „-ListImports“ BAS

ERPAdmin -ListImports /Datei:<CSV-Datei> /Header /NoDefaults /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller **büro+** Import-Layouts aus und speichert diese in einer **CSV-Datei** ab. Wird kein **Dateiname** angegeben wird in dem aktuellen **ERPAdmin**-Verzeichnis die Datei **Layouts_<Rechnername>.csv** erzeugt.

/Datei:<CSV-Datei>
Speichert eine Liste aller vorhandenen **büro+** Import-Layouts im CSV-Format unter **CSV-Datei** ab.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/NoDefaults **(optional)**
Sollen nur die veränderten Layouts ausgegeben werden und nicht die Standard **büro+**-Layouts (Layouts die in Klammern stehen) kann diese Option verwendet werden.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

> **Hinweis:**
> Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListImports /Datei:Liste_Import-Layouts.csv /Header /Email:Demo@ERPAdmin.de "/Subject: Liste mit büro+ Import-Layouts"
Erzeugt eine CSV-Datei mit allen vorhandenen **büro+** Layouts und speichert diese in der CSV-Datei

„**Liste_Import-Layouts.cscv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste mit büro+ Import-Layouts“**.

Tipp:
Dieser Befehl steht **auch ohne** eine **ERPAdmin** -Lizenz zur Verfügung und kann **ohne VAR.INI** aufgerufen werden. Es erscheint dann der Anmeldedialog zu **büro+** um die Benutzerdaten abzufragen.

82 Der Befehl „-ListLayouts“ BAS

ERPAdmin -ListLayouts /Datei:<CSV-Datei> /Header /NoDefaults /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller **büro+** Druck-Layouts aus und speichert diese in einer **CSV-Datei** ab. Wird kein **Dateiname** angegeben wird in dem aktuellen **ERPAdmin**-Verzeichnis die Datei **Layouts_<Rechnername>.csv** erzeugt.

/Datei:<CSV-Datei>
Speichert eine Liste aller vorhandenen **büro+** Druck-Layouts im CSV-Format unter **CSV-Datei** ab.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/NoDefaults **(optional)**
Sollen nur die veränderten Layouts ausgegeben werden und nicht die Standard **büro+**-Layouts (Layouts die in Klammern stehen) kann diese Option verwendet werden.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListLayouts /Datei:Liste_Druck-Layouts.csv /Header /Email:Demo@ERPAdmin.de "/Subject: Liste mit büro+ Druck-Drucklayouts"
Erzeugt eine CSV-Datei mit allen vorhandenen **büro+** Layouts und speichert diese in der CSV-Datei

„**Liste_Druck-Layouts.cscv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste mit büro+ Druck-Layouts“**.

Tipp:
Dieser Befehl steht **auch ohne** eine **ERPAdmin** -Lizenz zur Verfügung und kann **ohne VAR.INI** aufgerufen werden. Es erscheint dann der Anmeldedialog zu **büro+** um die Benutzerdaten abzufragen.

83 Der Befehl „-ListPOPConnect“ PRO

ERPAdmin -ListPOPConnect /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Liest den Inhalt **POPConnect-Postfaches** aus und speichert das Ergebnis in einer **CSV-Datei**. In der **VAR.INI** müssen dazu die folgenden Variablen mit gültigen Zugangsdaten gefüllt sein:

- **C_POPConnectSERVER** Name oder IP-Adresse der POP-Servers
- **C_POPConnectSERVERPORT** Server-Port: = 110, 995
- **C_POPConnectSSL** TRUE/FALSE wenn POP mit SSL-Verschlüsselung
- **C_POPConnectUser** POP-Postfach
- **C_POPConnectPwd** POP-Kennwort

/Datei:<CSV-Datei> **(optional)**
Name der **CSV-Datei** in der das Ergebnis gespeichert wird. Wird kein Name angegeben wird das Ergebnis in der Datei „**POPInbox.csv**“ gespeichert.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die **CSV-Datei** an die angegebene **E-Mail-Adresse**. Mehrere Empfänger müssen mit einem Semikolon getrennt werden.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

> **Hinweis:**
> Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListPOPConnect /Datei:POPConnect.csv /Header /Email:Demo@ERPAdmin.de
Liest das **POPConnect**-Postfach aus und speichert das Ergebnis in der Datei **POPConnect.csv** mit **Header**. Anschließend wird die Datei an Demo@ERPAdmin.de gesendet.

84 Der Befehl „-ListPOPControl“ PRO

ERPAdmin -ListPOPControl /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Liest den Inhalt des **POPControl-Postfaches** aus und speichert das Ergebnis in einer **CSV-Datei**. In der **VAR.INI** müssen dazu die folgenden Variablen mit gültigen Zugangsdaten gefüllt sein:

- **C_POPSERVER** Name oder IP-Adresse der POP-Servers
- **C_POPSERVERPORT** Server-Port: = 110, 995
- **C_POPSSL** TRUE/FALSE wenn POP mit SSL-Verschlüsselung
- **C_POPUser** POP-Postfach
- **C_POPPwd** POP-Kennwort

/Datei:<CSV-Datei> **(optional)**
Name der **CSV-Datei** in der das Ergebnis gespeichert wird. Wird kein Name angegeben wird das Ergebnis in der Datei „**POPInbox.csv**“ gespeichert.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die **CSV-Datei** an die angegebene **E-Mail-Adresse**. Mehrere Empfänger müssen mit einem Semikolon getrennt werden.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

> **Hinweis:**
> Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListPOPControl /Datei:POPControl.csv /Header /Email:Demo@ERPAdmin.de
Liest das **POPControl**-Postfach aus und speichert das Ergebnis in der Datei **POPControl.csv** mit **Header**. Anschließend wird die Datei an Demo@ERPAdmin.de gesendet.

85 Der Befehl „-ListPrinter“ BAS

ERPAdmin -ListPrinter /Datei:<Text-Datei> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

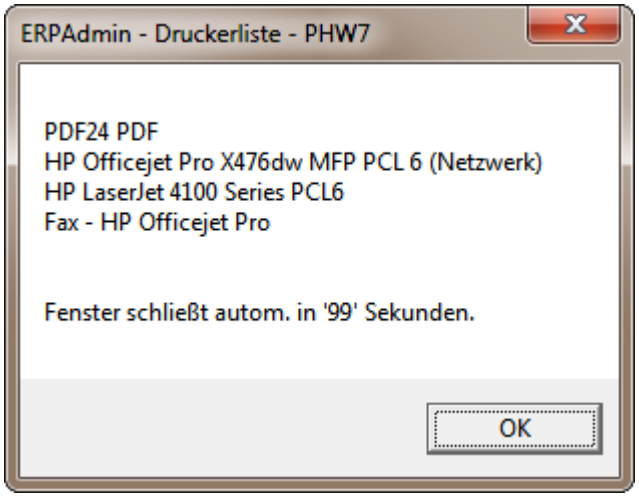
ERPAdmin gibt eine Liste aller auf dem Rechner vorhandenen Drucker aus und speichert diese in der **Texst-Datei**.

/Datei:<Text-Datei> **(optional)**
Speichert eine Liste aller auf dem System befindlichen Druckernamen im Textformat unter den angegebenen Dateinamen ab. Wird kein Dateiname angegeben wird in dem aktuellen **ERPAdmin** -Verzeichnis die Datei **Drucker_<Rechnername>.txt** erzeugt.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen. Wird KEINE E-Mail-Adresse übergeben erscheint eine Meldung am Bildschirm.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den **Dateinamen** einfügt.

Wurde keine E-Mail-Adresse übergeben wird die Liste der verfügbaren Drucker am Bildschirm ausgegeben.



[Auflistung der vorhandenen Drucker am PC \\PHW7]

/EmailCC:<E-Mail-Adresse>
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse>
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListPrinter /Datei:Verfuegbare_Drucker.txt
Erzeugt eine Liste aller verfügbaren Drucken des aktuellen PC und gibt diese am Bildschirm aus. Das Ergebnnis wird zusätzlich in der Datei „**Verfuegbare_Drucker.txt**“ gespeichert.

ERPAdmin -ListPrinter /Datei:Verfuegbare_Drucker.txt /Email:Demo@ERPAdmin.de
Erzeugt eine Liste aller verfügbaren Drucken des aktuellen PC und speichert diese in der Datei „**Verfuegbare_Drucker.txt**. Anschließend wird die Datei an Demo@ERPAdmin.de versendet.

Tipp:
Dieser Befehl steht **auch ohne** eine **ERPAdmin** -Lizenz zur Verfügung und kann **ohne VAR.INI** aufgerufen werden. Es erscheint dann der Anmeldedialog zu **büro+** um die Benutzerdaten abzufragen.

86 Der Befehl „-ListReguläreausdrücke“ BAS

ERPAdmin -ListReguläreausdrücke /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin Erzeugt eine CSV-Datei mit Daten aus dem Bereich **Parameter->Sonstige->Reguläre Ausdrücke** der **büro+** und speichert diese in einer CSV-Datei ab.

Benu.	Bezeichnung
✓	Adressnummer 5-Stellig numerisch mit "1" beginnend
✓	Adressnummer 5-Stellig numerisch mit "7" beginnend
✓	Belegnummer 6-Stellig numerisch mit "6" beginnend
✓	Belegnummer 6-Stellig numerisch mit Einleitung "Rxxxx"
✓	Belegnummer 10-Stellig mit "RE1" beginnend und 7-Stellig numerisch
✓	Belegnummer 10-Stellig mit "RE2" beginnend und 7-Stellig numerisch

/Datei:<CSV-Datei> **(optional)**
Hier kann ein beliebiger Dateiname übergeben werden unter dem **ERPAdmin** die CSV-Datei abspeichert. Wird kein Dateiname übergeben, wird das Ergebnis unter „**Liste-Reguläre-Ausdrücke.csv**“ gespeichert.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit Überschriften.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

87 Der Befehl „-ListRules“ BAS

ERPAdmin -ListRules /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /CharSet:<Zeichensatz>

ERPAdmin Erzeugt eine CSV-Datei mit allen Regeln der **büro+** und speichert diese in einer CSV-Datei ab.

/Datei:<CSV-Datei> **(optional)**
Hier kann ein beliebiger Dateiname übergeben werden unter dem **ERPAdmin** die CSV-Datei abspeichert. Wird kein Dateiname übergeben, wird das Ergebnis unter „**Regeln.csv**“ gespeichert.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit Überschriften.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

> **Hinweis:**
> Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

/CharSet:<Zeichensatz> **(optional)**
Soll das Ergebnis nicht im **ANSI** Format gespeichert werden kann auf **UNICODE** umgeschaltet werden. Dies ist dann hilfreich, wenn beim Schreiben der CSV-Datei ein Fehler auftritt weil UNICODE-Zeichen in den Regeln enthalten sind. Der Parameter heißt dann: **/ChartSet:UNICODE**

Beispiele:

ERPAdmin -ListRules
Erzeugt die CSV-Datei „**Regeln.csv**“ ohne Kopfzeile mit allen Regeln der **büro+**.

ERPAdmin -ListRules /Datei:ERP-Regeln.csv /Header

Erzeugt eine CSV-Datei mit Kopfzeile und speichert alle Regeln aus **büro+** in der Datei „**ERP_Regeln.csv**“ ab.

> **Tipp:**
> Dieser Befehl steht **auch ohne** eine **ERPAdmin** -Lizenz zur Verfügung und kann **ohne VAR.INI** aufgerufen werden. Es erscheint dann der Anmeldedialog zu **büro+** um die Benutzerdaten abzufragen.

88 Der Befehl „-ListSelektionsFelder“ BAS

ERPAdmin -ListSelektionsFelder <Tabelle> /Datei:<CSV-Datei> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin erzeugt eine „**CSV-Datei**“ mit allen Selektionen und Selektionsgruppen aus dem Bereich **Tabelle**. Neben den eigentlichen Selektionsfeldern werden auch alle Selektions-Gruppen-Bezeichnungen sowie die Feldarten und deren Vorbelegungen (Soweit vorhanden) mit ausgegeben. Damit haben Sie eine komfortable Möglichkeit, eine Dokumentation für **büro+** zu erstellen.

<Tabelle> **(optional)**
Tabelle aus **büro+** deren Selektionsfelder und Selektionsgruppen in eine CSV-Datei exportiert werden sollen. Wird keine Tabelle angegeben, wird für alle **büro+-**Tabellen jeweils eine CSV-Datei **Tabelle**.csv erzeugt sofern eine Selektion oder eine Selektionsgruppe vorhanden ist.

/Datei:<CSV-Datei> **(optional)**
Übergibt einen Dateinamen für die CSV-Datei in der das Ergebnis gespeichert werden soll. Wird kein Dateiname angegeben wird das Ergebnis unter **Tabelle.csv** abgespeichert.

/Email:<E-Mail-Adresse> **(optional)**
Mit diesem Parameter kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei gesendet wird. Wurden keine Daten gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

> **Hinweis:**
> Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiele:

ERPAdmin -ListSelektionsFelder Artikel /Datei:c:\ERPAdmin\Artikel-Selektionen.csv
Erzeugt die CSV-Datei „c:\ERPAdmin\Artikel-Selektionen.csv“. Diese könnte z.B. wie folgt aussehen:

Name	Bez	BenutztKz	RecArt	Art	Size	EKz	CKz	VgbBez
SelGrp_16	ERPAdmin	1	fraGruppe	fdaUnicodeS	2	ekzAlles	ekzCBNone	VgbValueKz=1
Sel67	Export für ERPAdmin-Mand	1	fraFeld	fdaBoolean	2	ekzBool	ekzCBNone	VgbValueKz=1
SelGrp_15	VariableStücklisten	1	fraGruppe	fdaUnicodeS	2	ekzAlles	ekzCBNone	VgbValueKz=1
Sel62	Höhe berechnern	1	fraFeld	fdaBoolean	2	ekzBool	ekzCBNone	VgbValueKz=1
Sel63	Breite berechnern	1	fraFeld	fdaBoolean	2	ekzBool	ekzCBNone	VgbValueKz=1
Sel64	Tiefe berechnern	1	fraFeld	fdaBoolean	2	ekzBool	ekzCBNone	VgbValueKz=1
Sel65	Anzahl berechnern	1	fraFeld	fdaBoolean	2	ekzBool	ekzCBNone	VgbValueKz=1
SelGrp_6	Wein	1	fraGruppe	fdaUnicodeS	2	ekzAlles	ekzCBNone	
Sel11	Land	1	fraFeld	fdaUnicodeS	25	ekzAlles	ekzCBAlles	VgbBez=, Argentinien, Australien, Chile, Deutschland, Frankreich, Italien, Portugal, Spanien,
Sel20	Rebsorte(n)	1	fraFeld	fdaUnicodeS	100	ekzAlles	ekzCBAllesLi	VgbBez=, Blauer Portugieser, Cabernet Sauvignon, Chardonnay, Gewürztraminer, Grauburgu
Sel27	Weingut	1	fraFeld	fdaUnicodeS	25	ekzAlles	ekzCBNone	
Sel21	Alkoholgehalt (%)	1	fraFeld	fdaDouble	8	ekzProzSz	ekzCBNone	
Sel22	Restsüße (g/liter)	1	fraFeld	fdaDouble	8	ekzMenge	ekzCBNone	
Sel23	Säuregehalt (g/liter)	1	fraFeld	fdaDouble	8	ekzMenge	ekzCBNone	
Sel24	Verschlusstyp	1	fraFeld	fdaUnicodeS	25	ekzAlles	ekzCBAllesLi	VgbBez=, Kork, Kunststoff, Glas, Schraubverschluss
SelGrp_5	Schuh-Verkauf	1	fraGruppe	fdaUnicodeS	3	ekzAlles	ekzCBNone	
Sel16	Größe (DE)	1	fraFeld	fdaUnicodeS	25	ekzAlles	ekzCBAlles	VgbBez=, 36, 37, 37.5, 38, 38.5, 39, 39.5, 40, 41, 42, 43, 44, 45, 46, 47, 38
Sel17	Grüße (US)	1	fraFeld	fdaUnicodeS	25	ekzAlles	ekzCBAlles	VgbBez=, 4.5, 5, 6, 6.5, 7, 7.5, 8, 8.5, 9, 9.5, 10, 10.5, 11, 11.5, 12, 13, 14, 15
Sel18	Größe (JN)	1	fraFeld	fdaUnicodeS	25	ekzAlles	ekzCBAlles	VgbBez=, 22, 22.5, 23, 23.5, 24, 24.5, 25, 25.5, 26, 26.5, 27, 27.5, 28, 28.5, 29, 29.5, 30, 31, 32, 33

Tipp:
Dieser Befehl steht **auch ohne** eine **ERPAdmin** -Lizenz zur Verfügung und kann **ohne VAR.INI** aufgerufen werden. Es erscheint dann der Anmeldedialog zu **büro+** um die Benutzerdaten abzufragen.

90 Der Befehl „-ListTextbausteine“ BAS

ERPAdmin -ListTextbausteine /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin Erzeugt eine CSV-Datei mit Daten aus dem Bereich **Parameter->Sonstige->Textbausteine** der **büro+** und speichert diese in einer CSV-Datei ab.

Kz.	Kurzbezeichnung	Beschreibung	Text	Memo
	EinleitungAN	Einleitung Angebot		
	VOB	Angebot: Hinweis auf VOB		
	SteuerhinweisPrivat	Rechnungen: Hinweis für Privatpersonen		
	Bauleistung	Rechnungen: Hinweis bei Bauleistungen		
	SteuerhinweisAusland	Rechnungen: Hinweis bei Rechnungen für Ausländer (Drittländer)		
	SteuerHinweisEG	Rechnungen: Hinweis für EG-Ausländer im UStID-Nr		

/Datei:<CSV-Datei> **(optional)**
Hier kann ein beliebiger Dateiname übergeben werden unter dem **ERPAdmin** die CSV-Datei abspeichert. Wird kein Dateiname übergeben, wird das Ergebnis unter „**Liste-Textbausteine.csv**“ gespeichert.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit Überschriften.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

91 Der Befehl „-ListUmsatzsteuer“ BAS

ERPAdmin -ListUmsatzsteuer /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin Erzeugt eine CSV-Datei mit Daten aus dem Bereich **Parameter->Sonstige->Umsatzsteuer** der **büro+** und speichert diese in einer CSV-Datei ab.

Std.	Schl.-Nr.	Bezeichnung	Suffix	DATEV	Satz (%)	Info	Steuerumstellung
	2	Mehrwertsteuer 7%	M7	2	7,0		bis 31.12.2020 Satz 5,0 %
✓	3	Mehrwertsteuer 19%	M19	3	19,0		bis 31.12.2020 Satz 16,0 %
	8	Vorsteuer 7%	V7	8	7,0		bis 31.12.2020 Satz 5,0 %
	9	Vorsteuer 19%	V19	9	19,0		bis 31.12.2020 Satz 16,0 %
	10	Umsatzsteuerfrei (Verkauf)	USF	10			
	11	EU Umsatzsteuerfrei mit ID-Nr.	EFI	11			

/Datei:<CSV-Datei> **(optional)**
Hier kann ein beliebiger Dateiname übergeben werden unter dem **ERPAdmin** die CSV-Datei abspeichert. Wird kein Dateiname übergeben, wird das Ergebnis unter „**Liste-Umsatzsteuer.csv**“ gespeichert.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit Überschriften.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

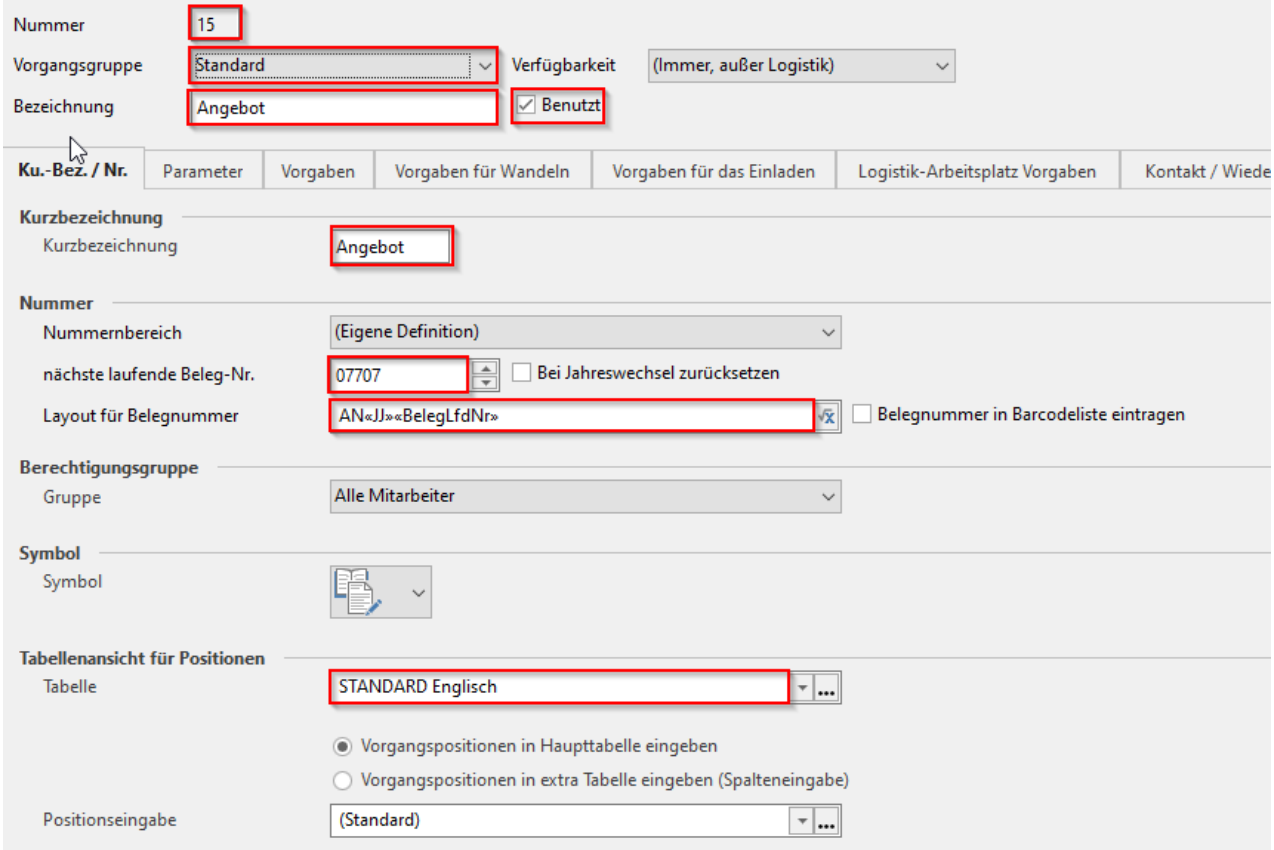
/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

92 Der Befehl „-ListVorgangArten“ BAS

ERPAdmin -ListVorgangArten /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin Erzeugt eine CSV-Datei mit Daten aus dem Bereich **Parameter->Vorgänge und Zwischenbelege->Arten** der **büro+** und speichert diese in einer CSV-Datei ab.



/Datei:<CSV-Datei> **(optional)**
Hier kann ein beliebiger Dateiname übergeben werden unter dem **ERPAdmin** die CSV-Datei abspeichert. Wird kein Dateiname übergeben, wird das Ergebnis unter „**Liste-VorgangArten.csv**“ gespeichert.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit Überschriften.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

93 Der Befehl „-ListWechselkurse“ BAS

ERPAdmin -ListWechselkurse /Datei:<CSV-Datei> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

ERPAdmin gibt eine Liste aller Wechselkurse der **büro+** aus und speichert diese in einer **CSV-Datei** ab. Wird kein **Dateiname** angegeben wird in dem aktuellen **ERPAdmin**-Verzeichnis die Datei **Liste-Wechselkurse.csv** erzeugt.

/Datei:<CSV-Datei>
Speichert eine Liste aller vorhandenen **büro+** Filter im CSV-Format unter **CSV-Datei** ab.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

/Header **(optional)**
Erzeugt eine Kopfzeile in der CSV-Datei.

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei an die angegebene E-Mail-Adresse. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den CSV-Dateinamen einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Hinweis:
Der E-Mail-Versand ist nur möglich, wenn eine **VAR.INI** existiert mit den SMTP-Versand-Variablen.

Beispiel:

ERPAdmin -ListWechselkurse /Datei:Liste_Wechselkurse.csv /Header /Email:Demo@ERPAdmin.de "/Subject:Liste mit büro+ Wechselkursen"
Erzeugt eine CSV-Datei mit allen vorhandenen **büro+** Wechselkurse und speichert diese in der CSV-Datei „**Liste_Wechselkurse.csv**“ ab. Anschließend wird die Datei an die E-Mail-Adresse **Demo@ERPAdmin.de** gesendet mit dem Betreff „**Liste mit büro+ Wechselkursen“**.

94 Der Befehl „-ListeDrucken“ PRO

ERPAdmin -ListeDrucken <Datenbank> <LayoutGruppe> <LayoutName> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /Drucker:<Druckername> /Datei:<Dateiname>

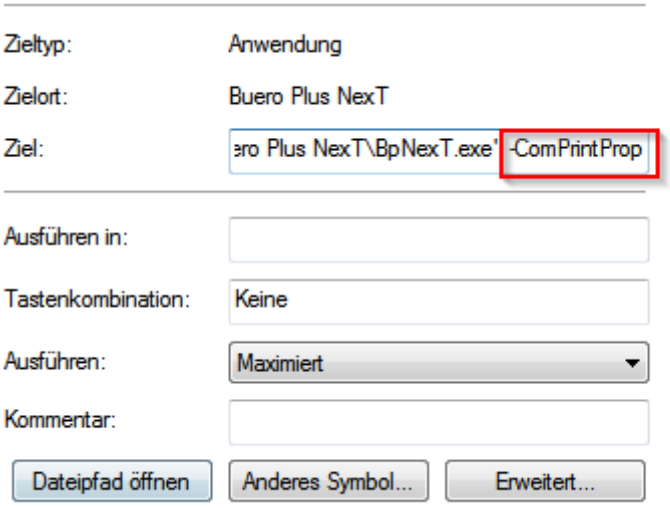
Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befahl haben Sie die Möglichkeit Listen aus **büro+** auszudrucken oder als PDF- oder Excel-Datei zu speichern. Dies ist besonders dann interessant, wenn das Erzeugen der Listen sehr lange dauert oder diese Listen regelmäßig benötigt werden.

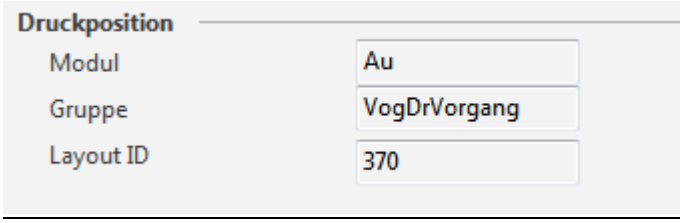
> **Hinweis:**
> Die Ausgabe von Drucken oder E-Mails aus dem **büro+** Programmbereich ist nicht möglich.

<Datenbank>
Geben Sie an dieser Stelle die Datenbank von **büro+** an, aus dem Bereich Sie die Liste drucken wollen.
Beispiel für Datenbanken: Artikel, Adressen, Warengruppen, Vorgang oder VorgangArchiv

<LayoutGruppe>
Hier muss das **Modul**- und die **Gruppe** aus **büro+** angegeben werden. Damit diese Werte angezeigt werden, muss **büro+** mit dem Parameter **„-ComPrintProp**“ (ohne die Anführungszeichen) gestartet werden.



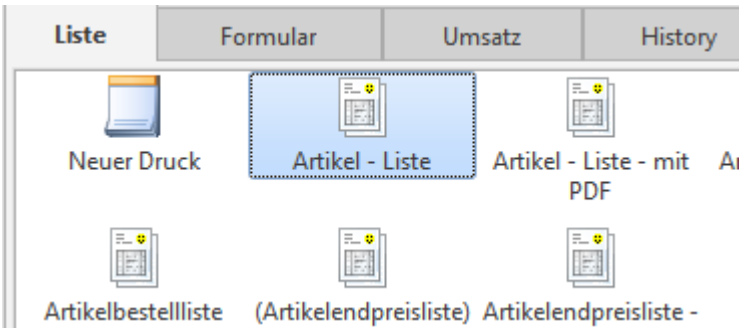
Die Angaben stehen in den Eigenschaften jedes Drucklayouts auf dem Register „**Parameter**“ in der Gruppe „**Druckoptionen**“.



[Beispiel für ein Drucklayout aus den Vorgängen]

Die **Layoutgruppe** setzt sich also aus dem Modul „**Au**“ und der Gruppe „**VogDrVorgang**“ zusammen. In obigem Beispiel würde die **Layoutgruppe** dann **AuVogDrVorgang** heißen.

<LayoutName>
Name eines Drucklayouts aus **büro+** z.B. „Artikel - Liste“. Bitte darauf achten, dass wenn Leerzeichen im **LayoutNamen** enthalten sind der ganze Ausdruck in Doppelte Anführungszeichen gesetzt werden muss.



[Liste mit Namen der Layouts aus dem Bereich Stammdaten->Artikel]

In obigem Beispiel würde der Parameter **"Artikel - Liste"** heißen.

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen, nach denen auch sortiert werden kann.

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

> **Hinweis:**
> Bei „**Auswertungen**“ z.B. Chefauswertung oder History-Auswertungen kann kein Bereich zur Einschränkung übergeben werden, da die Einschränkung von **büro+** selber durchgeführt werden.
>
> Bei der Verwendung von **Artikel-Katalogen** muss in den Auswertungen in der Artikel-Datenbank der „**0 Hauptkatalog**“ ausgewählt sein.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option wird eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **Dateiname** mit den Daten sendet. Wurde aufgrund des gewählten Dateiformates mehrere Dateien erzeugt, wird nur die erste Datei versendet. Wurde kein **Dateiname** übergeben jedoch im Drucklayout die Ausgabe in eine Datei hinterlegt, wird diese Datei an **E-Mail-Adresse** gesendet. Wurden keine Daten gefunden um eine Liste zu erzeugen wird keine E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/Drucker:<Druckername> **(optional)**
Für die Ausgabe des Drucklayout **LayoutName** kann ein abweichender Drucker verwendet werden. Es muss der Druckername angegeben werden unter dem der Drucker in der Systemsteuerung angezeigt wird.

Eine Liste der verfügbaren Drucker kann mit dem Befehl **ERPAdmin -ListPrinter** erzeugt werden.

Die Option **/Drucker=** darf nicht zusammen mit der Option /**Datei=** verwendet werden.

/Datei:<Dateiname> **(optional)**
ERPAdmin leitet die Druckausgabe in die angegebene Datei um, auch wenn im Drucklayout keine Dateiumleitung hinterlegt wurde. Anhand der Datei-Endung erkennt **ERPAdmin** um welches Datei-Format es sich handelt. Folgende Dateiendungen werden unterstützt:

- .LL List & Label
- .XLS Excel Format
- .PDF Adobe Portable Document Format
- .RTF Rich Text Format
- . HTML Hypertext Markup Language
- .MHTML Mimetype Hypertext Markup Language
- .XML Extendable Markup Language
- .JPG JPG Bildformat (kann mehrere Dateien erstellen
- .BMP BMP Bildformat (kann mehrere Dateien erstellen
- .EMF EMF Bildformat (kann mehrere Dateien erstellen
- .TIF Multipage TIFF Bildformat

- .TXT Text basierter Export
- .XLSX Microsoft Excel im neuen Format
- .DOC Microsoft Word
- .PPT Microsoft PowerPoint

Hinweis:
Bitte beachten Sie, dass je nach Ausbaustufe und Programmversion der **büro+** die zur Verfügung stehenden Formate unterschiedlich sein können. Zusätzlich können **büro+** Benutzern die Rechte fehlen um ein vorhandenes Format zu benutzen.

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien zur Historisierung in das angegebene History-Verzeichnis kopiert.

Im Dateinamen könne zwei Platzhalter verwendet werden: **@VON@** und **@BIS@** welche mit den übergebenen Werten aus **/Von:** bzw. **/Bis:** ersetzt werden. Wird einer der Platzhalter **@VON@** oder **@BIS@** verwendet muss der Daten-Bereich (Sortierung) der Liste auch mit **/Von:** und **/Bis:** eingeschränkt werden! In den übergebenen Werten **/Von:** und **/Bis:** dürfen sich keine Semikolons befinden.

Achtung:
Die Option **/Datei:** darf nicht zusammen mit der Option /**Drucker:** verwendet werden.

Beispiele:

ERPAdmin -ListeDrucken Artikel AuArtDrListe "Artikel - Liste"
Druckt eine Artikelliste auf dem Drucker aus.

ERPAdmin -ListeDrucken Vorgang AuVogDrListe Vorgangsliste
Gibt die Liste **Vorgangsliste** aus.

ERPAdmin -ListeDrucken Vorgang AuVogDrListe Vorgangsliste c:\ERPAdmin\Vorgang.xls
Erzeugt eine Exceldatei (.xls) mit dem Inhalt der Liste **Vorgangsliste** und speichert diese unter **c:\ERPAdmin\Vorgang.xls**

ERPAdmin -ListeDrucken Vorgang AuVogDrListe Vorgangsliste c:\ERPAdmin\Vorgang.xls
Erzeugt eine Exceldatei (.xls) mit dem Inhalt der Liste **Vorgangsliste** und speichert diese unter **c:\ERPAdmin\Vorgang.xls**

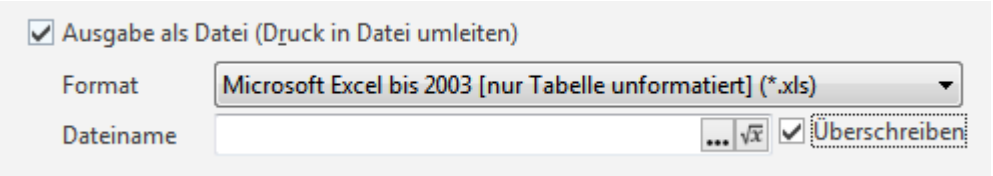
ERPAdmin -ListeDrucken Vorgang AuVogDrListe Vorgangsliste /Von:BelegNr=AN190000 /Bis:BelegNr=AN199999 c:\ERPAdmin\Vorgang_@VON@-@BIS@.xls
Erzeugt eine Exceldatei (.xls) mit dem Inhalt der Liste **Vorgangsliste** mit den Vorgängen AN190000 bis An199999 und speichert diese unter **c:\ERPAdmin\Vorgang_AN190000-AN199999.xls**

ERPAdmin -ListeDrucken Vorgang AuVogDrListe Vorgangsliste "/Drucker:HP OfficeJet x476"
Druckt die Liste **Vorgangsliste** auf dem Drucker **HP OfficeJet x476** aus. Nach der Ausgabe der der Drucker **HP OfficeJet x476** im Drucklayout als Standarddrucker hinterlegt.

ERPAdmin -ListeDrucken Vorgang AuVogDrListe Vorgangsliste /Datei:Rechnungen.xlsx /Email:Demo@ERPAdmin.de "/Subject:Vorgangs-Liste aller Rechnungen (@FILENAME@)"
Erzeugt die **Vorgangsliste** und speichert das Ergebnis unter **Rechnungen.xlsx** ab. Anschließend versendet **ERPAdmin** eine Email an Demo@ERPAdmin.de mit dem Betreff: „**Vorgangs-Liste aller Rechnungen (Rechnungen.xlsx)**“

Hinweis:
Achten Sie darauf, dass sich der Parameter **<LayoutGruppe>** aus den beiden Inhalten der Felder „**Modul**“ und „**Gruppe**“ zusammensetzt!

Tipp: Sie können einen Listendruck auch in eine im Layout in eine Datei umleiten und so direkt ein PDF oder eine Excel-Datei erzeugen. Aktivieren Sie dazu in den Eigenschaftes des Layouts im Register „**Druck**“ das Feld „**Ausgabe als Datei (Druck in Datei umleiten)**“ und hinterlegen Sie einen passenden Dateinamen.



Hinweis:
Bei der Ausgabe von Chefauswertungen muss als Tabelle „**Mandant**“ angegeben werden.

Bei „**Auswertungen**“ in denen bereits durch **büro+** Daten aggregiert werden z.B. Chefauswertung oder History-Auswertung, etc. sind keine Einschränkungen **/Von:**, **/Bis:** und **/Filter:** zulässig.

95 Der Befehl „-ListeDruckenDatei“ PRO

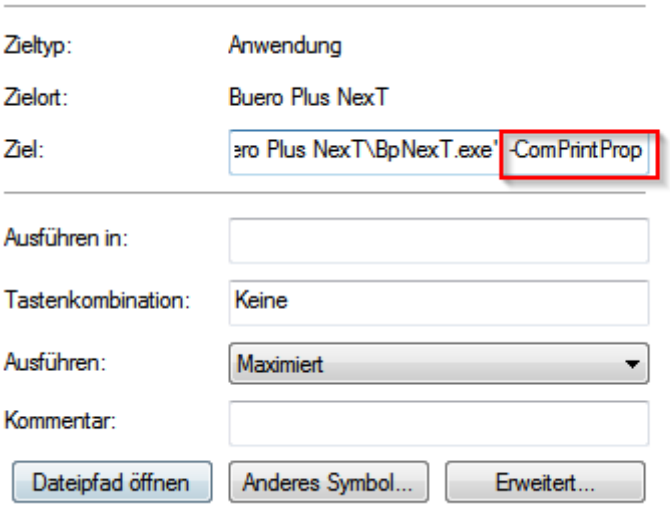
ERPAdmin -ListeDruckenDatei <Datenbank> <LayoutGruppe> <LayoutName> <büro+-Feldname> <CSV-Datei> <CSV-Spaltenbezeichner> /Filter:<Filterbedingung> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /Drucker: /Datei:<Dateiname>

Der Befehl **ERPAdmin -ListeDruck** wurde um die Funktionalität erweitert, dass mehrere Listen anhand einer CSV-Datei erzeugt werden und die Bereichs-Daten aus dieser CSV-Datei verwendet werden. Dies kann z.B. dazu verwendet werden, wenn für mehrere Kunden Stammblätter erzeugen möchte oder für Artikel Datenblätter als PDF erzeugt werden sollen.

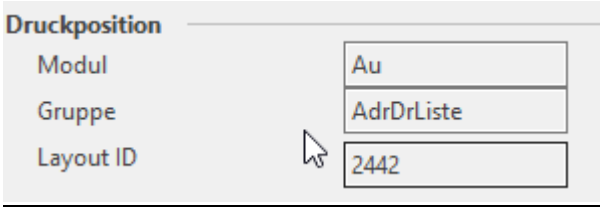
> **Hinweis:**
> Die Ausgabe von E-Mails-Layouts ist nicht möglich.

<Datenbank>
Geben Sie an dieser Stelle die Datenbank von **büro+** an, aus dem Bereich Sie die Liste drucken wollen. Beispiel für Datenbanken sind: **Artikel, Adressen, Warengruppen, Vorgang oder VorgangArchiv, etc.**

<LayoutGruppe>
Hier muss das Modul- und der Layout-Name aus **büro+** angegeben werden. Damit diese Werte angezeigt werden, muss **büro+** mit dem Parameter **„-ComPrintProp**“ (ohne die Anführungszeichen) gestartet werden.



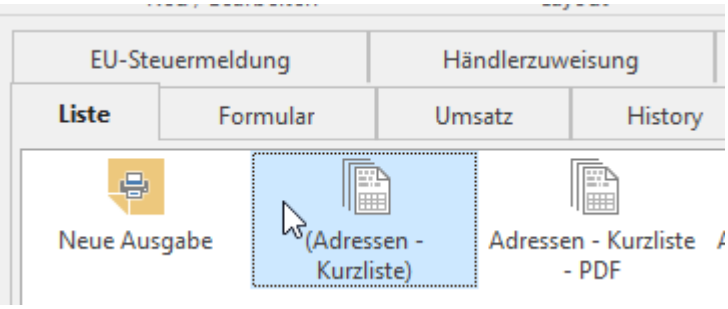
Die Angaben stehen in den Eigenschaften jedes Drucklayouts auf dem Register „**Parameter**“ in der Gruppe „**Druckoptionen**“.



[Beispiel für ein Drucklayout aus den Adressen]

Die **Layoutgruppe** setzt sich also aus dem Modul „**Au**“ und der Gruppe „**AdrDrListe**“ zusammen. In obigem Beispiel würde die Layoutgruppe dann **AuAdrDrListe** heißen.

<LayoutName>
Name eines Drucklayouts aus **büro+** z.B. „Artikel - Liste“. Bitte darauf achten, dass wenn Leerzeichen im **LayoutNamen** enthalten sind der ganze Ausdruck in Doppelte Anführungszeichen gesetzt werden muss.



[Liste mit Namen der Layouts aus dem Bereich Stammdaten->Adressen]

In obigem Beispiel würde das Layout **"(Adressen – Kurzliste)"** heißen.

<büro+-Feldname>
Um die Daten für die Druckausgabe einzuschränken muss hier ein gültiges Datenbank-Feld mit **Such-Index** angegeben werden. Im Bereich Adressen wäre das z.B. das Feld Adressnummer (**AdrNr**).

<CSV- Datei>
Gibt den Namen der ANSI-CSV-Datei mit Spaltenbezeichner an in der eine Auflistung der Adress-Nummer enthalten ist. Zu jedem dieser Werte soll dann die Liste **LayoutName** erzeugt werden.

Diese Datei könnte wie folgt aussehen:

	A	B	C	D
1	Adressnummer	Suchbegriff	Status	Inklusive
2	20000	Environ Architectural Design GmbH	Kunden	FALS
3	20001	Oshmans GmbH	Kunden	FALS
4	20002	Landskip Yard Service Gbr.	Kunden	FALS
5	20003	Coon Chicken Inn GmbH & Co KG	Kunden	FALS
6	20004	Lindsleys Lumber AG	Kunden	FALS
7	20005	Waves Music Ic.	Kunden	FALS
8	20006	Bold Ideas GmbH	Kunden	FALS
9	20007	Moplinks GmbH	Kunden	FALS

[ANSI-CSV-Datei mit Adress-Daten. Die Daten kommen aus der Spalte Adressnummer]

<CSV- Spaltenbezeichner >
Gibt den Spaltenbezeichner (Spalten-Überschrift) aus der CSV-Datei an in der sich die Daten für die Selektion befinden. In unserem Beispiel ist dies die erste Spalte **Adressnummer**.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **Dateiname** mit den Daten sendet. Wurde aufgrund des gewählten Dateiformates mehrere Dateien erzeugt, wird nur die erste Datei versendet. Wurde kein **Dateiname** übergeben jedoch im Drucklayout die Umleitung in eine Datei hinterlegt, wird diese Datei an **Email** gesendet. Wurden keine Daten gefunden um eine Liste zu erzeugen wird auch keine E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/Drucker: **(optional)**
Für die Ausgabe des Drucklayout **LayoutName** kann ein abweichender Drucker verwendet werden. Es muss der Druckername angegeben werden unter dem der Drucker in der Systemsteuerung angezeigt wird.

Eine Liste der verfügbaren Drucker kann mit dem Befehl **ERPAdmin -ListPrinter** erzeugt werden.

Die Option **/Drucker=** darf nicht zusammen mit der Option /**Datei=** verwendet werden.

/Datei:<Dateiname> **(optional)**
ERPAdmin leitet die Druckausgabe in die angegebene Datei um, auch wenn im Drucklayout keine Dateiumleitung hinterlegt wurde. Anhand der Datei-Endung erkennt **ERPAdmin** um welches Datei-Format es sich handelt. Folgende Dateiendungen werden unterstützt:

- .LL List & Label
- .XLS Excel Format
- .PDF Adobe Portable Document Format
- .RTF Rich Text Format
- . HTML Hypertext Markup Language

- .MHTML Mimetype Hypertext Markup Language
- .XML Extendable Markup Language
- .JPG JPG Bildformat (kann mehrere Dateien erstellen
- .BMP BMP Bildformat (kann mehrere Dateien erstellen
- .EMF EMF Bildformat (kann mehrere Dateien erstellen
- .TIF Multipage TIFF Bildformat
- .TXT Text basierter Export
- .XLSX Microsoft Excel im neuen Format
- .DOC Microsoft Word
- .PPT Microsoft PowerPoint

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien zur Historisierung in das angegebene History-Verzeichnis kopiert.

Enthält die Datei **CSV-Datei** mehr als einen Datensatz, werden mehrere Ergebnisdateien erzeugt. In diesem Fall muss im Dateinamen ein Platzhalter übergeben werden damit die verschiedenen Druckergenbisse in verschiedenen Dateien abgespeichert werden können. Dieser Platzhalter hat das Format **@büro+Feldbezeichner@** wobei **büro+Feldbezeichner** durch den zuvor übergeben Feldbezeichner zu ersetzten ist. In unserem Beispiel wäre das: **@ADRNR@**.

Beispiel:

ERPAdmin -ListeDruckenDatei Adressen AuAdrDrList "(Adressen – Kurzliste)" AdrNr Kundenliste.csv Adressnummer /Datei:Kundenstammblatt_@ADRNR@.pdf

Pro Datensatz/Adressnummer (Spalte mit der Bezeichnung **Adressnummer**) aus der Datei **Kundenliste.csv** wird je ein PDF der Liste **(Adressen – Kurzliste)** erzeugt und unter dem Dateinamen **Kundenstammblatt_@ADRNR@.pdf** gespeichert. In unserem Beispiel werden so die folgenden Dateien erzeugt:

Kundenstammblatt_20000.pdf
Kundenstammblatt_20001.pdf
Kundenstammblatt_20002.pdf
etc.

In gleicher Weise könne Sie auch für die Artikel oder Warengruppen Datenblätter automatisiert erzeugen.

Erzeugen Sie die Datei Kundenliste.csv durch einen Export der Adressen bei denen eine Selektion „Stammkunde“ gesetzt ist, können Sie selektiv für Ihre Kunden Stammblätter erzeugen.

Hinweis:
Bei der Verwendung von „**Katalogen**“ im Bereich **Stammdaten->Artikel** ist drauf zu achten, dass im Drucklayout immer „**(Alle Kataloge)**“ ausgewählt sind.

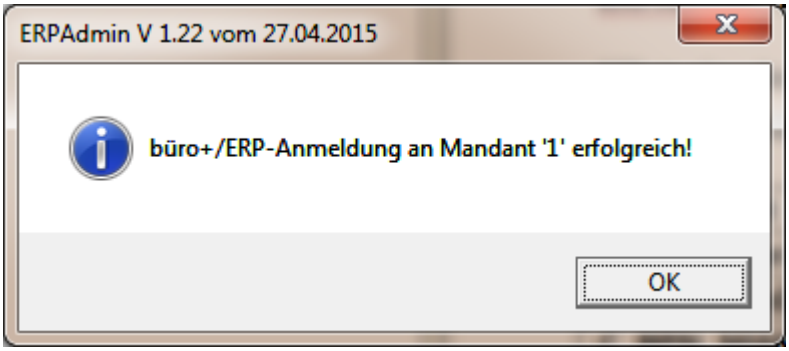
96 Der Befehl „-Login“ BAS

ERPAdmin -Login /MAND1 /MAND2 /MAND3 /MAND4 /MAND5
Mit diesem Befehl kann die Anmeldung an **büro+** getestet werden.

Zur Anmeldung werden die Benutzerdaten aus der **VAR.INI** verwendet.

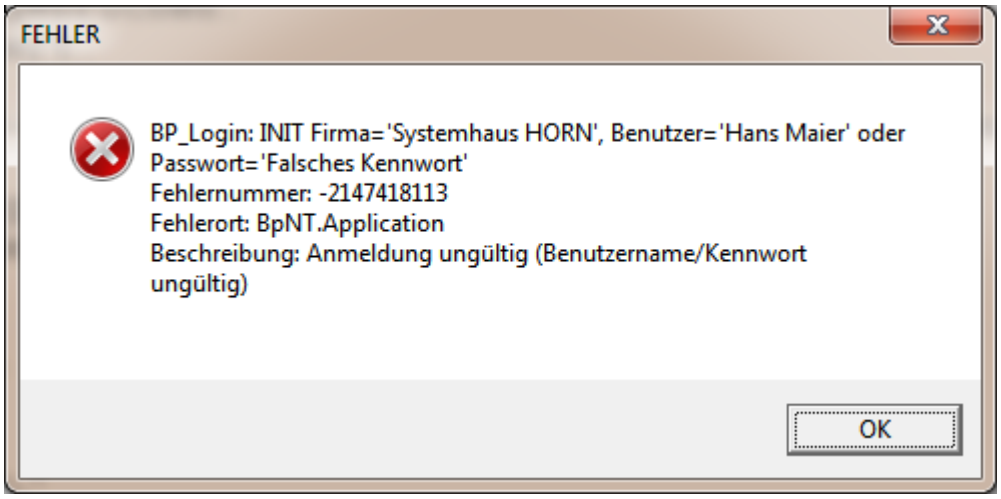
C_BPN_User	Benutzername für Anmeldung in Mandant 1
C_BPN_Pwd	Passwort für Benutzeranmeldung in Mandant 1
C_BPN_MANDNR	Mandanten-Nummer 1

Nach erfolgreicher Anmeldung erscheint folgende Meldung*:



[Die Anmeldung an **büro+** hat funktioniert.]

Trat bei der Anmeldung ein Fehler auf, weil z. B. ein falsches Kennwort verwendet wurde, erscheint folgende Meldung*:



[Fehlermeldung bei Anmeldung an **büro+**: Kennwort ungültig!]

*Beachten Sie dazu die Einstellungen des Befehls **/DEBUG.**

/MAND1, /MAND2, /MAND3, MAND4 PRO**, MAND5** PRO **(optional)**
Wird keine diese Optionen nicht angegeben wird automatisch der Wert **/MAND1** verwendet. Für die Anmeldung in Mandant 2, 3, 4 oder Mandant 5 können die Optionen **/MAND2**, **/MAND3**, **/MAND4** oder **/MAND5** verwendet werden.

ERPAdmin -Login /Mand2

ERPAdmin meldet sich in **büro+** an Mandant 2 an. Zur Anmeldung werden die Benutzerdaten für den 2. Mandanten aus der **VAR.INI** verwendet.

C_BPN_User2	Benutzername für Anmeldung für Mandant 2
C_BPN_Pwd2	Passwort für Benutzeranmeldung in Mandant 2
C_BPN_MANDNR2	Mandanten-Nummer 2

ERPAdmin -Login /Mand3

ERPAdmin meldet sich in **büro+** an Mandant 3 an. Zur Anmeldung werden die Benutzerdaten für den 3. Mandanten aus der **VAR.INI** verwendet.

C_BPN_User3	Benutzername für Anmeldung für Mandant 3
C_BPN_Pwd3	Passwort für Benutzeranmeldung in Mandant 3
C_BPN_MANDNR3	Mandanten-Nummer 3

ERPAdmin -Login /Mand4

ERPAdmin meldet sich in **büro+** an Mandant 4 an. Zur Anmeldung werden die Benutzerdaten für den 4. Mandanten aus der **VAR.INI** verwendet.

C_BPN_User4	Benutzername für Anmeldung für Mandant 4
C_BPN_Pwd4	Passwort für Benutzeranmeldung in Mandant 4
C_BPN_MANDNR4	Mandanten-Nummer 4

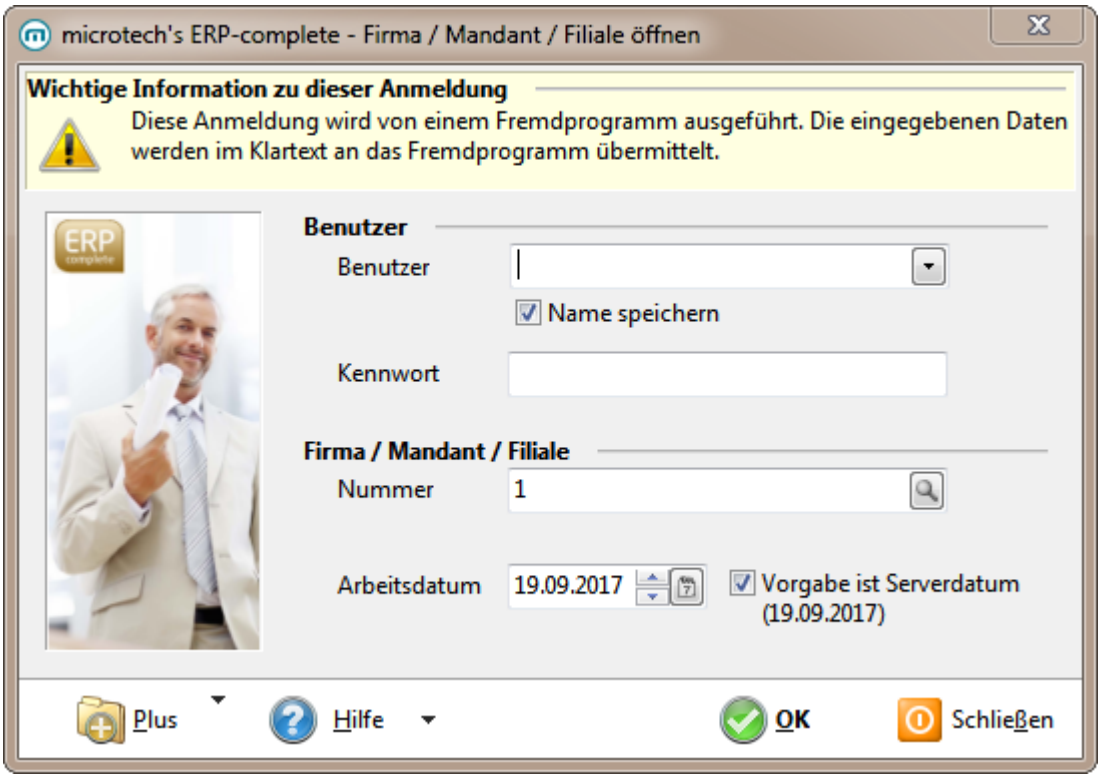
ERPAdmin -Login /Mand5

ERPAdmin meldet sich in **büro+** an Mandant 5 an. Zur Anmeldung werden die Benutzerdaten für den 5. Mandanten aus der **VAR.INI** verwendet.

C_BPN_User5	Benutzername für Anmeldung für Mandant 5
C_BPN_Pwd5	Passwort für Benutzeranmeldung in Mandant 5
C_BPN_MANDNR5	Mandanten-Nummer 5

Abfrage Login-Daten mit büro+-Anmelde-Dialog

Sind die Anmeldedaten in der **VAR.INI** noch nicht eingetragen (auskommentiert) oder mit einen Leerstring belegt öffnet sich der **büro+** Anmeldedialog und erfragt Benutzername, Kennwort und die Mandanten-Nummer vom Benutzer.



[Dialog zur Abfrage der Zugangsdaten]

Diese Daten werden nach erfolgreicher Anmeldung in die **VAR.INI** zurückgeschrieben. Das Kennwort wird dabei verschlüsselt abgespeichert.

Hinweis:
Es werden nur fünf Mandanten von **ERPAdmin** unterstützt. Angaben wie **/MAND6** etc. sind nicht zulässig und werden ignoriert! Benötigen Sie zu mehr als fünf Mandanten zeitgleich Zugang, verwenden Sie die Option **/VAR.INI:** um die vorhandenen Zugangsdaten zu überschreiben.

97 Der Befehl „-MassMail“ STD

ERPAdmin -MassMail <CSV-Datei> <E-Mail-Adresse> <Betreff> <E-Mail-Vorlage> <E-Mail Anhang> /NoFileNotExistError

Mit diesem Befehl können personalisierte Serien-E-Mails oder Status-Meldungen aus **büro+** versendet werden.

<CSV-Datei>
Mit diesem Parameter wird eine Datei im Format ANSI-CSV ausgewählt. Alle Spalten aus dieser Datei können im E-Mail-Betreff und im E-Mail-Template durch die Verwendung von Platzhaltern ausgegeben werden. Die Spaltenbezeichner müssen dazu nur mit „**@“** vor und hinter dem Spaltenbezeichner versehen werden z.B. **@Vog.BelegNr@**. Die erste Zeile der Datei muss die Spaltenbezeichnungen enthalten.

<E-Mail-Adresse>
Hier kann entweder eine einzelne E-Mail-Adresse übergeben werden oder die Spaltenüberschrift aus der Datendatei welche die E-Mail-Adressen der Empfänger enthält.

Wird eine E-Mail-Adresse anstelle einer Spaltenüberschrift übergeben, wird an die übergebene E-Mail eine Nachricht gesendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen. Im Template werden alle Dateien im Detail-Bereich der E-Mail gesammelt und in Form einer Liste zwischen den Platzhaltern **@DETAILS_BEGIN@** und **@DETAILS_END@ ausgegeben.**

Wird eine Spaltenüberschrift z. B. „**Email“** übergeben, müssen in dieser Spalte gültige E-Mail-Adressen enthalten sein. In diesem Fall wird pro Datensatz eine E-Mail erzeugt und versendet.

<Betreff>
Mit diesem Parameter wird der Betreff der E-Mail angegeben. Hier können auch Daten aus der Excel- oder CSV-Datei hinterlegt werden indem Platzhalten eingefügt werden (Siehe dazu nächsten Absatz).

<E-Mail-Vorlage>
Gib eine Vorlagendatei mit dem Inhalt der E-Mail-Nachricht an die versendet werden soll. Hat die Datei die Endung **.HTML** oder **.HTM** erfolgt der E-Mail-Versand automatisch im HTML-Format. Alle anderen Endungen werden als ANSI-Text interpretiert und die E-Mails dann als Text-Nachricht verwendet.

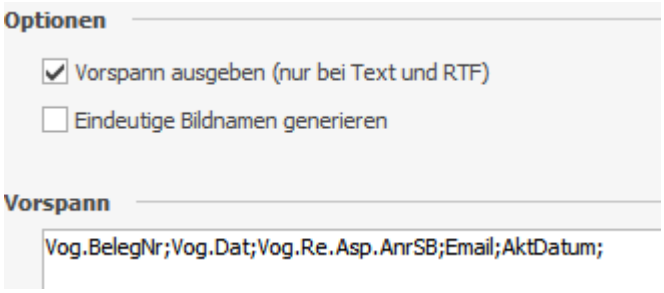
<E-Mail Anhang> **(optional)**
Datei inkl. Pfadangabe welche an die Emails angehängt werden sollen.

/NoFileNotExistError **(optional)**
Ist keine **Datendatei** zur Verarbeitung vorhanden und soll in diesem Fall keine Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** angegeben werden. Wurde keine Datei zur Verarbeitung gefunden bekommt der Benutzer eine E-Mail mit dem Hinweis „**Es wurden keine Daten gefunden.**“.

Beispiel 1:
Serien-E-Mail: Alle Kunden, die vor zehn Tagen etwas im Webshop gekauft haben also z.B. einen Lieferschein bekommen haben, sollen angeschrieben werden. Diese Kunden sollen eine Zufriedenheitsbewertung abgeben. Dazu wird eine E-Mail versendet mit persönlicher Anrede des Kunden, dem Kaufdatum und der Belegnummer. Mit der E-Mail sollen auch aktuelle Sonderangebot im Anhang als PDF mitgesendet werden.

Hierfür muss zunächst in **büro+** ein Lieferschein-Export erstellt werden für den Bereich „**Archiv Vorgänge**“ mit folgendem Aufbau:

«Vog.BelegNr»;«Vog.Dat»;«Vog.Re.Asp.AnrSB»;«Vog.Re.Asp.EMail1»;$GetAktDate();
Das Export-Layout erhält den Namen „**Export-Lieferscheine.BPL**“ und speichert das Ergebnis als CSV-Datei unter dem Pfad: „**c:\ERPAdmin\Lieferschein.csv**“. Bei CSV-Dateien muss im Export eine Kopfzeile hinzugefügt werden. Dies sieht dann wie folgt aus:



Das Export-Ergebnis könnte dann wie folgt aussehen:

	A	B	C	D	E
1	Vog.BelegN	Vog.Dat	Vog.Re.Asp.AnrSB	Email	AktDatum
2	LI1502235	12.01.2015	Sehr geehrter Herr Andreas,	[redacted]	18.08.2015
3	LI1502236	13.01.2015	Sehr geehrter Herr Abeska,	[redacted]	18.08.2015
4	LI1502237	18.02.2015	Sehr geehrter Herr Faber,	[redacted]	18.08.2015
5	LI1502238	24.02.2015	Sehr geehrter Herr Stuss,	[redacted]	18.08.2015
6	LI1502239	14.04.2015	Sehr geehrter Herr Kiel,	[redacted]	18.08.2015
7	LI1502244	31.03.2015	Sehr geehrte Frau Werwie,	[redacted]	18.08.2015
8	LI1502245	20.04.2015	Sehr geehrter Herr Bzyl,	[redacted]	18.08.2015
9	LI1502246	07.05.2015	Sehr geehrter Herr Bachmann,	[redacted]	18.08.2015
10	LI1502248	22.06.2015	Sehr geehrte Frau Nunes,	[redacted]	18.08.2015
11	LI1502249	18.08.2015	Sehr geehrte Frau Aitken,	[redacted]	18.08.2015

Weiterhin wird eine E-Mail-Vorlage benötigt, aus der die E-Mails erzeugt werden. Die Vorlage-Dateien beginnen mit **@DETAILS_BEGIN@** und enden mit **@DETAILS_END@**

Hier ein Beispiel für eine E-Mail-Vorlage im **Text-Format**.

@DETAILS_BEGIN@
@Vog.Re.Asp.AnrSB@
vielen Dank für Ihre Bestellung, welche wir am **@Vog.Dat@** mit der Lieferscheinnummer **@Vog.BelegNr@** an Sie versendet haben.
Wir würden uns freuen, wenn Sie unseren Service unter www.Systemhaus-HORN.de bewerten würden.
Sollten Sie Rückfragen zu Ihrer Bestellung haben, kontaktieren Sie bitte unseren Service unter 06074/881046.
Mit besten Grüßen
Ihr Team vom Systemhaus HORN
@DETAILS_END@

Die Vorlagendatei wird unter dem Namen „**c:\ERPAdmin\Vorlage.txt**“ gespeichert.

In der Vorlage befinden sich Platzhalter. Diese beginnen und enden mit einem Klammeraffen **(@).** Dazwischen befinden sich die Variablennamen, welche mit den Spaltenüberschriften unserer Export-Datei übereinstimmen müssen.

Nun wird **ERPAdmin** zweimal wie folgt aufgerufen:

ERPAdmin -export c:\ERPAdmin\Export-Lieferscheine.BPL c:\ERPAdmin\Lieferschein.csv /HIS

ERPAdmin -Massmail "c:\ERPAdmin\Lieferschein.csv" "E-Mail" "Bewerten Sie uns" " c:\ERPAdmin\Vorlage.txt" "c:\ERPAdmin\Angebot.pdf"
Der erste Aufruf erzeugt die Datei „**c:\ERPAdmin\Lieferschein.csv**“ zum Befüllen der E-Mails.
Der zweite Aufruf befüllt die E-Mail-Vorlage „**c:\ERPAdmin\Vorlage.txt**“ und versendet dann die E-Mails an die E-Mail-Adresse in der Spalte „**Email**“.

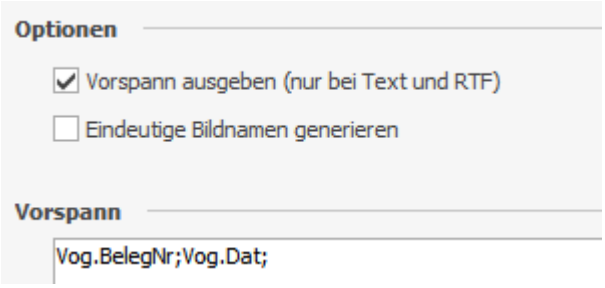
Weitere Einstellungsmöglichkeiten
Möchten Sie die E-Mail optisch ansprechender gestalten, erstellen Sie eine Vorlagendatei im HTML-Format mit der Endung „**.HTML**“. **ERPAdmin** erkennt an der Endung automatisch, dass es sich um eine HTML-Datei handelt und versendet dann die E-Mail auch im HTML-Format.

Beispiel 2:
Statusmeldung: Einmal am Tag soll Herr Bauer aus dem Einkauf darüber informiert werden, welche „**Bestellungen an Lieferant**“ überfällig sind. Hierzu soll eine E-Mail erzeugt werden, welche die Liste aller offenen überfälligen Bestellungen als Link enthält. Per Doppelklick soll Herr Bauer dann aus der E-Mail direkt in die entsprechende Bestellung von **büro+** springen können.

Dazu wird ein Export aller „**Bestellung an Lieferanten**“ erstellt, welche überfällig sind, z. B. mit folgendem Aufbau:

«Vog.BelegNr»;«Vog.Dat»;

Die benötigten Spaltenüberschriften werden wie folgt erzeugt:



Das Export-Ergebnis könnte z. B. so aussehen:

	A	B
1	Vog.BelegNr	Vog.Dat
2	BEL152604	10.08.2015
3	BEL152605	11.08.2015
4	BEL152606	14.08.2015

Im Anschluss wird eine E-Mail-Vorlage im HTML-Format erzeugt, welche eine Liste von Link-Aufrufen zu **büro+** enthält.

Hier ein Beispiel für diese E-Mail-Vorlage im HTML-Format:

Sehr geehrter Herr Bauer,

nachfolgende Lieferantenbestellungen sind überfällig:
@DETAILS_BEGIN@
<a
href='bpref://Vorgang:suchen@|Action=hacGoto,DbsKz=dbsVorgang,Index=BelegNr,Suche=**@Vog.BelegNr@**'> **@Vog.BelegNr @** vom **@Vog.Dat@**
@DETAILS_END@

Diese E-Mail wurde automatisch am **@DATETIME@** erzeugt.

Im Gegensatz zu „**Beispiel 1**“ befinden sich die Platzhalter „**@DETAILS_BEGIN@“** und „**@DETAILS_END@“** nicht mehr am Anfang und am Ende der Datei, sondern in der Mitte der E-Mail-Vorlage. Der Grund ist, dass nun alle Datensätze in Form einer Liste in dieser E-Mail ausgeben werden sollen.

Es können außerdem alle Platzhalter verwendet werden. Lesen Sie dazu das Kapitel „**Platzhalter und variable Parameter**“

An Stelle der Spaltenbezeichnung wird der Platzhalter **@Vog.BelegNr@** mit der Belegnummer des Vorgangs befüllt. Dadurch erkennt **ERPAdmin**, dass nur noch eine E-Mail mit dem gesamten Inhalt der Daten-Datei erstellt und versendet werden soll.

Die Aufrufe von **ERPAdmin** sehen dann z.B. so aus:

Für den Daten-Export:
ERPAdmin -export c:\ERPAdmin\Export-Bestellung-Lieferant.BPL c:\ERPAdmin\Bestellungen.csv

Für den E-Mail-Versand an Hr. Bauer mit dem Betreff „Überfällig Lieferantenbestellungen vom @DATE@“:
ERPAdmin -Massmail c:\ERPAdmin\Bestellungen.csv bauer@IhreDomaene.de "Überfällig Lieferantenbestellungen vom @DATE@" c:\ERPAdmin\Vorlage.txt

Die Ergebnis-E-Mail, die so erzeugt und versendet wird, sieht dann wie folgt aus:

Betreff: Überfällig Lieferantenbestellungen vom 18.08.2015"

Sehr geehrter Herr Bauer,

nachfolgende Lieferantenbestellungen sind überfällig:
BEL152604 vom 10.08.2015
BEL152605 vom 11.08.2015
BEL152606 vom 14.08.2015

Diese E-Mail wurde automatisch am 18.08.2015 19:02:09 erzeugt.

Läuft im Hintergrund **büro+** kann der Anwender durch Doppelklick auf die Links direkt in die Vorgangsübersicht springen und hat den entsprechenden Vorgang ausgewählt. Dazu muss in der aktuellen Vorgangsansicht das Suchfeld „**BelegNr**“ verfügbar sein.

98 Der Befehl „-MKDir“ PRO

ERPAdmin -MKDir <Spaltenname> </Datei:<CSV-Datei>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Erzeugt alle Verzeichnisse die in der Datei **CSV-Datei** in der Spalte **Spaltenname** angegeben sind.

<Spaltenname>
Spaltenbezeichnung aus der CSV-Datei.

<CSV-Datei>
Datei in ANSI-CSV-Format mit Spaltenbezeichnungen.

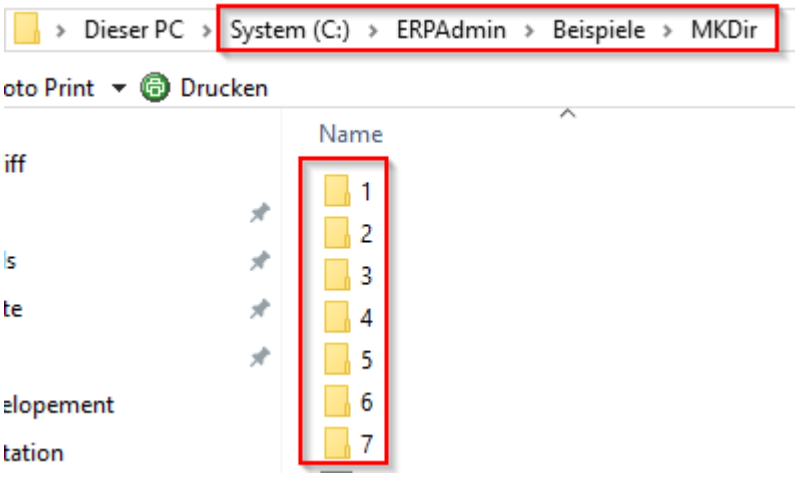
Beispiel:

Die CSV-Datei „**Verzeichnisliste.csv**“ hat folgenden Inhalt:

	A	B	C
1	Spalte1	Verzeichnis	Spalte3
2	Diese Spalte wird ignoriert	c:\ERPAdmin\Beispiele\MKDir\1\	Diese Spalte wird ignoriert
3	Diese Spalte wird ignoriert	c:\ERPAdmin\Beispiele\MKDir\2\	Diese Spalte wird ignoriert
4	Diese Spalte wird ignoriert	c:\ERPAdmin\Beispiele\MKDir\3\	Diese Spalte wird ignoriert
5	Diese Spalte wird ignoriert	c:\ERPAdmin\Beispiele\MKDir\4\	Diese Spalte wird ignoriert
6	Diese Spalte wird ignoriert	c:\ERPAdmin\Beispiele\MKDir\5\	Diese Spalte wird ignoriert
7	Diese Spalte wird ignoriert	c:\ERPAdmin\Beispiele\MKDir\6\	Diese Spalte wird ignoriert
8	Diese Spalte wird ignoriert	c:\ERPAdmin\Beispiele\MKDir\7\	Diese Spalte wird ignoriert

ERPAdmin -MKDir Verzeichnis c:\ERPAdmin\Beispiele\MKDir\Verzeichnisliste.csv
Erzeugt die Unterverzeichnisse aus der Spalte „**Verzeichnis**“ aus der CSV-Datei „**c:\ERPAdmin\Beispiele\MKDir\Verzeichnisliste.csv**“

Das Ergebnis aus Sicht des Windows-Explorers sieht wie folgt aus:



99 Der Befehl „-OPAusgleich“ PRO

ERPAdmin -OPAusgleich <AdrNrVon> <AdrNrBis> <Befehl> <CSV-Ergebnis-Datei> /Filter:<Filterbedingung> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

In den Offenen-Posten von für **büro+** werden sowohl Guthaben als auch Forderungen verwaltet. Besitzt ein Kunde in gleicher Höhe Forderungen wie auch Guthaben, können die Offenen-Posten gegeneinander ausgebucht werden. Dies kommt besonders dann vor, wenn ein Kunde Ware zurück sendet die er zuvor gekauft hat und um Erstattung bittet.

ERPAdmin kann dabei wie folgt unterstützen. Ergibt sich rechnerisch bei einer Adresse ein OP-Saldo von 0 kann **ERPAdmin** alle OP-Datensätze dieses Kunden ins OP-Archiv verschieben. Dies kann in zwei Schritten erfolgen:

1.) Prüfung der Offenen-Posten und Erzeugung einer **CSV-Ergebnis-Datei** mit allen Adressen und deren OP-Einträge die in der Summe 0 ergeben.
2.) Verschieben aller OP-Einträge von Adresse aus den Offenen-Posten in das OP-Archiv deren OP-Summe 0 ergibt.

<AdrNrVon>
Bereichsanfang der geprüft werden soll. Beispiel: 10000

<AdrNrBis>
Bereichsende der geprüft werden soll. Beispiel: 69999

<Befehl> **(optional)**
Mit dem Befehl „**CHECK**“ wird nur eine Prüfung durchgeführt. Das Ergebnis wird in einer CSV-Datei zurück geliefert. Erst nach Prüfung dieser Datei sollte der Befehl „**MOVE**“ ausgeführt werden. Wird kein Befehl angegeben wird immer ein **CHECK** durchgeführt.

<CSV-Ergebnis-Datei> **(optional)**
Dateiname (inkl. Pfadangabe) in der das Ergebnis zurückgegeben werden soll.
Beispiel: c:\ERPAdmin\OPLIsteDebitoren.csv

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **<CSV-Ergebnis-Datei>** mit den OP-Datensätzen sendet. Wurden kein Daten gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe

„**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Beispiele:

ERPAdmin -OPAusgleich 10000 69999
Führt eine Prüfung der Adressen im Bereich **10000-69999** durch und liefert alle mit einem 0-Saldo in der Datei **C:\ERPadmin\OffenePosten.csv** zurück.

ERPAdmin -OPAusgleich 10000 69999 CHECK C:\ERPadmin\OPListeDebitoren.csv
Führt eine Prüfung der Adressen im Bereich **10000-69999** durch und liefert alle mit einem 0-Saldo in der Datei **C:\ERPadmin\OPListeDebitoren.csv** zurück.

ERPAdmin -OPAusgleich 10000 69999 MOVE C:\ERPadmin\OPListeDebitoren.csv
Verschiebt alle OP-Datensätze der Adressen im Bereich **10000-69999** ins OP-Archiv, deren Adress-Saldo 0 ergibt. In der Datei **C:\ERPadmin\OPListeDebitoren.csv** wird eine Liste der Adressnummern und der OP-Einträge zurück geliefert.

Nachfolgend eine Beispieldatei die das Ergebnis eines Prüflaufes (**CHECK**) wiedergibt. Eine gleichartige Datei wird auch beim Aufruf mit dem Befehl „MOVE“ erzeugt.

	A	B	C
1	**AdrNr**	**Belegnummern**	**Status**
2	10014	RE1733420(107,10 S),RE1733420(107,10 H)	CHECK
3	10195	RE1733430(207,06 H),RE1733430(207,06 S)	CHECK
4	10586	RE1733409(4.891,85 H),RE1733409(4.891,85 S),RE1733429(160,65 H),RE1733429(160,65 S),RE1733458(582,62 H),RE1733458(582,62 S)	CHECK

100 Der Befehl „-OffenePostenInArchiv“ PRO

ERPAdmin -OffenePostenInArchiv /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin verschiebt die ausgewählten offene Posten (**/Von:, /Bis:, /Filter:**) in das Offene-Posten-Archiv.

/Von:<Bereich von> …. /Bis:<Bereich bis> **(optional)**
Über **/Von:** kann eine Einschränkung auf die Tabelle gelegt werden um einen einzelnen Wert zu selektieren wie z.B. **"/Von:AdrNr=10000"**. Soll ein ganzer Bereich selektiert werden muss auch ein Wert **/Bis:** übergeben werden z.B. **"/Bis:AdrNr=69999".**

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

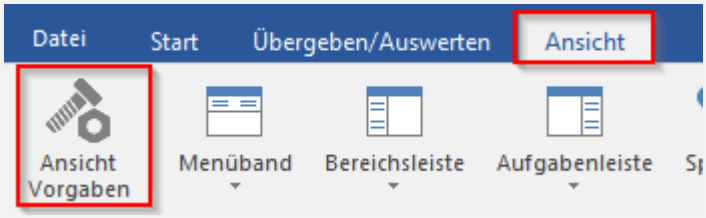
Beispiele:

ERPAdmin -OffenePostenInArchiv /Von:AdrNr=10000 /Bis:AdrNr=69999 "/Filter:ZahlBed='PayPal'"
Es werden alle Offenen-Posten der Adressen im Bereich **10000-69999** deren Zahlungsbedingung „**PayPal**“ ist in das Offene-Posten-Archiv verschoben.

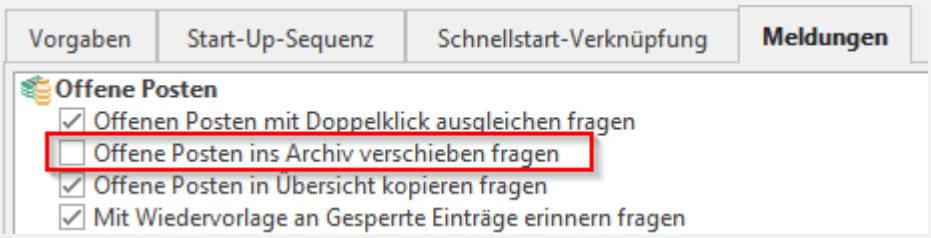
Hinweis:

Vor dem ersten Verschieben von Daten in einen Archiv- oder History-Bereich muss in **büro+** die entsprechende Sicherheitsmeldung für den COM-Benutzer deaktiviert werden.

Melden Sie sich dazu mit dem COM-Benutzer an und wechseln Sie in das Menü „**Ansicht**“ -> „**Ansicht Vorgaben**“.



Suchen Sie hier den Eintrag im Bereich „**Offene Posten**“-> „**Offene Posten ins Archiv verschieben fragen**“ und deaktivieren Sie diesen.



Erst jetzt kann das Verschieben über COM durchgeführt werden!

101 Der Befehl „-OffenePostenInArchivDatei“ PRO

ERPAdmin -OffenePostenInArchivDatei <CSV-Datei> <SpalteNr-BelegNr> <SpalteNr-Betrag> /NoFileNotExistError /Header /MoveToHis

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin verschiebt Offene Posten die mit den Angaben aus der Datei **CSV-Datei** übereinstimmen in das Offene-Posten-Archiv, wenn die Belegnummer und der Betrag 1:1 identisch sind. Skonto oder Preisnachlässen werden nicht berücksichtigt.

<CSV-Datei>
Dateiname inkl. Pfadangabe einer ANSI-CSV-Datei mit mind. zwei Spalten mit Angaben zur Belegnummer und zum Betrag. Die Position dieser Spalten ist beliebig. Es ist ein Wildcard (*) im Dateinamen erlaubt.

<SpalteNr-BelegNr>
Spalte der **CSV-Datei** in der sich Angaben zur Belegnummer befinden. Die Belegnummer muss genau so enthalten sein wie diese auch in den Offenen-Posten im Feld **OP.BelegNr** steht.

Alle Leerzeichen werden automatisch aus dem Feld entfernt.

<SpalteNr-Betrag>
Spalte der **CSV-Datei** in der sich Angaben zum Betrag befinden. Der betrag muss gleich dem Betrag im Offenen-Posten Feld **OP.OPBetBt** sein. Bei den Beträgen wird nicht zwischen SOLL oder HABEN unterschieden. Es wird nur der Betrag verglichen.

Alle Leerzeichen, „S“, „H“, „EUR“ und Punkt (.) werden automatisch entfernt.

/NoFileNotExistError **(optional)**
Ist keine Datei vorhanden und soll in diesem Fall **keine** Fehlermeldung erzeugt werden, kann diese Option verwendet werden.

/Header **(optional)**
Mit dieser Option wird die erste Zeile (Kopfzeile) ignoriert und nur die Datenzeilen der Datei bearbeitet. Datenzeilen sind dann alle Zeilen ab der Zeile 2 und Folgende.

/MoveToHis **(optional)**
Verschiebt die **CSV-Datei** nach erfolgreicher Verarbeitung in das History-Verzeichnis (**C_HistoryDir**).

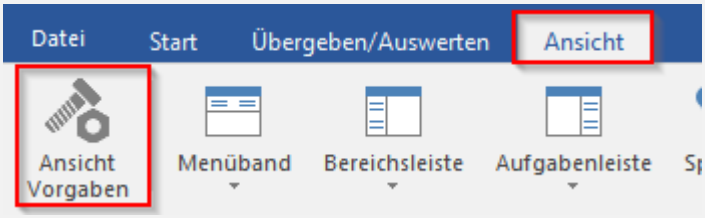
Beispiel:
ERPAdmin -OffenePostenInArchivDatei OPAusgleich.csv 3 7 /Header /MoveToHis

Verarbeitet den Inhalt der Datei **OPAusgleich.csv**. In der Spalte **3** befinden sich die Belegnummern, in der Spalte **7** die Beträge. Die erste Zeile der Datei wird ignoriert, weil es sich um eine Kopfzeile handelt (**/Header**). Ist bei der Verarbeitung kein Fehler aufgetreten wird die Datei im History-Verzeichnis (**C_HISTORYDIR**) archiviert.

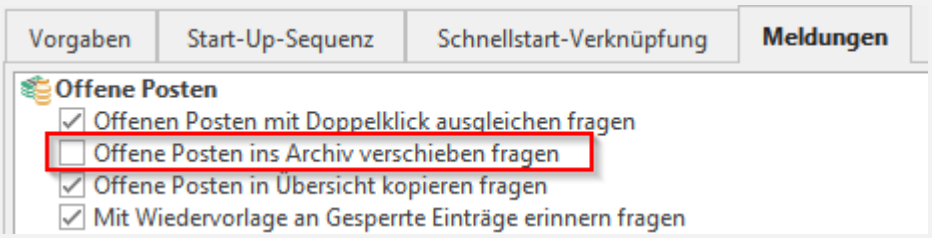
Hinweis:

Vor dem ersten Verschieben von Daten in einen Archiv- oder History-Bereich muss in **büro+** die entsprechende Sicherheitsmeldung für den COM-Benutzer deaktiviert werden.

Melden Sie sich dazu mit dem COM-Benutzer an und wechseln Sie in das Menü „**Ansicht**“ -> „**Ansicht Vorgaben**“.



Suchen Sie hier den Eintrag im Bereich „**Offene Posten**“-> „**Offene Posten ins Archiv verschieben fragen**“ und deaktivieren Sie diesen.



Erst jetzt kann das Verschieben über COM durchgeführt werden!

102 Der Befehl „-OpenURL“ STD

ERPAdmin -OpenURL <URL> /SILENTMODE

Mit diesem Befehl öffnet **ERPAdmin** eine Webseite im **Internet-Explorer**. Zusätzlich kann noch die Option /**SILENTMODE** übergeben werden, welcher das Öffnen des **Internet-Explorer** unterdrückt. Beim Aufruf über den Scheduler wird dieser Modus unbedingt benötigt!

Die Aufrufe von **ERPAdmin** sehen dann so aus:

Beispiel: **ERPAdmin -OpenURL http://www.Systemhaus-HORN.de**

Es wird die Webseite www.Systemhaus-HORN.de im Internet-Explorer geöffnet. Nachdem die Seite geladen wurde bleibt diese geöffnet stehen.

Beispiel:
ERPAdmin -OpenURLwww. http://Systemhaus-horn.de /SILENTMODE

Es wird die Webseite www.Systemhaus-HORN.de aufgerufen jedoch wird dazu keine Instanz des Internet-Explorers geöffnet.

Hinweis:
Es werden nur die Protokolle **http** oder **https** unterstützt. Die Übergabe eines Ports beim Aufruf ist nicht möglich.

104 Der Befehl „-OutlookImportKontakte“ PRO

ERPAdmin -OutlookImportKontakte <ExportLayout> <Outlook EntryID> <Outlook StoreID> /Datei:<CSV-Datei> /Path:<Bilderpfad> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /LtzAend /Delete /Email:<E-Mail-Adresse> /Subject:<Email-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin exportiert bestimmte Adressen oder Ansprechpartner aus **büro+** und importiert diese direkt in Outlook in den Default-Kontakt-Ordner. Bereits vorhandene Kontakte bleiben bestehen. Wurden Kontakte zu einem früheren Zeitpunkt mit **ERPAdmin** importiert werden diese Kontakte zuerst gelöscht und anschließend neu importiert. Gelöscht werden alle Kontakte die in dem Outlook-Feld **CustomerID** (deutsch **Kundennr**.) einen Wert stehen haben. Dieses Outlook-Feld darf also nicht anders verwendet werden.

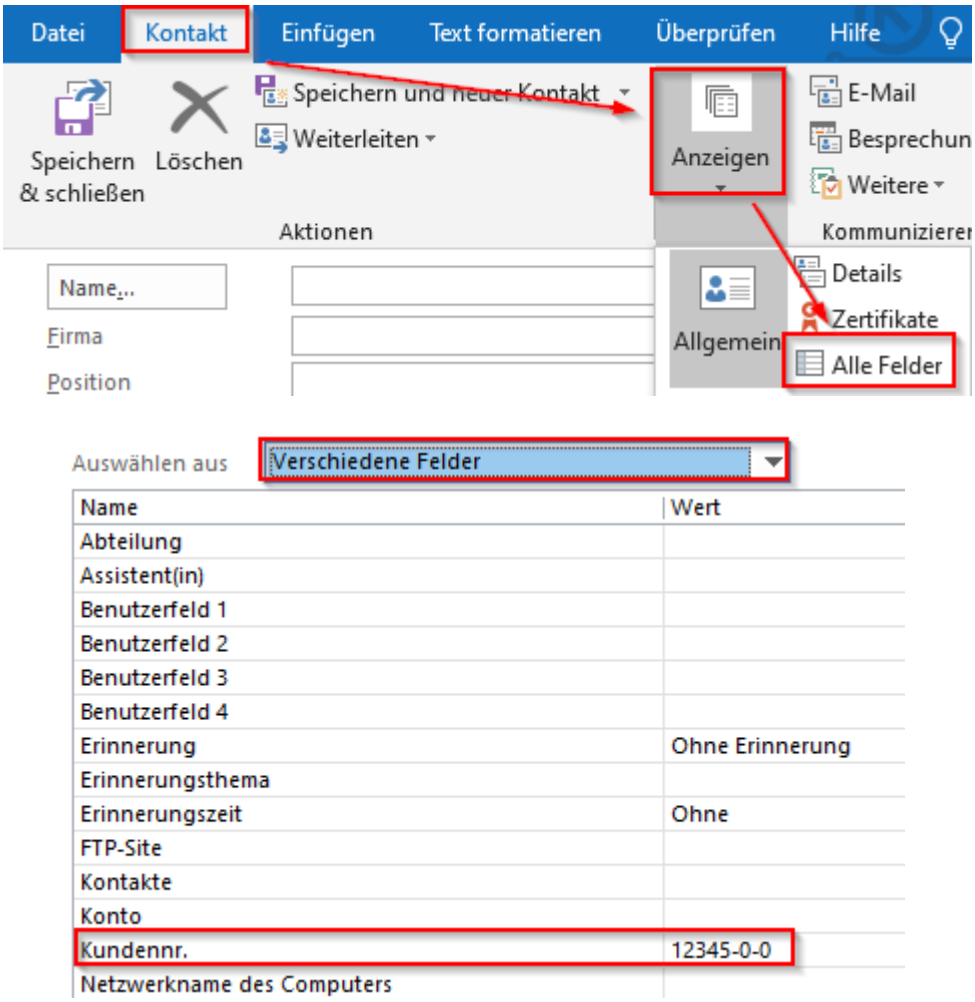
Nachfolgende ein Beispiel, wie ein Kontakt nach der Übergabe an Outlook aussehen kann:

<Export-Layout>
Für den Import der Kontakte können die Datenfelder frei gewählt werden. Die Zuordnung zu den Outlook-Feldern erfolgt anhand der Kopfzeile der CSV-Datei. Damit **ERPAdmin** einen Adress-Datensatz in Outlook zuordnen kann, muss das Outlook-Feld „**CustomerID**“ mit einer **eindeutigen Referenz** belegt werden.

Wenn der Export aus dem Bereich **Adressen** erfolgt kann als eindeutige Referenz z.B. die **Adressnummer** verwendet werden.

Erfolgt der Export aus dem Bereich **Ansprechpartner** kann eine Kombination aus **Adressnummer**, **Anschriftennummer** und **Ansprechpartnernummer** verwendet werden. Wichtig ist nur, dass die Referenz eindeutig ist.

Diese Zuordnung der Daten erfolgt in Outlook über das Feld **CustomerID** (deutsch **Kundennr**.):



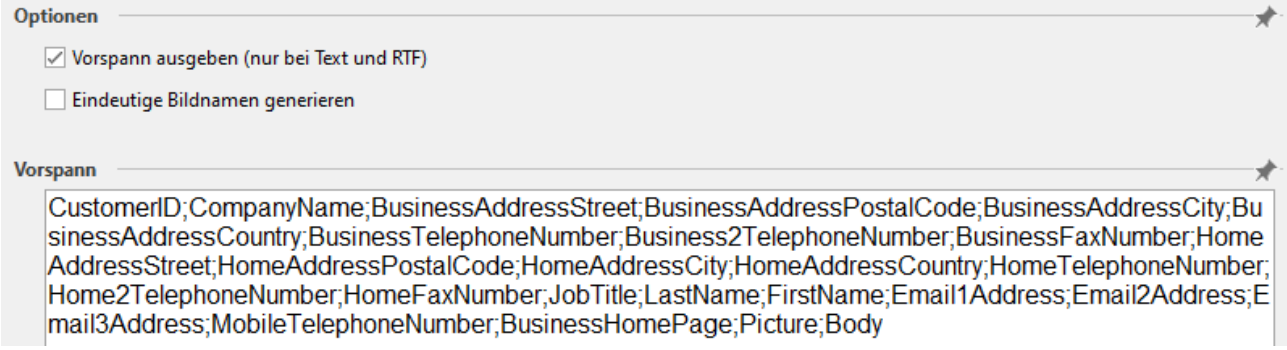
Nachfolgend die Feldliste von Outlook die von **ERPAdmin** unterstützt wird. Es kann frei gewählt werden welche Felder übergeben werden sollen. Die Reihenfolge ist dabei unwichtig.

Outlook Feldbezeichner	**büro+ Beispiel Zuordnung**
CustomerID*	**AdrNr-AnschNr-AspNr***
CompanyName	Re.Na2
BusinessAddressStreet	Re.Str
BusinessAddressPostalCode	Re.PLZ
BusinessAddressCity	Re.Ort
BusinessAddressCountry	Re.LandBez
BusinessTelephoneNumber	Re.Tel1
Business2TelephoneNumber	Re.Tel2
BusinessFaxNumber	Re.Fax
BusinessHomePage	Re.Wseite
HomeAddressStreet	Li.Str
HomeAddressPostalCode	Li.PLZ
HomeAddressCity	Li.Ort
HomeAddressCountry	Li.LandBez
HomeTelephoneNumber	Li.Tel1
Home2TelephoneNumber	Li.Tel2
HomeFaxNumber	Li.Fax

JobTitle	Re.Asp.Pos
LastName	Re.Asp.VNa
FirstName	Re.Asp.NNa
Email1Address	Re.Asp.Email1
Email2Address	Re.Asp.Email2
Email3Address	LI.Asp.Email1
MobileTelephoneNumber	Re.Asp.MTel
Picture	Re.Asp.Bild
Body	Re.Memo

* Pflichtfelder

Im Export-Layout müssen dann die ausgewählten Spaltenbezeichner eingetragen werden und die Option „**Vorspan ausgeben (nur bei Text und RTF)**“ muss aktiviert werden.



Die Spaltenüberschriften aus dem Vorspann steuern, welche Inhalte in Outlook in welche Felder geschrieben werden.

/Datei:<CSV-Datei> **(optional)**
Hier kann ein abweichender Dateiname für den CSV-Export übergeben werden. Wird kein Dateiname übergeben wird der Name aus dem Export verwendet. Achten Sie in diesem Fall auf die korrekte Angabe des Export-Verzeichnisses.

</Path:<Bilderpfad> **(optional)**
Sollen auch Bilder aus der Warenwirtschaft in die Outlook Kontakte übergeben werden, muss ein Verzeichnis angegeben, in dem die Bilder gespeichert werden. Im Export-Layout muss eine entsprechende Bildvariable mit angegeben und als Feld-Art „**JPG Bild**“ ausgewählt werden.

Zusätzlich muss im Export-Layout das Verzeichnis für die Ablage der Bilder angegeben werden.

Dieser Dateiname muss mit dem Parameter **/Path:** an **ERPAdmin** übergeben werden.

Beispiel: **/Path:C:\ERPAdmin\Beispiele\OutlookImportKontakte\Bilder**

<Outlook EntryID> und <Outlook StoreID> **(optional)**
Sollen die Daten nicht in den Default Kontakt-Ordner von Outlook importiert werden sondern in einen anderen Kontakt-Ordner, muss dieser Ordner als Kontakt-Ordner im Postfach angelegt werden und die **EntryID** und **StoreID** übergeben werden.

Die Werte für **EntryID** und **StoreID** können mit dem Befehl **ERPAdmin.exe -OutlookListFolders** ermittelt. Ergebnis ist eine CSV-Datei „Outlook-Folders.csv“ mit allen Ordnern des aktuellen Outlook-Postfachs. In den Spalten **EntryID** und **StoreID** stehen die entsprechenden Werte alle Ordner.

> **Hinweis**:
> Unterordner der **Outlook-Kontakte** können leider nicht verwendet werden.

/Von:<Bereich von> **(optional)**
Wie bei den Exporten aus **büro+** üblich kann dieser auf bestimmte Daten eingeschränkt werden. Hier wird der Anfangsbereich „**Sortierung von…**“ vorbelegt.

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches „**Sortierung bis…**“ vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

> **Hinweis:**
> Wird eine oder mehrere der Optionen **/Von: /Bis**: oder **/Filter:** verwendet, werden alle Sortierungen und Filter im Exportlayout überschrieben.

/Delete **(optional)**
Löscht die Export-Datei von **büro+** wenn der Import der Kontakte in Outlook erfolgreich war.

/Email:<E-Mail-Adresse> **(optional)**
Sendet das Export-Ergebnis an die angegebene E-Mail sofern der Export eine Ergebnisdatei erzeugt hat. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe

Kapitel „**Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/LtzAend **(optional)**
Führt einen Export unter Berücksichtigung des Feldes „**Letzte Datensatzänderung**“ durch. Diese Option muss zunächst in **DBManager** für die entsprechende Datenbank aktiviert werden und es muss eine entsprechende Sortierung vorhanden sein. Beim ersten Aufruf werden alle Datensätze ab dem **30.12.1899** aus der Datenbank exportiert. **ERPAdmin** speichert nach dem Export in der Datei **<Exportlayout>.INI** den aktuellen Timestamp und verwenden diesen beim nächsten Aufruf als Startdatum. Damit werden bei allen Folgeaufrufen nur noch die Datensätze exportiert, bei denen sich seit dem letzten Export eine Änderung ergeben hat oder die neu dazugekommen sind.

> **Hinweis**:
> Wenn Sie **ERPAdmin** nicht auf dem **BPServer** ausführen, müssen Sie dafür sorgen, dass die Systemzeiten beider Rechner synchron sind.

Beispiele

ERPAdmin.exe -OutlookImportKontakte Ansprechpartner-Export.BPL /Path:c:\ERPAdmin\Beispiele\OutlookImportKontakte\Bilder /LtzAend /Filter:Sel5=1
Der büro+-Export **Ansprechpartner-Export.BPL** für Ansprechpartner wird ausgeführt. Dabei werden nur Ansprechpartner exportiert bei denen die Selektion **SEL5=1** gesetzt ist. Durch die Option **/LtzAend** werden nur geänderten Datensätze seit dem letzten Aufruf exportiert. Vorhandene Bilder werden im Verzeichnis **c:\ERPAdmin\Beispiele\OutlookImportKontakte\Bilder** übergeben.

105 Der Befehl „-POP2CSV“ PRO

ERPAdmin -POP2CSV <Verzeichnis E-Mail Anhänge> <Verzeichnis E-Mail Ablage> /Datei:<Dateiname> /Header /Delete

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin liest alle vorhandenen E-Mails eines POP-Postfaches aus und speichert das Ergebnis in einer CSV-Datei. Enthält der E-Mail-Body Zeilenumbrüche, werden diese durch ein **#z** ersetzt.

Die Zugangsdaten für das POP-Postfach müssen in der **VAR.INI** hinterlegt werden.

C_POP2CSVSERVER	Adresse POP-Server/IP-Adresse
C_POP2CSVSERVERPORT	POP-Server-Port (995= SSL)
C_POP2CSVSSL	SSL/TLS-Verschlüsselung (TRUE/FALSE)
C_POP2CSVUSER	POP-Benutzer-Name
C_POP2CSVPWD	POP-Kennwort

<Verzeichnis E-Mail-Anhänge> **optional**
Enthaltene E-Mail Anhänge werden in diesem Verzeichnis abgelegt.

<Verzeichnis E-Mail Ablage> **optional**
In diesem Verzeichnis können die E-Mails im EML-Format historisiert werden.

/Datei:<Dateiname> **optional**
Name der CSV-Datei unter dem das Ergebnis gespeichert werden soll. Wurde kein Dateiname angegeben wird das Ergebnis unter **POP-Inhalte.csv** gespeichert.

/Delete **optional**
Mit dieser Option werden die E-Mails im POP-Postfach nach dem auslesen gelöscht

Beispiele

ERPAdmin -POP2CSV
Ruft alle E-Mails vom POP-Server ab und speichert das Ergebnis in der Datei **POP-Inhalte.csv**. Es werden keine Anhänge gespeichert und es findet auch keine Historisierung der E-Mails statt.

ERPAdmin -POP2CSV c:\ERPAdmin\Attachments\ c:\ERPAdmin\\Emails
Ruft alle E-Mails vom POP-Server ab und speichert das Ergebnis in der Datei **POP-Inhalte.csv**. Anhänge werden im Unterverzeichnis **c:\ERPAdmin\Attachments** gespeichert und die E-Mails werden als EML-Datei im Verzeichnis **c:\ERPAdmin\\Emails** Historisiert.

106 Der Befehl „-POPConnect“ PRO

ERPAdmin -POPConnect /Delete

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin holt E-Mails aus einem POP3-Postfach ab und legt diese in **büro+** in den Dokumenten einer Adresse ab. Die Zuordnung erfolgt anhand der E-Mail-Adressen die bei den Anschriften oder Ansprechpartner hinterlegt sind.

Die E-Mails werden im EML-Format abgespeichert und können mit Outlook oder einem anderen EML-fähigen Browser gelesen werden. Sollten Sie kein Outlook verwenden finden Sie im Internet kostenfreie Viewer zum Anzeigen von EML-Emails.

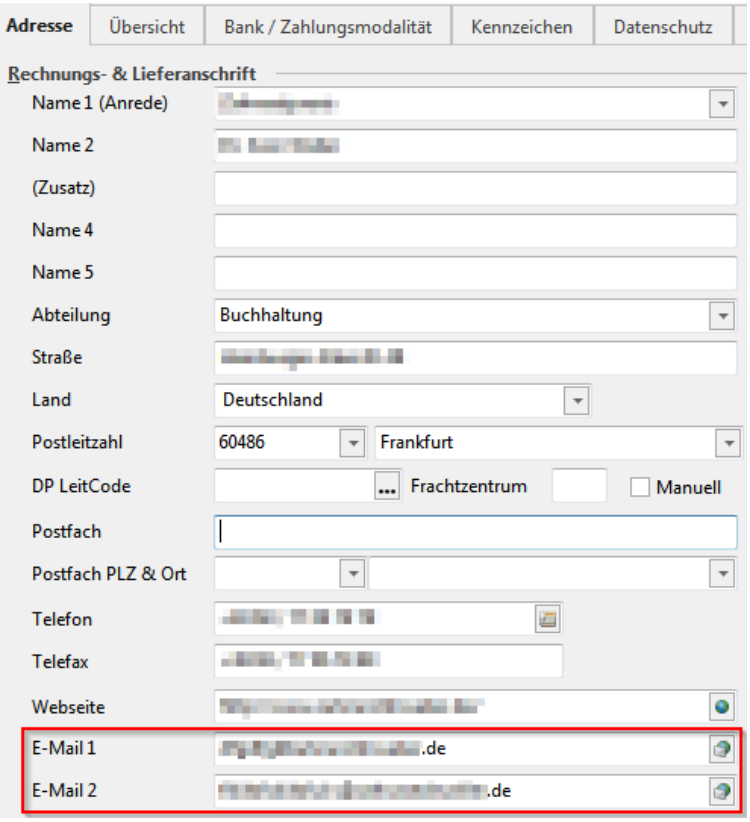
Konnte eine E-Mail-Adresse zugeordnet werden, wird die E-Mail als Dokument in **büro+** gespeichert und im Verzeichnis **C_HISTORDIR** abgelegt welches in der **VAR.INI** definiert sein muss.

/Delete (optional)
Wurde zusätzlich der Parameter **/Delete** übergeben wird die E-Mail nach erfolgreicher Verarbeitung anschließend aus dem Postfach gelöscht. Kann die E-Mail-Adresse nicht zugeordnet werden bleibt die E-Mail im Postfach liegen und muss ggf. vom Benutzer manuell gelöscht werden.

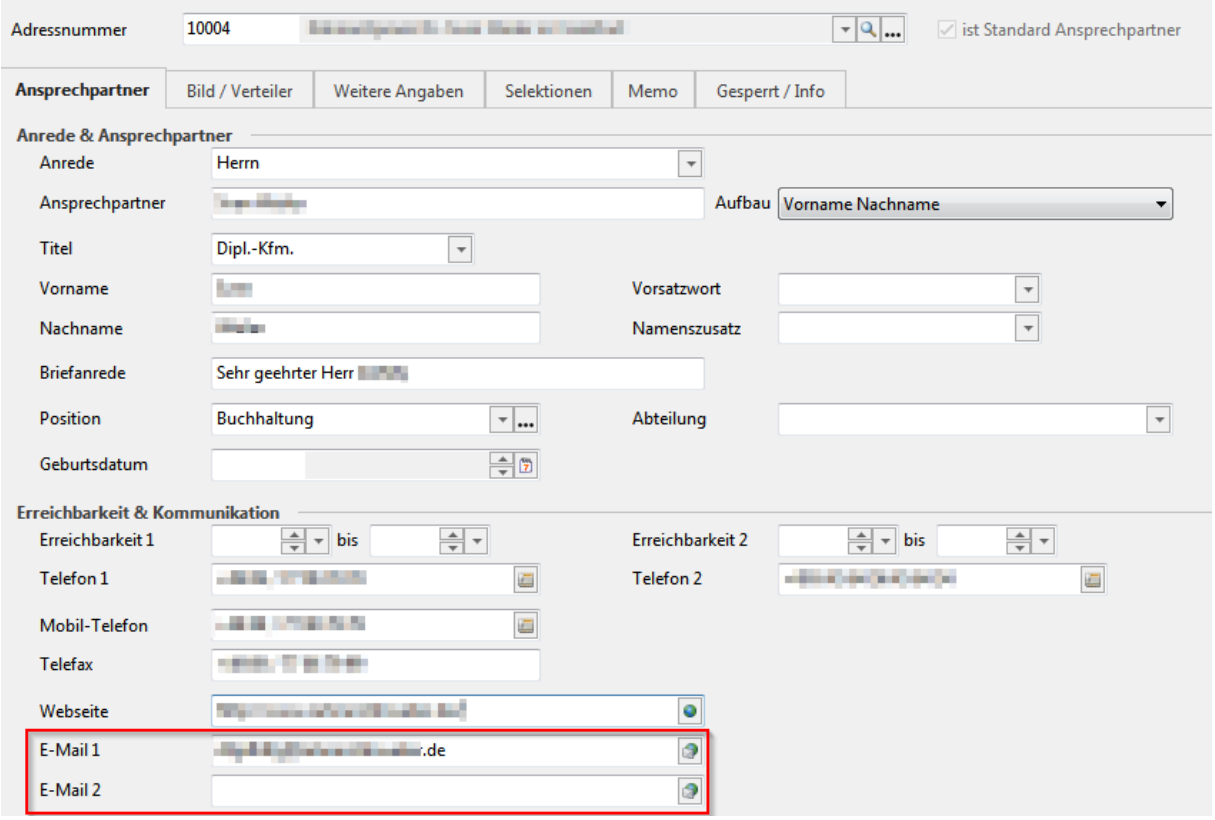
Tritt beim Speichern der E-Mail in **büro+** ein Fehler auf, wird die E-Mail im Verzeichnis **C_BADDIR** abgespeichert.

Der Benutzer bekommt in jedem Fall eine Rückmeldung per E-Mail über den Status der Verarbeitung.

Zuordnung über E-Mail-Felder in den Anschriften:



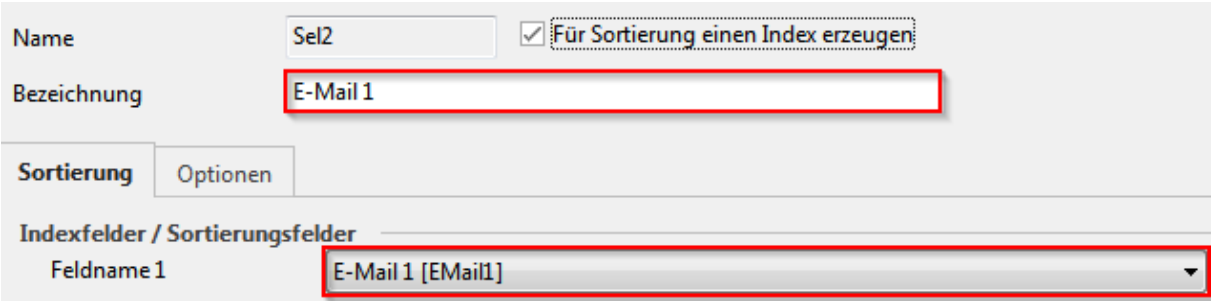
Zuordnung über E-Mail Felder in den Ansprechpartnern



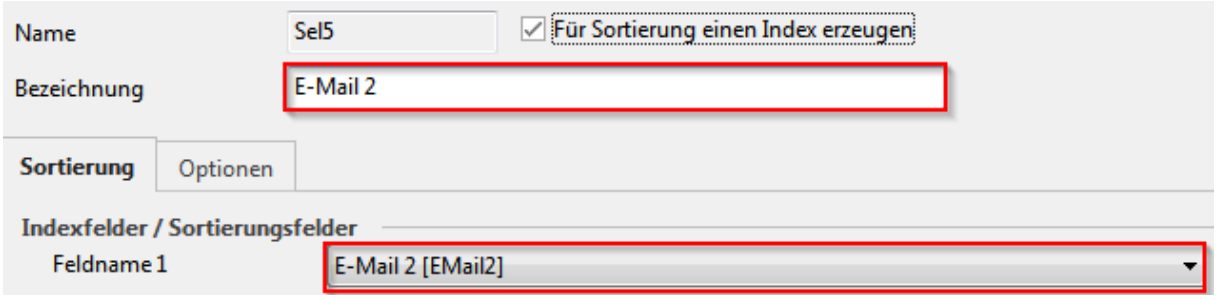
Voraussetzungen

Damit **ERPAdmin** nach den E-Mail-Adressen suchen kann, müssen diese im **DBManager** als Suchfeld angelegt werden. Dies muss für die **Anschriften** und für die **Ansprechpartner** erfolgen. Dabei ist es unerheblich welche Selektion Sie verwenden. Wichtig ist nur, dass die Felder **EMail1** und **Email2** nicht in einer weiteren Suche enthalten sind!

Erweiterung der Anschriften oder Ansprechpartner-Sortierungen um **E-Mail 1**



Erweiterung der Anschriften oder Ansprechpartner Sortierungen um **E-Mail 2**



In der **VAR.INI** sind folgende Einstellungen vorzunehmen:

C_POPConnectServer	IP-Adresse oder Name des POP3-Servers
C_POPConnectServerPort	Server-Port für Kommunikation (143, 993, 995, etc.)
C_POPConnectSSL	**TRUE**=SSL, **FALSE** sonst
C_POPConnectUser	Benutzername für das POP3-Postfach
C_POPConnectPwd	Kennwort für das POP3-Postfach
C_POPConnectAdrANSEmail1	**TRUE** wenn die Email1 der Anschrift beachtet werden soll. **FALSE** sonst
C_POPConnectAdrANSEmail2	**TRUE** wenn die Email2 der Anschrift beachtet werden soll. **FALSE** sonst
C_POPConnectAdrANSASPEmail1	**TRUE** wenn die Email1 des Ansprechpartners beachtet werden soll. **FALSE** sonst
C_POPConnectAdrANSASPEmail2	**TRUE** wenn die Email2 des Ansprechpartners beachtet werden soll. **FALSE** sonst
C_POPConnectAdrDokName	Schema des Dokumentennamens mit Platzhaltern
C_POPConnectAdrDokNameMaxLength	Max. länge der Dokumenten-Bezeichnung. Überzählige Zeichen werden abgeschnitten

C_POPConnectAdrANSEmail1, C_POPConnectAdrANSEmail2, C_POPConnectAdrANSASPEmail1, C_POPConnectAdrANSASPEmail2
Setzten Sie die Felder auf **TRUE** bei denen **ERPAdmin** den Inhalt auf Übereinstimmung prüfen soll. Die Überprüfung erfolgt in der oben angegebenen Reihenfolge (erst Anschriften, dann Ansprechpartner). Überprüft wird die Absende-Adresse einer E-Mail. Wird **ERPAdmin** in der ERP-Datenbank fündig, wird die E-Mail in der Dokumentendatenbank unter der Adress-Nummer eingefügt unter der die E-Mail-Adresse zuerst gefunden wurde.

C_POPConnectAdrDokName
Hier wird der Name definiert der in der Dokumentendatenbank eingefügt werden soll. Dazu stehen folgende Platzhalter zur Verfügung:

@DATE@	Datum der E-Mail

@TIME@	Uhrzeit der E-Mail
@YYYY@	Jahreszahl 4-Stellig: 2015
@YY@	Jahreszahl 2-Stellig: 15
@MM@	Monat 2-Stellig: 10 (für Oktober)
@DD@	Tag 2-Stellig: 13
@h@	Stunde 2-Stellig: 10
@m@	Minute 2-Stellig: 30
@s@	Stunde 2-Stellig: 00
@SUBJECT@	Betreff der E-Mail
@EMAILFROM@	Absende-Adresse der E-Mail

Hinweis:
Bitte beachten Sie die Groß-Klein-Schreibung bei den Platzhaltern

Beispiel
Sie möchten die E-Mails sortiert mit dem Betreff in den Dokumenten ablegen. Dann könnet das Schema wie folgt sein:
C_POPConnectAdrDokName = "@YYYY@.@MM@.@DD@ @h@:@m@:@s@ @SUBJECT@"

In **büro+** würde der Dokumenten-Name dann wie folgt erscheinen:
2017.07.27 13:53:17 AW: Bestellung BL1234567 vom 15.07.2017

C_POPConnectAdrDokNameMaxLength
Da viele E-Mails sehr lange Betreffzeilen haben, kann die max. länge des Dokumentennamen vorgegeben werden. Die Länge betrifft die Gesamtlänge des Dokumentennamens innerhalb von **büro+.** Zu lange Namen werden abgeschnitten.

Tipp:
Verwenden Sie für **ERPAdmin -POPConnect** ein eignes E-Mail-Postfach welches Sie von einem PC aus überwachen. So können Sie Fehlerhafte E-Mails die im Postfach liegen bleiben, weil z.B. keine passende Betreff-Zeile gefunden wurde schneller erkennen und ggf. löschen.

POP-Postfach Inhalt auslesen

Kommt es beim Abrufen des POP-Postfaches zu Fehlermeldungen, ist es hilfreich, wenn man sich den Inhalt des POP-Postfaches ansehen kann. Besteht kein direkter Zugriff auf das Postfach kann der Inhalt mit **ERPAdmin** wie folgt ausgelesen werden:

ERPAdmin -ListPOPConnect /Datei:POPConnect.csv /Header

Als Ergebnis wird eine CSV-Datei „**POPConnect.csv**“ erzeugt die den Inhalt des POP-Postfaches wiedergibt. Dabei bleiben die Daten auf dem POP-Server unverändert. Der Zugriff erfolgt nur lesend.

107 Der Befehl „-POPControl“ PRO

ERPAdmin -POPControl /NoOkEmail /Delete /Datei:<Dateiname> /MoveToBad /NoLockError

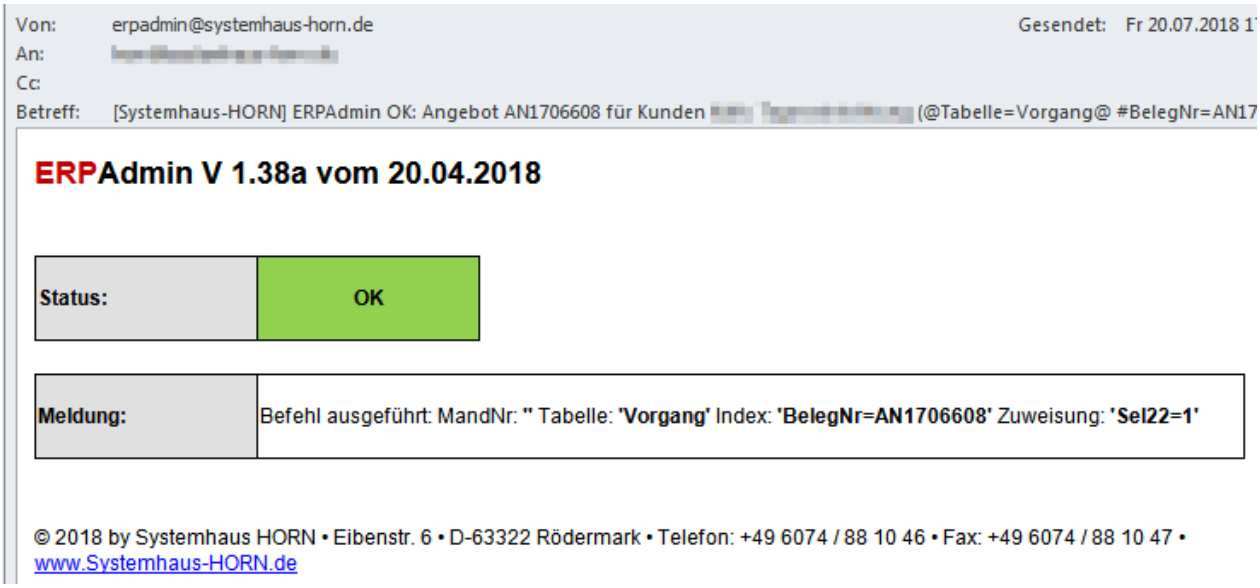
Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin holt E-Mails aus einem POP3-Postfach ab und führt anhand von Angaben in der Betreffzeile Feldzuweisungen in **büro+** durch. Damit können Sie Feldwerte innerhalb von **büro+** per E-Mail verändern. Anwendung findet dieser Befehl z.B.: bei der Freigabe von Vorgängen durch Außendienstmitarbeiter welche Angebote zur Freigabe per E-Mail zugesendet bekommt und nur auf Antworten Klicken müssen um diese damit freizugeben.

Alle erfolgreich abgearbeiteten Emails werden zur Historisierung im Verzeichnis **C_HISTORDIR** abgelegt welches in der **VAR.INI** ebenfalls definiert sein muss. Danach werden die E-Mails aus dem Postfach gelöscht.

Fehlerhafte E-Mails bleiben in dem Postfach stehen und werden beim nächsten Aufruf erneut von **ERPAdmin** verarbeitet. Soll dies nicht passieren, muss der Benutzer die fehlerhaften E-Mails aus dem Postfach löschen. Zusätzlich können diese E-Mails auch im Verzeichnis **C_BADDIR** historisiert werden mit dem Parameter **/MoveToBad**.

Standardmäßig wird der Absender per E-Mail darüber informiert ob seine Änderung in **büro+** erfolgreich durchgeführt wurde oder es einen Fehler gab. Die Rückmeldung im Erfolgsfall sieht dann in etwa wie folgt aus:


Von: erpadmin@systemhaus-horn.de Gesendet: Fr 20.07.2018 1
An:
Cc:
Betreff: [Systemhaus-HORN] ERPAdmin OK: Angebot AN1706608 für Kunden (@Tabelle=Vorgang@ #BelegNr=AN17

ERPAdmin V 1.38a vom 20.04.2018

Status:	**OK**

Meldung:	Befehl ausgeführt: MandNr: '' Tabelle: **'Vorgang'** Index: **'BelegNr=AN1706608'** Zuweisung: **'Sel22=1'**

© 2018 by Systemhaus HORN • Eibenstr. 6 • D-63322 Rödermark • Telefon: +49 6074 / 88 10 46 • Fax: +49 6074 / 88 10 47 • www.Systemhaus-HORN.de

Die E-Mail im Fehlerfall kann wie folgt aussehen:

ERPAdmin V 2.51a vom 08.05.2020

Status:	**FEHLER**

Meldung:	Der Datensatz in Tabelle **"Kalender"** ist durch den Benutzer **"Peter.Horn"** vom Arbeitsplatz **"PHW10"** gesperrt. Der Zugriff auf den Kalender-Datensatz **"Termin: Service am 06.12.2017 von 09:00 bis 10:00 Uhr"** ist zur Zeit nicht möglich. Datum: 20.05.2020 13:00:15 Betreff: WG: Termin bei Kunde Ralf Geisler(10239) - @Tabelle=Kalender@ #ID=15200# $ErledigtKz=1$!EmailBody=Info!

ERPAdmin © 2020 by Systemhaus HORN • Eibenstr. 6 • D-63322 Rödermark • Telefon: +49 6074 / 88 10 46 • Fax: +49 6074 / 88 10 47 • www.Systemhaus-HORN.de

/NoOkEMail **(optional)**
Soll der Absender nicht darüber informiert werden, dass seine E-Mail von **ERPAdmin** erfolgreich verarbeitet wurde kann die Option **/NoOkEmail** übergeben werden.

/Delete **(optional)**
Löscht alle korrekt verarbeiteten E-Mails aus dem Postfach des E-Mail-Servers.

/Datei:<Dateiname> **(optional)**
Angabe eines Dateinamens in der **ERPAdmin** ein Protokoll der Abarbeitung im CSV-Format speichert. Wird kein Dateiname übergeben speichert **ERPAdmin** das Protokoll unter „**POPControl_Protokoll.csv**“ im **ERPAdmin** Verzeichnis.

/MoveToBad **(optional)**
Verschiebt fehlerhafte E-Mail-Dateien aus dem **ERPAdmin**-Verzeichnis in das BAD-Verzeichnis (C_BADDIR).

/NoLockError **(optional)**
Kann ein Datensatz nicht geändert werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

Einstellungen in der VAR.INI
Damit die E-Mails aus dem POP3-Postfach abgerufen werden können sind in der **VAR.INI** folgende Einträge vorzunehmen:

C_POPSERVER	IP-Adresse oder Name des POP3-Servers
C_POPSERVERPORT	Server-Port für Kommunikation (143, 993, 995, etc.)
C_POPSSL	**TRUE**=SSL, **FALSE** sonst
C_POPUSER	Benutzername für das POP3-Postfach
C_POPPWD	Kennwort für das POP3-Postfach
C_POPControlDokName	Schema des Dokumentennamens mit Platzhaltern für Historisierung. Beschreibung siehe Befehl (**POPConnect**). Achtung: Der Name muss eindeutig sein!
C_POPControlDokNameMaxLength	Max. länge der Dokumenten-Bezeichnung für Historisierung. Überzählige Zeichen werden abgeschnitten
C_HistoryDir	Verzeichnis in dem alle verarbeiteten E-Mails archiviert werden
C_BadDir	Verzeichnis in dem alle fehlerhaften E-Mails archiviert werden **PFLICHTANGABE**

C_POPControlSignatur	Zuweisung: **<Tabelle>=<Feldname>** um eine Echtheitsprüfung durchzuführen. Typischer Weise wird ein Selektionsfeld verwendet. Mehrere Einträge sind durch ein Semikolon zu trennen.
C_POPControlWriteOnce	**TRUE** wenn nur einmal in die Datenbank geschrieben werden darf. **FALSE** sonst. Setzt die Verwendung einer Signatur voraus. Nach dem Schreiben in die Datenbank wird ein Zeitstempel an das Signaturfeld angehängt.

Achtung:

Die beiden Befehle **POPControl** und **POPConnect** verwenden teilweise die gleichen **VAR.INI** Einträge. Bitte beachten Sie das bei der Verwendung.

C_POPControlDokName
Hier wird der Name definiert der zur Archivierung der E-Mails verwendet werden soll. Dazu stehen folgende Platzhalter zur Verfügung:

@DATE@	Datum der E-Mail
@TIME@	Uhrzeit der E-Mail
@YYYY@	Jahreszahl 4-Stellig: 2015
@YY@	Jahreszahl 2-Stellig: 15
@MM@	Monat 2-Stellig: 10 (für Oktober)
@DD@	Tag 2-Stellig: 13
@h@	Stunde 2 Stellig: 10
@m@	Minute 2 Stellig: 30
@s@	Stunde 2 Stellig: 00
@SUBJECT@	Betreff der E-Mail
@EMAILFROM@	Absende-Adresse der E-Mail

Bitte beachten Sie die Groß-Klein-Schreibung

Aufbau der Betreffzeile

Im Betreff der Email wird **ERPAdmin** mitgeteilt, welcher Feldwert in welcher Tabelle und in welchem Mandanten gesetzt werden soll. Folgende Platzhalter müssen übergeben werden:

- **@TABELLE=<Tabellen-Name>@** z.B: **@Tabelle=Artikel@**
- **#<SuchIndex>=Suchwert#** z.B: **#ArtNr=12345#**
- **$<Feldname>=Feldwert$** z.B. **$WgrNr=100$**
- **!<Befehl>!** Z.B: **!EmailCC=Erpadmin@Erpadmin.de!**

Die Reihenfolge der Platzhalter in der Betreffzeile darf nicht verändert werden. Es können aber Kommentierungen vor, zwischen oder nach den Platzhaltern angegeben werden. Diese werden von **ERPAdmin** ignoriert. Jedoch dürfen in den Kommentierungen nicht die Steuerzeichen **@ # $** enthalten sein!

Sollen in der Feldzuweisung mehrere Felder zeitgleich übergeben werden, sind die einzelnen Felder durch ein Semikolon zu trennen.

Innerhalb der Betreffzeile dürfen die Steuerzeichen **@**, **#**, **$** und **!** sonst nicht verwendet werden und auch nicht in den Feldzuweisungen vorkommen!

Achtung: Eine Feldzuweisung überschreibt immer bereits vorhandene Werte.

POPControl-Befehle

ERPAdmin kennt vier Befehle über die **POPControl** gesteuert werden kann. Der Aufbau eines Befehle muss wie folgt sein: **!Befehl=<Wert>!**

EmailCC

Mit diesem Befehl wird **ERPAdmin** mitgeteilt, dass die Bestätigung-E-Mail (OK-Email) an eine weitere E-Mail-Adresse gesendet werden soll.

Beispiel: **!EmailCC=Horn@Systemhaus-Horn.de!**

EmailBCC

Mit diesem Befehl wird **ERPAdmin** mitgeteilt, dass die Bestätigung-E-Mail (OK-Email) an eine weitere E-Mail-Adresse gesendet werden soll.

Beispiel: **!EmailBCC=Horn@Systemhaus-Horn.de!**

EmailBody

Mit diesem Befehl wird **ERPAdmin** mitgeteilt, dass der Text-Inhalt der E-Mail in einem Datenbankfeld von **büro+** gespeichert werden soll. Da es sich bei E-Mails in der Regel um Mehrzeiligen Text handelt, sollte hier immer ein RTF Feld (z.B. Info- oder Memo-Feld) angegeben werden. Sind in dem angegebenen Feld bereits Daten vorhanden wird der Eintrag unter Angabe des aktuellen Timestamp angehängt. Vorhandene Formatierungen gehen dabei verloren.

Beispiel: **!EmailBody=Memo!**

Signatur

Mit diesem Befehl wird **ERPAdmin** ein Signatur-Wert aus einem Selektionsfeld übergeben der vor dem Schreiben in die Datenbank auf Richtigkeit geprüft wird. Damit soll ein unberechtigter Schreib-Zugriff unterbunden werden. Die Erzeugung des Wertes kann durch eine Regel in **büro+** realisiert werden. Dazu sollte das Datum oder Die Uhrzeit als verwendet werden.

Beispiel: !Signatur=<Selektionsfeld>!

Zusätzlich muss in der **VAR.INI** ein Feldname aus **büro+** zugeordnet werden:

C_POPControlSignatur="<Tabelle>=<Selektionsfeld>"

Wurde in der **VAR.INI** eine Zuweisung vorgenommen ist die Übergabe einer Signaturwertes in der Betreffzeile der E-Mail und die Übereinstimmung dieser Werte zwingend notwendig. Der im Betreff übergebene Wert wird also mit dem Feldinhalt der Datenbank auf Gleichheit geprüft.

Wird **POPControl** für mehrere Datenbanken verwendet sind die verschiedenen Zuweisungen durch Semikolon zu trennen. Pro Datenbank (Tabelle) kann nur ein Signaturfeld ausgewählt werden.

Soll z.B. in der Tabelle „**Vorgang**“ der Inhalt des Selektionsfeldes „**Sel5**“ als Signaturwert herangezogen werden, lautet der **VAR.INI** Eintrag: **C_POPControlSignatur = "Vorgang= Sel5"**

Platzhalter

Es ist möglich das Absende-Datum der E-Mail und die E-Mail-Adresse des Absenders in der Feldzuweisung zu verwenden. Benutzen Sie dazu die beiden Platzhalter **@EmailDate@** oder **@EmailFrom@.** Bitte beachten Sie die Groß-Kleinschreibung!

Sicherheit durch Signatur-Werte

Damit von **ERPAdmin** keine beliebigen E-Mails verarbeitet werden kann ein Signaturwert aus der Datenbank von **büro+** übergeben werden. Dazu muss in der **VAR.INI** ein Feld definiert werden in dem die Werte gespeichert sind. Dazu kann ein Selektionsfeld verwendet werden. Der Wert aus diesem Selektionsfeld muss dann in der Betreffzeile mitgegeben werden. Bevor **ERPAdmin** in die **büro+** Datenbank schreibt prüft es ab ob der übergebene Wert mit dem Feldinhalt des Datensatzes übereinstimmt.

Der Inhalt dieser Signaturfelder müssen vorher mit einem Wert befüllt werden. Dies kann z.B. durch eine Regel erfolgen.

Beispiel 1:

Alle Angebote sollen von den Vertretern vor Versendung an den Kunden durch die Vertreter freigegeben werden. Dazu wurde in den Vorgängen eine Selektion (SEL1) angelegt, welche die Werte „**Freigegeben**“ oder „**Abgelehnt**“ enthalten kann. Jedes Angebot wird an den jeweiligen Vertreter per E-Mail versendet. In der E-Mail sind zwei Links hinterlegt auf die der Vertreter Klicken kann. Jeder der beiden Links erzeugt eine E-Mail-Antwort. Einmal für die „**Freigegeben**“ und einmal für die „**Abgelehnt**“.

Die E-Mail könnte wie folgt aussehen. Die Platzhalter werden beim Versand durch **büro+** mit Daten befüllt.

Sehr geehrter Herr Müller,

anbei das Angebüt für die Firma Vog.Re.Na2 für das Projekt **"Vog.Bez"**

Zustimmung

Ablehnung

Mit besten Grüßen

AktBzr.VNa AktBzr.NNa

.....

SYSTEMHAUS **HORN**
Computer - Netzwerke - Beratung - Service
AktMand.Str
AktMand.PLZ AktMand.Ort
Telefon: AktMand.Tel1
Telefax: AktMand.Fax

Der Link „**Zustimmung**“ sieht wie folgt aus:

```
<a href="mailto:erpadmin@systemhaus-horn.de&subject=@Tabelle=Vorgang@#BelegNr=«Vog.BelegNr»#$Sel1=Freigegeben$">
```

Mit dem Klick auf den Link wird eine E-Mail an den Empfänger erpadmin@systemhaus-horn.de gesendet. Der Betreff enthält dabei die folgenden Angaben:

Wähle die Datenbanktabelle: **@Tabelle=Vorgang@**
Wähle den Vorgang aus: **#BelegNr=«Vog.BelegNr»#**
Speichere folgende Daten: **$Sel1=Freigegeben$**

Der Link „**Ablehnung**“ sieht wie folgt aus:

```
<a href="mailto:erpadmin@systemhaus-horn.de&subject=@Tabelle=Vorgang@#BelegNr=«Vog.BelegNr»#$Sel1=Abgelehnt$">
```

Der Betreff enthält dabei die folgenden Angaben:
Wähle die Datenbanktabelle: **@Tabelle=Vorgang@**
Wähle den Vorgang aus: **#BelegNr=«Vog.BelegNr»#**
Speicher folgende Daten: **$Sel1=Abgelehnt$**

Beispiel 2:

Um ein beliebiges Datenbankfeld in **büro+** mit einem Wert zu befüllen kann man eine E-Mail an das von **ERPAdmin** überwachte POP3-Postfach senden. Im Betreff der E-Mail muss dann angegeben werden, welches Feld für welchen Datensatz in welcher Tabelle beschrieben werden soll.

Speicher bei Artikel-Nummer „**12345**“ die Warengruppe=„**100**“. Der Betreff sieht dann wie folgt aus:
@Tabelle=Artikel@ #ArtNr=12345# $WgrNr=100$

Speicher bei Vorgang „**AN1801234**“ in das Feld **SEL1** der Wert „**Freigegeben**“. Der Betreff der E-Mail sieht dazu wie folgt aus:
@Tabelle=Vorgang@ #BelegNr=AN18012345# $Sel1=Freigegeben$

Speicher bei Vorgang „**AN1801234**“ in das Feld **SEL1** der Wert „**Freigegeben**“ und Speicher zusätzlich den E-Mail Inhalt als Kommentar im Memo-Feld des Vorgangs. Der Betreff der E-Mail sieht dazu wie folgt aus:
@Tabelle=Vorgang@ #BelegNr=AN18012345# $Sel1=Freigegeben$!EmailBody=Memo!

Speicher bei Adresse-Nummer „**11223**“ in das Feld **SEL1** das aktuellem Tagesdatum“. Der Betreff der E-Mail sieht dazu wie folgt aus:
@Tabelle=Adresse@ #AdrNr=11223# $Sel1=01.07.2018$
oder
@Tabelle=Adresse@ #AdrNr=11223# $Sel1=@DATE@$

Speicher bei Artikel-Nummer „**22344**“ das Absende-Datum in das Feld **Sel1** und die Absender-E-Mail-Adresse in dem Feld **SEL2**. Die Bestätigungs-E-Mail soll im CC an die Adresse Vertrieb@Kunden.de gesendet werden. Der Betreff der E-Mail sieht dazu wie folgt aus:
@Tabelle=Artikel@ #ArtNr=22344# $Sel1=@EmailDate@;Sel2=@EmailFrom@$
!EmailCC=Vertrieb@Kunde.de!

> **Hinweis:**
> Die farbliche Hervorhebung der Sonderzeichen **@ # $** dient hier lediglich einer besseren Unterscheidbarkeit.

Tipp:
Verwenden Sie für **ERPAdmin -POPControl** ein eignes E-Mail-Postfach welches Sie von einem PC aus überwachen. So können Sie Fehlerhafte E-Mails die im Postfach liegen bleiben, weil z.B. keine passende Betreff-Zeile gefunden wurde schneller erkennen und ggf. löschen.

POP-Postfach Inhalt auslesen

Kommt es beim Abrufen des POP-Postfaches zu Fehlermeldungen, ist es hilfreich, wenn man sich den Inhalt des POP-Postfaches ansehen kann. Besteht kein direkter Zugriff auf das Postfach kann der Inhalt mit **ERPAdmin** wie folgt ausgelesen werden:

ERPAdmin -ListPOPControl /Datei:POPControl.csv /Header

Als Ergebnis wird eine CSV-Datei „**POPControl.csv**“ erzeugt die den Inhalt des POP-Postfaches wiedergibt. Dabei bleiben die Daten auf dem POP-Server unverändert. Der Zugriff erfolgt nur lesend.

108 Der Befehl „-PrintFile“ PRO

ERPAdmin -PrintFile /Datei:<Dateinamen> /Drucker:<Druckername> /MoveToHis

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Gibt die angegebene **Word-** oder **PDF-Datei** auf dem Standard Drucker aus.

/Datei:< Dateiname>
Name der PDF oder Word-Datei inkl. Pfadangabe die gedruckt werden sollen. Die Angabe eines Wildcard (*****) ist erlaubt (z.B. *.pdf). Folgende Dateiformate können gedruckt werden und setzten eine installierte Version der benötigten Software z.B. Acrobat Reader oder Microsoft Word voraus:

- **PDF** (Acrobar Reader)
- **Doc, Docx** (Microsoft Word)

/Drucker:<Druckername> **(optional)**
Für die Ausgabe der **Datei** kann ein abweichender Drucker verwendet werden. Wird kein abweichender Drucker angegeben erfolgt der Druck auf dem Standard Drucker. Es muss der Druckername angegeben werden unter dem der Drucker in der Systemsteuerung angezeigt wird.

Eine Liste der verfügbaren Drucker kann mit dem Befehl **ERPAdmin -ListPrinter** erzeugt werden.

/MoveToHis **(optional)**
Zur Verwendung dieser Option lesen Sie bitte das Kapitel „**Optionen und Parameter von ERPAdmin**“.

Beispiel:

ERPAdmin -PrintFile /Datei:*.pdf /MoveToHis
Druckt alle PDF-Dateien im aktuellen Verzeichnis auf dem Stand-Drucker aus und verschiebt diese nach erfolgreicher Druckausgabe in das History-Verzeichnis.

ERPAdmin -PrintFile /Datei:*.pdf "/Drucker:HP OfficeJet 9540dlw" /MoveToHis
Druckt alle PDF-Dateien im aktuellen Verzeichnis auf dem Drucker „**HP OfficeJet 9540dlw**“ aus und verschiebt diese nach erfolgreicher Druckausgabe in das History-Verzeichnis.

109 Der Befehl „-ProjekteInArchiv“ PRO

ERPAdmin -ProjekteInArchiv /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl verschiebt **ERPAdmin** Projektdatensätze in das Projekt-Archiv.

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Die zu verschiebenden Versand-Daten können über die Werte **/Von:** bis **/Bis:** eingeschränkt werden.
Wird kein Bereich angegeben, werden alle Projekt-Datensätze in das Projekt-Archiv verschoben.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Beispiele:

Sie möchten alle Projektdatensätze aus dem Zeitraum **01.01.2022** bis **31.12.2022** in das Projekt-Archiv verschieben.
ERPAdmin -ProjekteInArchiv /Von:Dat=01.01.2022 /Bis:Dat=31.12.2022

110 Der Befehl „-ReadData“ PRO

ERPAdmin -ReadData <Datenbank> <Feldliste> /Datei: /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl stellt **ERPAdmin** eine Möglichkeit zur Verfügung Daten aus **büro+** in eine CSV-Datei zu schreiben ohne einen Export erstellen zu müssen. Dazu müssen die Feldnamen aus **büro+** in einer Liste mit Semikolon getrennt übergeben werden.

> **Hinweis:**
> Es können keine Datenfelder vom Typ „Array“ oder „BlobTable“ abgefragt werden. Eine Liste der Datenbankfelder und der Feldtypen können Sie mit dem Befehl **ERPAdmin -ListDatenbankFelder** erzeugen.

<Datenbank>
Bereich aus **büro+** aus dem die Daten ausgelesen werden sollen. Z.B: Artikel, Adressen Vorgang, etc.

<Feldliste>
Übergibt eine Liste von Feldbezeichner aus **büro+** an **ERPAdmin**. Die einzelnen Feldbezeichner werden mit Semikolon getrennt.

/Datei:<CSV-Dateiname> **(optional)**
Name der CSV-Datei inkl. Pfadangabe in der das Ergebnis gespeichert werden soll. Wird kein Dateiname angeben wird das Ergebnis in der Datei **Datenbank**.csv gespeichert.

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen, nach denen auch sortiert werden kann.

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/Header **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit den Spaltenbezeichnungen.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **<CSV-Ergebnis>** mit den Daten sendet. Wurden keine Daten gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Beispiele:

ERPAdmin -ReadData Artikel "ArtNr;KuBez1;Suchbeg" /Datei:Artikelliste.csv /Von:ArtNr=1000 /Bis:ArtNr=9999
Erzeugt die CSV-Datei **Artikelliste.csv** und speichert die ausgewählten Daten der Datenbankfelder „**ArtNr;KuBez1;Suchbeg**“ der **Artikel 1000 bis 9999**.

ERPAdmin -ReadData Artikel "ArtNr;KuBez1;Suchbeg" /Datei:Artikelliste.csv /Filter:WShopKz=1
Erzeugt die CSV-Datei **Artikelliste.csv** und speichert alle **Artikel** mit WebShop-Kennzeichen.

ERPAdmin -ReadData Artikel "ArtNr;KuBez1;Suchbeg" /Datei:Artikelliste.csv /Filter:WShopKz=1 /Header
Erzeugt die CSV-Datei **Artikelliste.csv** und speichert alle **Artikel** mit WebShop-Kennzeichen und einer Kopfzeile mit den Spaltenbezeichnungen.

ERPAdmin -ReadData Artikel "ArtNr;KuBez1;Suchbeg" /Datei:Artikelliste.csv /Filter:WShopKz=1 /Header /Email:Kunden@Firma.de
Erzeugt die CSV-Datei **Artikelliste.csv** und speichert alle **Artikel** mit WebShop-Kennzeichen und einer Kopfzeile mit den Spaltenbezeichnungen und sendet die Datei anschließend an die E-Mail **Kunde@Firma.de**.

111 Der Befehl „-ReadDataSetValue“ PRO

ERPAdmin -ReadDataSetValue <Datenbank> <Feldliste> <Zuweisung> /Datei: /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /Header /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /NoLockError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl stellt **ERPAdmin** eine Möglichkeit zur Verfügung Daten aus **büro+** in eine CSV-Datei zu schreiben ohne einen Export erstellen zu müssen. Dazu müssen die Feldnamen aus **büro+** in einer Liste mit Semikolon getrennt übergeben werden. Nachdem ein Datensatz geschrieben wurde wird in diesem Datensatz ein Feldwert gesetzt. So können neue Datensätze selektiert werden die seit dem letzten Aufruf dazugekommen sind.

<Datenbank>
Bereich aus **büro+** aus dem die Daten ausgelesen werden sollen. Z.B: Artikel, Adressen Vorgang, etc.

<Feldliste>
Übergibt eine Liste von Feldbezeichner aus **büro+** an **ERPAdmin**. Die einzelnen Feldbezeichner werden mit Semikolon getrennt.

<Zuweisung>
Führt eine **Feldzuweisung** in der **Datenbank** aus nachdem die Werte „**Feldliste**“ ausgelesen wurden. Sollen mehrere Feldzuweisungen ausgeführt werden müssen die einzelnen Zuweisungen mit einem **Semikolon** getrennt werden.

/Datei:<CSV-Dateiname> **(optional)**
Name der CSV-Datei inkl. Pfadangabe in der das Ergebnis gespeichert werden soll. Wird kein Dateiname angeben wird das Ergebnis in der Datei **Datenbank**.csv gespeichert.

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen, nach denen auch sortiert werden kann.

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/Header **(optional)**
Erzeugt in der CSV-Datei eine Kopfzeile mit den Spaltenbezeichnungen.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **<CSV-Ergebnis>** mit den Daten sendet. Wurden keine Daten gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/NoLockError **(optional)**
Kann ein Datensatz nicht geändert werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

Beispiele:

ERPAdmin -ReadDataSetValue Artikel ArtNr;KuBez1;Suchbeg Sel1=1;Sel2=@Date@ /Datei:Artikelliste.csv /Filter:Sel1=0
Schreibt den Inhalt der Felder „**ArtNr;KuBez1;Suchbeg**“ aus der Datenbank **Artikel** in die Datei **Artikelliste.csv**. Anschließend werden die Werte **SEL1=1** (True) und **Sel2=@Date@** (Aktuelles Tagesdatum) in die Datensätze geschrieben. Bei einem 2. Aufruf werden nur noch die Datensätze des vorherigen Aufrufs nicht mehr Selektiert.

ERPAdmin -ReadDataSetValue Artikel ArtNr;KuBez1;Suchbeg Sel1=1;Sel2=@Date@ /Datei:Artikelliste.csv /Filter:Sel1=0 /Header
Erzeugt das gleiche Ergebnis wie das vorherige Beispiel jedoch mit einer Kopfzeile in der CSV-Datei.

113 Der Befehl „-ReplaceQuotation“ PRO

ERPAdmin -ReplaceQuotation <CSV-Datei> <Ersetzung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Führt in einer **CSV-Datei** eine Ersetzung aller Anführungszeichen durch und ersetzt diese durch die Zeichenfolge **Ersetzung**. Wird kein Wert für **Ersetzung** übergeben, werden alle Anführungszeichen aus der Datei gelöscht.

Beispiele:

ERPAdmin -ReplaceQuotation Order.csv
Löscht alle Anführungszeichen aus der Datei **Order.csv**

Datei vorher:

```
"111";"22222";"3333333"
"111";"22222";"3333333"
"111";"22222";"3333333"
```

Datei nachher

```
111;22222;3333333
111;22222;3333333
111;22222;3333333
```

ERPAdmin -ReplaceQuotation Order.csv Inch
Ersetzt alle Anführungszeichen in der Daten **Order.csv** durch das Wort **Inch**

Datei vorher:

```
111;LCD-Fernsehen mit 32" Diagonale
111;LCD-Fernsehen mit 40" Diagonale
111;LCD-Fernsehen mit 60" Diagonale
```

Datei nachher

```
111;LCD-Fernsehen mit 32Inch Diagonale
111;LCD-Fernsehen mit 40Inch Diagonale
111;LCD-Fernsehen mit 60Inch Diagonale
```

114 Der Befehl „-RTFExport“ PRO

ERPAdmin -RTFExport <Datenbank> <RTF-Feld><Export-Verzeichnis> /Von:<Bereich von>/Bis:<Bereich bis> /Filter:<Filterbedingung>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin erzeugt pro Datensatz aus dem Bereich **Datenbank** eine RTF-Datei mit dem RTF-Inhalt aus dem Datenbankfeld **RTF-Feld**. Die RTF-Dateien werden im Verzeichnis **Export-Verzeichnis** im RTF-Format abgespeichert. Auf diese Weise können Text-Auszeichnungen wie **Fett**, *Kursiv*, Unterstreichungen, Zeichen-**Größen,** Schriftarten und Farben aber auch Bilder und Tabellen aus RTF-Feldern übertragen werden was mit dem Standard Export/Import von **büro+** nicht möglich ist.

<Datenbank>
Gibt die Datenbank aus **büro+** an aus der RTF-Feld-Inhalte exportiert werden sollen.
Beispiele: Artikel, Adressen Vorgang

<RTF-Feld>
Feldbezeichner aus Datenbank von **büro+** dessen Inhalt im RTF-Format exportiert werden soll.
Beispiele: Bez1, Memo, Info

<Export-Verzeichnis>
Verzeichnispfad in dem die RTF-Dateien erzeugt werden sollen. Pro Datensatz aus der **Datenbank** wird eine RTF-Datei erzeugt. Diese tragen dann den Namen/Nummer aus den Datenbankbereich. Bei einem Export aus dem Bereich der Stammdaten- Artikeln ist dies z.B. die Artikelnummer.

/Von:<Bereich von> **(optional)**
Selektiert einen Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt. Verwendet werden können alle Variablen nach denen auch sortiert werden kann.
Beispiel: AdrNr=10000

/Bis:<Bereich bis> **(optional)**
Hier wird das Ende des Bereiches vorbelegt. Wird kein Ende definiert wird der Wert **/Von:** verwendet.
Beispiel: AdrNr=19999

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.
Beispiel: Status=‘Kunde‘

Beispiele:

ERPAdmin -RTFExport Artikel Bez1 C:\ERPAdmin\Dok
Erzeugt für alle **Artikel** eine RTF-Datei im Verzeichnis **c:\ERPAdmin\Dok** mit dem RTF-Inhalt aus dem Feld **Bez1**.

> **Hinweis:**
> Die Export-Dateien werden mit der Endung **.rtf** gespeichert und können mit Word oder WordPad direkt bearbeitet werden.

115 Der Befehl „-RTFImport“ PRO

ERPAdmin -RTFImport <Datenbank> <RTF-Feld> <Import-Verzeichnis> /Datei:<CSVDatei> /MoveToHis

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin importiert den Inhalt der RTF-Dateien aus dem **Import-Verzeichnis** in die Datensätze im Bereich **Datenbank**. Der Inhalt der RTF-Dateien wird dabei in das **RTF-Feld** geschrieben. Die **Zuordnung** einer Datei zu einem Datensatz innerhalb von **büro+** erfolgt anhand des Dateinamens ohne Endung.

<Datenbank>
Gibt die Datenbank aus **büro+** an in die RTF-Felder importiert werden sollen.
Aktuell werden folgende Datenbanken unterstützt (**Zuordnung**):
Artikel (**ArtNr**), Adressen (**AdrNr**), Vorgang (**BelegNr**)

<RTF-Feld>
Feldbezeichner aus Datenbank von **büro+** in die der RTF-Inhalt importiert werden soll.
Beispiele: Bez1, Memo, Info

<Import-Verzeichnis>
Verzeichnispfad in dem die RTF-Dateien für den Import liegen. Wir keine Steuer-Datei mit der Option /Datei:<CSV-Datei> übergeben erfolgt die Zuordnung zum Datensatz anhand des Dateinamens.

/Datei:<CSVDatei> **(optional)**
Kann die Zuordnung einer RTF-Datei nicht anhand des Dateinamens erfolgen, kann eine ANSI-CSV-Datei übergeben werden welche in der ersten Spalte die Zuordnung zu **büro+** enthält und in der zweiten Spalte den Dateinamen der RTF-Datei aus dem **Import-Verzeichnis**.

/MoveToHis **(optional)**
Wurde eine **CSVDatei** mit der Option **/Datei:<CSVDatei>** angegeben wird diese nach erfolgreicher Verarbeitung in das History-Verzeichnis verschoben.

Beispiele:

ERPAdmin -RTFImport Artikel Bez1 C:\ERPAdmin\Dok
Importiert den Inhalt der RTF-Datei im Verzeichnis **c:\ERPAdmin\Dok** in das Feld **Bez1** im Bereich **Artikel**.

ERPAdmin -RTFImport Artikel Bez1 C:\ERPAdmin\Dok\ /Datei:Artikelzuordnung.csv
Importiert den Inhalt der RTF-Datei im Verzeichnis **c:\ERPAdmin\Dok** in das Feld **Bez1** im Bereich **Artikel**. Die Zuordnung erfolgt anhand der Datei Artikelzuordnung.csv. Diese könnte wie folgt aussehen:

	A	B
1	1000	RTFText1.rtf
2	1001	RTFText1.rtf
3	1002	RTFText3.rtf

Der Inhalt der Datei **RTFText1.rtf** wird in den Artikel ‘**1000**‘ in das Feld **Bez1** geschrieben.
Der Inhalt der Datei **RTFText2.rtf** wird in den Artikel ‘**1001**‘ in das Feld **Bez1** geschrieben.
Der Inhalt der Datei **RTFText3.rtf** wird in den Artikel ‘**1002**‘ in das Feld **Bez1** geschrieben.

116 Der Befehl „-SendAtOnce“ PRO

ERPAdmin -SendAtOnce <Hinweistext>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Nach jedem Aufruf von **ERPAdmin** bekommt der Benutzer eine Rückmeldung per E-Mail, wenn kein Fehler aufgetreten ist und wenn die Rückmeldung für den OK-Fall nicht deaktiviert wurde (**/NoOkEmail**).

Sollen alle diese OK-Rückmeldungen gesammelt werden und zu einem definierten Zeitpunkt an den Benutzer versendet werden kann dies durch einen Eintrag in der **VAR.INI** eingestellt werden:

C_EMAIL_SENDATONCE=TRUE

Ist diese VAR.INI-Variable gesetzt, werden keine **OK-Emails** mehr an den Benutzer gesendet und alle Ergebnisse (außer Fehlermeldungen) in einer Datei **ERPAdmin <PC-Name> SendAtOnce.TXT** im ERPAdmin-Verzeichnis gesammelt.

Treten während der Verarbeitung Fehler auf werden diese ungeachtet dieser Einstellung direkt an den Benutzer gesendet.

Sollen alle so gesammelten Rückmeldungen dann zu einem definierten Zeitpunkt an den Benutzer gesendet werden, muss folgenden Aufruf durchgeführt werden:

ERPAdmin -SendAtOnce

Nach dem Aufruf sendet **ERPAdmin** den Inhalt der Datei **ERPAdmin <PC-Name> SendAtOnce.TXT** an den Benutzer und löscht anschließend die Datei aus dem ERPAdmin-Verzeichnis.

<Hinweistext> **(optional)**
Damit die E-Mail-Rückmeldungen von verschiedenen **ERPAdmin –SendAtOnce** unterschieden werden können, kann ein Hinweistext beim Aufruf mit übergeben werden. Dieser wird dann im Betreff der E-Mail mit ausgegeben.

Beispiel:

Aus einer CMD-Datei werden nacheinander mehrere Aufrufe von **ERPAdmin** durchgeführt.

ERPAdmin -FTPGet *.Orders.csv
ERPAdmin -Import c:\ERPAdmin\Bestellungen.bpl c:\ERPAdmin\Order*.txt
ERPAdmin -Export c:\ERPAdmin\Lagerbestand.bpl
ERPAdmin -FTPPut Lagerbestand.csv
ERPAdmin -SendAtOnce "Import WebShop Bestellungen"

Die vier ersten Aufrufe schreiben Ihre Rückmeldung an den Benutzer in eine Datei solange es sich nicht um eine Fehlermeldung handelt. Mit dem letzten Aufruf von **ERPAdmin** wird eine E-Mail mit den Rückmeldungen an den Benutzer verwendet.

117 Der Befehl „-SendMsg“ und „/SendMsg“ PRO

ERPAdmin -SendMsg <Rechnername> <Nachricht> <Anzeigedauer>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Es wird eine Nachricht über das Netzwerk an einen anderen Rechner gesendet und dort angezeigt. Dazu wird die Windows Funktion „**Msg**“ verwendet. Damit das funktioniert muss der Nachrichtenversand von einem PC zu einem anderen PC unter Windows funktionieren. Getestet werden kann dies mit folgendem Aufruf von der Windows Kommando-Zeile:

MSG /Server:<PC-Name> * Dies ist eine Test Nachricht

Anpassung der Registry

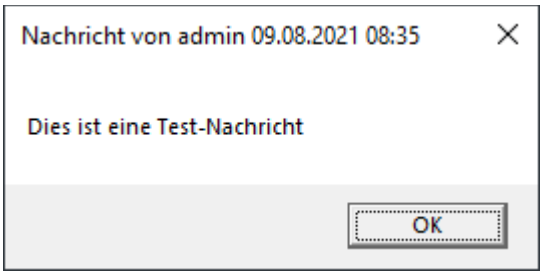
Um unter Windows Nachrichten empfangen zu können muss dies in der Registry aktiviert werden. Gehen Sie dazu wie folgt vor:

- Öffnen sie das Startmenü und geben sie "**regedit**" unten in die Suchleiste ein. Starten sie das Programm
- Navigieren sie zu **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server**
- Suchen sie dort den Eintrag "**AllowRemoteRPC**" und führen sie einen Doppelklick darauf aus
- Ändern sie im sich anschließend öffnenden Fenster den Wert auf **1** und klicken sie auf **OK**
- Gehen sie oben auf "**Ansicht**" und wählen sie den Punkt "**Aktualisieren**", alternativ drücken sie einfach **F5** oder starten sie den Rechner neu

Funktioniert der Nachrichtenversand innerhalb des Windows-Netzwerkes kann auch **ERPAdmin** Nachrichten an Computer senden.

Beispiel:

ERPAdmin -SendMsg BPServer "Dies ist eine Test-Nachricht"



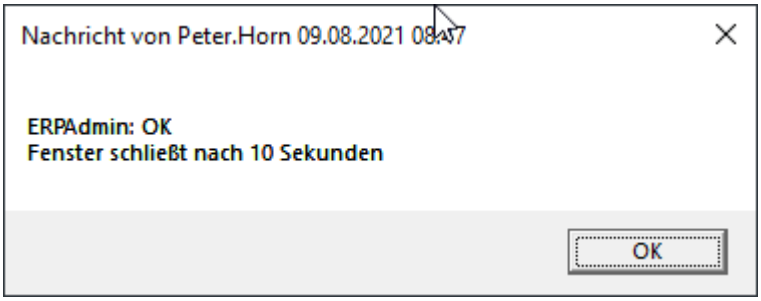
[Auf dem Rechner „**BPServer**“ erscheint ein Nachrichtenfenster]

/SendMsg:<Rechnername>

ERPAdmin sendet eine kurze Statusmeldung nach Ausführung eines **ERPAdmin**-Befehls an den Rechner **Rechnername**.

Beispiel:

ERPAdmin -Init /SendMsg:BPServer



[OK-Statusnachricht von ERPAdmin]

118 Der Befehl „-SetAktDate“ PRO

ERPAdmin -SetAktDate <NewDate> <MonatsWechsel> <JahresWechsel>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin führt einen Wechsel des Arbeitsdatums in **büro+** aus.

Achtung:
Mit diesem Befehl wird nur das Datum des Mandanten erhöht. Die Funktionen des Clients bei Monats- und Jahreswechsel wie z.B. „**Zurücksetzten der Nummernkreise bei Jahreswechsel**“ oder „**SEPA Mandate bei Jahreswechsel zurücksetzten**“ werden **nicht** ausgeführt. Das gleiche gilt für Regel mit der Auswertungsposition „**Nach einem Tages-/Monats-/Jahreswechsel**“.

Wir empfehlen daher die Monats- und Jahres- Wechsel in jedem Fall am Client manuell durchzuführen!

<NewDate>
Gibt das neue Arbeitsdatum für **büro+** an das gesetzt werden soll. Das Datum darf nicht in der Vergangenheit aus Sicht des aktuellen Arbeitsdatum liegen.

<MonatsWechsel> **(optional)**
Soll ein neues Arbeitsdatum gesetzt werden welches einen Monatswechsel zur Folge hat muss der Wert **TRUE** übergeben werden. Wird kein Wert übergeben wird autom. der Wert **FALSE** übergeben. Für einen Jahreswechsel muss auch der Wert für den Monatswechsel auf **TRUE** gesetzt werden.

<JahresWechsel> **(optional)**
Soll ein neues Arbeitsdatum gesetzt werden welches einen Jahreswechsel zur Folge hat muss der Wert **TRUE** übergeben werden. Wird kein Wert übergeben wird autom. der Wert **FALSE** übergeben.

Beispiele:

ERPAdmin -SetAktDate 16.03.2019
Das aktuelle Arbeitsdatum wird auf den **16.03.2019** gesetzt. Das angenommene Arbeitsdatum vor dem Aufruf war der **15.03.2019**.

ERPAdmin -SetAktDate @DATE@
Das aktuelle Arbeitsdatum wird auf das aktuelle Datum **(16.03.2019**) gesetzt. Das angenommene Arbeitsdatum vor dem Aufruf war der **15.03.2019**.

ERPAdmin -SetAktDate @DATE@ TRUE
Das aktuelle Arbeitsdatum wird auf das aktuelle Datum (**01.04.2019**) gesetzt. Das angenommene Arbeitsdatum vor dem Aufruf war der **31.03.2019**. Es wird ein Monatswechsel durchgeführt.

ERPAdmin -SetAktDate @DATE@ TRUE TRUE
Das aktuelle Arbeitsdatum wird auf das aktuelle Datum (**03.01.2019**) gesetzt. Das angenommene Arbeitsdatum vor dem Aufruf war der **21.12.2018**. Es wird ein Jahreswechsel durchgeführt.

119 Der Befehl „-SFTPConvertPuttyKey“ PRO

ERPAdmin -SFTPConvertPuttyKey <Verschlüsselte-ppk-Datei> <unverschlüsselte-ppk-Datei> <Passwort>

Erzeugt aus einer mit **Passwort** verschlüsselten ppk-Datei (privat) eine unverschlüsselte (public) ppk-Datei.

Das Gegenstück zur Publik-Key-Datei (.pub) muss dem SFTP Server bekannt und hinterlegt sein um das Verfahren beim SFTP-Transfer zu nutzen.

<Verschlüsselte ppk-Datei>
Dateiname der **verschlüsselten ppk-Datei**. Diese enthält einen Header der wie folgt aussieht:

```
PuTTY-User-Key-File-2: ssh-rsa
Encryption: aes256-cbc
Comment: rsa-key-20210827
Public-Lines: 6
```

<Unverschlüsselte ppk-Datei>
Ergebnis-Dateiname der **unverschlüsselten ppk-Datei**. Diese enthält einen Header der wie folgt aussieht:

```
PuTTY-User-Key-File-2: ssh-rsa
Encryption: none
Comment: rsa-key-20210827
Public-Lines: 6
```

<Passwort>
Passwort mit dem die **Verschlüsselte ppk-Datei** gesichert wurde.

120 Der Befehl „-SFTPDelete“ PRO

ERPAdmin -SFTPDelete <Dateinamen> <SFTP-Verzeichnis> /NoFileNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin löscht aus dem **SFTP-Verzeichnis** alle **Dateinamen**.

<Dateinamen>
Gibt die Dateien an welche aus dem Verzeichnis SFTP-Verzeichnis gelöscht werden sollen. Die Verwendung eines Wildcards (*) ist erlaubt.

<SFTP-Verzeichnis> **(optional)**
Name des SFTP-Verzeichnisses aus den Dateien gelöscht werden sollen. Wird kein SFTP-Verzeichnis angegeben verwendet ERPAdmin „**./**“

/NoFileNotExistError **(optional)**
Existiert die keine Datei **Dateiname**auf dem SFTP-Server und soll in diesem Fall **keine** Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

Beispiele

ERPAdmin -SFTPDelete Bestellung2001356.csv /NoFileNotExistError
Löscht die Datei **Bestellung2001356.csv** aus dem SFTP-Verzeichnis ./. Existiert die Datei nicht wird keine Fehlermeldung ausgegeben (**/NoFileNotEristError**)

ERPAdmin -SFTPDelete *.jpg ./Artikelbilder/
Löscht alle Dateien ***.jpg** aus dem SFTP-Verzeichnis **./ArtikelBilder/**

121 Der Befehl „-SFTPDir“ PRO

ERPAdmin -SFTPDir <SFTP-Verzeichnis> /Datei:<CSV-Datei> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /Filter:<Dateiname>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin liest den Inhalt eines Verzeichnisses vom SFTP-Server am Bildschirm auf oder schreibt ihn in eine CSV-Datei. Sind mehr als 20 Verzeichniseinträge vorhanden wird die Liste bei der Bildschirmausgabe auf 20 Datensätze begrenzt. Bei der Ausgabe in eine CSV-Datei werden immer alle Datensätze des SFTP-Verzeichnisses ausgegeben.

<SFTP-Verzeichnis>
Verzeichnis auf dem SFTP-Server. Wurde kein **SFTP-Verzeichnis** angegeben wird das aktuelle Verzeichnis **./** verwendet. Wurde in der **VAR.INI** die Variable **C_SFTPDefaultDownloadDir** mit einem Verzeichnis vorbelegt, wird der Inhalt dieses Verzeichnisses ausgegeben.

/Datei:<CSV-Datei>
Name der CSV-Datei in der das Ergebnis gespeichert werden soll. Wird kein Dateiname angegeben wird eine CSV-Datei mit dem Namen „SFTP-Verzeichnis.csv“ erzeugt. Existiert die Datei bereits wird diese überschrieben. Wird keine CSV-Datei angegeben, erfolgt die Rückmeldung am Bildschirm.

Im Gegensatz zur Bildschirmausgabe wird neben dem Dateinamen (Filename) noch die Art (FileType) („directory“ oder „file“), die Dateigröße (Size) und das Erstelldatum (Date) mit ausgegeben.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **<CSV-Datei>** sendet. Wurden keine Datei gefunden wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail Versand an **ERPAdmin**. Neben den standard Platzhaltern (Siehe „Kapitel Platzhalter und variable Parameter“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/Filter:<Dateiname> **(optional)**
Das Ergebnis kann über einen Datei-Filter eingeschränkt werden. Es ist genau ein Wildcard (*) erlaubt. Beispiel: *.csv

Beispiele:

ERPAdmin -SFTPDir
Listet den Inhalt des Login-Verzeichnisses am Bildschirm auf. Es werden nur die ersten 20 Einträge angezeigt.

ERPAdmin -SFTPDir /Bilder/ "/Datei: c:\ERPAdmin\Export.csv" /Email:Demo@ERPAdmin.de
Exportiert den Inhalt des SFTP-Verzeichnisses **/Bilder/** in die CSV-Datei **c:\ERPAdmin\Export.csv** und versendet diese an die E-Mail-Adresse Demo@ERPAdmin.de**.**

	A	B	C	D
1	**Filename**	**FileType**	**Size**	**Date**
2	RE205173148.pdf	file	19076	Wed, 26 Feb 2020 16:19:26 +0100
3	RE205172987_20202402_074457.csv	file	649	Wed, 26 Feb 2020 16:13:26 +0100
4	RE205173168_20202602_133320.csv	file	575	Wed, 26 Feb 2020 16:20:23 +0100
5	RE205173262.pdf	file	17182	Fri, 28 Feb 2020 09:40:36 +0100
6	RE205173181_20202602_174859.csv	file	620	Wed, 26 Feb 2020 18:10:51 +0100
7	RE205172891.pdf	file	17137	Wed, 26 Feb 2020 16:11:02 +0100
8	RE205173163.pdf	file	17805	Wed, 26 Feb 2020 16:20:08 +0100
9	RE205173241_20202702_171536.csv	file	987	Thu, 27 Feb 2020 17:40:53 +0100
10	RE205172922.pdf	file	17310	Wed, 26 Feb 2020 16:11:40 +0100
11	RE205173126.pdf	file	17340	Wed, 26 Feb 2020 16:18:25 +0100
12	RE205173030_20202402_080609.csv	file	919	Wed, 26 Feb 2020 16:14:51 +0100
13	RE205173008_20202402_075332.csv	file	591	Wed, 26 Feb 2020 16:14:07 +0100
14	RE205172913_20202402_070352.csv	file	577	Wed, 26 Feb 2020 16:11:29 +0100

Hinweis:
Die notwendigen Voreinstellungen in der **VAR.INI** sind im Befehl „**SFTPPut“** beschrieben.

122 Der Befehl „-SFTPPut“ PRO

ERPAdmin -SFTPPut <Lokale-Datei> <SFTP-ZielDatei> <SFTP-Zielverzeichnis> <DoubletFile> /NoFileNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Damit die Daten der SFT übertragen werden können, ist die Registrierung der zwei DLLs „**ChilkatAx-9.5.0-win32.dll**“ und „**ChilkatAx-9.5.0-x64.dll**“ notwendig. Mit dem Aufruf „**ERPAdmin -Init**“ können Sie prüfen, ob die DLL korrekt registriert wurde.

Das Programm stellt eine Verbindung zu dem in der **VAR.INI** angegebenen SFTP-Server her und überträgt dann Daten an diesen Server. Dazu werden die ebenfalls in der **VAR.INI** hinterlegten Zugangsdaten verwendet.

Es gibt zwei mögliche Authentifizierungsverfahren:

- **Benutzername** und **Passwort**
- **Benutzername** und **Authentifizierung mit dem Publik Key Verfahren**

Bei der Authentifizierung mit **Benutzername** und **Passsswort** müssen die beiden **VAR.INI** Einträge **C_SFTPUSER** und **C_SFTPPWD** mit gültigen Werten gefüllt sein. Die Variable **C_SFTPPKFILE** muss leer sein.

Public-Key-Verfahren

Erfolgt die Authentifizierung mit dem **Publik Key Verfahren** wird eine Public-Key-Datei (.ppk) und der **Benutzername** benötigt.

Diese ppk-Datei **darfs nicht verschlüsselt sein**. Dies ist in der ppk-Datei in der zweiten Zeile zu erkennen:

```
PuTTY-User-Key-File-2: ssh-rsa
Encryption: none
Comment: rsa-key-20210827
Public-Lines: 6
AAAAB3NzaC1yc2EAAAABJQAAAQEAvn76jYnhNzFGJpuqEwrLvUwS4+Ds97MyVDYF
T2q[illegible]V8Tkm
BWv[illegible]k+1V
NVg[illegible]E/Ti
IZF[illegible]ob58
bYADILE2UIkNJnoJYbJ/l1oxjWZ66k1JtW7HIcAxbQB+cqZg6Q==
Private-Lines: 14
AAABAQC0Mu0CbauHPHLKk0AELM6QeGT1ZhCXSKU65v4asN4AY1UCy5ar7In81Lqz
rw[illegible]JbbEWJ
1R[illegible]F1rsO
J0[illegible]N+XN2
57[illegible]0ROS+
YG[illegible]EnIoL
3I[illegible]+1w7Z
Az[illegible]EjsZj
6o[illegible]Kls61
aH[illegible]mGvNZ
8E[illegible]zBwUA
AA[illegible]gs1Se
Xc[illegible]OMJTN
ac[illegible]=
Private-MAC: 9ae79ceafbd58728e8b820658fec044956fe9fd8
```

[Unverschlüsselte ppk-Datei]

Sollte die Datei verschlüsselt sein muss diese zuvor mit dem folgenden **ERPAdmin** Befehl konvertiert werden in eine unverschlüsselte Datei:

ERPAdmin -SFTPConvertPuttyKey <Verschlüsselte-ppk-Datei> <unverschlüsselte-ppk-Datei> <Passwort>

Das Gegenstück zur Publik-Key-Datei (.pub) muss dem SFTP Server bekannt und hinterlegt sein um das Verfahren zu nutzen.

In der **VAR.INI** wird in der Variablen **C_SFTPPKFILE** der Dateinamen der Schlüssel-Datei (*.ppk) hinterlegt. Diese muss im Aufrufverzeichnis des **ERPAdmin** liegen. Zusätzlich muss der **Benutzername** in der Variablen **C_SFTPUSER** übergeben werden.

Folgende Einstellungen können in der **VAR.INI** vorgenommen werden:

C_SFTPSERVER	SFTP-Server oder IP-Adresse
C_SFTPUSER	SFTP-Benutzername
C_SFTPPWD	SFTP-Passwort
C_SFTPPORT	SFTP-Port, 22=Standard
C_SFTPPKFILE	SFTP-Public-Key-File
C_SFTPDefaultUploadDir	SFTP-Default-Verzeichnis für Upload (-SFTPPut)
C_SFTPDefaultDownloadDir	SFTP-Default-Verzeichnis für Download (-SFTPGet, -SFTPDir)

Bei Verwendung mehrerer SFTP-Server (**/SFTP2 /SFTP3**, etc.) sind die jeweils passenden Variablen zu befüllen.

C_SFTPSERVER2	SFTP-Server oder IP-Adresse 2
C_SFTPUSER2	SFTP-Benutzername 2
C_SFTPPWD2	SFTP-Passwort 2
C_SFTPPORT2	SFTP-Port 2, 22=Standard
C_SFTPPKFILE2	SFTP-Public-Key-File 2
C_SFTPSERVER3	SFTP-Server oder IP-Adresse 3
C_SFTPUSER3	SFTP-Benutzername 3
C_SFTPPWD3	SFTP-Passwort 3
C_SFTPPORT3	SFTP-Port 3, 22=Standard
C_SFTPPKFILE3	SFTP-Public-Key-File 3
C_SFTPSERVER4	SFTP-Server oder IP-Adresse 4
C_SFTPUSER4	SFTP-Benutzername 4
C_SFTPPWD4	SFTP-Passwort 4
C_SFTPPORT4	SFTP-Port 4, 22=Standard
C_SFTPPKFILE4	SFTP-Public-Key-File 4
C_SFTPSERVER5	SFTP-Server oder IP-Adresse 5
C_SFTPUSER5	SFTP-Benutzername 5
C_SFTPPWD5	SFTP-Passwort 5
C_SFTPPORT5	SFTP-Port 5, 22=Standard

C_SFTPPKFILE5 SFTP-Public-Key-File 5

<Lokale-Datei>
Übergibt eine Datei inkl. Pfadangabe die per SFTP übertragen werden soll.
Beispiel: **c:\ERPAdmin\Bild.jpg**

Sollen mehrere Dateien übertragen werden kann ein Wildcard (*) verwendet werden. Es ist nur ein Wildcard (*) erlaubt Angaben wie ***.*** sind **NICHT** zulässig.
Beispiel: **c:\ERPAdmin*.jpg**

<SFTP-Zieldatei>
Dateiname ohne Verzeichnisangabe auf dem SFTP-Zielserver. Wird hier kein Name übergeben wird derselbe Name verwendet die in **Lokale-Datei** angegeben wurde.

<SFTP-Zielverzeichnis>
Unterverzeichnis auf dem SFTP-Server in dem die Datei abgelegt werden soll.
Beispiel: **/Bilder/**

<DoubletFile> **(optional)**
Dateiname inkl. Pfadangabe in dem alle übertragenen Dateien protokolliert werden. Vor jeder Übertragung prüft **ERPAdmin** ob die zu übertragende Datei in der Datei **DoubletFile** vorhanden ist. Die Datei wird nur dann übertragen, wenn Sie noch nicht vorhanden ist. Weitere Informationen zur Doubletten-Prüfung entnehmen Sie bitte dem Kapitel „**Doubletten-Prüfung**“.
Beispiel: **C:\ERPAdmin\Doubletten_SFTPPut.log**

/NoFileNotExistError **(optional)**
Existiert die **Quelldatei** nicht auf dem Server und soll in diesem Fall **keine** Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

> **Hinweis:**
> Die Datei **DoubletFile** sollte nach einer bestimmten Zeit bereinigt werden da Sie über die Zeit immer größer wird und damit unnötigen Speicherplatz belegt und auch mit zunehmender Größe Rechenzeit verbraucht um durchsucht zu werden. Mit der Variablen **C_DoubletFilesMax** in der Datei **VAR.INI** können Sie eine max. Anzahl an Einträgen festlegen. Wird diese überschritten wird eine E-Mail an **C_ERROR_TO** versendet.

Beispiele

ERPAdmin -SFTPPut "c:\ERPAdmin\Bild.jpg"
Die Datei „**c:\ERPAdmin\bild.jpg**“ wird auf den SFTP-Server transferiert und im Root-Verzeichnis des Benutzers unter gleichem Namen abgelegt.

ERPAdmin -SFTPPut "c:\ERPAdmin\Bild.jpg" "Artikel12345.jpg" /images/
Die Datei „**c:\ERPAdmin\Bild.jpg**“ wird per SFTP übertragen und im Verzeichnis „**/images/**“ unter dem Namen „**Artikel12345.jpg**“ abgespeichert.

ERPAdmin -SFTPPut "c:\ERPAdmin\Bild*.jpg" " " /images/
Die Dateien „**c:\ERPAdmin\Bild*.jpg**“ werden per SFTP übertragen und im Verzeichnis „**/images/**“ abgespeichert.

> **Hinweis:**
> Bei einigen SFTP-Servers ist es notwendig den Pfad zu einem Unterverzeichnis mit einem Punkt beginnen zu lassen. Beispiel: **"./images/"**

Wird kein **SFTP-Zielverzeichnis** über die Kommandozeile übergeben kann in der **VAR.INI** eine Vorgabewert hinterlegt werden:

Beispiel: **C_SFTPDefaultUploadDir = "./UploadDir/".**

Wird in der **VAR.INI** kein Vorgabewert hinterlegt wird der Default-Wert **"./"** verwendet.

Zugriff auf mehrere SFTP-Server

Soll **ERPAdmin** auf verschiedene SFTP-Server zugreifen, können die Zugangsdaten für zwei zusätzliche SFTP-Server in der **VAR.INI** zu hinterlegt werden. Der Aufruf von **ERPAdmin** muss dann um die Option **/SFTP2, /SFTP3, /SFTP4** oder **/SFTP5** erweitert werden.

ERPAdmin -SFTPPut "c:\ERPAdmin\Bild.jpg" /SFTP2
Überträgt die Datei **c:\ERPAdmin\Bild.jpg** an den in der **VAR.**INI unter **SFTP2** angegebenen Server und dessen Zugangsdaten.

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung dieser Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“.

123 Der Befehl „-SFTPGET“ PRO

ERPAdmin -SFTPGet <SFTP-QuellDateien> <SFTP-Quellverzeichnis> <Lok.Verzeich.> <DoubletFile> /DELETE /NoFileNotExistError /Ignore0ByteFiles

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Damit die Daten der SFT übertragen werden können, ist die Registrierung der zwei DLLs „**ChilkatAx-9.5.0-win32.dll**“ und „**ChilkatAx-9.5.0-x64.dll**“ notwendig. Mit dem Aufruf „**ERPAdmin -Init**“ können Sie prüfen, ob die DLL korrekt installiert wurde.

Das Programm stellt eine Verbindung zu dem in der **VAR.INI** angegebenen SFTP-Server her und überträgt dann Daten von diesem SFTP-Server. Dazu werden die ebenfalls in der **VAR.INI** hinterlegten Zugangsdaten verwendet.

Die benötigten Einstellungen in der **VAR.INI** sind im Befehl „SFTPGet“ beschrieben.

Die Benutzung erfolgt analog zum **ERPAdmin** Befehl: **-SFTPPut**.

<SFTP-QuellDatei>
Dateiname ohne Verzeichnisangabe auf dem SFTP-Server.
Beispiel: **Bild.jpg**

Zum Download mehrerer Dateien ist auch die Angabe eines Wildcards (*****) erlaubt:
Beispiel: **Bild*.jpg**

Wichtig: Bei der Verwendung von Wildcards (*****) darf immer nur ein Stern verwendet werden. Angaben wie ***.*** sind **NICHT** zulässig.

<SFTP-Quellverzeichnis>
Verzeichnis auf dem SFTP-Server aus dem die Datei geladen werden soll.
Beispiel: **/Bilder/**

<Lok.Verzeichnis>
Lokales Verzeichnis in dem die Datei abgelegt werden soll.
Beispiel: **C:\ERPAdsmin\Bilder**

<DoubletFile> **(optional)**
Dateiname inkl. Pfadangabe in der alle übertragenen Dateien protokolliert werden. Nach jeder Übertragung prüft **ERPAdmin** ob die übertragende Datei in der Datei **DoubletFile** vorhanden ist. und verschiebt in diesem Fall die Datei nach **C_BADDIR**. Weitere Informationen zur Doubletten-Prüfung entnehmen Sie bitte dem Kapitel „**Doubletten-Prüfung**“.
Beispiel: **C:\ERPAdmin\Doubletten_SFTPGet.log**

/DELETE **(optional)**
Zusätzlich kann die Option **/DELETE** angegeben werden, welcher nach dem Download die Datei auf dem SFTP-Server löscht. Wird die Option **/DELETE** Verwendet, sollte die Historisierung aktiviert sein, damit alle per SFTP geladenen Dateien als Sicherungskopie im Historie-Verzeichnis abgelegt werden. Das hilft bei einer späteren Fehlersuche.

/NoFileNotExistError **(optional)**
Existiert die S**FTP-Quelldatei** nicht auf dem Server und soll in diesem Fall **keine** Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** verwendet werden.

/Ignore0ByteFiles **(optional)**
Vor dem Download wird geprüft ob die zu übertragende Datei eine Größe von 0-Byte hat. Ist dies der Fall wird die Datei nicht übertragen und eine Fehlermeldung erzeugt.

Beispiele

ERPAdmin -SFTPGet "Bild.jpg" "/Bilder/" "c:\ERPAdmin\Bilder\"
Transferiert die Datei **„Bild.jpg“** aus dem SFTP-Verzeichnis **„/Bilder/“** in das lokale Verzeichnis **„c:\ERPAdmin\Bilder“** unter dem Namen **„Bild.jpg“**.

ERPAdmin -SFTPGet "*.jpg" "/Bilder/" "c:\ERPAdmin\Bilder\"
Dieser Aufruf transferiert alle Dateien aus dem FTP-Verzeichnis **„/Bilder/“** mit der Endung **„.jpg“** in das lokale Verzeichnis **„c:\ERPAdmin\ Bilder“**.

ERPAdmin -SFTPGet "*.jpg" "/Images/" "c:\ERPAdmin\FTP-Bilder\" /DELETE
Dieser Aufruf transferiert alle Dateien wie im vorherigen Beispiel. Nach der FTP-Übertragung werden die Dateien dann auf dem SFTP-Server gelöscht.

ERPAdmin -SFTPGet "Order*.csv" "/Bestellungen/" "c:\ERPAdmin\" "c:\ERPAdmin\Order.log" /DELETE
Dieser Aufruf transferiert alle SFTP-Dateien mit dem Namen **Order*.csv** aus dem SFTP- Verzeichnis **/Bestellungen/**. Anschließend wird eine Doubletten-Prüfung mit dem Dateinamen durchgeführt anhand der Daten in der Datei **c:\ERPAdmin\Order.log**. Wird eine Doublette gefunden wird diese nach **C_BADDIR** verschoben. Alle transferierten Dateien werden in der Datei **c:\ERPAdmin\Order.log** mit Zeitstempel der ersten Übertragung protokolliert.

Hinweise:
Bei einigen SFTP-Servers ist es notwendig den Pfad zu einem Unterverzeichnis mit einem Punkt beginnen zu lassen. Beispiel: **"./images/"**

Wird kein **SFTP-Quellverzeichnis** über die Kommandozeile übergeben kann in der **VAR.INI** eine Vorgabewert hinterlegt werden:

Beispiel: **C_SFTPDefaultDownloadDir = "./DowloadDir/".**

Wird in der **VAR.INI** kein Vorgabewert hinterlegt wird der Default-Wert **"./"** verwendet.

124 Der Befehl „-SplitCSVToFiles“ PRO

ERPAdmin -SplitCSVToFiles <CSV-Datei> <Templatedatei> <Dateiname> <Format> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin erzeugt für jeden Datensatz aus der **CSV-Datei** eine einzelne Datei. Der Aufbau dieser Dateien wird durch die **Templatedatei** bestimmt. Darin enthaltene Platzhalten der Form **@Header@** (wobei **Header** ein Spaltenbezeichner aus der **CSV-Datei** ist) werden mit dem Inhalt der entsprechenden Spalte des Datensatzes aus der **CSV-Datei** ersetzt. **Dateiname** gibt den Aufbau der Dateinamen an unter dem die Ergebnisdateien abgespeichert werden.

<CSV-Datei>
Name einer CSV-Datei im ANSI-Format mit Kopfzeile, deren Datensätze in einzelne Dateien zerlegt werden sollen.

<Templatedatei>
Vorlage der Ergebnisdatei im ANSI Format. Darin enthalten können Platzhalter sein im Format **@Header@** die durch den Feldinhalt aus der **CSV-Datei** ersetzt werden.

<Dateiname>
Vorlage für Dateinamen unter dem die Ergebnisse gespeichert werden. Erlaubt die Verwendung von Platzhalter in der Form **@Header@** welche dann dynamisch mit dem Feldinhalt des jeweiligen Datensatzes befüllt werden. Wird kein Platzhalter übergeben werden die Ergebnisdateien mit einer Nummerierung versehen beginnen mit „_1“.

Beispiel: Es wird der Dateiname „**Artikel.csv**“ übergeben. Die Ergebnisdateien heißen dann „**Artikel_01.csv**“, „**Artikel_02.csv**“, u.s.w.

<Format> **(optional)**
Gibt an ob die Ergebnisdateien im **ANSI**, **UNICODE** oder **UTF8** Format abgespeichert wird. Wird kein Format angegeben wird das Ergebnis im **ANSI** Format abgespeichert.

/Email:<E-Mail-Adresse> **(optional)**
Sendet das Export-Ergebnis an die angegebene E-Mail sofern der Export eine Ergebnisdatei erzeugt hat. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare

zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/MoveToHis, /MoveToBad, /Delete (optional)
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

Beispiel:

Artikel.csv:

Dateiname_XML	Laenge	Breite	Winkel
LI1902965-1.csv	15	25	90
LI1902965-2.csv	100	200	90
LI1902965-3.csv	200	100	90
LI1906909-2.csv	153	11	90

Template.xml

```
<Artikel>
        <Laenge>@Laenge@</Laenge>
        <Breitse>@Breite@</Breite>
        <Winkel>@Winkel@</Winkel>
        <Schablone>@Schblonde@</Sc
```

ERPAdmin -SplitCSVToFiles Artikel.csv Template.xml @Dateiname_XML@
Erzeugt pro Datensatz aus der Datei **Artikel.csv** eine Ergebnisdatei mit dem Dateinamen aus der Spalte „**Dateiname_XML**“ mit dem Inhalt aus der Datei **Template.xml** im **ANSI**-Format ab. Alle Platzhalter für die es einen Wert in der jeweiligen Spalte gibt werden ersetzt.

Diese sind in dem obigen Beispiel die Platzhalter **@Laenge@, @Breite@, @Winkel@**, etc.

Ergebnisdatei **LI902965-1.csv** aus Datensatz 1:

```
<Artikel>
        <Laenge>15</Laenge>
        <Breite>25</Breite>
        <Winkel>90</Winkel>
```

Hinweis:
Bitte achten Sie bei den Platzhaltern auf Groß-Kleinschreibung und verwenden Sie KEINE Umlaute oder Sonderzeichen.

125 Der Befehl „-Update“ BAS

ERPAdmin -Update <Ziel-Verzeichnis> <Quell-Datei> <Ausnahmen>/Datei:<Ziel-Datei> /Force

Mit diesem Befehl kopiert **ERPAdmin** die **Quell-Datei** in alle Unterverzeichnisse (erste Ebene) des **Ziel-Verzeichnisses**. Wird der Name einer **Ziel-Datei** zusätzlich angegeben, erfolgt das Kopieren in das entsprechende Unterverzeichnis nur dann, wenn sich die **Ziel-Datei** bereits im Unterverzeichnis befindet. Diese wird dann überschrieben.

Ausnahmen für Verzeichnisse
Beginnt ein Unterverzeichnis mit einem Unterstrich (_)wird dieses Verzeichnis ignoriert und übersprungen.

<Ziel-Verzeichnis> **(optional)**
Gibt ein Verzeichnis inkl. Laufwerk an in dessen Unterverzeichnissen Dateien kopiert werden sollen. Existieren diese Dateien bereits werden Sie überschrieben.

Wird kein Zielverzeichnis angegeben, wird das aktuelle Verzeichnis von verwendet.

<Quell-Datei> **(optional)**
Gibt die Datei inkl. Pfad an welche Kopiert werden soll. Wird keine Quell-Datei angegeben wird die **ERPAdmin**.exe verwendet.

<Ausnahmen> **(optional)**
Hier kann eine Liste (durch Semikolon getrennt) von Verzeichnissen angegeben werden, welche übersprungen werden sollen. Es muss der vollständige Verzeichnis-Pfad mit Laufwerk angegeben werden und jede Angabe muss mit einem „**\;**“ enden.

Beispiel: **"C:\ERPAdmin\Logfiles\;C:\ERPAdmin\History\;"**
Die beiden Verzeichnisse **C:\ERPAdmin\Logfiles** und **C:\ERPAdmin\History** werden übersprungen.

/Datei:<Ziel-Datei> **(optional)**
Wird eine Zieldatei angegeben prüft **ERPAdmin** vor dem Kopieren in dem Unterverzeichnis, ob die **Ziel-Datei** bereits existiert. Ist dies nicht der Fall, erfolgt KEIN kopieren der **Quell-Datei** in dieses Unterverzeichnis des **Ziel-Verzeichnisses**.

/Force **(optional)**
Wurde diese Option übergeben, ignoriert **ERPAdmin** die **Ausnahmen für Verzeichnisse** und führt die Aktualisierung in allen Unterverzeichnissen durch.

Beispiele:

Folgende Annahme gilt für die ersten beiden Beispiele: Sie befinden sich im Verzeichnis **c:\ERPAdmin** worin sich auch eine Datei **ERPAdmin.exe** befindet. Das Verzeichnis **C:\ERPAdmin** besitzt mehrere Unterverzeichnisse in den Sie getrennt verschiedene befehle ausführen wollen.

Sie möchten die **ERPAdmin.exe** in alle Unterverzeichnisse des Verzeichnis **C:\ERPadmin** kopieren.
ERPAdmin -Update

Sie möchten die Datei ***ERPAdmin.exe*** nur in die Unterverzeichnisse kopieren in denen sich bereits eine (evtl. ältere Version) von **ERPAdmin.exe** befindet.
ERPAdmin -Update /Datei:ERPAdmin.exe

Sie möchten alle **ERPAdmin** Dateien auf Ihrem Server aktualisieren. Diese befinden sich jeweils in Unterverzeichnissen des Ordners **m:\ERPAdmin**. Es sollen nur die Unterverzeichnisse aktualisiert werden in denen sich bereits eine Datei **ERPAdmin.exe** befindet.
ERPAdmin -Update m:\ERPAdmin\ /Datei:ERPAdmin.exe

126 Der Befehl „-UpdateExchangeRate“ PRO

ERPAdmin -UpdateExchangeRate <Währungsbasis> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse>

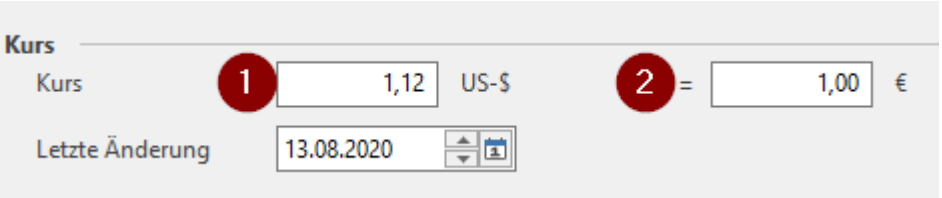
Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin lädt von der Webseite der Europäischen Zentralbank eine Auswahl an Wechselkursen im XML-Format. Der Wechselkurs wird auf Basis der Fremdwährung bereitgestellt. Die Daten werden automatisch in **büro+** importiert (**Fremdwährungstabelle**).

Eine Liste der aktuell unterstützen Wechselkurse finden Sie unter
https://www.ecb.europa.eu/stats/eurofxref/eurofxref-daily.xml

> **Hinweis:**
> **Die Verwendung dieses Befehls setzt die Bereitstellung der Daten über die Webseite der EZB voraus! Diese Funktion ist nur so lange nutzbar, wie die EZB diesen kostenfreien Service in unveränderter Form anbietet!**

<Währungsbasis> **(optional)**
Mit dieser Option kann festgelegt werden ob der Wechselkurs auf Basis der **Fremdwährung** oder des **EUR** hinterlegt werden soll. Wird kein Wert übergeben erfolgt der Wechselkurs auf Basis der Fremdwährungseinheit.



Beispiel: **1 US$ = 1,27 EUR** **Auf Basis der Fremdwährung (1)**

Soll der Wechselkurs auf Basis des **EUR (2)** interpretiert werden, ist als Parameter der Wert **EUR** anzugeben. Andere Werte als **EUR** sind nicht zulässig.

Beispiel: **1 EUR** = **0,78 US$**

Folgende Wechselkurse werden aktuell durch die EZB bereitgestellt (**Stand 01/2021**):

ISO Währung	Währung	Land/Länder
AUD	Dollar	Australien, Kiribati, Nauru, Tuvalu
BGN	Lew	Bulgarien
BRL	Real	Brasilien
CAD	Dollar	Kanada
CHF	Franken	Schweiz, Lichtenstein
CNY	Renminbi Yuan	Volksrepublik China
CZK	Krone	Tschechien
DKK	Krone	Dänemark (inkl. Färöer und Grönland)

GBP	Pfund	Vereinigtes Königreich
HKD	Dollar	Hong Kong
HRK	Kuna	Kroatien
HUF	Forint	Ungarn
IDR	Rupiah	Indonesien
ILS	Schekel	Israel, Staat Palästina
INR	Rupie	Indien
ISK	Krone	Island
JPY	Yen	Japan
KRW	Won	Süd Korea
MXN	Peso	Mexiko
MYR	Ringgit	Malaysia
NOK	Krone	Norwegen
NZD	Dollar	Neuseeland, Niue, Pitcairninseln
PHP	Peso	Philippine
PLN	Zloty	Polen
RON	Leu	Rumänien
RUB	Rubel	Russland
SEK	Krone	Schweden
SGD	Dollar	Singapur
THB	Bath	Thailand
TRY	Lira	Türkei, Nord-Zypern
USD	Dollar	Vereinigte Staaten, Ecuador, El Salvador, Föderierte Staaten von Mikronesien, Marshallinseln, Osttimor, Palau sowie die britischen Überseegebiete Britische Jungferninseln und Turks- und Caicosinseln
ZAR	Rand	Südafrika, Lesotho, Eswatini

/Email:<E-Mail-Adresse> **(optional)**
Sendet die CSV-Datei mit den Wechselkursen an die angegebene E-Mail sofern der Download ein Ergebnis hatte. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen des Exportes einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

Beispiel:

ERPAdmin -UpdateExchangeRate /Email:Demo@ERPAdmin.de

Lädt die aktuellen Wechselkurse von der Webseite der EZB und aktualisiert diese in **büro+.** Anschließend wird die Ergebnis-Datei mit den Wechselkursen an die E-Mail-Adresse Demo@ERPAdmin.de gesendet.

Hinweis:
Nach dem Update der Wechselkurse muss ggf. manuell eine Neuberechnung der Preise angestossen werden. Gehen Sie dazu in den Bereich **Parameter->Sonstige->Fremdwährungen** und öffnen und speichern Sie die gewünschte Fremdwährung. Sie werden anschließend gefragt ob die Preise neu kalkuliert werden sollen. Eine autom. Neuberechnung der Preise ist leider nicht möglich.

127 Der Befehl „-UTF82ANSI“ STD

ERPAdmin -UTF82ANSI <UTF8-Dateiname> <ANSI-Dateiname>

Dieser Befehl wandelt die Datei **UTF8-Dateiname** in das Dateiformat ANSI um und speichert diese unter **ANSI-Dateinamen** ab. Es dürfen keine speziellen UTF8-Zeichen in der Datei vorhanden sein da eine Darstellung / echte Konvertierung nicht möglich ist. Es wird lediglich das Dateiformat konvertiert.

Beispiel:
ERPAdmin -UTF82ANSI c:\ERPAdmin\UTF8-Artikel.CSV c:\ERPAdmin\ANSI-Artikel.CSVv

128 Der Befehl „-UTF8OB2ANSI“ STD

ERPAdmin -UTF8OB2ANSI <Dateiname>

Dieser Befehl wandelt die UTF8-Datei- ohne-BOM **Dateiname** in das Dateiformat ANSI um und speichert diese unter gleichem Dateinamen ab. Es dürfen keine UTF8-Zeichen in der Datei vorhanden sein da eine Darstellung / echte Konvertierung nicht möglich ist. Es wird lediglich das Dateiformat konvertiert.

Beispiel:
ERPAdmin -UTF8OB2ANSI c:\ERPAdmin\Artikel.CSV

129 Der Befehl „-UNICODE2ANSI“ STD

ERPAdmin -UNICODE2ANSI <Dateiname>

Dieser Befehl wandelt die UNICODE-Datei **Dateiname** in das Dateiformat ANSI um und speichert diese unter gleichem Dateinamen ab. Es dürfen keine UNICODE-Zeichen in der Datei vorhanden sein da eine Darstellung / echte Konvertierung nicht möglich ist. Es wird lediglich das Dateiformat konvertiert.

Beispiel:
ERPAdmin -UNICODE2ANSI c:\ERPAdmin\Artikel.CSV

Hinweis:
Die Konvertierung von **UTF8** oder **UNICODE** nach **ANSI** ist nur dann möglich, wenn in der Datei keine Zeichen aus dem **UTF8** oder **UNICODE**-Zeichensatz verwendet wurden.

130 Der Befehl „-VersandInArchiv“ STD

ERPAdmin -VersandInArchiv /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

Mit diesem Befehl verschiebt **ERPAdmin** die Versanddatensätze in das Versand-Archiv.

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Die zu verschiebenden Versand-Daten können über die Werte **/Von:** bis **/Bis:** eingeschränkt werden. Wird kein Bereich angegeben, werden alle Versand-Datensätze in das Versand-Archiv verschoben.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Beispiele:

Sie möchten alle Versanddatensätze mit den Belegnummern „**LI1611500**“ bis „**LI1611999**“ in das Versand-Archiv verschieben.
ERPAdmin -VersandInArchiv /Von:BelegNr=LI1611500 /Bis:BelegNr=LI1611999

Sie möchten alle Versanddatensätze die **Erledigt** sind in das Versandarchiv verschieben.
ERPAdmin -VersandInArchiv /Filter:ErledigtKZ=1

Hinweis: Um Vorgänge über **/Von:** und **/Bis:** einzuschränken können alle Suchfelder verwendet werden die von **büro+** zur Vorgangsselektion verwendet werden und die einen eigenen Suchindex besitzen. Dies sind: **BelegNr**, **PktNr, AdrNr**, etc.

131 Der Befehl „-VorgangAusArchiv“ STD

ERPAdmin -VorgangAusArchiv <VogArtNr> <BuchKZ> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

Die Funktionsweise ist analog zum Befehl „-**VorgangInArchiv**“. Beim Verschieben wird der Vorgang aus dem Archiv in die Vorgänge verschoben. Danach existiert dieser nicht mehr im Archiv.

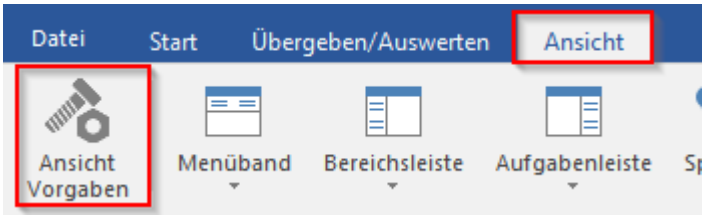
Beispiele:

Sie möchten alle Angebote des Kunden **11500** aus den **Archiv-Vorgängen** in die Vorgänge kopieren.
ERPAdmin -VorgangAusArchiv 15 FALSE /Von:AdrNr=11500

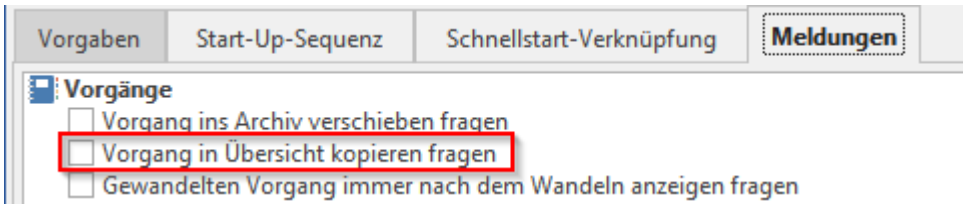
Hinweis:

Vor dem ersten Verschieben von Daten aus einen Archiv- oder History-Bereich muss in **büro+** die entsprechende Sicherheitsmeldung für den COM-Benutzer deaktiviert werden.

Melden Sie sich dazu mit dem COM-Benutzer an und wechseln Sie in das Menü „**Ansicht**“ -> „**Ansicht Vorgaben**“.



Suchen Sie hier den Eintrag im Bereich „**Vorgänge**“-> „**Vorgang in Übersicht kopieren fragen**“ und **deaktivieren** Sie diesen.



Erst jetzt kann das Verschieben über COM durchgeführt werden!

133 Der Befehl „-VorgangBuchen“ STD

ERPAdmin -VorgangBuchen <VogArtNr> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /NoLockError

Mit diesem Befehl bucht **ERPAdmin** alle Vorgänge **VogArtNr** bei denen das **GebuchtKZ** noch nicht gesetzt ist und die **nicht gesperrt** sind.

<VogArtNr>
Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.**

15		Angebot
20		Auftragsbestätigung
30		Bestellung vom Kunden
40		Bestellung an Lieferant
41		Eingangsrechnung
50		Lieferschein
70		Rechnung
90		Gutschrift

Klartextnamen der Vorgänge sind nicht zugelassen.

Hinweis: Gesperrte Vorgänge werden nicht gebucht.

/Von:<Bereich von> /Bis:<Bereich bis>> **(optional)**
Die Vorgänge können über **/Von:** bis **/Bis:** eingeschränkt werden. Um Vorgänge über **/Von:** und **/Bis:** einzuschränken können alle Suchfelder verwendet werden die von **büro+** zur Vorgangsselektion verwendet werden und die einen eigenen Suchindex besitzen. Dies sind: **BelegNr, AuftrNr, Suchbeg, AdrNr, ProjNr**.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/NoLockError **(optional)**
Kann ein Vorgang nicht gebucht werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

Hinweis:
Würde beim Buchen eines Vorgangs der Bestand negativ werden und ist dies laut Buchungs-Parametern nicht zulässig, wird eine Fehlermeldung an den Benutzer erzeugt.
Ist jedoch in der **VAR.INI** die Variable **C_BuchenMeldeMinusBestand=FALSE** gesetzt wird nur eine normale Meldung an den Benutzer erzeugt (Kein Fehler) welche mit der Option **/NoOkEmail** unterdrückt werden kann. Der Standardwert für die Variable ist **TRUE.**

Beispiele:
Sollen alle noch nicht gebuchten Lieferscheine verbucht werden lautet der Aufruf:
ERPAdmin -VorgangBuchen 50

Sollen nur die Lieferscheint mit bestimmten Auftragsnummern (AB1601000 bis AB1609999) gebucht werden sieht der Aufruf wie folgt aus:
ERPAdmin -VorgangBuchen 50 /Von:AuftrNr=AB1601000 /Bis:AuftrNr=AB1609999

Sollen nur die Lieferscheine eines Auftrags AB1601234 gebucht werden lautet der Aufruf:
ERPAdmin -VorgangBuchen 50 /Von:AuftrNr=AB1601234

Soll keine Vorgangsart ausgewählt werden und die Selektion der Vorgänge nur über die Belegnummer erfolgen muss als **VorArtNr** der Wert **0** übergeben werden. Der Aufruf sieht der Aufruf wie folgt aus:
ERPAdmin -VorgangBuchen 0 /Von:BelegNr=LI1602500 /Bis:BelegNr=LI1609999

Hinweis:
Das stornieren eines bereits verbuchten Vorgangs ist mit dieser Funktion nicht möglich. Verwenden Sie dazu den Befehl **ERPAdmin -VorgangStornieren**

134 Der Befehl „-VorgangArchivDrucken“ PRO

ERPAdmin -VorgangArchivDrucken <VogArtNr> <DruckLayout> /Von: /Bis:<Bereich bis> /Filter:<Filterbedingung> /Drucker:<Druckername> /Datei:<Dateiname>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Beispiel:

Gibt den Beleg RE1812345 auf dem hinterlegten Drucker aus.
ERPAdmin -VorgangArchivDrucken 70 Rechnung /Von:BelegNr=RE1812345

Die Funktionsweise ist analog zum Befehl „-**VorgangDrucken**“ jedoch bezieht sich die Druckausgabe nur auf die Vorgänge im Bereich **Archiv-Vorgänge**.

135 Der Befehl „-VorgangArchivDruckenDatei“ PRO

ERPAdmin -VorgangArchivDruckenDatei <VorgangArtNr> <DruckLayout> <DateiSpaltenBezeichner> <ERP-FeldBezeichner> <CSV-Datei> <DoubletFile> /Filter:<Filterbedingung> /Drucker:<Druckername> /Datei:<Dateiname>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Die Funktionsweise ist analog zum Befehl „-**VorgangDruckenDatei**“ jedoch bezieht sich die Druckausgabe nur auf die Vorgänge im Bereich **Archiv-Vorgänge**.

136 Der Befehl „-VorgangDrucken“ PRO

ERPAdmin -VorgangDrucken <VogArtNr> <DruckLayout> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /Drucker:<Druckername> /Datei:<Dateiname>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl druckt **ERPAdmin** die Vorgänge mit der Vorgangs-Art-Nummer **VogArtNr** aus.

<VogArtNr>
Gibt an von welcher Vorgangsart eine Druckausgabe erfolgen soll. Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.**

	15		Angebot
	20		Auftragsbestätigung
	30		Bestellung vom Kunden
	40		Bestellung an Lieferant
	41		Eingangsrechnung
	50		Lieferschein
	70		Rechnung
	90		Gutschrift

Klartextnamen der Vorgänge sind nicht zugelassen.

Mehrere Vorgangsarten können durch Angabe eines Semikolons getrennt übergeben werden. Im **Drucklayout** und im **Dateinamen** können die Variablen **@VORGANGARTEN.BEZ@** und **@VORGANGARTEN.KUBEZ@** verwendet werden um **Drucklayout** und **Dateinamen** dynamisch je Vorgangsart anzupassen.

<DruckLayout>
DruckLayout gibt den Namen des Layouts an das zum Drucken verwendet werden soll. Es sind hier nur Drucklayouts zulässig die auch in der Vorgangsart zur Verfügung stehen.

/Datei:<Dateiname> **(optional)**
Die Druckausgabe kann auch direkt in eine Datei umgeleitet werden. Im Dateinamen können die Platzhalter **@BELEGNR@**, **@AUFTRNR@** und **@DAT@** verwendet werden welcher bei der Ausgabe mit der Belegnummer, Auftragsnummer oder dem Vorgangsdatum des Vorgangs ersetzt werden. Der Dateiname muss inkl. Pfadangabe übergeben werden.

Achtung:
Wird der Platzhalter **@AuftrNr@** verwendet, muss bei der Erfassung der Auftragsnummer in büro darauf geachtet werden, dass nur Zeichen verwendet werden welche auch in einem Dateinamen zulässig sind.

Nicht zulässig sind z.B. folgende Zeichen: **? : < > * " / **

Anhand der Endung erkennt **ERPAdmin** um welches Dateiformat es sich handelt. Die folgenden Endungen werden unterstützt:

- .LL List & Label
- .XLS Excel Format
- .PDF Adobe Portable Document Format
- .RTF Rich Text Format
- . HTML Hypertext Markup Language
- .MHTML Mimetype Hypertext Markup Language
- .XML Extendable Markup Language
- .JPG JPG Bildformat (kann mehrere Dateien erstellen
- .BMP BMP Bildformat (kann mehrere Dateien erstellen
- .EMF EMF Bildformat (kann mehrere Dateien erstellen
- .TIF Multipage TIFF Bildformat
- .TXT Text basierter Export
- .XLSX Microsoft Excel im neuen Format
- .DOC Microsoft Word
- .PPT Microsoft PowerPoint

Wurde in der **VAR.INI** ein Historie-Verzeichnis angegeben (**C_HISTORYDIR=**) und wurde die Druckausgabe in eine Datei umgeleitet (**/Datei:**) werden alle Dateien in das angegebene Verzeichnis kopiert.

> **Hinweis:**
> Die Option **/Datei:** darf nicht zusammen mit der Option /**Drucker:** verwendet werden.

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Die zu druckenden Vorgänge können über die Werte **/Von:** bis **/Bis:** eingeschränkt werden. Wird kein Bereich angegeben, werden alle Vorgänge der gewählten Art gedruckt.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/Drucker:<Druckername> **(optional)**
Für die Ausgabe des Drucklayout **DruckLayout** kann ein abweichender Drucker verwendet werden. Es muss der Druckername angegeben werden unter dem der Drucker in der Systemsteuerung angezeigt wird.

Eine Liste der verfügbaren Drucker kann mit dem Befehl **ERPAdmin -ListPrinter** erzeugt werden.

Beispiele

Sie möchten alle Rechnungen aus den Vorgängen mit dem Layout „**Rechnung_neutral**“ drucken.
ERPAdmin -VorgangDrucken 70 Rechnung_neutral

Zusätzlich können Sie die Rechnungen auch über die Belegnummer eingrenzen.
ERPAdmin -VorgangDrucken 70 Rechnung_neutral /Von:BelegNr=RE16000 /Bis:BelegNr=RE16999

Das Angebot AN180001 soll in eine PDF-Datei umgeleitet werden.
ERPAdmin -VorgangDrucken 15 Angebot /Von:BelegNr=AN180001 /Datei= c:\ERPAdmin\Angebot.pdf

Werden mit einem Aufruf mehrere Angebote in eine Datei umgeleitet Nummeriert **ERPAdmin** diese eigenständig durch. Soll an Stelle der aufsteigenden Nummerierung die Belegnummer im Dateinamen verwendet werden kann dies durch einfügen des Platzhalters **@BELEGNR@** erzwungen werden.
ERPAdmin -VorgangDrucken 15 Angebot /Datei:c:\ERPAdmin\Angebot@BELEGNR@.pdf

Die Druckausgabe der Angebote soll auf den Drucker „**HP OfficeJet x476**“ umgeleitet werden.
ERPAdmin -VorgangDrucken 15 Angebot "/Drucker:HP OfficeJet x746"

Es sollen alle **Angebote (AN), Auftragsbestätigungen (AB)** und **Rechnungen (RE)** des Tages aus dem Vorgangs-Archiv als PDF in dem Verzeichnis ./PDF/ gespeichert werden unter dem Dateinamen **„<Vorgangs-Bezeichnung>_<Belegnummer>.pdf“**. Dazu wurde für jede Vorgangsart ein Drucklayout „**XX-PDF**“ erstellt wobei **XX** die Kurzbezeichnung der jeweiligen Vorgangsart entspricht.
ERPAdmin -VorgangArchivDrucken 15;20;70 "@VORGANGARTEN.KUBEZ@-PDF" /Filter:Dat=@DATE@ "/Datei:@VORGANGARTEN.BEZ@_@BELEGNR@.PDF

Hinweis:
Um Vorgänge über **/Von:** und **/Bis:** einzuschränken können alle Suchfelder verwendet werden die von **büro+** zur Vorgangsselektion verwendet werden und die einen eigenen Suchindex besitzen. Dies sind z.B.: **BelegNr, AuftrNr, Suchbeg, AdrNr, ProjNr**.

137 Der Befehl „-VorgangDruckenDatei“ PRO

ERPAdmin -VorgangDruckenDatei <Vorgang-ArtNr> <DruckLayout> <DateiSpaltenBezeichner> <büro+FeldBezeichner> <CSV-Datei> <Doubletten-Datei> /Filter:<Filterbedingung> /Drucker:<Druckername> /Datei:<Dateiname>

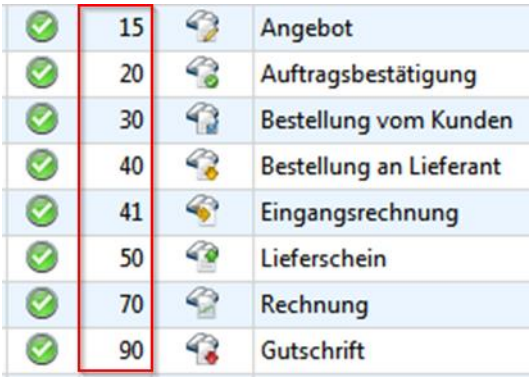
Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Sie können eine Liste von Vorgängen ausgeben die in einer **CSV-Datei** übergeben werden. Dabei ist egal ob es sich um einen Druck oder um den Versand per E-Mail handelt.

Wenn Sie z.B. vom Versanddienstleister eine Liste mit den versendeten Trackingnummern bekommen, können Sie die dazugehörigen Versandbestätigungen mit dem Anhang der Rechnung aus **büro+** per E-Mail mit **ERPAdmin** versenden lassen. Dazu wird eine **CSV-Datei** benötigt welche die **Belegnummer** der Vorgänge enthält.

Ausgabe eines zusätzlichen Dokumentes nach dem Druck eines Vorgangs
Bei der Druckausgabe auf einen Drucker kann nach jedem Vorgangsdruck ein weiteres Dokument (PDF, DOC oder DOCX) zusätzlich ausgegeben werden. Im Drucker liegen diese Dokumente dann korrekt sortiert hintereinander. Dazu muss in der **CSV-Datei** eine zusätzliche Spalte mit dem Bezeichner „**@PrintFileName@“** erzeugt werden. In dieser Spalte stehen dann die entsprechenden Dokumentennamen inkl. Pfadangabe.

<Vorgang-ArtNr>
Gibt an von welcher Vorgangsart eine Druckausgabe erfolgen soll. Die **Vogang-ArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.**



15	Angebot	
20	Auftragsbestätigung	
30	Bestellung vom Kunden	
40	Bestellung an Lieferant	
41	Eingangsrechnung	
50	Lieferschein	
70	Rechnung	
90	Gutschrift	

Klartextnamen der Vorgänge sind **nicht** zugelassen.

<DruckLayout>
DruckLayout gibt den Namen des Layouts an das zum Drucken verwendet werden soll. Es sind hier nur Drucklayouts zulässig die auch in der Vorgangsart zur Verfügung stehen.

<DateiSpaltenBezeichner>
Name der Spaltenüberschrift der **CSV-Datei** aus der ein Wert ausgelesen werden soll.
Beispiel: **KundeRef**

<büro+Feldbezeichner>
Variablenname aus **büro+** mit dem der Wert aus der CSV-Datei verglichen werden soll. Das Präfix **Vog.** der Variable „**Vog.BelegNr**“ ist dabei zu ignorieren.
Beispiel: **BelegNr** statt **Vog.BelegNr**
AuftrNr statt **Vog.AuftrNr**

<CSV-Datei>
Dateiname inkl. Pfadangabe in der sich die Liste mit Daten für den Vorgangs-Druck befinden.

<Doubletten-Datei> **(optional)**
Dateiname inkl. Pfadangabe in dem alle Belegnummern protokolliert werden. Vor jeder Ausgabe prüft **ERPAdmin** ob die Belegnummer in der Datei **Doubletten-Datei** bereits vorhanden ist. Ist dies der Fall wird due Ausgabe übersprungen. Weitere Informationen zur Doubletten-Prüfung entnehmen Sie bitte dem Kapitel „**Doubletten-Prüfung**“.
Beispiel: **C:\ERPAdmin\Doubletten_Vorgang_Drucken.log**

/FILTER:<Filterbedingung> **(optional)**
Hier kann ein zusätzlicher Filterausdruck in der Notation von **büro+** übergeben werden.

/Drucker: <Druckername> **(optional)**
Für die Ausgabe des **DruckLayouts** kann ein abweichender Drucker verwendet werden. Es muss der Druckername angegeben werden unter dem der Drucker in der Systemsteuerung angezeigt wird.

Eine Liste der verfügbaren Drucker kann mit dem Befehl **ERPAdmin -ListPrinter** erzeugt werden.

/Datei:<Dateiname> **(optional)**
ERPAdmin leitet die Druckausgabe in die angegebene Datei um, auch wenn im Drucklayout keine Dateiumleitung hinterlegt wurde. Anhand der Datei-Endung erkennt **ERPAdmin** um welches Datei-Format es sich handelt. Als Platzhalter müssen **@BELEGNR@** oder **@AUFTRNR@** verwendet werden. Es muss mind. einer der beiden Platzhalter verwendet werden, wenn ein Dateiname übergeben wird.

> **Achtung:**
> Wird der Platzhalter **@AuftrNr@** verwendet, muss bei der Erfassung der Auftragsnummer in büro darauf geachtet werden, dass nur Zeichen verwendet werden welche auch in einem Dateinamen zulässig sind.
>
> Nicht zulässig sind z.B. folgende Zeichen: **? : < > * " / **

Dateinamen für Ergebnis-Datei aus CSV-Datei verwenden
Es besteht auch die Möglichkeit, den Dateinamen direkt aus einer Spalte der **CSV-Datei** mit zu übergeben. Dazu muss eine Spalte z.B. „**@Dateiname@“** vorhanden sein mit einem eindeutigen Namen und die Option **/Datei:@Dateiname@** beim Aufruf von **ERPAdmin**. Die Verwendung der Platzhalter **@BELEGNR@** und **@AUFTRNR@** sind in diesem Fall nicht zulässig.

Beispiel: **/Datei:@Dateiname@**
In diesem Beispiel würde der Dateiname aus der Spalte „**@Dateiname@**“ der CSV-Datei verwendet.

Folgende Dateiendungen werden unterstützt:

- .LL List & Label
- .XLS Excel Format
- .PDF Adobe Portable Document Format
- .RTF Rich Text Format
- . HTML Hypertext Markup Language
- .MHTML Mimetype Hypertext Markup Language
- .XML Extendable Markup Language
- .JPG JPG Bildformat (kann mehrere Dateien erstellen
- .BMP BMP Bildformat (kann mehrere Dateien erstellen
- .EMF EMF Bildformat (kann mehrere Dateien erstellen
- .TIF Multipage TIFF Bildformat
- .TXT Text basierter Export
- .XLSX Microsoft Excel im neuen Format
- .DOC Microsoft Word
- .PPT Microsoft PowerPoint

Hinweis:
Bitte beachten Sie, dass je nach Ausbaustufe und Programmversion der **büro+** die zur Verfügung stehenden Formate unterschiedlich sein können. Zusätzlich können **büro+** Benutzern die Rechte fehlen um ein vorhandenes Format zu benutzen.

Beispiel:

In den CSV-Dateien „**Tracking_xxxxxx.csv**“ werden mehrmals am Tag neben der Trackingnummer auch die Belegnummer und noch einige andere Daten vom Versender zurückgegeben. **xxxxx** ist dabei eine fortlaufende Nummer. Die Spalte mit den Belegnummern der Rechnungen aus büro+ trägt den Namen „**KundenRef**“.

Pro Rechnung (ArtNr=70) soll eine E-Mail an den Kunden verwendet werden und eine Rechnungskopie als PDF im Anhang versendet werden. Das zu verwendende Drucklayout heißt „**Rechnung_Email**“

ERPAdmin -VorgangDruckenDatei 70 Rechnung_Email KundenRef BelegNr "C:\ERPAdmin\Tracking*.csv"

Soll sichergestellt werden, dass jeder Vorgang nur einmal ausgegeben wird, muss der Name einer eine **Doubletten-Datei** übergeben werden.

ERPAdmin -VorgangDruckenDatei 70 Rechnung_Email KundenRef BelegNr " " "C:\ERPAdmin\Tracking*.csv" "C:\ERPAdmin\Doubletten.log"

/MoveToHis, /MoveToBad, /Delete **(optional)**
Damit **CSV-Dateien** nicht doppelt verarbeitet werden können die Optionen **/MoveToHis** und **/MoveToBad** mit übergeben werden.

Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“.

138 Der Befehl „-VorgangErzeugen“ PRO

ERPAdmin -VorgangErzeugen <VogArtNr> <AdrNr> <Artikelliste>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Manchmal ist es wünschenswert einen Vorgang mit minimalen Angaben zu erzeugen, weil alle wesentlichen Daten bereits in den Stammdaten von **büro+** vorhanden sind. Dazu dient dieser Befehl. Es werden lediglich die Vorgangs-Art, die Adressnummer und jeweils Artikel-Nummer und Menge einer Position benötigt. Alle anderen Daten zieht sich **büro+** aus den Stammdaten.

<VogArtNr>
Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.**

	15		Angebot
	20		Auftragsbestätigung
	30		Bestellung vom Kunden
	40		Bestellung an Lieferant
	41		Eingangsrechnung
	50		Lieferschein
	70		Rechnung
	90		Gutschrift

<AdrNr>
Adressnummer für die der Vorgang erzeugt werden soll.
Beispiel: **11500**

<Artikelliste>
Auflistung von Artikel die im neu erzeugten Vorgang (**VogArtNr**) hinzugefügt werden sollen. Mehrere Artikelnummern sind durch ein Semikolon zu trennen.

Zusätzlich kann die Menge mit übergeben werden, wenn es sich um mehr als einen Artikel handelt. Dazu ist zuerst die Artikelnummer anzugeben und dann durch ein Gleichheitszeichen getrennt die gewünschte Menge. Leerzeichen sind NICHT erlaubt.

Beispiel:
Der Artikel 30100 soll mit Menge 2 hinzugefügt werden.
Der Artikel 45600soll mit Menge 5 hinzugefügt werden.
Der Artikel 59101 soll mit Menge 1 hinzugefügt werden.
30100=2;45600=5;59101=1

ERPAdmin -VorgangErzeugen 30 11500 "30100=2;45600=5;59101=1"
Es wird eine „**Bestellung vom Kunden**“ (**30**) für die Adressnummer **11500** erzeugt mit den angegebenen Artikeln (30100, 45600, 59101) und Mengen (2, 5, 1).

139 Der Befehl „-VorgangImport“ PRO

ERPAdmin -VorgangImport <CSV-Datei> <Zeile 1. Datensatz> <CSV-Beschreibungsdatei> <Vorgang ArtNr> <Vorgabe AdrNr> <Vorgabe Einheit> <Wechsel Spalte> <DoubletFile> /Buchen /Drucken:<Drucklayout> /Drucker:<Druckername> /Datei:<Dateiname> /NoAdrNrNotExistError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Beim CSV-Import von Vorgängen kann es vorkommen, dass die Daten nicht in einem asynchronen Format geliefert werden wo sich Kopf- und Fußzeilen abwechseln, sondern wo sich in jeder Zeile sowohl Vorgangs- als auch Positions-Daten befinden. Solche Daten kommen häufig aus Webshops oder aus anderen ERP-Systemen.

ERPAdmin kann diese Daten direkt importieren ohne dass eine Aufbereitung oder Konvertierung durchgeführt werden muss. In der CSV-Datei müssen folgende Feldwerte enthalten sein:

- **Vog.AdrNr** (Vorgangs-Adress-Nummer)
- **VPo.ArtNr** (Artikelnummer)

Diese Daten reichen aus um einen Vorgangsimport durchführen zu können. Es können weitere Daten übergeben werden. Alle Daten die nicht aus der CSV-Datei importiert werden, werden aus den Stammdaten der Adressen und Artikel verwendet.

Damit **ERPAdmin** weiß welche Felder sich in der CSV-Datei befinden, muss zunächst eine Datei mit der gleichen Anzahl an Spalten erstellt werden, welche als Werte die Import-Variablen aus **büro+** besitzt. Dabei dürfen nur die eigentlichen Bezeichner angegeben werden ohne die Anfangs- und Endmarkierungen aus dem Import-Layout.

Achtung:
Die **Infoblattbezeichner** eines Vorgangs befinden sich nicht direkt in der Vorgangsdatenbank und können nicht auf diesem Weg importiert werden.

Funktionen wie **Zwischensummen, Artikel-Reservieren, Artikel zurücklegen, etc.** können nicht auf diesem Weg in einem Vorgang erzeugt werden.

Beispiel:
Vog.AuftrNr;Vog.AdrNr;Vog.Li.Na2;Vog.Li.Str;Vog.Li.PLZ;Vog.Li.Ort;VPo.ArtNr;Vpo.Mge

Diese Daten werden in einer Beschreibungsdatei abgespeichert. Die Datei darf nur aus dieser einen Zeile bestehen. Sollen einzelne Spalten ignoriert werden ist die Spaltenüberschrift **DUMMY** zu verwenden.

Anhand der Präfixe „**Vog.**“ und „**VPo.**“ erkennt **ERPAdmin,** ob es sich um die Daten für den Vorgangskopf oder die Vorgangspositionen handelt.

<CSV-Datei>
Name der CSV-Datei die Importiert werden soll. Ist keine Datei für einen Import vorhanden wird **keine** Fehlermeldung erzeugt.

<Zeile 1. Datensatz>
Enthält die **CSV-Datei** eine Kopfzeile mit den original Feld-Bezeichner beginnt der Importbereich mit der 2. Zeile. In diesem Fall ist der Wert „**2**“ zu übergeben. Besitzt die CSV-Datei **KEINE** Kopfzeile übergeben Sie den Wert „**1**“. Wollen Sie die ersten 10 Zeilen überspringen übergeben Sie den Wert „**11**“.

<CSV-Beschreibungsdatei> **(bedingt optional)**
CSV-Datei in der die Spaltenbezeichnungen von **büro+** übergeben werden. Wird keine Beschreibungsdatei mit Überschriften benötig, weil die Überschriften bereits in der Import-Datei vorhanden sind, kann dieser Wert als Leerzeichen **" "** übergeben werden.

<Vorgang ArtNr>
Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.** Sie gibt an, welchen Vorgang Sie importieren wollen.

	15		Angebot
	20		Auftragsbestätigung
	30		Bestellung vom Kunden
	40		Bestellung an Lieferant
	41		Eingangsrechnung
	50		Lieferschein
	70		Rechnung
	90		Gutschrift

<Vorgabe AdrNr>
Hier wird eine Default-Adresse definieret falls in der Datei an der erwarteten Stelle keine Adressnummer steht oder eine ungültige Adressnummer. In der Warenwirtschaft können diese Vorgänge per Farbregel schnell identifiziert werden.

<VorgabeEinheit>
Geben Sie hier eine Default Einheit an, welche immer dann verwendet wird, wenn in der Import-Datei keine Einheit übergeben wurde. Wenn Sie keinen Wert übergeben also **" "** wir die Einheit aus den Artikelstammdaten verwendet.

Falls Sie die Einheiten konvertieren möchten, können Sie in der **VAR.INI** eine Ersetzungsliste hinterlegen.

C_VorgangImportEinhErsetzung **"10=kg;20=Stk;30=liter"**

Mit obiger Liste werden die Werte wie in der Liste angegeben vor dem Import ersetzt

<Wechsel Spalte> **(optional)**
Damit **ERPAdmin** erkennt, wann ein neuer Vorgang beginnt muss ein Spaltennamen angegeben werden. Aufeinanderfolgende gleich lautenden Einträge gehören zum gleichen Vorgang. Ändert sich der Wert, wird ein neuer Vorgang erzeugt. Wird kein Wert angegeben, werden alle Datensätze der **CSV-Datei** in denselben Vorgang importiert.

Hinweis:
Es ist nicht vorteilhaft, wenn Daten für mehreren Vorgängen in derselben **CSV-Datei** übergeben werden. Kommt es während des Importes zu einem Fehler muss die komplette CSV-Datei ggf. neu importiert. Bereits Importierte Vorgänge wären dann ggf. doppelt vorhanden.

Dies kann vermieden werden durch Sperrung der Vorgangstabelle für den Import. Dazu muss in der Datei **VAR.INI** der folgende Eintrag hinzugefügt werden: **C_BP_LockTables=TRUE**

<DoubletFile > **(optional)**
Dateiname inkl. Pfadangabe in dem alle importierten Dateien protokolliert werden. Vor jedem Import prüft **ERPAdmin** ob die zu importierende Datei in **DoubletFile** vorhanden ist und verschiebt in diesem Fall die Datei nach **C_BADDIR** wenn der Parameter **/MoveToBad** übergeben wurde. Weitere Informationen zur Doubletten-Prüfung entnehmen Sie bitte dem Kapitel „**Doubletten-Prüfung**“
Beispiel: **C:\ERPAdmin\Doubletten_Import.log**

/BUCHEN **(optional)**
Wird die Option /**BUCHEN** übergeben, wird der Vorgang nach dem Erzeugen automatisch verbucht.

/Drucken:<Drucklayout> **(optional)**
Sollen die importierten Vorgänge ausgegeben werden, kann ein **Drucklayout** übergeben werden. Diese Option darf nicht zusammen mit der Option **/BUCHEN** verwendet werden.

/Drucker:<Druckername> **(optional)**
Soll die Druckausgabe auf einem Abweichenden Drucker erfolgen, kann hier der entsprechende Drucker **(Druckername)** übergeben werden. Die genaue Schreibweise der Drucker kann mit dem Befehl **ERPAdmin -ListDrucker** ausgelesen werden.

/Datei:<Dateiname> **(optional)**
Das Ergebnis der Druckausgabe kann in eine Datei umgeleitet werden. Der **Dateiname** muss den Platzhalter **@BELEGNR@** enthalten damit die Ausgabe eindeutig ist. Diese Option muss zusammen mit der Option**/Drucken:<Drucklayout>** verwendet werden.

/NoAdrNrNotExistError **(optional)**
Wurde eine Adress-Nummer in der **CSV-Datei** gefunden welche **nicht** in büro+ existiert, wird **keine** Fehlermeldung an den Benutzer zurückgegeben. Der Vorgang wird mit der **Vorgabe AdrNr** erzeugt.

Beispiele:

Die Datei „**WebShopImport.csv**“ soll als „Bestellung vom Kunden“ (Vorgang ArtNr=30) importiert werden. In der 1. Zeile befinden sich noch alte Überschriften welche ignoriert werden sollen. Die Spaltenbezeichnungen wurden in der Datei „**ImportLayout.csv**“ wie folgt gespeichert:

Vog.AuftrNr;Vog.AdrNr;Vog.Li.Na2;Vog.Li.Str;Vog.Li.PLZ;Vog.Li.Ort;VPo.ArtNr;Vpo.Mge;DUMMY

Falls keine Adressnummer in der CSV-Datei übergeben wurde, soll die Adress-Nummer **60000** verwendet werden. Als Einheit soll „**Stk**“ verwendet werden falls keine Einheit in der CSV-Datei übergeben wurde. Anhand der Werte in der Spalte **AuftrNr** soll erkannt werden welche Datensätze zum gleichen Vorgang gehören.

ERPAdmin -VorgangImport WebShopImport.csv 2 ImportLayout.csv 30 60000 Stk Vog.AuftrNr

Die Datei „**WebShopImport.csv**“ enthält folgende Daten:

KD-Ref.Nr.	KD-Nr	Lief.Ansch	Lief.Str	Lief.Plz	Lief.Ort	ArtikelNr	Menge	Preis
10013455	14539	Hans Meiser	Bachstr. 7	63322	Rödermark	12350	1	19,5
10013455	14539	Hans Meiser	Bachstr. 7	63322	Rödermark	23455	2	7,99
10013455	14539	Hans Meiser	Bachstr. 7	63322	Rödermark	25343	1	14,5
10013456	23441	Nina Baum	Kornweg 4	63500	Mainhausen	15456	2	29,9
10013456	23441	Nina Baum	Kornweg 4	63500	Mainhausen	17554	1	19,5

Mit obigem Import werden zwei Vorgänge in **büro+** erzeugt. Einer mit drei Positionen für den Kunden „**Hans Meiser**“ aus Rödermark und ein Vorgang mit zwei Positionen für den Kunden „**Nina Baum**“.

Datum	Nummer ▲	Auftragsnummer	Art	Anschrift
06.03.2017	BK1700002	10013455	Bestellung vom Kunden	14539 Herr Hans Meiser - Bachstr. 7, 63322 Rödermark
06.03.2017	BK1700003	10013456	Bestellung vom Kunden	23441 Herr NIna Baum - Kornweg 4, 63500 Mainhausen

<u>Weitere Beispiele:</u>

Alle Artikel der Datei „**WebShopImport.csv**“ sollen in einen Vorgang importiert werden.
ERPAdmin -VorgangImport WebShopImport.csv 2 ImportLayout.csv 30 60000 Stk

Alle Artikel der Datei „**WebShopImport.csv**“ erfolgt ohne Beschreibungsdatei weil alle Überschriften bereits in der Import-Datei vorhanden sind.
ERPAdmin -VorgangImport WebShopImport.csv 1 " " 30 60000 Stk

<u>/MoveToHis, /MoveToBad, /Delete</u> **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“.

140 Der Befehl „-VorgangInArchiv“ STD

ERPAdmin -VorgangInArchiv <VogArtNr> <BuchKZ> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

Mit diesem Befehl verschiebt **ERPAdmin** Vorgänge der **VogArtNr** in das Vorgangs-Archiv. Gesperrte Vorgänge werden NICHT verschoben.

< VogArtNr>
Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.** Sollen alle Vorgänge verschoben werden kann der Wert **0** übergeben werden.

✓	15		Angebot
✓	20		Auftragsbestätigung
✓	30		Bestellung vom Kunden
✓	40		Bestellung an Lieferant
✓	41		Eingangsrechnung
✓	50		Lieferschein
✓	70		Rechnung
✓	90		Gutschrift

Klartextnamen der Vorgänge sind nicht zugelassen.

<BuchKZ>
TRUE: Gibt an, dass die Vorgänge gebucht sein müssen bevor sie verschoben werden, **FALSE** sonst.

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Die zu verschiebenden Vorgänge können über die Werte **/Von:<BereichsVon>** bis **/Bis:<BereichBis>** eingeschränkt werden. Wird kein Bereich angegeben, werden alle Vorgänge der gewählten Art verschoben.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Beispiele:

Sie möchten alle gebuchten Rechnungen aus den Vorgängen in das Archiv verschieben.
ERPAdmin -VorgangInArchiv 70 TRUE

Alle Rechnungen unabhängig von Ihrem Buchungsstatus sollen ins Archiv
ERPAdmin -VorgangInArchiv 70 FALSE

Alle Auftragsbestätigungen mit den Auftragsnummern **AUF16000** bis **AUF16999** sollen ins Archiv verschoben werden ungeachtet des Buchungskennzeichens.
ERPAdmin -VorgangInArchiv 20 FALSE /Von:AuftrNr=AUF16000 /Bis:AuftNr=AUF16999

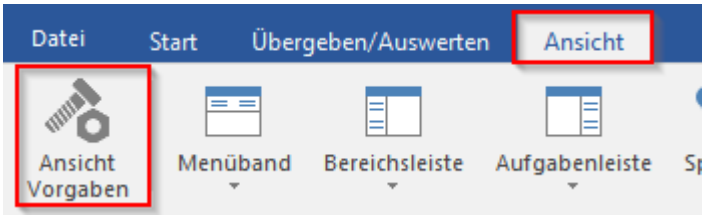
Verschiebe alle Angebote aus dem Jahr **2016** ins das Archiv.
ERPAdmin -VorgangInArchiv 15 FALSE /Von:Dat=01.01.2016 /Bis:Dat=31.12.2016

Hinweis: Um Vorgänge über **/Von:** und **/Bis:** einzuschränken können alle Suchfelder verwendet werden die von **büro+** zur Vorgangsselektion verwendet werden und die einen eigenen Suchindex besitzen. Dies sind: **BelegNr**, **AuftrNr**, **Suchbeg**, **AdrNr, ProjNr.**

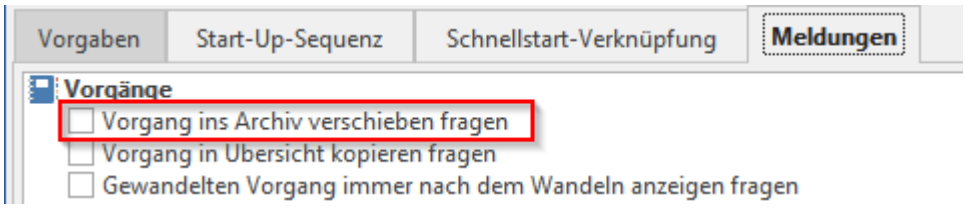
Hinweis:

Vor dem ersten Verschieben von Daten in einen Archiv- oder History-Bereich muss in **büro+** die entsprechende Sicherheitsmeldung für den COM-Benutzer deaktiviert werden.

Melden Sie sich dazu mit dem COM-Benutzer an und wechseln Sie in das Menü „**Ansicht**“ -> „**Ansicht Vorgaben**“.



Suchen Sie hier den Eintrag im Bereich „**Vorgänge**“-> „**Vorgang ins Archiv verschieben fragen**“ und **deaktivieren** Sie diesen.



Erst jetzt kann das Verschieben über COM durchgeführt werden!

141 Der Befehl „-VorgangPositionAdd“ PRO

ERPAdmin -VorgangPositionAdd <Artikel-Nr> <Bedingung> <Vorgang-Feldzuweisung> <DoubletFile> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /NoLockError /AbrPosKz /Einheit:<Einheit> /Menge:<Menge> /Max:<Max Anzahl>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin überprüft für alle Vorgänge im Bereich **/Von…. /Bis… /Filter…** ob **Bedingung** erfüllt ist und hängt in diesem Fall den Artikel **Artikel-Nr** an den Vorgang mit **Menge** an, wenn der Artikel nicht bereits in **Max. Anzahl Positionen** (nicht Menge) im Vorgang vorhanden ist.

<Artikel-Nr>
Artikel-Nummer des Artikels der bei zutreffender **Bedingung** als letzte Position an den Vorgang angehängt werden soll.

<Bedingung> **(optional)**
Die Bedingung besteht aus einem Feldbezeichner aus dem Bereich der Vorgangs-Positionen z.B. **Sel5** (Lieferdatum)**,** einem Vergleichs-Operator z.B. **„=“ „<>“ „<“, „>“, „<=“, „>=“** und einem Vergleichswert z.B. **4500**. Die Verknüpfung von mehreren Bedingungen mit **AND** oder **OR** wird nicht unterstützt. Die Verwendung von Anführungszeichen oder einfachen Hochkommas ist nicht erlaubt.

Wird keine Bedingung angegeben wird der Artikel immer hinzugefügt.

Beispiel-Bedingung:

Sel5<=4500

oder

ArtNr=4347

<Vorgang-Feldzuweisung> **(optional)**
Schreibt **Vorgang-Feldzuweisung** in jeden gefunden Vorgang unabhängig ob ein Artikel hinzugefügt wurde oder nicht. Damit kann die erneute Verarbeitung zu einem späteren Zeitpunkt unterbunden oder auch ein Kennzeichen zur Weiterverarbeitung gesetzt werden. Mehrere Feldzuweisungen sind durch ein Semikolon zu trennen. Dem Feldbezeichner muss das Prefix **Vog.** vorangestellt werden.

Beispiel: Vog.ErledigtKz=1

<DoubletFile > **(optional)**
Dateiname inkl. Pfadangabe in dem alle Belegnummern protokolliert werden die verarbeitet wurden. Vor dem Hinzufügen eines Artikels prüft **ERPAdmin,** ob der Vorgang bereits in der Datei **DoubletFile** vorhanden ist. So kann sichergestellt werden, dass jeder Vorgang nur einmal verarbeitet wird. Wurde ein Vorgang einmal von **ERPAdmin** angefasst, egal ob ein Artikel hinzugefügt wurde oder nicht, wird dieser Vorgang kein zweites Mal verarbeitet.

/Von:<Bereich von> und **/Bis:<Bereich von>** **(optional)**
Die zu verarbeitenden Vorgänge können über die Werte **/Von:<BereichVon>** bis **/Bis:<BereichBis>** eingeschränkt werden. Wird kein Bereich angegeben, werden alle Vorgänge verarbeitet.

Beispiel: /Von:BelegNr=BL1900000 /Bis:BelegNr=BL1909999

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel „**Benutzung von Filtern**“ weiter oben beschrieben.

/AbrPosKz **(optional)**
Sollen nur abrechenbare Vorgangs-Positionen beachtet und Positionen wie „**Geliefert**“, „**Alternativ**“, „**Optional**“, etc. nicht beachtet werden, muss die Option **/AbrPosKz** verwendet werden.

Ist die Option gesetzt, werden nur Vorgangspositionen berücksichtigt die das Abrechnungskennzeichen „**VPo.AbrPosKz**“ gesetzt haben. Alle anderen Positionen werden ignoriert.

Ist die Option **NICHT** gesetzt, werden alle Positionen berücksichtig ungeachtet vom Abrechnungs-Positions-Kennzeichen.

/NoLockError **(optional)**
Kann ein Vorgang nicht bearbeitet werden, weil diese gerade von einem anderen Benutzer in Verwendung ist, wird keine Fehlermeldung zurückgegeben.

/Einheit:<Einheit> **(optional)**
Überschreibt die Einheit aus den Artikelstammdaten.

/Menge:<Menge> **(optional)**
Gibt die Menge für den angehängten Artikel an. Wurde kein Wert übergeben wird „**1**“ als Default verwendet.

/Max:<Max Anzahl Positionen> **(optional)**
ERPAdmin zählt die Anzahl der Vorgangspositionen bei denen die Bedingung erfüllt ist. Nur wenn die **Max. Anzahl Positionen** noch nicht erreicht ist, wird der Artikel hinzugefügt. Wird kein Wert übergeben wird der Wert **999999** verwendet was zu führt, dass immer der Artikel **Artikel-Nr** hinzugefügt wird.

Beispiele:

ERPAdmin -VorgangPositionAdd A1456 /Menge:1 /Von:BelegNr=AN190000 /Bis:BelegNr=AN190000
Fügt im Vorgang mit den Belegnummern **AN190000** den Artikel **A1456** mit Menge **1** hinzu.

ERPAdmin -VorgangPositionAdd A1456 /Menge:2 /Von:BelegNr=AN190000 /Bis:BelegNr=AN199999
Läuft über alle Vorgänge mit den Belegnummern **AN190000** bis **AN199999** und fügt den Artikel **A1456** mit Menge **2** hinzu.

ERPAdmin -VorgangPositionAdd A1456 ArtNr=A2121 /Menge:2 /Von:BelegNr=AN190000 /Bis:BelegNr=AN199999
Läuft über alle Vorgänge mit den Belegnummern **AN190000** bis **AN199999** und fügt den Artikel **A1456** mit Menge **2** hinzu, wenn in dem Vorgang der Artikel **A2121** in min. **1** Position vorhanden ist.

ERPAdmin -VorgangPositionAdd A1456 ArtNr=A4313 Vog.ErledigtKz=1 /Menge:2 /Von:BelegNr=AN190000 /Bis: BelegNr=AN199999 /Filter:ErledigtKz<>1
Läuft über alle Vorgänge im Bereich **AN190000** bis **AN199999** (bei denen **ErledigtKz <> 1**) und fügt den Artikel **A1456** mit Menge **2** hinzu, wenn in dem Vorgang der Artikel **A4313** in min. **1** Position vorhanden ist. Anschließend wird das ErledigtKennzeichen gesetzt (**ErledigtKz=1**).

ERPAdmin -VorgangPosition A1456 Add ArtNr=A4313 Vog.ErledigtKz=1 Vorgang.log /Menge:2 /Von:BelegNr=AN190000 /Bis: BelegNr=AN199999 /Filter:ErledigtKz<>1
Läuft über alle Vorgänge im Bereich **AN190000** bis **AN199999** (bei denen Vog.**ErledigtKz <> 1**) und fügt den Artikel **A1456** mit Menge **2** hinzu, wenn in dem Vorgang der Artikel **A4313** in min. **1** Position vorhanden ist. Anschließend wird das Erledigt-Kennzeichen gesetzt (**ErledigtKz=1**). Aller verarbeiteten Vorgänge werden in der Datei **Vorgang.log** protokolliert.

142 Der Befehl „-VorgangPositionAddFromFile“ PRO

ERPAdmin -VorgangPositionAddFromFile <CSV-Datei> /DeletePos /NoLockError /Filter:<Filterbedingung> /NoSort /MoveToHis /MoveToBad /Delete /GebuchtKz

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

ERPAdmin fügt aus einer **CSV-Datei** Artikel in die Vorgangs-Positionen von bereits existierenden Vorgängen ein. Im Gegensatz zu einem Import werden dabei alle Artikel-Daten wie Zusätze, Rabatte, Abweichende Preise etc. im Vorgang mit erzeugt die bei einem Import nicht eingefügt werden. Die Artikel aus der **CSV-Datei** werden dabei an das Ende der existierenden Positionen angehängt.

Das Einfügen neuer Vorgangs-Positionen ist nur bei **ungebuchten** Vorgängen möglich. Sonst muss die Option **/GebuchtKz** verwendet werden.

<CSV-Datei>
Bei der **CSV-Datei** muss es sich um eine ANSI-CSV-Datei handeln mit Kopfzeile. Die Kopfzeilen enthalten dabei die Feldbezeichner von büro+. Folgende Spaltenbezeichner müssen in der **CSV-Datei** vorhanden sein:

- **Vog.BelegNr** (Belegnummer eines Vorgangs. Diese muss eindeutig sein)
- **VPo.ArtNr** (Artikel-Nummer aus dem Bereich Stammdaten)
- **VPo.Mge** (Menge/Anzahl)
- **VPo.Einh** (Einheit des Artikels)

Achtung:
Achten Sie bei den Spaltenbezeichnern unbedingt auf die Groß-/Kleinschreibung.
Erlaube Präfixe sind nur: **Vog.** und **VPo.**

Hinweis:
Die **CSV-Datei** wird nach dem Einladen von **ERPAdmin** nach der Spalte **Vog.BelegNr** sortiert. Nur wenn die Positionen einer Belegnummer aufeinander folgen, werden diese im gleichen Vorgang eingefügt.

Weitere Werte für die Vorgangs-Position können in den Spalten der **CSV-Datei** übergeben werden und müssen dann das Präfix „**VPo.**“ tragen. Soll der Inhalt einer Spalte ignoriert werden, muss diese Spalte die Spalten-Bezeichnung „**DUMMY**“ bekommen.

Beispiel für eine **CSV-Datei**:

	A	B	C	D	E	F
1	**Vog.BelegNr**	**Vog.AdrNr**	**VPo.ArtNr**	**VPo.Mge**	**VPo.EPreis**	**VPo.Einh**
2	AN2208144	10000	10150	1	9,95	Stk
3	AN2208144	10000	10000	2	19,8	Stk
4	AN2208144	10000	10001	3	147,5	Stk

/DeletePos **(optional)**
Sollen die bereits vorhandenen Vorgangs-Positionen vor dem Hinzufügen gelöscht werden, muss die Option **/DeletePos** übergeben werden. Damit werden zuerst alle vorhandenen Positionen aus dem Vorgang entfernt und danach die Artikel aus der **CSV-Datei** eingefügt.

/NoLockError **(optional)**
Kann ein Vorgang nicht bearbeitet werden, weil diese gerade von einem anderen Benutzer in Bearbeitung ist, wird keine Fehlermeldung zurückgegeben.

/Filter:<Filterbedingung> **(optional)**
Wenn nicht alle in der **CSV-Datei** übergebenen Vorgänge verarbeitet werden sollen, können die Vorgänge noch über einen Filter eingeschränkt werden. Es werden nur die Vorgänge aus der **CSV-Datei** verarbeitet welche durch den Filter passen.

/NoSort **(optional)**
Deaktiviert die Sortierung nach der Belegnummer durch **ERPAdmin**. Es muss in diesem Fall eine nach der Belegnummer sortierte **CSV-Datei** übergeben werden!

/MoveToHis, /MoveToBad, /Delete **(optional)**
Die **CSV-Datei** Datei kann nach der Verarbeitung gelöscht oder in das History-Verzeichnis verschoben werden. Zur Verwendung der **Optionen** lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“.

/GebuchtKz **(optional)**
Sollen Positionen auch in bereits gebuchten Vorgängen hinzugefügt werden, kann diese Option verwendet werden.

Beispiel für eine CSV-Datei:

	A	B	C	D	E	F	G	H	I	J
1	**Vog.BelegNr**	**DUMMY**	**Vog.Dat**	**VPo.ArtNr**	**VPo.Mge**	**VPo.EPreis**	**VPo.Einh**	**VPo.LiefMge**	**DUMMY**	**Vog.Bez**
2	BK2100033	10000	18.11.2021	3268	1	1	Stk	1	Nein	Vorgang-Positionen hinzugefügt
3	BK2100033	10000	18.11.2021	3269	2	2	m	1	Ja	
4	BK2100033	10000	18.11.2021	3270	3	3	Stk	1	Nein	
5	BK2100049	10000	18.11.2021	3268	3	4	Liter	1	Nein	Vorgang-Positionen hinzugefügt
6	BK2100049	10000	18.11.2021	3270	3	5	Stk	2	Ja	
7	BK2100032	10000	18.11.2021	3270	3	5	Stk	2	Nein	Vorgang-Positionen hinzugefügt

Beispiel:

ERPAdmin -VorgangPositionAddFromFile Neue-Positionen.csv /DeletePos
Entfernt existierenden Positionen innerhalb der Vorgänge die über die Spalte **Vog.BelegNr** angesprochen werden und fügt anschließend alle Artikel zu den entsprechenden Vorgängen hinzu.

143 Der Befehl „-VorgangPositionCount“ PRO

ERPAdmin -VorgangPositionCount <Bedingung> <Ergebnis-Feld> <Vorgang-Feldzuweisung> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /NoLockError /AbrPosKz

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl ermittelt **ERPAdmin** für alle Positionen eines Vorgangs, wie oft die angegebene **Bedingung** erfüllt ist und speichert die ermittelte Anzahl in dem **Ergebnis-Feld**. Damit ist es möglich festzustellen ob z.B. ein Artikel in einem Vorgang vorhanden ist um z.B. in Abhängigkeit einen weiteren Artikel hinzuzufügen (analog Feste-Artikel aus den Vorgängen).

<Bedingung>
Die Bedingung besteht aus einem Feldbezeichner aus dem Bereich der Vorgangs-Positionen (VPO.) z.B. **VPo.LiefDat (Lieferdatum)** oder aus dem Bereich Artikel (Art) z.B. **Art.WgrNr (Warengruppe)**, einem Vergleichs-Operator z.B. „**=**“ „**<>**“ „**<**“, „**>**“, „**<=**“, „**>=**“ und einem Vergleichswert z.B. **13.05.2019**. Die Verknüpfung von mehreren Bedingungen mit **AND** oder **OR** wird nicht unterstützt.

Beispiel: VPo.LiefDat<=13.05.2019 oder **Art.WgrNr=100**

<Ergebnis-Feld>
Es wird ein Feldbezeichner aus dem Bereich der Vorgängen (**Vog**) übergeben z.B. ein Selektionsfeld, in dem das Ergebnis abgespeichert werden soll. Das Feld **muss** ein **Textfeld** sein.

Beispiel: Vog.SEL28

<Vorgang-Feldzuweisung> **(optional)**
Schreibt **Vorgang-Feldzuweisung** in jeden gefunden Vorgang unabhängig ob **Bedingung** erfüllt wurde. Damit kann die erneute Verarbeitung zu einem späteren Zeitpunkt unterbunden oder auch ein Kennzeichen zur Weiterverarbeitung gesetzt werden. Mehrere Feldzuweisungen sind durch ein Semikolon zu trennen.

Beispiel: ErledigtKz=1

/Von:<Bereich von> und **/Bis:<Bereich von>** **(optional)**
Die zu verarbeitenden Vorgänge können über die Werte **/Von:<BereichVon>** bis **/Bis:<BereichBis>** eingeschränkt werden. Wird kein Bereich angegeben, werden alle Vorgänge verarbeitet.

Beispiel: /Von:BelegNr=BL1900000 /Bis:BelegNr=BL1909999

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/NoLockError **(optional)**
Kann ein Vorgang nicht bearbeitet werden, weil diese gerade von einem anderen Benutzer in Bearbeitung ist , wird keine Fehlermeldung zurückgegeben.

/AbrPosKz **(optional)**
Sollen nur abrechenbare Vorgangs-Positionen beachtet und Positionen wie z.B. „**Geliefert**“, „**Alternativ**“, „**Optional**“, etc. nicht beachtet werden, muss die Option **/AbrPosKz** verwendet werden.

Ist die Option gesetzt, werden nur Vorgangspositionen berücksichtigt die das Abrechnungskennzeichen „**VPo.AbrPosKz**“ gesetzt haben. Alle anderen Positionen werden ignoriert.

Ist die Option NICHT gesetzt, werden alle Positionen berücksichtig ungeachtet vom Abrechnungs-Kennzeichen.

Beispiele:

ERPAdmin -VorgangPositionCount VPo.LiefDat<=13.05.2019 Vog.Sel28 /Von:BelegNr=BL1900000 /Bis:BelegNr=BL1909999
In den Vorgängen „**Bestellung Lieferant**“ (**BL1900000** bis **BL1909999**) wird die Anzahl der Positionen ermittelt, bei denen das Lieferdatum überschritten wurde. Das Ergebnis wird in dem Feld **Vog.Sel28** gespeichert. Über eine Farbregel können diese Vorgänge dann hervorgehoben werden damit der Benutzer schneller erkennt, bei welchen Bestellungen der Liefertermin überschritten wurde.

ERPAdmin -VorgangPositionCount Art.WgrNr=100 Vog.Sel28 /Von:BelegNr=AN1900000 /Bis:BelegNr=AN1909999
In den Vorgängen „**Angebot**“ (**AN1900000** bis **AN1909999**) wird die Anzahl der Positionen ermittelt, in denen Artikel aus der Warengruppe **100** vorkommen. Das Ergebnis wird in dem Feld **Vog.Sel28** gespeichert.

ERPAdmin -VorgangPositionCount Art.WgrNr=100 Vog.Sel28 ErledigtKz=1 /Von:BelegNr=AN1900000 /Bis:BelegNr=AN1909999 /Filter:ErledigtKZ<>1
In den Vorgängen „**Angebot**“ (**AN1900000** bis **AN1909999**) bei denen das **ErledigtKz** nicht gesetzt ist, wird die Anzahl der Positionen ermittelt, in denen Artikel aus der Warengruppe **100** vorkommen. Das Ergebnis wird in dem Feld **Vog.Sel28** gespeichert und das **ErledigtKz** im Vorgang wird gesetzt. Beim nächsten Durchlauf wird dieser Vorgang ignoriert, weil das **ErledigtKz** bereits gesetzt ist.

144 Der Befehl „-VorgangPositionMengeCount“ PRO

ERPAdmin -VorgangPositionMengeCount <Bedingung> <Ergebnis-Feld> <Vorgang-Feldzuweisung> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /NoLockError /AbrPosKz

Die Aufruf-Parameter sind 1:1 identisch mit dem vorherigen Befehl **VorgangPositionCount.** Gezählt werden hier aber die Mengen in den Vorgangspositionen und nicht die Anzahl der Positionen.

145 Der Befehl „-VorgangStornieren“ STD

ERPAdmin -VorgangStornieren <Datenbank> <VogArtNr> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

Mit diesem Befehl storniert **ERPAdmin** Vorgänge im Bereich „**Vorgang**“ oder „**VorgangArchiv**“ sofern die Vorgänge ein aktives Gebucht-Kennzeichen haben und nicht gesperrt sind. Befindet sich der Vorgang im Archiv, wird dieser zuerst in die Vorgänge verschoben (nicht kopiert). Nach dem Verschieben befindet sich keine Kopie des Vorgangsmehr im Archiv!

<Datenbank>
Kann die Werte „**Vorgang**“ oder „**VorgangArchiv**“ haben und gibt an aus welcher Datenbank die Vorgänge storniert werden sollen.

<VogArtNr>
Gibt die Vorgangs-ArtNr. an die storniert werden sollen. Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.**

15		Angebot
20		Auftragsbestätigung
30		Bestellung vom Kunden
40		Bestellung an Lieferant
41		Eingangsrechnung
50		Lieferschein
70		Rechnung
90		Gutschrift

Klartextnamen der Vorgänge wie z.B. „Angebot“ sind **nicht** zulässig.

Hinweis: Gesperrte Vorgänge werden nicht storniert.

/Von:<Bereich von> und **/Bis:<Bereich von>** **(optional)**
Mit **/Von:** und **/Bis:** kann eine Einschränkung auf die Tabelle gelegt werden wie z.B. **"/Von:BelegNr=RE17000"** oder **"/Von:AdrNr=10000"**. Soll ein ganzer Bereich selektiert werden muss auch ein Wert **/Bis:** übergeben werden z.B. **"/Bis:BelegNr=RE17999"** oder **"/Bis:AdrNr=19999"**

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

Beispiele:

ERPAdmin -VorgangStornieren VorgangArchiv 70 /Von:BelegNr=RE17000 oder
ERPAdmin -VorgangStornieren VorgangArchiv 0 /Von:BelegNr=RE17000
Storniere die Rechnung im Vorgangs-Archiv mit der Beleg-Nummern RE17000

ERPAdmin -VorgangStornieren VorgangArchiv 90 /Von:BelegNr=GSE17000 /Bis:BelegNr=GS17999
Storniere alle Gutschriften im Vorgangs-Archiv mit der Beleg-Nummern GS17000 bis GS17999

ERPAdmin - VorgangStornieren Vorgang 50 /Von:AdrNr=11500
Storniere alle Lieferscheine des Kunden mit der Adress-Nummer 11500 in den Vorgängen

146 Der Befehl „-VorgangSVArtikelAdd“ PRO

ERPAdmin -VorgangSVArtikelAdd <Artikel-Selektion> <Mengen-Selektion> <Feldzuweisungen Vorgang> /Von:BelegNr=<BelegNr> /Bis:BelegNr=<BelegNr> /Filter:<Filterbedingung> /Datei:<Doubletfile>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

> **Hinweis:**
> **ERPAdmin** unterstützt aktuell nur die Abrechnung von Serviceverträgen „**Über Laufzeit**“.

Für die Abrechnung von Service-Verträgen (SV) müssen in **büro+** zusätzliche Artikel mit der Art „**Servicevertragsartikel**“ angelegt werden. Das kann einige Nachteile haben:

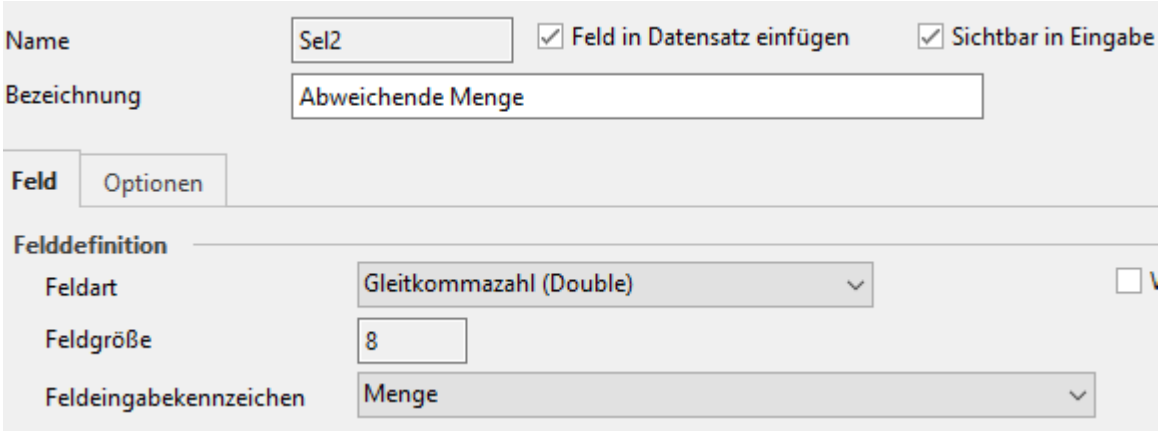
- Werden alle Artikel als SV-Artikel benötigt müssen alle Artikel doppelt angelegt werden
- Lagerbestände müssen ggf. auf zwei Artikeln geführt und dauern umgebucht werden
- Artikel-Umsätze müssen auf mehreren Artikeln gezählt werden (statt auf einem Artikel)
- In der Historie müssen immer zwei Artikel ausgewertet werden
- Stücklisten und Varianten-Artikel können nicht als SV-Artikel verwendet werden
- Abweichende Preise müssen doppelt geführt werden
- Abweichende Mengen können erst ab **büro+ XL** verwendet werden

Alle diese Punkte können mit **ERPAdmin** gelöst werden und das ab **büro+ M**.

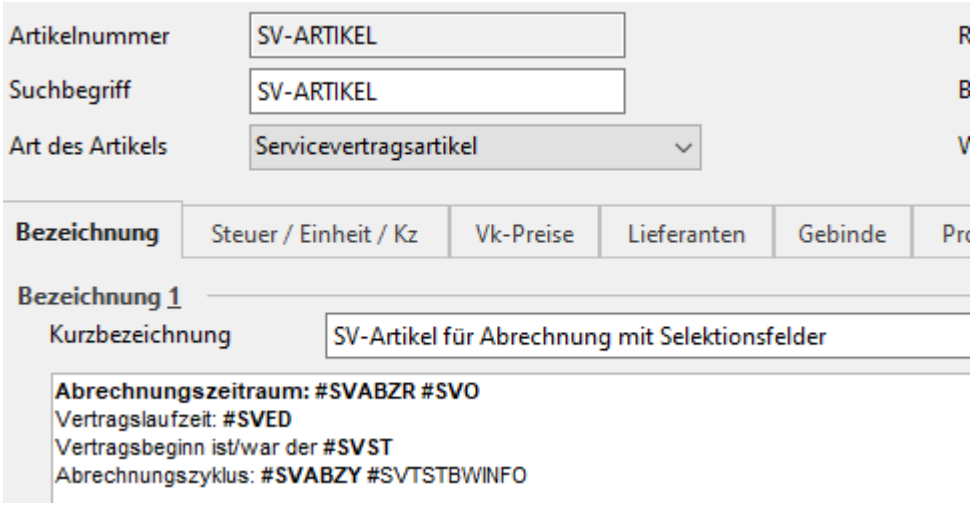
Dazu müssen zwei Selektionsfelder in den Serviceverträgen angelegt werden:

- **Standard Artikelnummer (Feldart: Text (Unicode-String))**

- **Abweichende Menge (Feldart: Gleitkommazahl (Double))**

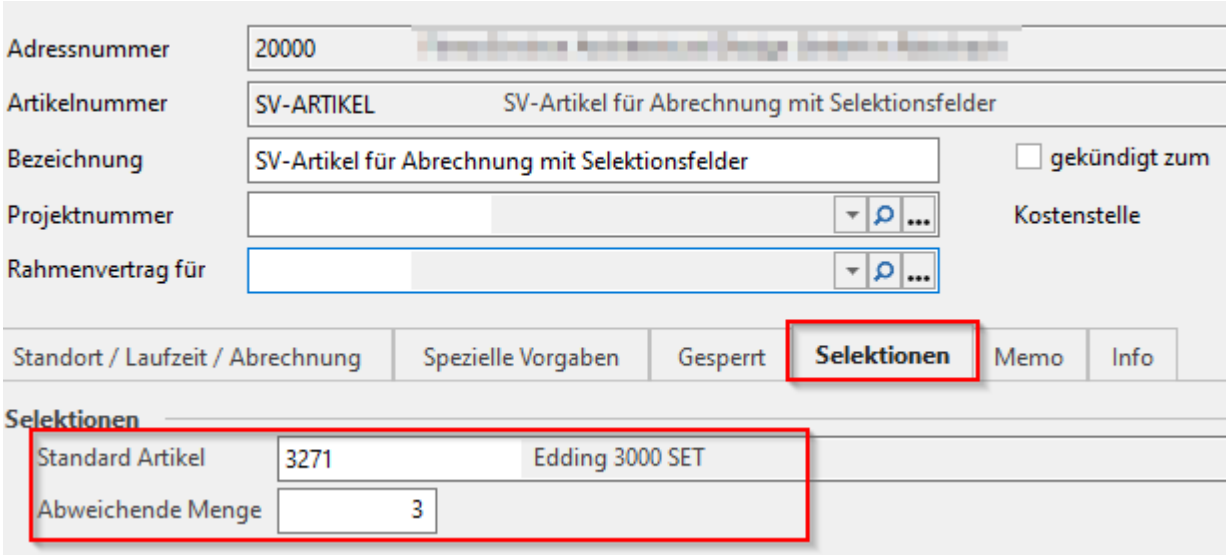


Weiterhin muss ein **SV-Artikel** angelegt werden um die Abrechnungszeiträume, den Abrechnungszyklus, Vertragsbeginn, etc. in den Vorgängen auszuweisen. Dieser Artikel könnte dazu wie folgt aussehen:



[SV-Artikel für die Abrechnung der Service-Verträge]

Nun können die Service-Verträge angelegt werden unter Verwendung des obigen **SV-Artikels**. Die Abrechnungszeiträume werden wie gehabt im Servicevertrag des Kunden hinterlegt.



Im Servicevertrag auf dem Reiter **Selektionen** werden dann der eigentliche Artikel und eine abweichende Menge in den Selektionsfeldern hinterlegt. Wird keine abweichende Menge hinterlegt wird der Wert „**1**“ angenommen.

> **Hinweis:**
> Die Felder auf dem Reiter „**Spezielle Vorgaben**“ (ab büro+ L/XL) bei Verwendung dieses **ERPAdmin** - Befehls nicht mehr verwendet werden.

<Artikel-Selektion>
Übergibt an **ERPAdmin** den Namen des Selektionsfeldes aus den Serviceverträgen in dem die Artikelnummer gespeichert wird.
Beispiel: Sel1

<Mengen-Selektion> **(optional)**
Übergibt an **ERPAdmin** den Namen des Selektionsfeldes aus den Serviceverträgen in dem die abweichende Menge gespeichert wird. Wird kein Selektionsfeld angegeben wird immer die Menge „**1**“ verwendet.
Beispiel: Sel2

<Feldzuweisungen Vorgang> **(optional)**
Diese Feldzuweisungen können im SV-Vorgang von **ERPAdmin** ausgeführte werden. Mehrere Feldzuweisungen sind durch Semikolons zu trennen. Damit können Werte im Vorgang (Nicht in den VorgangsPositionen) geändert werden.
Beispiel: Vog.ErledigtKz=Ja

/Von:BelegNr=<BelegNr>
Selektiert einen Vorgangs-Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt.
Beispiel: /Von:BelegNr=SV2100000

/Bis:BelegNr=<BelegNr>
Hier wird das Ende des Vorgangs-Bereiches vorbelegt.
Beispiel: /Bis:BelegNr=SV2199999

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/Datei:<DoubletFile > **(optional)**
Dateiname inkl. Pfadangabe in dem alle bereits verleiteten Vorgänge protokolliert werden. Vor jeder Vorgangsverarbeitung prüft **ERPAdmin** ob die Belegnummer in **DoubletFile** vorhanden ist. Nur wenn die Belegnummer nicht in der Doublettendatei vorhanden ist wurde der Vorgang verarbeitet.
Beispiel: /Datei:Doubletten_SV-Artikel.log

Weitere Informationen zur Verwendung von Doubletten-Prüfungen finden Sie Kapitel „**Doubletten-Prüfung**“. Die Doublettenprüfung darf nicht verwendet werden wenn in einem Vorgang mehrere SV-Artikel abgerechnet werden.

Ablauf der Servicevertragsabrechnung

Die Serviceverträge werden ganz normal im Bereich Vorgänge abgerechnet.

> **Hinweis:**
> Verwenden Sie eine eigene Vorgangsart für die Erstellung der SV-Abrechnungsvorgänge.

Nachdem alle SV-Vorgänge erzeugt wurden befindet sich in diesen Vorgängen die hinterlegten SV-Artikel mit den Abrechnungszeiträumen. Der SV-Artikel selber darf dabei weder einen Preis noch eine Menge haben.

Adresse | **Positionen** | Infoblatt | Vor- / Nachtext | Adr.-Kennzeichen | Vorgaben | Selektionen

Position	SW	ID des Servicev	Artikelnummer	Bezeichnung
1.	☐	1506	SV-ARTIKEL	**Abrechnungszeitraum: 01.01.2020 - 31.12.2020** Vertragslaufzeit: **31.12.2020** Vertragsbeginn ist/war der **01.01.2020** Abrechnungszyklus: **12 Monate**

Nun wird **ERPAdmin** aufgerufen. Nach dem Aufruf von **ERPAdmin** fügt dieses in jedem Vorgang mit SV-Artikeln den in dem Selektionsfeld (**Sel1**) hinterlegten Artikel in den Vorgang ein unter Beachtung der Menge aus dem Selektionsfelde (**Sel2**). Wurde kein Selektionsfeld für die Menge angegeben oder wurde in dem Selektionsfeld (**Sel2**) keine Menge hinterlegt wird die Menge „**1**“ verwendet.

Adresse | **Positionen** | Infoblatt | Vor- / Nachtext | Adr.-Kennzeichen | Vorgaben | Selektionen | Parameter

Posit	SW	ID des Se	Artikelnummer	Bezeichnung	Menge
1.	☐	1506	SV-ARTIKEL	**Abrechnungszeitraum: 01.01.2020 - 31.12.2020** Vertragslaufzeit: **31.12.2020** Vertragsbeginn ist/war der **01.01.2020** Abrechnungszyklus: **12 Monate**	
2.	☐		3271	Edding 3000 Set klein (4 Stifte) - 1 x schwarz (ED3000SW) - 1 x grün (ED3000GR)	1

Der Vorgang kann nun entweder vom Benutzer oder durch **ERPAdmin -VorgangWandeln** weiterverarbeitet werden.

> **Hinweis:**
> Über eine Regel „**Eigene Abläufe**“ kann **ERPAdmin** direkt aus **büro+** heraus aufgerufen werden.

Beispiele:

ERPAdmin -VorgangSVArtikelAdd Sel1 /Von:BelegNr=SV2100000 /Bis:BelegNr=SV2199999
Es werden alle Vorgänge **SV210000** bis **SV2199999** auf enthaltene Servicevertragsartikel geprüft. Wird ein SV-Artikel gefunden fügt **ERPAdmin** den hinterlegten Artikel aus **Sel1** mit der Menge **1** zum Vorgang hinzu.

ERPAdmin -VorgangSVArtikelAdd Sel1 Sel2 /Von:BelegNr=SV2100000 /Bis:BelegNr=SV2199999
Es werden alle Vorgänge **SV210000** bis **SV2199999** auf enthaltene Servicevertragsartikel geprüft. Wird ein SV-Artikel gefunden fügt **ERPAdmin** den hinterlegten Artikel aus **Sel1** mit der Menge **Sel2** zum Vorgang hinzu.

ERPAdmin -VorgangSVArtikelAdd Sel1 Sel2 Vog.ErledigtKz=Ja /Von:BelegNr=SV2100000 /Bis:BelegNr=SV2199999
Die SV-Artikel werden wie im vorherigen Beispiel hinzugefügt. Anschließend wird im Vorgang das Erledigt-Kennzeichen (**Vog.ErledigtKz=Ja**) gesetzt.

147 Der Befehl „-VorgangUstIDCheck“ PRO

ERPAdmin -VorgangUstIDCheck <Selektion1> <Selektion2> /Von:BelegNr=<BelegNr> /Bis:BelegNr=<BelegNr> /Filter:<Filterbedingung> /NoLockError /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /Force /MoveToHis /ReAns /CharSet:<Zeichensatz>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Vor jedem Auftrag mit Lieferanschrift in ein EU-Empfängerland muss die Umsatzsteuer-ID des Empfängers auf Gültigkeit geprüft werden. Dabei unterstützt Sie **ERPAdmin**. Es können alle Vorgänge im Bereich **/Von:BelegNr=<BelegNr>** bis **/Bis:BelegNr=<BelegNr>** überprüft werden, bei denen eine Umsatzsteuer-ID hinterlegt ist und wo das Lieferland NICHT Deutschland ist.

Bei der Prüfung wird sowohl die Richtigkeit der Umsatzsteuer-ID überprüft aus auch die Richtigkeit der Angabe der **Lieferadresse**. Als Ergebnis für die Umsatzsteuer-ID-Prüfung wird entweder ein „**OK**“ zurückgegeben oder ein Fehlercode.

> **Hinweis:**
> Die Prüfung der Umsatzsteuer-ID ist **nur für Unternehmen aus Deutschland** mit deutsche Umsatzsteuer-ID möglich und nur für Kunden die sich aus deutscher Sicht im EU-Ausland befinden.

Für die Überprüfung der Lieferanschrift werden die Meldungen „**A**“, „**B**“, „**C**“ oder „**D**“ zurückgegeben die sich auf die **Firmierung**, die **Straße**, **PLZ** oder den **Ort** beziehen.

Bei der Adress-Prüfung werden die Angaben **Firmierung**, **Straße**, **PLZ** und **Ort** der **Lieferanschrift** geprüft und eine Rückmeldung erzeugt, wenn diese von den erwarteten Angaben zum Unternehmens abweichen.

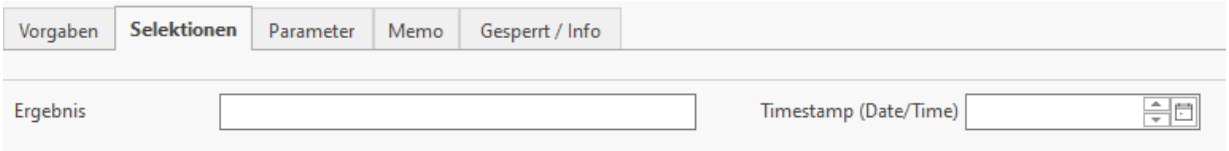
> **Achtung:**
> **ERPAdmin** führt nur dann eine Prüfung für einen Vorgang durch, wenn das Feld **Selektion1** leer ist. Alle Vorgänge mit nicht leerem Inhalt werden bei der Prüfung ignoriert.
>
> Mit der Option **/Force** kann eine erneute Prüfung erzwungen werden.

Diese Funktionalität von **ERPAdmin** setzt den Web-Server des „**Bundesamtes für Steuer**“ (BFS) voraus bei dem diese Daten online abgefragt werden. Weitere Informationen finden Sie unter https://evatr.bff-online.de/eVatR/. Hier können Sie auch eine manuelle Prüfung der Umsatzsteuer-ID durchführen.

DBManager – Benötigte Felder

Es werden zwei Felder in den Vorgängen benötigt in denen **ERPAdmin** die Informationen der Umsatzsteuer-ID-Prüfung speichern kann. Hierfür sollten am besten zwei Selektionsfelder wie folgt angelegt werden:

- Name: **Ergebnis** – Typ: String, Länge 400 Zeichen (**Selektion1**)
- Name: **Timestamp** – Typ: Date/Time (**Selektion2**)



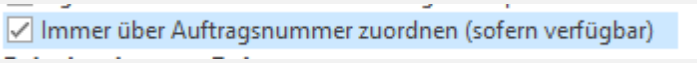
Dokumentation

Jede Abfrage am BFR-Server wird mit einer XML-Datei beantwortet. Diese XML-Dateien werden dann in den Dokumenten der Vorgänge als Dokumentation hinterlegt. Die Zuordnung in den Vorgängen erfolgt über die Auftragsnummer (sofern vorhanden) sonst über die Belegnummer.

Zusätzlich werden die XML-Dateien im History-Verzeichnis von **ERPAdmin** (wenn in der **VAR.INI** angegeben) für eine bestimmte Zeit archiviert.

Hinweis:

Damit Dokumente beim Wandeln der Vorgänge nicht verloren gehen muss in den Parametern der Dokumente folgende Einstellung aktiviert werden:



<Selektion1>

Name des Selektionsfeldes im Vorgang für die Rückmeldung der Prüfung. Das Feld muss **min. 50 Zeichen** groß sein und vom Typ „**String**“.

Ist das Feld **min. 400 Zeichen** groß, wird zusätzlich zum Fehlercode die komplette Fehler-Meldungen im Selektionsfeld gespeichert.

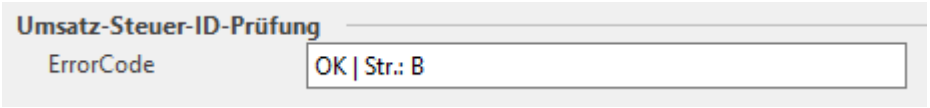
Ist die Umsatzsteuer-ID korrekt, wird ein „**OK**“ zurückgegeben.

Trat ein Fehler auf wird ein Fehlercode zurückgegeben**. Eine Liste aller Fehlercode finden Sie weiter unten.**

Neben der Umsatzsteuer-ID werden auch die Anschriften-Felder Firma/Name (**Li.Na2**), Straße (**Li.Str**), PLZ (**Li.PLZ**) und Ort (**LI.Ort**) geprüft. Kam es hier zu einer Abweichung, wird dieses wie folgt zurückgemeldet:

- A = stimmt überein
- B = stimmt nicht überein
- C = nicht angefragt
- D = vom EU-Mitgliedsstaat nicht mitgeteilt

Ist die Umsatzsteuer-ID korrekt aber die Straße stimmt nicht überein, sieht die Rückmeldung z.B. wie folgt aus:



<Selektion2>
Name des Selektionsfeldes im Vorgang für den Timestamp der Abfrage. Hier wird ein Timestamp im Format **tt.mm.jjjj hh:mm:ss** eingetragen. Die Uhrzeit des Timestamp ist Teil der Rückmeldung des BFS-Servers.

/Von:BelegNr=<BelegNr> **(optional)**
Selektiert einen Vorgangs-Bereich aus **büro+**. Hier wird der Anfangsbereich vorbelegt.
Beispiel: /Von:BelegNr=AB2100000

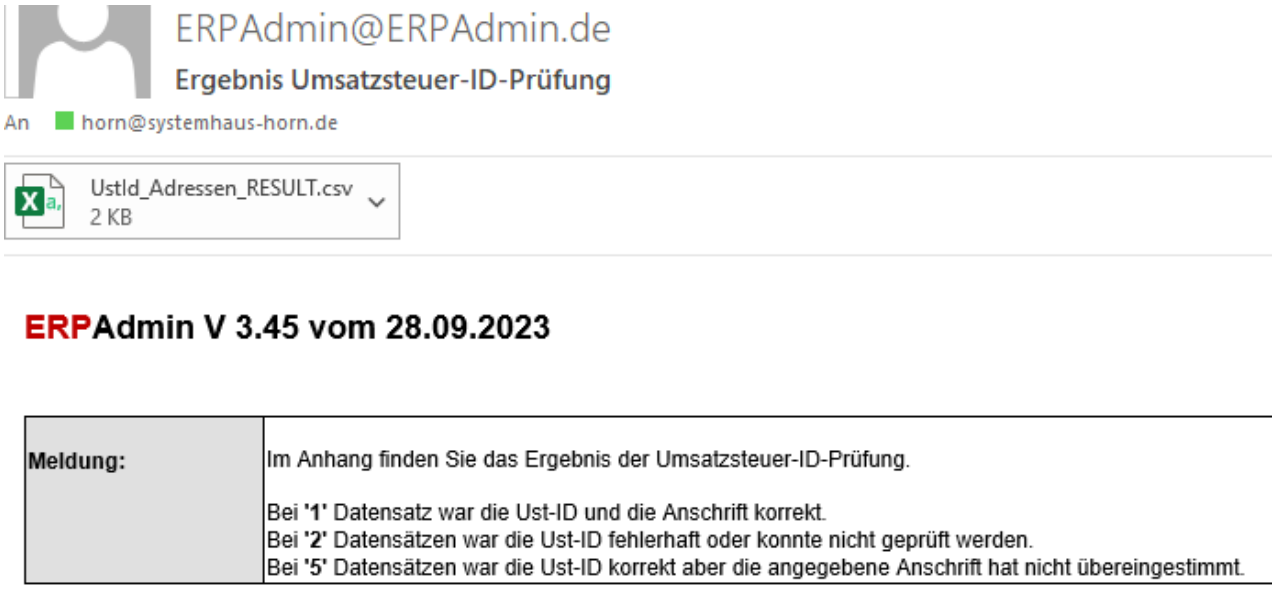
/Bis:BelegNr=<BelegNr> **(optional)**
Hier wird das Ende des Vorgangs-Bereiches vorbelegt.
Beispiel: /Bis:BelegNr=AB2199999

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/NoLockError **(optional)**
Kann in einen Vorgang nicht geschrieben werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei „**UstID_Vorgang_Result.csv**“ mit allen Ergebnissen der Umsatzsteuer-ID-Prüfung sendet. Wurden keine Datei erzeugt wird keine E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

Die E-Mail kann dann wie folgt aussehen:



/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/Force **(optional)**
Eine Prüfung der Umsatzsteuer-ID wird nur dann für einen Vorgang ausgeführt, wenn der Inhalt des Feldes **<Selektion1>** leer ist. Durch Angabe der Option **/Force** kann dennoch eine Prüfung erzwungen werden. Bitte beachten Sie, dass in den Dokumenten des Vorgangs durch wiederholten Aufruf mit dieser Option mehrere XML-Ergebnisdateien hinterlegt sein können. Diese sind durch den Timestamp unterscheidbar.

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

/ReAns **(optional)**
Soll anstelle der Lieferanschrift die **Rechnungsanschrift** geprüft werden muss die Option **/ReAns** übergeben werden. Ist im Vorgang das Kennzeichen „**Dreieckgsgeschäft**“ aktiv, wird trotz des Parameters **/ReAns** die **Lieferanschrift** geprüft.

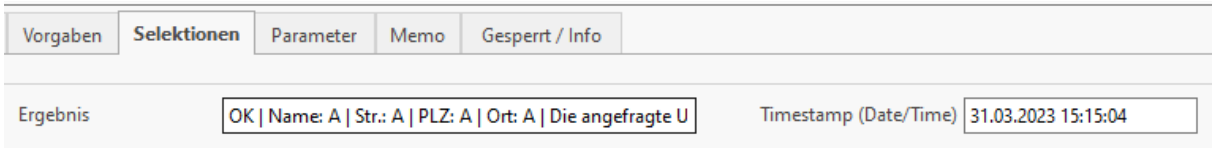
/ChartSet: **<Zeichensatz>**
Wurde die Option **/Email:** verwendet, wird eine Ergebnisdatei per E-Mail verwendet. Das Format dieser CSV-Datei ist **ANSI,** damit die Datei mit Excel geöffnet werden kann. Soll diese CSV-Datei als **UTF-8** versendet werden, muss die Option **/CharSet:UTF-8** übergeben werden.

Beispiele:

ERPAdmin -VorgangUstIDCheck Sel1 Sel2 /Von:BelegNr=AN2100000 /Bis:BelegNrAN2199999
Es werden alle Vorgänge mit den Belegnummern **AN210000** bis **AN219999** geprüft die eine Umsatzsteuer-ID enthalten und wenn das Selektionsfeld **SEL1** leer ist.

ERPAdmin -VorgangUstIDCheck Sel1 Sel2 /Von:BelegNr=AN2100000 /Bis:BelegNrAN2199999 /Filter:LiOrt<>‘Wien‘
Es werden alle Vorgänge mit den Belegnummern **AN210000** bis **AN219999** geprüft die eine Umsatzsteuer-ID enthalten, wenn das Selektionsfeld **SEL1** leer ist und wenn der Ort der Lieferanschrift nicht „**Wien**“ ist.

Das Ergebnis einer Prüfung kann dann in büro+ wie folgt aussehen:


Vorgaben
Selektionen
Parameter
Memo
Gesperrt / Info
Ergebnis
OK | Name: A | Str.: A | PLZ: A | Ort: A | Die angefragte U
Timestamp (Date/Time)
31.03.2023 15:15:04

Liste der Fehlercodes

ErrorCode	Beschreibung
OK	**Die angefragte USt-IdNr. ist gültig.**
201	Die angefragte USt-IdNr. ist ungültig.
202	Die angefragte USt-IdNr. ist ungültig. Sie ist nicht in der Unternehmerdatei des betreffenden EU-Mitgliedstaates registriert. **Hinweis**: Ihr Geschäftspartner kann seine gültige USt-IdNr. bei der für ihn zuständigen Finanzbehörde in Erfahrung bringen. Möglicherweise muss er einen Antrag stellen, damit seine USt-IdNr. in die Datenbank aufgenommen wird.
203	Die angefragte USt-IdNr. ist ungültig. Sie ist erst ab dem ... gültig (siehe Feld 'Gueltig_ab').
204	Die angefragte USt-IdNr. ist ungültig. Sie war im Zeitraum von ... bis ... gültig (siehe Feld 'Gueltig_ab' und 'Gueltig_bis').
205	Ihre Anfrage kann derzeit durch den angefragten EU-Mitgliedstaat oder aus anderen Gründen nicht beantwortet werden. Bitte versuchen Sie es später noch einmal. Bei wiederholten Problemen wenden Sie sich bitte an das Bundeszentralamt für Steuern - Dienstsitz Saarlouis.
206	Ihre deutsche USt-IdNr. ist ungültig. Eine Bestätigungsanfrage ist daher nicht möglich. Den Grund hierfür können Sie beim Bundeszentralamt für Steuern - Dienstsitz Saarlouis - erfragen.
207	Ihnen wurde die deutsche USt-IdNr. ausschließlich zu Zwecken der Besteuerung des innergemeinschaftlichen Erwerbs erteilt. Sie sind somit nicht berechtigt, Bestätigungsanfragen zu stellen.
208	Für die von Ihnen angefragte USt-IdNr. läuft gerade eine Anfrage von einem anderen Nutzer. Eine Bearbeitung ist daher nicht möglich. Bitte versuchen Sie es später noch einmal.
209	Die angefragte USt-IdNr. ist ungültig. Sie entspricht nicht dem Aufbau der für diesen EU-Mitgliedstaat gilt. (Aufbau der USt-IdNr. aller EU-Länder)
210	Die angefragte USt-IdNr. ist ungültig. Sie entspricht nicht den Prüfziffernregeln die für diesen EU-Mitgliedstaat gelten.
211	Die angefragte USt-IdNr. ist ungültig. Sie enthält unzulässige Zeichen (wie z.B. Leerzeichen oder Punkt oder Bindestrich usw.).
212	Die angefragte USt-IdNr. ist ungültig. Sie enthält ein unzulässiges Länderkennzeichen.
213	Die Abfrage einer deutschen USt-IdNr. ist nicht möglich.
214	Ihre deutsche USt-IdNr. ist fehlerhaft. Sie beginnt mit 'DE' gefolgt von 9 Ziffern.
215	Ihre Anfrage enthält nicht alle notwendigen Angaben für eine einfache Bestätigungsanfrage (Ihre deutsche USt-IdNr. und die ausl. USt-IdNr.). Ihre Anfrage kann deshalb nicht bearbeitet werden.

216	Ihre Anfrage enthält nicht alle notwendigen Angaben für eine qualifizierte Bestätigungsanfrage (Ihre deutsche USt-IdNr., die ausl. USt-IdNr., Firmenname einschl. Rechtsform und Ort). Es wurde eine einfache Bestätigungsanfrage durchgeführt mit folgenden Ergebnis: Die angefragte USt-IdNr. ist gültig.
217	Bei der Verarbeitung der Daten aus dem angefragten EU-Mitgliedstaat ist ein Fehler aufgetreten. Ihre Anfrage kann deshalb nicht bearbeitet werden.
218	Eine qualifizierte Bestätigung ist zur Zeit nicht möglich. Es wurde eine einfache Bestätigungsanfrage mit folgendem Ergebnis durchgeführt: Die angefragte USt-IdNr. ist gültig.
219	Bei der Durchführung der qualifizierten Bestätigungsanfrage ist ein Fehler aufgetreten. Es wurde eine einfache Bestätigungsanfrage mit folgendem Ergebnis durchgeführt: Die angefragte USt-IdNr. ist gültig.
220	Bei der Anforderung der amtlichen Bestätigungsmitteilung ist ein Fehler aufgetreten. Sie werden kein Schreiben erhalten.
221	Die Anfragedaten enthalten nicht alle notwendigen Parameter oder einen ungültigen Datentyp. Weitere Informationen erhalten Sie bei den Hinweisen zum Schnittstelle - Aufruf.
223	Die angefragte USt-IdNr. ist gültig. Die Druckfunktion steht nicht mehr zur Verfügung, da der Nachweis gem. UStAE zu § 18e.1 zu führen ist.
999	Eine Bearbeitung Ihrer Anfrage ist zurzeit nicht möglich. Bitte versuchen Sie es später noch einmal.

148 Der Befehl „-VorgangWandeln“ STD

ERPAdmin -VorgangWandeln <AusVogArtNr> <NachVogArtNr> <LieferbarStatus> /LiefKz:<Liefer-Mengen-Schlüssel> /DRUCKEN: /LIEFDAT: /BUCHEN /BEZAHLT /NoLockError /CheckArtikelLock /GebuchtKz /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung>

Für die Verwendung einzelner Optionen in diesem Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

<AusVogArtNr> und **<NachVorArtNr>**
Mit diesem Befehl wandelt **ERPAdmin** alle Vorgänge die **nicht gesperrt** sind **AusVogArtNr** in **NachVorArtNr** in Anhängigkeit des **Lieferbarstatus**. Die **VogArtNr** der Vorgangsarten findet man in den Parametern „**Vorgängen und Zwischenbelegen -> Arten**“ von **büro+.**

	15		Angebot
	20		Auftragsbestätigung
	30		Bestellung vom Kunden
	40		Bestellung an Lieferant
	41		Eingangsrechnung
	50		Lieferschein
	70		Rechnung
	90		Gutschrift

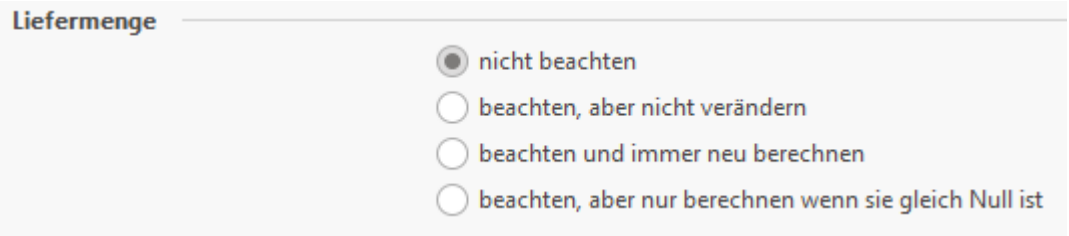
Klartextnamen der Vorgänge sind **nicht** zugelassen.

<Lieferbarstatus> **(optional)**
Der **Lieferbarstatus** entspricht dem Ampelsystem aus dem Bereich der Vorgänge. Dabei steht:

0	Immer Wandeln unabhängig vom Lieferbarstatus	
1	für rot	nicht lieferbar
2	für gelb	teilweise lieferbar
3	grün	komplett lieferbar

Wird kein Wert angegeben wird „**0**“ (Immer Wandeln unabhängig vom Lieferbarstatus) verwendet.

/LiefKz:<Liefer Mengen Schlüssel> **(optional)**
Hier wird **ERPAdmin** mitgeteilt, wie das Feld **Liefermenge** beachtet werden soll. Die Funktionsweise ist analog zu den Einstellungen aus dem Bereich **Parameter->Vorgänge und Zwischenbelege->Arten** auf dem Register „**Vorgaben für Wandeln**“.



0 nicht beachten
1 beachten, aber nicht verändern
2 beachten und immer neu berechnen
3 beachten, aber nur berechnen wenn sie gleich Null ist

Wird hier kein Wert angegeben, wird der Wert „**0**“ (nicht beachten) verwendet!

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Sollen nicht alle Vorgänge einer Vorgangs-Art gewandelt werden, sondern nur bestimmte Vorgänge, können diese über die beiden Abfragen **/Von:** bis **/Bis:** eingegrenzt werden. Um Vorgänge über **/Von:** und **/Bis:** einzuschränken können alle Suchfelder verwendet werden die von **büro+** zur Vorgangsselektion verwendet werden. Dies sind: BelegNr, AuftrNr, Suchbeg, AdrNr, ProjNr. Soll keine Einschränkung gemacht werden bleiben diesen beiden Werte leer.

Parameter /DRUCKEN / BEZAHLT /DRUCKEN, /LIEFDAT, /NoLockError, /CheckArtikelLock, /GebuchtKz
Über die Optionen /**BEZAHLT**, **/BUCHEN, /DRUCKEN:<Layoutname>** und **/LiefDat:<Datum>** können weitere Befehle an **ERPAdmin** übergeben werden.

Die Optionen **/BUCHEN** und **/DRUCKEN** dürfen dabei **nicht** zusammen verwendet werden.

/BEZAHLT **(optional)**
Wird der Parameter /**BEZAHLT** übergeben prüft **ERPAdmin** ob der Vorgangs-Wert im Feld „**Kunde hat bezahlt“ >= dem Vorgangsendbetrag** ist oder **„Kunde hat bezahlt“ > 0** und das Kennzeichen „**Restbetrag ist Preisnachlass**“ gesetzt ist. Die Werte findet man in den Vorgängen auf dem Vorgangsregister „**Parameter**“.

Damit ist es möglich einen Vorgang automatisch zu wandeln, wenn dieser komplett bezahlt wurde. Dies ist z.B. bei Aufträgen mit Bezahlung per Vorkasse der Fall.

/BUCHEN **(optional)**
Wird die Option /**BUCHEN** übergeben, wird der Ziel-Vorgang nach dem Wandeln automatisch verbucht.

/DRUCKEN:<Layoutname> PRO **(optional)**
Mit der Option /**DRUCKEN=<Layout>** wird der Ziel-Vorgang nach dem Wandeln automatisch mit dem übergebenen Layout ausgedruckt. Für die Verwendung dieses Befehls benötigen Sie die Professional-Lizenz von **ERPAdmin**!

/LIEFDAT:<Datum> **(optional)**
Soll beim Wandeln das Lieferdatum beachtet werden, kann dieses über den Parameter **/LIEFDAT:<Datum>** an **ERPAdmin** übergeben werden. Soll immer das Tagesdatum verwendet werden kann an Stelle des Datums der Platzhalter **@DATE@** übergeben werden. Weitere Informationen zu den Platzhaltern ist im Kapitel „**Der Befehl -Export und -Import**“ zu finden

/Filter:<Filterbedingung> **(optional)**
Die zu wandelnden Vorgänge können über einen Filter eingeschränkt werden. Weitere Informationen zu Filtern entnehmen Sie bitte dem entsprechenden Kapitle „**Benutzung von Filtern**“

/NoLockError PRO **(optional)**
Ist ein Vorgang der gewandelt werden soll durch einen anderen Benutzer in Bearbeitung, kommt es zu einer Fehlermeldung und der Vorgang wird übersprungen. Soll diese Fehlermeldung unterdrückt werden kann die Option **/NoLockError** gesetzt werden. Es wird dann eine Warnmeldung an den Benutzer versendet. Für die Verwendung dieses Befehls benötigen Sie die Professional -Lizenz von **ERPAdmin**!

/CheckArtikelLock PRO **(optional)**
Enthält ein Vorgang Artikel die in den Stammdaten gesperrt sind erscheint beim Wandeln eine Fehlermeldung am Bildschirm. Da **ERPAdmin** unbeaufsichtigt läuft, wird mit jedem Aufruf eine **ERPAdmin** Instanz gestartet welche beim Wandeln hängen bleiben würde. Es ist dafür zu sorgen, dass kein Artikel eines Vorgangs gesperrt ist wenn diese mit **ERPAdmin** gewandelt werden sollen. Ist das nicht möglich, kann die Option **/CheckArtikelLock** angegeben werden. **ERPAdmin** prüft dann vor dem Wandeln ob es gesperrte Artikel im Vorgang gibt und liefert eine entsprechende Meldung an den Benutzer zurück. Für die Verwendung dieses Befehls benötigen Sie die Professional -Lizenz von **ERPAdmin**!

/GebuchtKz PRO **(optional)**
Sollen nur gebuchte Vorgänge gewandelt werden muss die Option **/GebuchtKz** angegeben werden.
Für die Verwendung dieses Befehls benötigen Sie die Professional-Lizenz von **ERPAdmin**!

Beispiele
Es sollen alle „**Bestellungen vom Kunden**“ die komplett lieferbar sind in einen „**Lieferschein**“ gewandelt werden. Anschließend sollen die so erzeugten Lieferscheine verbucht werden damit der Lagerbestand abgebucht wird.

ERPAdmin -VorgangWandeln 30 50 3 /BUCHEN

Soll der Ziel-Vorgang nicht nur gebucht sondern auch mit dem Layout „Lieferschein“ gedruckt werden, lautet der Aufruf wie folgt:
ERPAdmin -VorgangWandeln 30 50 3 /DRUCKEN:Lieferschein

Sollen nur Vorgänge von Endkunden (die haben in unserem Fall die Adress-Nummern 10000-19999) gewandelt und gebucht werden sieht der Aufruf wie folgt aus:
ERPAdmin -VorgangWandeln 30 50 3 /Von:AdrNr=10000 /Bis:AdrNr=19999 /BUCHEN

Wandel alle „**Bestellungen vom Kunden Vorkasse**“ die Bezahlt sind in einen Lieferschein und verbuche den Lieferschein nach dem Wandeln.
ERPAdmin -VorgangWandeln 30 50 0 /BEZAHLT /BUCHEN

Soll der Zielvorgang nach dem Wandeln auch gedruckt werden ist der Aufruf um das zu verwendende Drucklayout zu ergänzen:
ERPAdmin -VorgangWandeln 30 50 0 /BEZAHLT /DRUCKEN:Lieferschein

Dürfen nur die Vorgänge gewandelt werden welche ein Lieferdatum von heute oder älter haben sieht der Aufruf wie folgt aus:
ERPAdmin -VorgangWandeln 30 50 0 /BEZAHLT /DRUCKEN:Lieferschein /LIEFDAT=@DATE@

Wandele nur Bestellungen in Lieferscheine bei denen die Selektion 5 aktiviert wurde (Boolean Feld):
ERPAdmin -VorgangWandeln 30 50 0 /Filter:Sel5=Wahr

Die mehrfache Verwendung des Gleichheitszeichens bei einem Filter ist korrekt und notwendig. Die Filterbedingung beginnt unmittelbar nach dem ersten Gleichheitszeichen.

Hinweis:
Nach Aussage von microtech (Stand 02.2017) funktioniert der „Lieferbar Status“ in büro+ nicht mehr, wenn es sich um einen gebuchten Vorgang handelt in dem Buchungsparameter abgearbeitet wurden. Bitte beachten Sie das bei Ihren Arbeiten!

Achtung:
Bei Ziel-Vorgänge in die gewandelt werden sollen darf das folgende Kennzeichen in den Vorgangsparametern „**Für das Wandeln in diesen Vorgang**“ **nicht** aktiviert sein:

Ist obiges Kennzeichen aktiviert, führt dies zu einer Verklemmung, weil **büro+** eine Bildschirm-Meldung erzeugt und auf Bestätigung wartet. Verwenden Sie stattdessen die Option **/CheckArtikelLock**

149 Der Befehl „-WriteData“ PRO

ERPAdmin -WriteData <Datenbank> <Feldwert> /Von:<Bereich von> /Bis:<Bereich bis> /Filter:<Filterbedingung> /NoLockError

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl können Datenfelder in **büro+** direkt mit **ERPAdmin** aktualisiert werden ohne dass ein Export/Import durchgeführt werden muss.

Es können nur vorhandene Datensätze beschrieben werden. Das Einfügen von neuen Datensätzen ist **nicht** möglich!

<Datenbank>
Gibt die Tabelle von **büro+** an in welcher die Daten geändert werden sollen z.B. „Adressen“, „Artikel“, „Warengruppen“, „Vorgang“, etc.

<Feldwert>
Ggibt den Feldnamen und den Wert an der in ein oder mehrere Felder geschrieben werden soll. Zum setzen des Skonto-Kennzeichens in den Artikel würde der Feldwert wie folgt aussehen:
"SktoKz=Ja".

Sollen mehrere Werte gesetzt werden, sind diese mit einem **Semikolon** zu trennen. Leerzeichen sind nicht erlaubt. Soll das Rabatt-Kennzeichen von VK0, VK1 und VK2 gesetzt werden sieht der Feldwert wie folgt aus:
"VK0.RabKz=Ja;VK1.RabKz=Ja;VK2.RabKz=Ja"

/Von:<Bereich von> und **/Bis:<Bereich bis>** **(optional)**
Mit **/Von:** und **/Bis:** wird ein Daten-Bereich selektiert der Felder bestimmt in den der **Feldwert** geschrieben werden soll.

/Filter:<Filterbedingung> **(optional)**
Mit einem **Filter** kann der Ergebnis-Bereich weiter eingeschränkt werden. Die Verwendung der Filter wird in einem extra Kapitel beschrieben.

/NoLockError **(optional)**
Kann ein Datensatz nicht geändert werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

Beispiele

Sie möchten bei den Artikeln 10000 bis 29999 das Skonto-Kennzeichen setzen.
ERPAdmin -WriteData Artikel SktoKz=Ja /Von:ArtNr=10000 /Bis:ArtNr=29999

Sie möchten bei den Artikel 10000 bis 29999 das Rabatt-Kennzeichen für VK0 setzten.
ERPAdmin -WriteData Artikel VK0.RabKz=Ja /Von:ArtNr=10000 /Bis:ArtNr=29999

Sie möchten bei allen Adressen 10000 bis 19999 mit dem Status „Endkunde“ das inkl. Steuer-Kennzeichen setzen.

ERPAdmin -WriteData Adressen IklStKz=Ja /Von:AdrNr=10000 /Bis:AdrNr=19999 /Filter:Status='Endkunde'

Sie möchten bei allen „Angeboten“ mit Land „Österreich“ in der Lieferanschrift die Steuerkategorie „3 Ausland-EU“ setzen.
ERPAdmin -WriteData Vorgang UStKat=3 /Von:Art=15 /Bis: Art=15 /Filter:LiLand=40

Alle gesperrten Artikel 10000 bis 99999 sollen dem Katalog „1 Gelöscht“ zugewiesen werden.
ERPAdmin -WriteData Artikel Katalog=1 /Von:ArtNr=10000 /Bis:ArtNr=99999 /Filter:GspGrp>0

Setzt das Standard-Lager-Kennzeichen bei allen Lagern HAULAG
ERPAdmin -WriteData Lager StdKz=Ja /Von:LagNr=HAULAG

Weise allen Adressen 30000 bis 39999 die Zahlungsbedingung „10 Tage 2%, 30 Tage netto“ zu.
ERPAdmin -WriteData "ZahlBed=10 Tage 2%%, 30 Tage netto;SkToTg1=10;SktoSz1=2;NettoTg=30" /Von:AdrNr=30000 /Bis:AdrNr=39999

Hinweis: Das doppelte %-Zeichen ist hier notwendig, da ein Einfaches %-Zeichen bei der Übergabe gefiltert wird.

Setzte bei allen Lagern das Lagerbuch-Kennzeichen
ERPAdmin -WriteData Lager LagBuchKz=Ja

150 Der Befehl „-WriteDataFromFile“ PRO

ERPAdmin -WriteDataFromFile <Datenbank> <SuchIndex> /Datei:<CSV-Datei> /NoFileNotExistError /NoLockError /MoveToHis /MoveToBad /Delete

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl können die Inhalte oder einzelne Feldwerte aus einer **CSV-Datei** direkt in der **Datenbank** von **büro+** geschrieben werden ohne dass ein Export/Import durchgeführt werden muss. Daten können so **nur aktualisiert** werden, neue Datensätze jedoch **NICHT** neu hinzugefügt werden. Um neue Daten einzufügen verwenden Sie den Befehl: **ERPAdmin -InsertDataFromFile**

Die Zuordnung der Daten zu den Datensätzen in **büro+** erfolgt über das Feld **SuchIndex**.

<Datenbank>
Tabelle aus **büro+** in der die Daten aktualisiert werden sollen. Die genaue Bezeichnung einer **büro+** Datenbank kann man mit dem Befehl **ERPAdmin -ListDatenbankfelder** herausfinden

<SuchIndex>
Eindeutiges Suchfeld mit dem die Datensätze in **büro+** referenziert werden können. Zu diesem Suchfeld muss ein **SuchIndex** (Sortierung) in der **Datenbank** existieren. In der CSV-Datei muss eine Spalte mit diesen Werten vorhanden sein um die Daten eindeutig zuzuordnen. **SuchIndexe** sind z.B.: **AdrNr** für die Datenbank **Adressen**, **ArtNr** für die Datenbank **Artikel** oder das Feld **ID** welches in fast allen Datenbanken vorhanden ist.

/Datei:<CSV-Datei
CSV-Datei im ANSI-Format mit den Feldbezeichnern aus **büro+** als Spaltenköpf.

Soll eine Spalte ignoriert werden muss als Spaltenbezeichner mit den Zeichen **DUMMY** beginnen.
Beispiel: „**DUMMY(ArtNr)**“ oder „**DUMMY ArtNr**“

/NoFileNotExistError **(optional)**
Ist keine CSV-Datei vorhanden und soll in diesem Fall keine Fehlermeldung erzeugt werden kann die Option **/NoFileNotExistError** angegeben werden. Dies kann dann sinnvoll sein, wenn z.B. dem Import ein Daten-Export vorausgegangen ist und dieser kein Ergebnis geliefert hat.

/NoLockError **(optional)**
Kann ein Datensatz nicht geändert werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

/MoveToHis, /MoveToBad, /Delete **(optional)**
Zur Verwendung der Optionen lesen Sie das Kapitel „**Optionen und Parameter von ERPAdmin**“

Beispiel:

Schreibt den Inhalt des Feldes **LiefDat** aus der CSV-Datei **Liefertermine.csv** in die Tabelle „**Bestelleingang**“ und ordnet die Datensätze anhand des Feldes **ID** zu.

ID	LiefDat
1538	30.11.2018
1539	14.10.2018
1537	02.12.2018

ERPAdmin -WriteDataFromFile Bestelleingang ID /Datei:Liefertermine.csv

151 Der Befehl „-XLS2CSV“ PRO

ERPAdmin -XLS2CSV <XLS-Datei> <CSV-Datei> <Excel-Tabellenblatt>

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

Mit diesem Befehl konvertiert **ERPAdmin** unter Verwendung von MS-Excel (**muss auf dem System installiert sein**) die **XLS-Datei** in eine **CSV-Datei**. Existiert die CSV-Datei bereits wird diese überschrieben.

<XLS-Datei>
Dateiname inkl. Endung und Pfadangabe für die Ergebnisdatei im Excel-Format.

<CSV-Datei>
Dateiname inkl. Pfadangabe für Datei im ANSI-CSV-Format

<Excel-Tabellenblatt>
Wird ein **Excel-Tabellenblatt** übergeben werden die Daten von diesem Tabellenblatt exportiert. Wird kein **Excel-Tabellenblatt** angegeben wird das aktuell gültige Excel-Tabellenblatt exportiert.

Beispiele

ERPAdmin –XLS2CSV C:\ERPAdmin\Adressen.xls C:\ERPAdmin\Adressen.csv
Konvertiert die Excel Datei **C:\ERPAdmin\Adressen.xls** in ein CSV-Datei **C:\ERPAdmin\Adressen.csv**. Dabei werden die Daten aus dem aktiven Tabellenblatt exportiert.

ERPAdmin –XLS2CSV C:\ERPAdmin\Adressen.xls C:\ERPAdmin\Adressen.csv Tabelle3
Konvertiert die Excel Datei **C:\ERPAdmin\Adressen.xls** in ein CSV-Datei **C:\ERPAdmin\Adressen.csv**. Dabei werden die Daten aus dem Tabellenblatt **Tabelle3** exportiert.

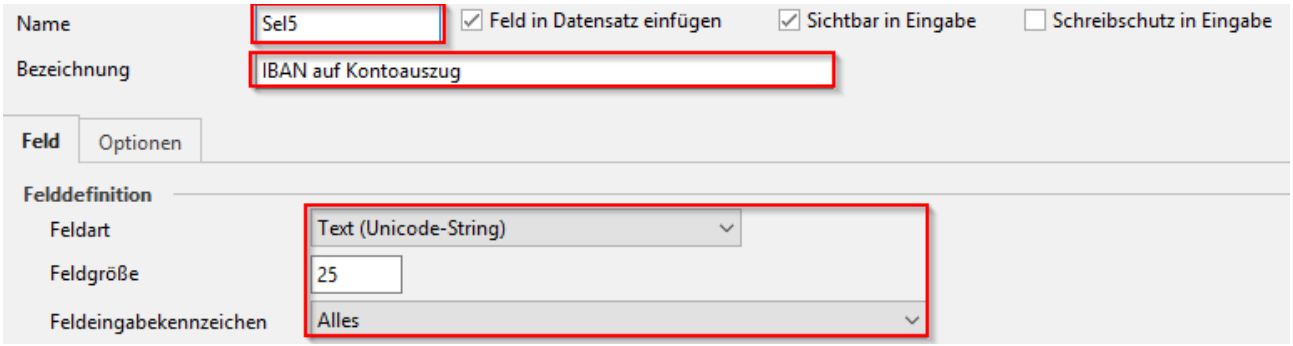
152 Die Befehle „-ZVEAdresseZuweisen“ und „ZVEEmpfaengerSpeichern“ PRO

Für die Verwendung dieses Befehls benötigen Sie die Professional-Version von **ERPAdmin**!

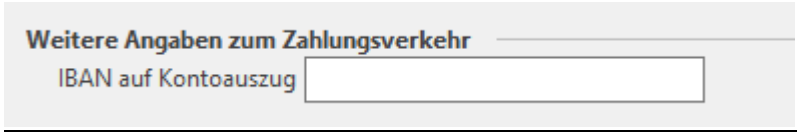
Nach dem Einlesen von Kontoauszügen im **büro+** Programmbereich **Kontoauszüge-Zahlungsverkehreingang** kann durch die Anwendung von Regel die Adress-Nummer erkannt und zugeordnet werden. Leider gestalte sich die Zuordnung dieser Daten oft aufwendig. **ERPAdmin** bietet Ihnen eine Möglichkeit, die einmal von Anwender getroffenen Zuordnungen zu einer Adresse zu speichern und später wiederholt anzuwenden ohne das Import-Regel oder Reguläre Ausdrücke bemüht werden müssen. Der Ablauf gliedert sich dabei in drei Schritte.

1. Schritt

Erzeugen Sie im **DBManager** in der Tabelle „**Adressen**“ ein Selektionsfeld vom Typ String mit min. 22 Zeichen Länge zur Speicherung einer IBAN. In unserem Fall hat das Selektionsfeld den Namen „**Sel5**“ und die Bezeichnung „**IBAN auf Kontoauszug**“.

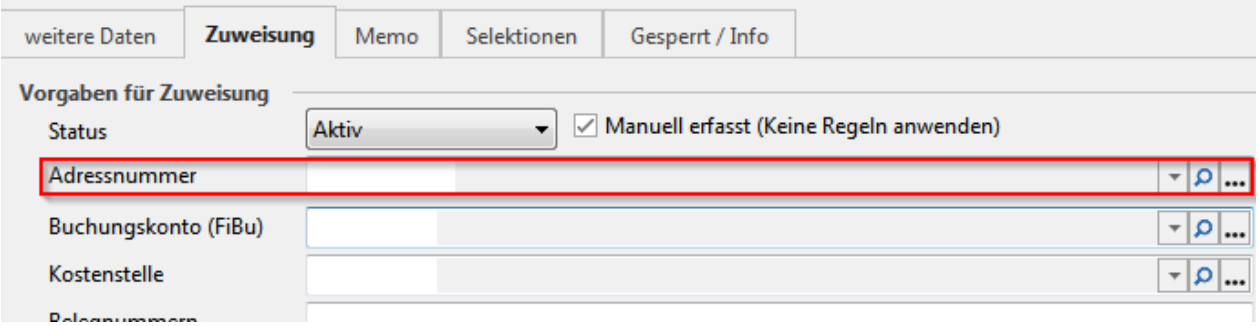


In den Selektionen der Adressen wird das Feld dann wie folgt angezeigt:



2. Schritt

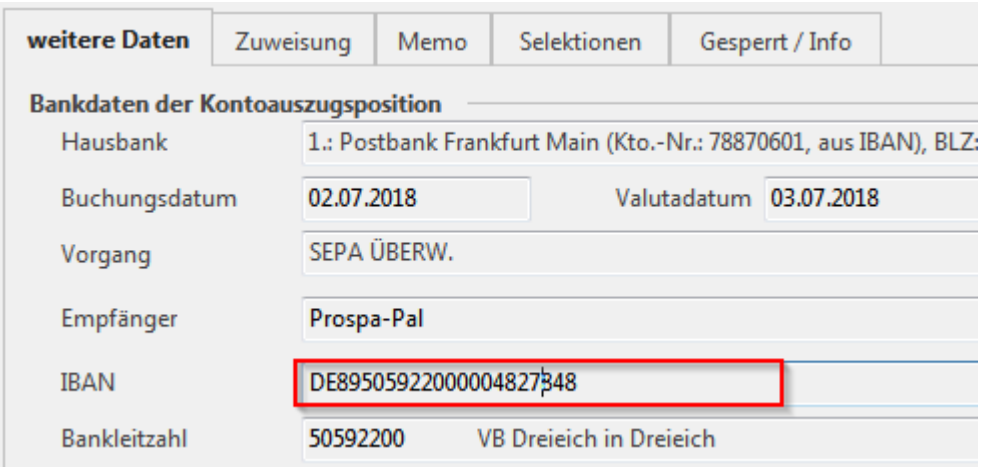
Nach dem Einlesen von Kontoauszügen im Bereich **Kontoauszüge-Zahlungsverkehrseingang** ordnet der Benutzer zunächst allen Datensätzen die entsprechenden offenen Posten zu. Dabei ist darauf zu achten, dass neben der Zuweisung der OP-Datensätze auch die **Adressnummer** des Kunden im **Zahlungsverkehrseingang** Datensatz hinterlegt wird.



Nachdem alle Datensätze bearbeitet wurden ruft der Anwender **ERPAdmin** mit folgendem Befehl auf:

ERPAdmin -ZVEEmpfaengerSpeichern <Feldname> /Datei:<Dateiname> /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff> /EmailCC:<E-Mail-Adresse> /EmailBCC:<E-Mail-Adresse> /NoLockError

Feldname
Gibt ein Feld aus den Stammdaten Adressen an, in dem die IBAN aus dem Zahlungsverkehrs-Datensatz gespeichert werden soll.



Wir empfehlen zu diesem Zweck ein eigenes Selektionsfeld in den Adress-Stammdaten anzulegen mit der Feldlänge 50 Zeichen. In diesem Feld werden dann die IBAN aus dem Zahlungsverkehr gespeichert.

/Datei:<CSV-Dateiname> **(optional)**
Gibt den Namen einer CSV-Datei an unter der ein Status-Report der gespeicherten IBANs angelegt wird. Wird kein Name angegeben, verwendet **ERPAdmin** den Dateinamen „**ZahlungsverkehrEingang_IBAN.csv**“.

/Email:<E-Mail-Adresse> **(optional)**
Mit dieser Option kann eine E-Mail-Adresse übergeben werden an die **ERPAdmin** die Ergebnisdatei **<CSV-Dateiname>** mit dem Ergebnis-Status sendet. Wurden keine Empfänger zugeordnet wird keine E-Mail versendet. Diese E-Mail wird ggf. zusätzlich zur OK-E-Mail versendet. Mehrere E-Mail-Adressen sind mit einem Semikolon zu trennen.

Wurde eine Ergebnisdatei erzeugt wird diese im History-Verzeichnis (**C_HistoryDir**) archiviert.

/Subject:<E-Mail-Betreff> **(optional)**
Übergibt eine Betreffzeile für den E-Mail-Versand an **ERPAdmin**. Neben den Standard Platzhaltern (Siehe „**Kapitel Platzhalter und variable Parameter**“) kann auch der Platzhalter **@FILENAME@** verwendet werden welcher den Dateinamen der Ergebnisdatei einfügt.

Nach dem Aufruf von **ERPAdmin** werden die Daten im Zahlungsverkehr ganz normal vom Anwender verarbeitet werden.

/EmailCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen E-Mail-Empfänger an. Soll die E-Mail an mehrere zusätzlichen Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/EmailBCC:<E-Mail-Adresse> **(optional)**
Gibt einen zusätzlichen nicht sichtbaren E-Mail-Empfänger an. Soll die E-Mail an mehrere nicht sichtbare zusätzliche Empfänger gesendet werden müssen die E-Mail-Adressen mit einem Semikolon getrennt werden.

/NoLockError **(optional)**
Kann ein Datensatz nicht geändert werden, weil diese gerade von einem anderen Benutzer bearbeitet wird, wird keine Fehlermeldung zurückgegeben.

Beispiel:
Zur Speicherung der IBAN wurde in den Adressen das Selektionsfeld **SEL5** angelegt.
ERPAdmin -ZVEEmpfaengerSpeichern SEL5

3. Schritt

Nach dem nächsten Einlesen neuer Kontoauszüge im Bereich **Kontoauszüge-Zahlungsverkehrseingang** ruft der Anwender als erstes **ERPAdmin** mit folgendem Befehl auf:

ERPAdmin -ZVEAdresseZuweisen <Feldname> /Datei: /Email:<E-Mail-Adresse> /Subject:<E-Mail-Betreff>

Die Verwendung der Parameter **Feldname** und **/Datei:** erfolgt analog zum Aufruf des Befehls **ZVEEmpfaengerSpeichern**

ERPAdmin ermittelt nun die Adressnummer anhand der bisher gespeicherten IBANs und trägt diese bei den jeweiligen Datensätzen ein.

Danach wiederholt der Anwender Schritt 1 für alle Datensätze bei denen noch keine Adressnummer zugeordnet wurde. Datensätze bei denen ein Konto zugeordnet wurde werden übersprungen.

Nach und nach lernt **büro+** so die IBANs der Adressen kennen und ordnet die Adressnummer den Zahlungsverkehrsdatensätzen automatisch zu.

Beispiel:
Zur Speicherung der IBAN wurde in den Adressen das Selektionsfeld **SEL5** angelegt. **ERPAdmin** vergleicht nun die IBAN aus dem Zahlungsverkehr mit dem in den in den Adressen gespeicherten IBANs. Bei einer Übereinstimmung wird die Adressnummer aus den Stammdaten in den Zahlungsverkehr übernommen.
ERPAdmin - ZVEAdresseZuweisen SEL5

Hinweis:
Nach der erstmaligen Ausführung von Schritt 1 und Schritt 2 dreht sich die Reihenfolge für alle nachfolgenden Bearbeitungen um. Der Benutzer führt dann nach dem Einlesen der Kontoauszüge immer erst Schritt 2 durch und nach der Zuordnung der Adressnummern Schritt 1.

153 Weitere Anwendungsbeispiele

Beispiel 1: Export und FTP-Upload
Artikeldaten aus **büro+** exportieren und per FTP an Webshop übertragen
ERPAdmin -Export "c:\ERPAdmin\ERP-Export.BPL" "c:\ERPAdmin\Artikelliste.xls"
ERPAdmin -FTPPUT "c:\ERPAdmin\Artikelliste.xls"

Beispiel 2: FTP-Downlad und Import
Download der Datei „**Bestellung.csv“** mit Webshop Bestellungen vom FTP-Server und Import in **büro+**
ERPAdmin -FTPGET "Bestellung.csv" "c:\ERPAdmin\ERP-Import.CSV"
ERPAdmin -Import "c:\ERPAdmin\Vorgang-Import.BPL" „c:\ERPAdmin\Bestellung.csv

Beispiel 3: Export und FTP-Upload
Exportiere Webshop-Artikel aus **büro+** und **übertrage** dieser per FTP zum Webshop
ERPAdmin -Export "c:\ERPAdmin\Artikel-Export.BPL" "c:\ERPAdmin\Webshopartikel.xls"
ERPAdmin -FTPPUT "c:\ERPAdmin\Webshopartikel.xls"

Starte danach den WebShop-Import durch Aufruf der URL http://www.webshop.de/Import.html

ERPAdmin -OpenURL " http://www.webshop.de/Import.html"

Beispiel 4: Reduziert die Anzahl von Dateien in einem Verzeichnis
Im Verzeichnis „**C:\Backup**“ sollen max. **5** Sicherungsdateien (die neuesten) erhalten bleiben.
ERPAdmin -DeleteFiles "c:\Backup" 5 "*.TIB"

Beispiel 5: Export/Import zwischen zwei büro+/ERP-Mandanten
Artikel aus Mandant 1 exportieren und direkt wieder in Mandant 2 importieren
ERPAdmin -Export "c:\ERPAdmin\Artikel-Export.BPL" /Mand1
ERPAdmin -Import "c:\ERPAdmin\Artikel-Import.BPL" /Mand2

Hinweis: Um Daten zwischen zwei Mandanten auszutauschen, müssen in der **VAR.INI** die Zugangsdaten für einen 2. Mandanten hinterlegt sein.

Beispiel 6: Vorgang Wandeln und Buchen
Alle Vorgänge „**Bestellung vom Kunden**“ (**30**) die komplett lieferbar sind (**3**) sollen in „**Lieferscheine**“ (**50**) gewandelt und anschließend verbucht (/**BUCHEN**) werden.
ERPAdmin -VorgangWandeln 30 50 3 /BUCHEN

Beispiel 7: Vorgang Wandeln wenn bezahlt und dann Buchen
Alle Vorgänge „**Bestellung Vorkasse**“ **(101**) sollen in einen „**Lieferschein**“ (**50**) gewandelt werden wenn diese bezahlt sind (/**BEZAHLT**) und dann verbucht werden.
ERPAdmin -VorgangWandeln 35 50 0 /BUCHEN /BEZAHLT

Beispiel 8: Kunden Zahlungsbedingung zuweisen
ERPAdmin allen Adressen mit Status „Kunde“ die Zahlungsbedingung „10 Tage 2%, 30 Tage netto“ zu
ERPAdmin -WriteData "ZahlBed=10 Tage 2%%, 30 Tage netto;SkToTg1=10;SktoSz1=2;NettoTg=30" Status=Kunde

Beispiel 9: Setzten bei allen Artikel das Rabatt-, Skonto- und Provisions-Kennzeichen
Bei alle Artikel im Programm soll das Rabatt-Kennzeichen für VK0, VK1, VK2 sowie das Skontofähig- und

das Provisionsfähig-Kennzeichen gesetzt werden.
ERPAdmin -WriteData Artikel "SktoKz=Ja;ProvKz=Ja;VK0.RabKz=Ja;VK1.RabKz=Ja;VK2.RabKz=Ja;"

Hinweis: Je nach Menge der Artikel kann die Verarbeitung mehrere Minuten dauern.

Beispiel 10: Negative bestände ausbuchen
Buche alle negativen Lagerbestände aus dem STAMMLAGER aus
ERPAdmin -LagerbestandAusbuchen STAMMLAGER STAMMLAGER /Filter:Mge<0

Beispiel 11: Buche alle Lagerbestände der Kundenlagern aus

Buche alle Lagerbestände die sich in Kundenlagern befinden aus
ERPAdmin -LagerbestandAusbuchen KD10000 KD69999

Beispiel 11: Verschiebe alle alten Angebote ins Archiv

Verschiebe alle Angebote aus dem Jahr 2016 ins Vorgangs-Archiv
ERPAdmin -VorgangInArchiv 15 FALSE /Von:Dat=01.01.2016 /Bis:Dat=31.12.2016

Beispiel 12: Fasse mehrere Dateien mit Bestellungen zu einer Datei zusammen.

Da alle Dateien einen Header enthalten muss dieser zuerst entfernt werden.
ERPAdmin -CSVKonverter c:\ERPAdmin\Bestelungen*.csv DeleteHeader
ERPAdmin -AppendANSIFIle c:\ERPAdmin\Bestelung.csv c:\ERPAdmin\Bestelungen*.csv

Wird in der Datei „Bestellung.csv“ eine Kopfzeile benötigt würde der Aufruf wie folgt lauten:

ERPAdmin -AppendANSIFile c:\ERPAdmin\Bestelung.csv c:\ERPAdmin\Kopfzeile.csv c:\ERPAdmin\Bestelungen*.csv

Beispiel 13: Lösche eine Liste von alten Artikel aus der Datenbank

Ein Lieferant liefert regelmäßig eine Liste mit Artikel-Update. Darunter auch eine Liste der Artikel die schon lange ausgelistet sind und die gelöscht werden können, weil diese älter sind als 10 Jahre. In Spalte 4 der CSV-Datei **c:\ERPAdmin\Löschliste.csv** befinden sich die Artikelnummern.

ERPAdmin -DeleteDataFromFile Artikel 4 ArtNr c:\ERPAdmin\Löschliste.csv

Beispiel 14: Exportiere Artikel-Liste und versende diese per Email

Kombiniert man den Export-Befehl mit dem E-Mail-Versand, kann ein Export an einen Benutzer per E-Mail gesendet werden. Dazu müssen die beiden Befehle nacheinander ausgeführt werden.

ERPAdmin -Export "Artikel.bpl" "C:\ERPAdmin\Artikel_Export.csv"
ERPAdmin -Email "Kunde@test.de" "Neue Artikeldaten" " " "C:\ERPAdmin\Artikel_Export.csv"

154 Automatisierung von ERPAdmin mit dem Windows Aufgabenplaner

Damit ein Export- oder Import automatisiert werden kann, muss sich gestellt sein, dass die Laufwerke auf die Exportiert bzw. Importiert werden soll, auch zu diesem Zeitpunkt zugegriffen werden kann.

Da gemappt Laufwerke zu diesem Zeitpunkt in der Regel nicht zur Verfügung stehen, sollten alle Pfadangaben im UNC Format (\\Servername\Freigabename\Dateiname) gemacht werden. Außerdem ist darauf zu achten, dass der verwendete Benutzer (Sicherheitsoptionen) auch die notwendigen Zugriffsrechte auf den Pfad besitz und zur Ausführung der Windows Aufgabenplanung.

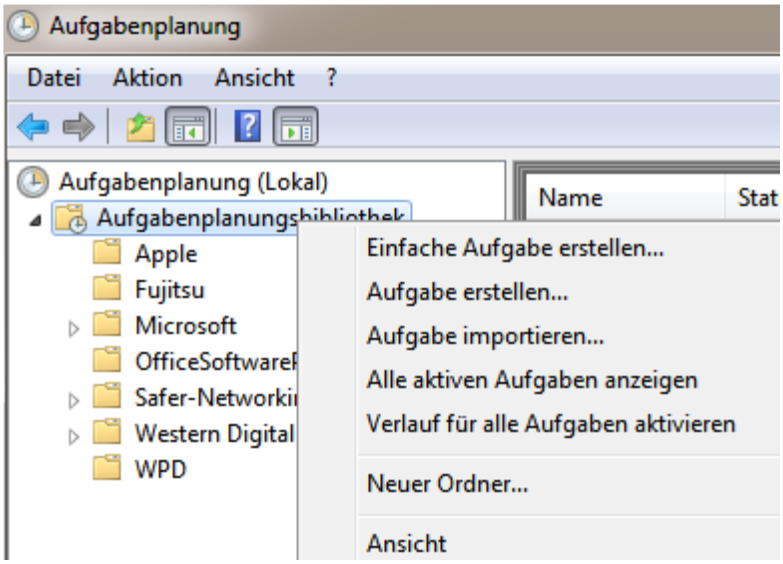
Einrichtung Aufgabenplaner

Der Windows Aufgabenplaner befindet sich in der „**Windows Programmverwaltung**“.

Um eine Aufgabe hinzuzufügen gehen Sie wie folgt vor:

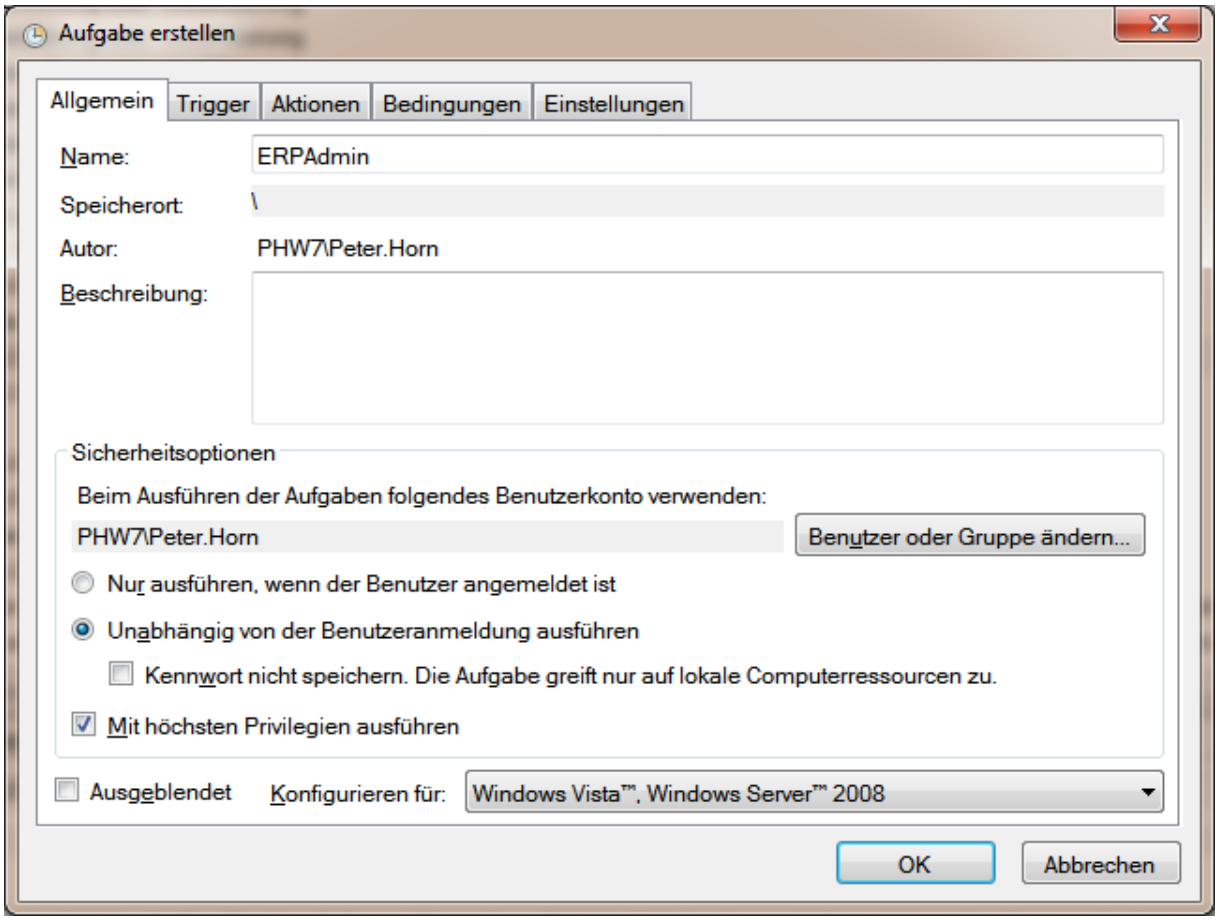
1.) Starten Sie die Windows **Aufgabenplanung**.

2.) Klicken Sie mit der rechten Maustaste auf „**Aufgabenplanungsbibliothek**“ und wählen Sie „**Aufgabe erstellen…**“ aus



3.) Geben Sie dem Auftrag einen „**Namen**“ und aktivieren Sie „**Unabhängig von der Benutzeranmeldung ausführen“** und „**Mit höchsten Privilegien ausführen**“.

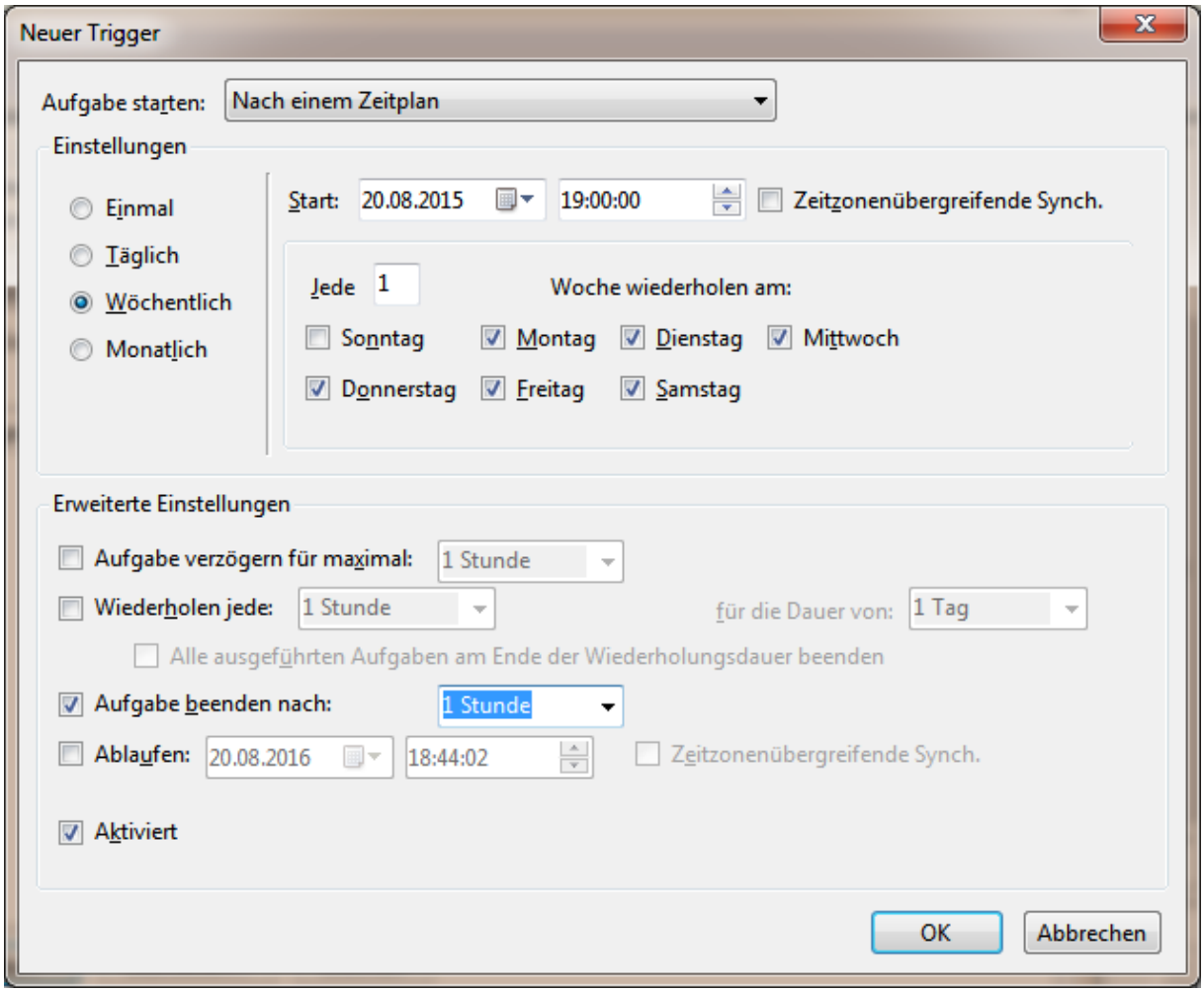
4.) Wählen Sie unter „**Konfiguriert für:**“ das verwendete Betriebssystem Ihres Rechners aus.



Falls Ihr Benutzer mit dem Sie aktuell angemeldet sind keine Administrator Privilegien besitzt, Klicken Sie auf den Button „**Benutzer und Gruppe ändern**“ und wählen Sie einen Administrator oder einen 100% äquivalenten Benutzer aus.

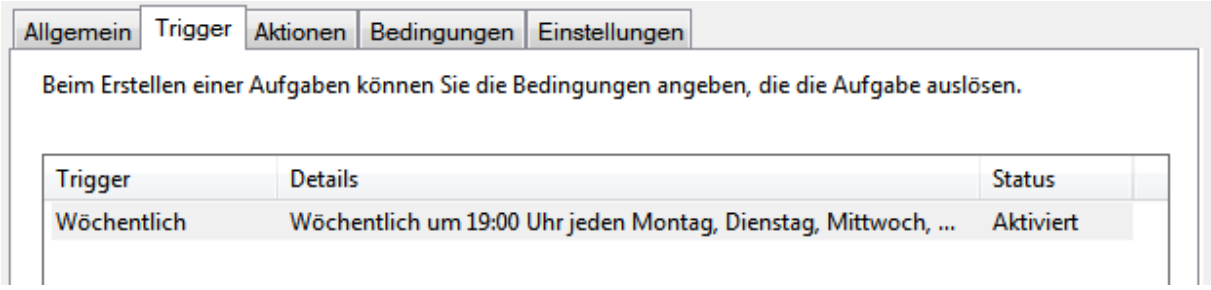
5.) Wechsel Sie auf das Register „**Trigger**” und Klicken Sie hier aus „**Neu**“ um den Zeitpunkt für die Ausführung festzulegen.

Wählen Sie Ihren Ausführungszeitunkt aus, Aktivieren Sie „**Aufgabe beenden nach:“** und tragen einen sinnvollen Wert ein. Damit soll verhindert werden, dass eine Instanz von **ERPAdmin** im Fehlerfall ewig läuft.

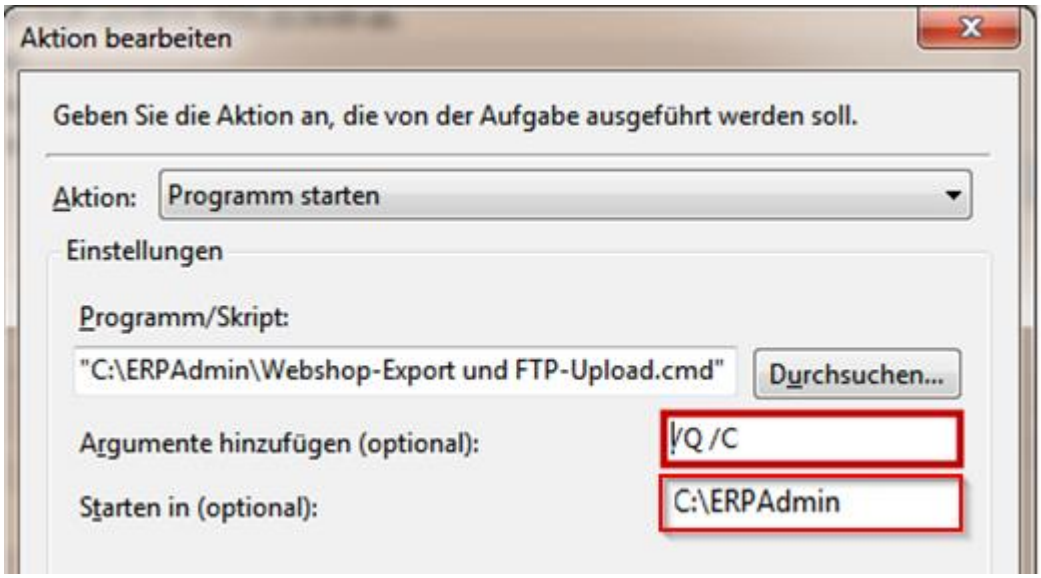


Bestätigen Sie mit „**OK**“

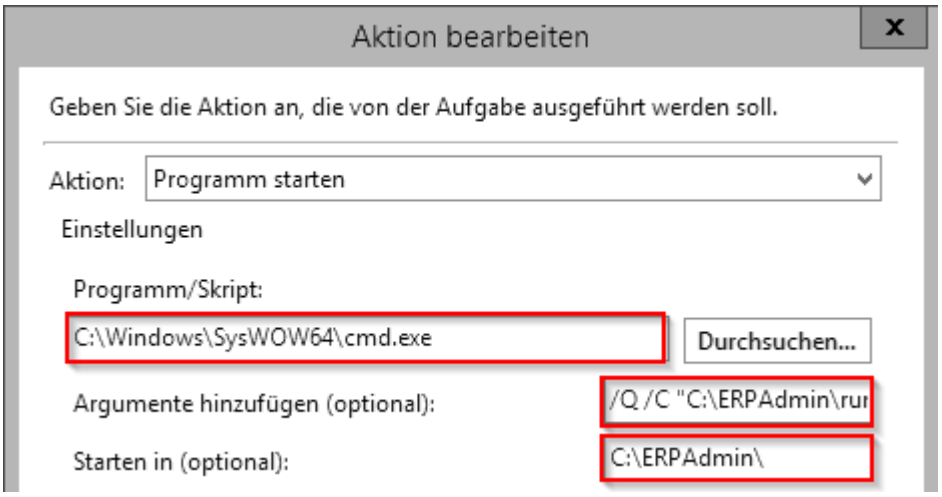
Es befindet sich jetzt ein Trigger in Ihrer Aufgabenliste.



6.) Wechseln Sie nun auf den Reiter „**Action**“ und Klicken auf den Button „**Neu**“ um die CMD-Datei auszuwählen Sie Sie aufrufen wollen. Wählen Sie durch Klick auf „**Durchsuchen**“ die gewünschte CMD-Datei aus.



Wenn bei **Windows Server 2012/2016/2019/2022** der Aufruf über die 32-Bit Umgebung erfolgen soll, sieht dieser wie folgt aus:



Hinweis: Bei Server-Betriebssystemen (z.B. Windows Server 2012/2016/2019/2022) kann es notwendig sein, als Argument **„/Q /C“** mit anzugeben.
/Q = Quiet Mode
/C = Beendet des Task nach Ausführung

Bestätigen Sie Ihre Auswahl durch Klick auf „**OK**”

Hinweis:
Bei Server 2012/2016/2019/2022 „**kann**“ die Option „**Mit höchsten Privilegien ausführen**“ manchmal zu Problemen bei der Ausführung führen. Deaktivieren Sie diese Option in diesen Fällen.

Hinweis 2:
In dem Feld „**Starten in (optional)**“ dürfen keine Anführungszeichen verwendet werden.

7.) Beenden Sie nun die Bearbeitung ihres Auftrags durch Klick auf „**OK**“.

8.) Testen Sie die Ausführung Ihrer Aufgabe durch rechtsklick auf die Aufgabe und wählen Sie „**Ausführen**“ aus

9.) Testen Sie ob die Aufgabenplanung alle Rechte besitzt um **ERPAdmin** korrekt auszuführen. Rufen Sie dazu aus der Aufgabenplanung den Befehl „**ERPAdmin -INIT“** auf und prüfen Sie das Logfile anschließend auf Fehlermeldungen.

155 Tipps & Tricks

Probleme beim ausführen von ERPAdmin über die Aufgabenplanung

Sie haben ein **ERPAdmin** Script erstellt in Form einer .**CMD** Datei. Diese lässt sich von Hand problemfrei aufrufen und ausführen. Wird diese jedoch über die Windows Aufgabenplanung ausgeführt, kommt es zu einem Fehler beim Initialisieren von Objekten.

Die Ursache hierfür sind meistens fehlende Berechtigungen. Führen Sie die Installation der beiden MSI-Installer erneut mit dem Benutzer durch, der auch später im Aufgabenplaner als Benutzer hinterlegt wird.

Haben Sie Probleme bei der Initialisierung von Office-Komponenten (Excel, Word, Outlook) führen Sie mit dem Benutzer Eine Reparatur-Installation durch der auch später im Aufgabenplaner verwendet wird.

Probleme bei der Verwendung von Umlauten

Es kann vorkommen, dass im Kontext der Eingabeaufforderung Umlaute nicht korrekt übergeben bzw. dargestellt werden. Dies kann durch die Umschaltung der Code-Page behoben werden. Ergänzen Sie dazu die aufrufende CMD-Datei am Anfang mit folgendem Befehlt: **CHPC 1252**

Probleme beim Verwenden von Freigaben

Bei dem Zugriff auf System-Ressourcen sollte immer darauf geachtet werden, dass diese auch im Kontext der Programmausführung zur Verfügung stehen. Wird **ERPAdmin** aus dem Windows Scheduler aufgerufen, stehen z.B. Freigaben nicht standardmäßig zur Verfügung. Daher empfehlen wir nur auf physisch vorhandene Laufwerke anzusprechen, die aus Sicht der Maschine auf der **ERPAdmin** vorhanden sind.

Probleme beim Export oder Import von Archiv-Vorgängen

Beim Export von Daten aus dem Vorgangsarchiv stimmt der Export von **ERPAdmin** nicht mit dem händischen Export/Import überein. Ursache hierfür ein eine fehlerhafte Angabe in den Export/Import Layouts von **büro+** welche sich immer auf den Bereich der „**Vorgänge“** beziehen. Ergänzen Sie Ihren Export/Import Aufruf um die Option **/HIS**.

Welche Tabelle/Datenbank muss angegeben werden bei der Ausgabe einer Inventur-Liste?

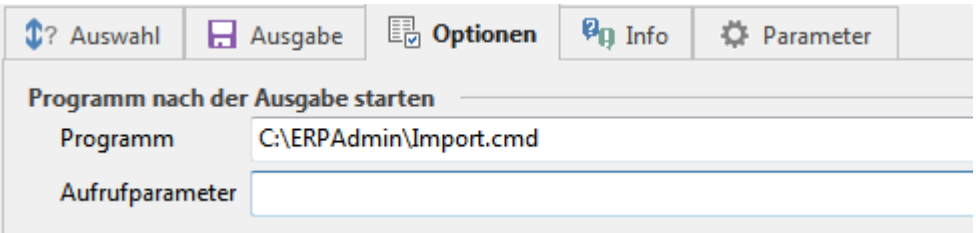
Für die Ausgabe von Inventur- oder Fehlmengenlisten muss als Tabelle/Datenbank „**Lagerbestand**“ angegeben werden, weil sich diese Listen aus diesen Bereich beziehen.

Eingabefenster schließt sich nicht nach Ausführungsende

Wird **ERPAdmin** aus einer CMD-Datei heraus aufgerufen, kommt es unter Windows 10 manchmal vor, dass dieses CMD.EXE-Fenster nicht geschlossen werden. Abhilfe kann der Befehl „**Exit /B**“ am Ende der CMD-Datei bringen.

Aufruf einer CMD-Datei nach einem Export aus büro+

Wird **ERPAdmin** direkt nach einem Export aus **büro+** ausgerufen müssen immer absolute Pfade in der aufgerufenen CMD-Datei verwendet werden.



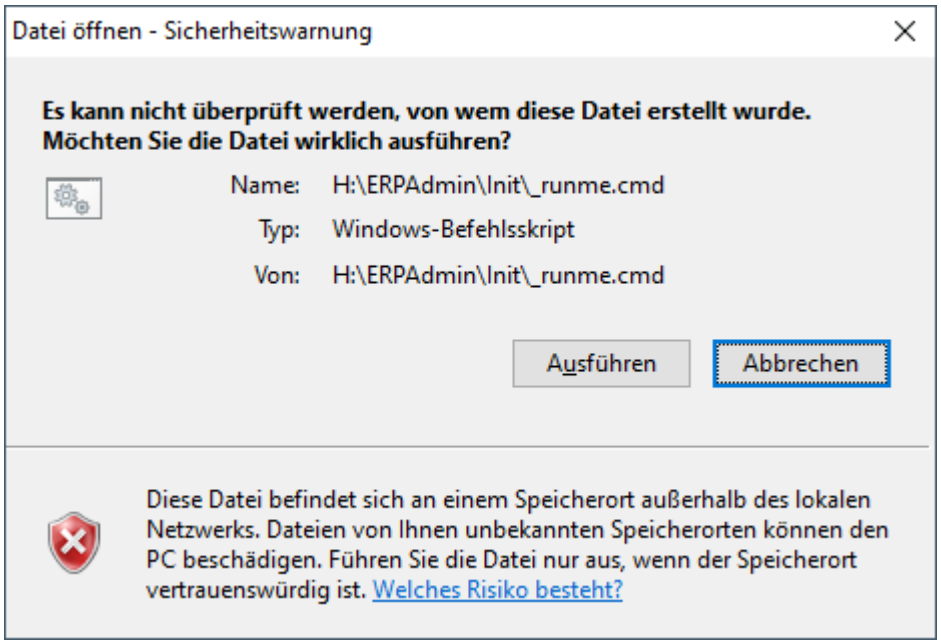
Deaktivierung von ERPAdmin während Wartungsarbeiten

Bei der Durchführung von Wartungsarbeiten am **ERPAdmin**-PC kann es hinderlich sein, wenn der Scheduler in dieser Zeit **ERPAdmin** mehrfach aufruft und damit Fehlermeldungen produziert. Jeder dieser Aufrufe der fehlschlägt erzeugt dann eine Fehler-E-Mail. Die Ausführung von **ERPAdmin** kann unterbunden werden indem in dem Verzeichnis die Datei „**ERPAdmin_STOP**“ erzeugt wird. Solange diese Datei existiert wird **ERPAdmin** sofort nach dem Start wieder beendet ohne eine Rückmeldung an den Benutzer zu erzeugen. Nach Beendigung der Wartungsarbeiten muss die Datei wieder gelöscht werden.

Alternativ kann auch in der **VAR.INI** der Eintrag **C_STOP=TRUE** hinzugefügt werden.

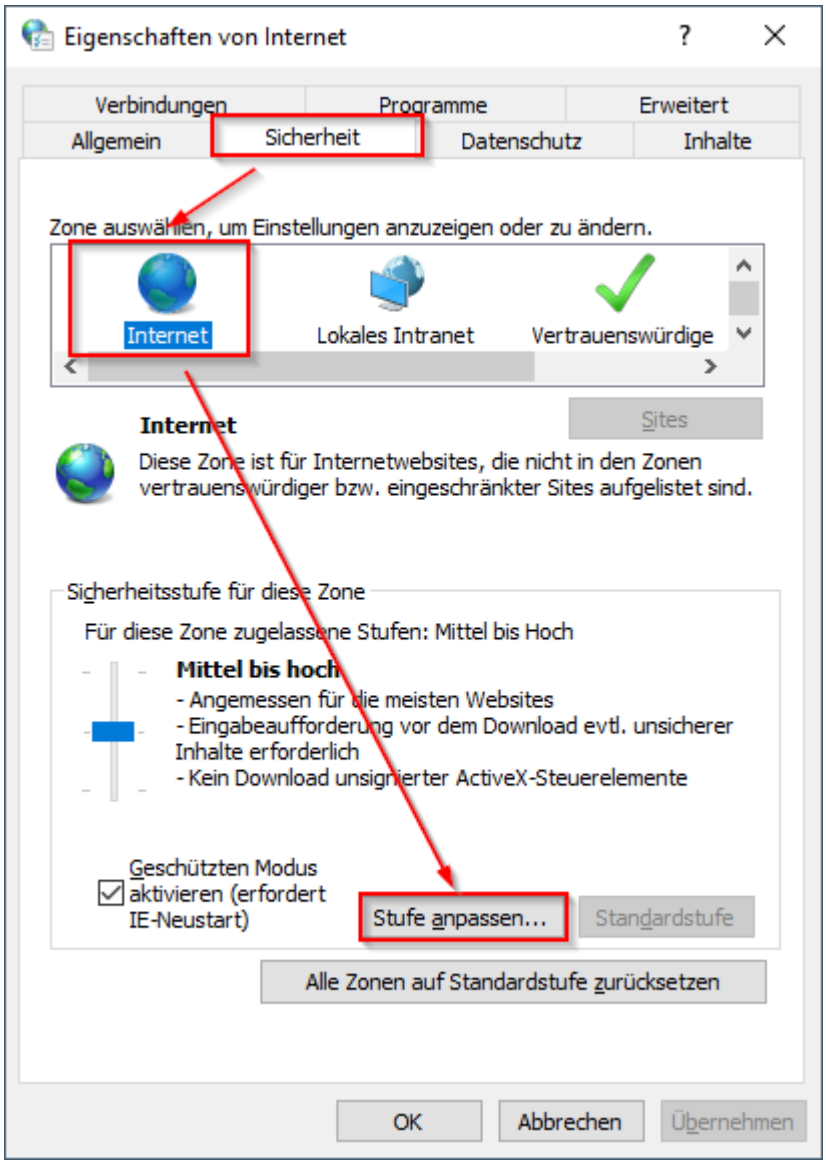
Beim Ausführen einer CMD-Datei auf einem Netzwerk-Laufwerk kommt eine Sicherheitswarnung

Es erscheint nachfolgende Sicherheitswarnung, wenn Sie eine Datei auf einem Netzwerk-Laufwerk aufrufen:

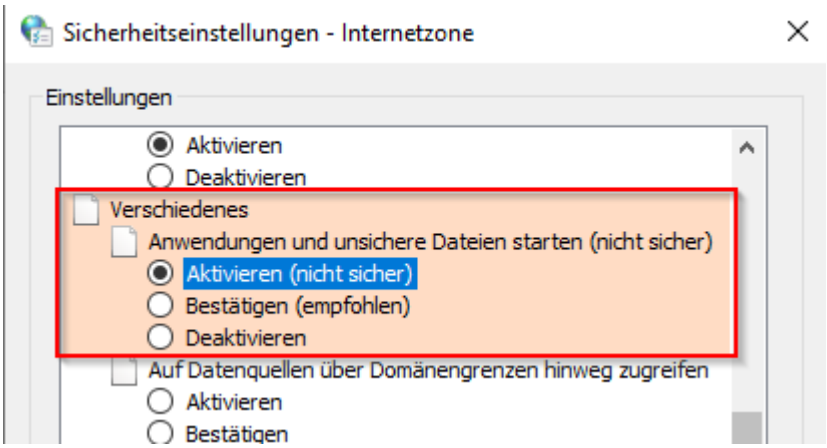


Um diese Sicherheitsabfrage zu deaktivieren, starten Sie über „Ausführen“ das Programm „**inetcpl.cpl**“ oder öffnen Sie in der Systemsteuerung die „**Internetoptionen**“.

Wechsel Sie auf das Register „**Sicherheit**“ und wählen dort „**Internet**“ und dann „**Stufe anpassen…**“ aus.



Aktivieren Sie im Bereich „**Verschiedenes**“ die Option „**Anwendungen und unsichere Dateien Starten (nicht sicher)**“ die Option.



Sie sollten nun CMD-Dateien auf Netzwerk-Laufwerken ohne Meldung ausführen können.

156 Systemvoraussetzungen

- **büro+** Version 22 ab BUILD **6810** (Alle Ausbaustufen)
- Mindestens Windows 10 oder Server 2016
- Direkter Zugriff auf **büro+**
- Benutzer mit **büro+** COM-Berechtigung
- Mind. eine freie COM-Zugriffslizenz während der Programmlaufzeit
- Für die Nutzung der Excel-Funktionen wird Microsoft Excel 2013 oder höher auf dem Rechner benötigt in einer 32-Bit Installation.
- Für die Nutzung der Word-Funktionen wird Microsoft Word 2013 oder höher auf dem Rechner benötigt in einer 32-Bit Installation.
- Für die Nutzung der Outlook-Funktionen wird Microsoft Outlook 2013 oder höher auf dem Rechner benötigt in einer 32-Bit Installation.
- Für Druckausgabe von PDF-Dateien durch **ERPAdmin** (nicht durch **büro+**) wird eine Installierte Version des Acrobat-Readers benötigt.
- Microsoft Collaboration Data Objects (CDO) für den E-Mail-Versand
- SMTP-Zugang (Mit oder ohne SSL Verschlüsselung) für den Versand von E-Mails (Office 365 mit OAuth2 wird **nicht** unterstützt)
- Installation der mitgelieferten Chilkat-DLLs, mindestens Version **9.5.0.94**

157 Anregungen, Kritik und Verbesserungsvorschläge

Über jede Form von Kritik, Anregungen oder Verbesserungsvorschlägen freuen wir uns. Teilen Sie uns Ihre Erfahrungen gerne mit und helfen Sie uns **ERPAdmin** zu verbessern. Selbstverständlich erhalten Sie die verbesserte bzw. geänderte Version von uns kostenfrei, sofern Sie bereits eine gültige Lizenz von **ERPAdmin** besitzen und sich als Beta-Tester zur Verfügung gestellt haben. Es besteht jedoch kein Anspruch auf die Umsetzung eingereichter Verbesserungsvorschläge.