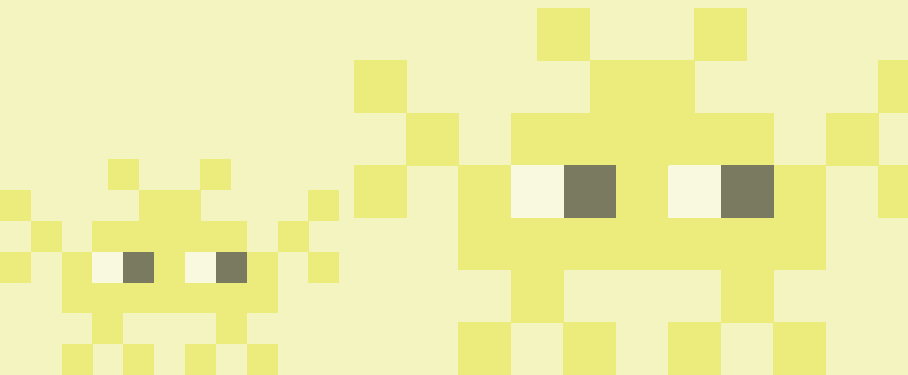


PROFESSIONAL



EURO-BEAMER

BANKDATEN KONVERTIEREN



Manual zum **EURO-BEAMER**.

Für den Inhalt verantwortlich: Heli Ammann.

Alle Rechte vorbehalten.

Grafik: www.allesgrafik.at

Zum Geleit:

Die prinzipielle Funktion des **EURO-BEAMER** entnehmen Sie bitte der Dokumentation „Manual EURO-BEAMER.pdf“.

Im Folgenden sind nur die Unterschiede der Professional-Version zur Standard-Version erläutert.

1.	Installation	4
	Installation auf dem Entwicklungsrechner	
	Installation auf dem Anwendungsrechner	
2.	Bedienung	6
	Eigenschaften	
	Methoden	
	Warnungen	
	Fehler	
3.	Anwendungsbeispiel	12
	Visual Fox Pro	
	ASP	
4.	Support	14

1. Installation

1.1. Installation auf dem Entwicklungsrechner

Mit Aufruf von setup.exe auf der Installations-CD wird **EURO-BEAMER Professional** installiert.

Es ist KEINE Eingabe einer Lizenznummer nötig.

Zusätzlich befinden sich auf der CD:

- Im Unterverzeichnis „EUROBEAMER_EXE“: die Einzelversion von **EURO-BEAMER**, damit kann die Funktion des **EURO-BEAMER** mit der Benutzeroberfläche getestet werden, ohne programmtechnisch eingebunden werden zu müssen (Installation und Bedienung: siehe „Manual EURO-BEAMER.pdf“).
- Im Unterverzeichnis: „EUROBEAMER_DLL“: Setup-Programm, das lediglich die **EURO-BEAMER Professional-DLL** („Runtime“) auf einem PC installiert, ohne Einbindung in das Startmenü und ohne Dokumentation.

1.2. Installation auf dem Anwendungsrechner

Die **EURO-BEAMER**-DLL kann in eigene Applikationen eingebunden und mitvertrieben werden. Dazu muss die **EURO-BEAMER**-DLL auf dem PC des Anwenders installiert werden.

Für diese Installation gibt es 2 Möglichkeiten:

- Das Setup-Programm im Unterverzeichnis „EUROBEAMER_DLL“ auf der mitgelieferten CD installiert die **EURO-BEAMER**-DLL und alle weiteren nötigen DLLs.
- Die **EURO-BEAMER**-DLL und alle von dieser benötigten weiteren DLLs werden manuell in die Installationsroutine der Anwendung übernommen:
 - ▶ eurobeam.dll (Standardverzeichnis c:\programme\eurobeam_dll)
 - ▶ gdiplus.dll, vfp8r.dll, vfp8renu.dll, vfp8t.dll, vfp8rdeu.dll (Standardverzeichnis c:\Programme\Gemeinsame Dateien\Microsoft Shared\VFP)
 - ▶ msvcr70.dll (Standardverzeichnis c:\windows\system32)

Hinweis: Wenn alle benötigten DLLs in einem Verzeichnis stehen und eurobeam.dll registriert wird (Eingabe: regsvr32 eurobeam.dll), werden die anderen DLLs automatisch registriert.



2. Bedienung

Die Konvertierung erfolgt, wie in „Manual EURO-BEAMER.pdf“ erläutert, statt der Eingabemaske steht aber die Klasse *ediconv* mit folgenden Eigenschaften und Methoden zur Verfügung:

2.1. Eigenschaften

2.1.1. Vom Benutzer einzugebende Eigenschaften

objekt.txtFrom.Value	Eingabe-Datei (Textfeld)
objekt.txtTo.Value	Ausgabe-Datei (Textfeld, optional)
objekt.cDataFrom	zu konvertierende Daten im CSV- oder V2-Format (DIRDEB- / PAYMUL-Konvertierung) oder EDIFACT-Format (CREMUL- / DEBMUL-Konvertierung). (Textfeld, Alternative zum Einlesen einer Datei mit txtFrom.Value)
objekt.chkV2U2	Gibt an, ob beim Erstellen von V2-Daten der Umsatz-Satz 2 mit Einzahler- und Empfängername generiert werden soll (kann von manchen Programmen nicht verarbeitet werden). (logisches Feld, nur bei Konvertierung ins V2-Format nötig).
objekt.txtTempPath.Value	Hier kann man den temporären Pfad bestimmen, den EURO-BEAMER verwenden soll (Schreib- und Löschrechte benötigt); z.B.: „;c:\temp\“ (mit abschl. Backslash).

Folgende Eigenschaften entsprechen den Optionen lt. „Manual EURO-BEAMER.pdf“:

Eigenschaft	max. Länge	Option
objekt.txtKontowortlaut.Value	35	Kontowortlaut
objekt.txtKontonummer.Value	11	Kontonr.
objekt.txtBLZ.Value	5	BLZ
objekt.txtBankkennung.Value	35	Bankkennung
objekt.txtStrasse.Value	35	Strasse
objekt.txtPLZ.Value	4	PLZ
objekt.txtOrt.Value	35	Ort
objekt.txtZahlzweck.Value	35	Zahlungszweck
objekt.txtBuchungstext1.Value	70	Zusatzzeilen
objekt.txtBuchungstext2.Value	70	
objekt.txtBuchungstext3.Value	70	
objekt.txtBuchungstext4.Value	70	
objekt.txtBuchungszeile.Value	12 ¹	Buchungszeile (Empf.)
objekt.txtDatum.Value	(„TT.MM.JJJJ“)	Datum

¹ Die Empfänger-Buchungszeile sollte nur Grossbuchstaben und Ziffern, keine Leer- und Sonderzeichen enthalten.

objekt.txtBelegart.Value	2	Belegart
objekt.txtTeilnehmer.Value	35	Teilnehmer- kennung
objekt.txtDatenaustauschreferenz.Value	14	Datenaustausch- referenz
objekt.txtNachrichtenreferenz.Value	35	Nachrichten- referenz
objekt.txtKundennummer.Value	35	Kundennummer
objekt.txtKontaktperson.Value	35	Kontaktperson
objekt.txtKontakttelefon.Value	128	Telefon Kontakt- person
objekt.chkDTZus.Value	Logisches Feld	DTAUS-Erweite- rungsteile immer = Verw.zweck; F=nein (Stan- dard), T=ja
objekt.txtLogPfad.Value	250	Log-Pfad



Alle Eigenschaften sind Textfelder.

Ein gesetzter Wert wird gespeichert und ist auch beim nächsten Aufruf der DLL vorhanden. Es wird empfohlen, vor jeder Konvertierung alle Eigenschaften zu initialisieren, da u.U. Werte vom letzten Aufruf aktiv sein könnten.

Die einzulesenden Daten können wahlweise via Datei (objekt.txtFrom.Value) oder gleich als Wert (objekt.cDataFrom) übergeben werden. Die konvertierten Daten können in eine Datei geschrieben werden (objekt.txtTo.Value) oder als Wert ausgelesen werden (objekt.cDataTo).

2.1.2. Vom Programm zurückgelieferte Eigenschaften

objekt.cDataTo	konvertierte Daten im CSV-Format (CREMUL- / DEBMUL-Konvertierung) oder EDIFACT-Format (DIRDEB- / PAYMUL-Konvertierung) (Textfeld, Alternative zum Schreiben der Ergebnisdaten in eine Datei mit txtTo.Value)
objekt.ErrorNum	Fehlernummer (numerisches Feld)
objekt.ErrorText	Fehlertext (Textfeld)
objekt.WarningNum	Warnungsnummer (numerisches Feld)
objekt.WarningText	Warnungstext (Textfeld)
objekt.nAnzahlDIRDEB	Anzahl der konvertierten Lastschriften (numerisches Feld)
objekt.nSummeDIRDEB	Summe der konvertierten Lastschriften (numerisches Feld)
objekt.nAnzahlPAYMUL	Anzahl der konvertierten Überweisungen (numerisches Feld)
objekt.nSummePAYMUL	Summe der konvertierten Überweisungen (numerisches Feld)
objekt.nAnzahl	Anzahl der gesamten konvertierten Buchungen (numerisches Feld)
objekt.nSumme	Summe der gesamten konvertierten Buchungen (numerisches Feld)

2.2. Methoden

objekt.cmdStart.Click	Beginnt die Konvertierung
objekt.cmdError.Click	Erzeugt einen Fehler (für Testzwecke)

Wenn mit txtFrom.Value oder cDataFrom die zu konvertierenden Daten definiert wurden, kann mit cmdStart.Click die Konvertierung begonnen werden.

2.3. Warnungen

Wenn einige der Lastschriften nicht vollständig sind, aber die restlichen Buchungen konvertiert werden können, wird eine Warnung ausgegeben (WarningNum und WarningText) und die Konvertierung fortgesetzt. Konvertierung und etwaige Warnungen werden in der Datei einz_log.txt (im Programmdatenverzeichnis, Standard c:\Dokumente und Einstellungen\All Users\Anwendungsdaten\EUROBEAM bzw. c:\ProgramData\EUROBEAM\ bei Vista) protokolliert. Eine Liste der nicht durchgeführten Buchungen wird in der Datei einz_err.txt protokolliert.

2.3.1. Warnungsnummer / Warnungsmeldungen

- 100 Falsche oder fehlende Informationen bei manchen Einzugsaufträgen
- 101 Buchungen werden ohne Zahlungszweck erstellt
- 102 Manche Buchungen werden ohne Zahlungszweck erstellt
- 103 Prüfsummenfehler (fehlerhafte EDIFACT-Datei oder Verarbeitungsfehler; bitte Kopie der Ursprungsdatei an: support@ammann.info senden)
- 104 Auslandslastschriften werden von EDIFACT eigentlich nicht unterstützt (Datei wird trotzdem erstellt)

2.4. Fehler

Wenn keine Konvertierung durchgeführt werden kann, weil wichtige Optionen fehlen oder ungenügende Daten vorhanden sind, wird der Fehler in ErrorNum und ErrorText geschrieben und die Konvertierung abgebrochen. Zum Reagieren auf Fehlermeldungen wird die Verwendung von ErrorNum (Fehlernummer) empfohlen, da sich der Inhalt von ErrorText in zukünftigen Programmversionen ändern kann.

2.4.1. Fehlernummern / Fehlermeldungen

- 0 keiner
- 1 Konvertierung erfolgreich beendet
- 10 VFP-Systemfehlermeldung
- 11 undefinierbarer Fehler
- 12 einzulesende Datei nicht gefunden
- 13 Fehler beim Verarbeiten der CREMUL/DEBMUL-Datei
- 14 Pfad für Ausgabe-Datei nicht korrekt
- 15 Zu konvertierende Daten nicht im korrekten Format (EDIFACT-, CSV- oder V2-Daten)
- 16 Kein einlesbares Dateiformat
- 17 Keine Daten in der Datei für DIRDEB/PAYMUL-Konvertierung
- 18 Daten enthalten nicht konvertierbare EDIFACT-Nachrichtentypen (es können ausschließlich CREMUL- und DEBMUL-Daten verarbeitet werden)
- 20 Fehlende Bankkennung
- 21 Zu lange Bankkennung (max. 35 Zeichen)
- 22 Fehlender Kontowortlaut
- 23 Zu langer Kontowortlaut (max. 35 Zeichen)
- 24 Zu lange Datenaustauschreferenz (max. 14 Zeichen)
- 25 Zu lange Teilnehmerkennung (max. 35 Zeichen)
- 26 Zu lange Nachrichtenreferenz (max. 35 Zeichen)
- 27 Zu lange Kundennummer (max. 35 Zeichen)
- 28 Keine korrekte Auftraggeberkontonummer
- 29 Keine korrekte Auftraggeber-BLZ
- 30 Zu lange Kontaktperson (max. 35 Zeichen)
- 31 Zu lange Kontakttelefonnummer (max. 128 Zeichen)
- 50 Falsche oder fehlende Informationen für den gesamten Auftragsbestand
- 51 Falsche oder fehlende Informationen bei allen Aufträgen
- 70 Fehler beim Einlesen der DTAUS-Datei (widersprüchliche Angaben in Feld A3 und C7a)
- 71 Prüfsummenfehler beim Einlesen der DTAUS-Datei
- 72 Prüfsummenfehler beim Einlesen der DTAZV-Datei

3. Anwendungsbeispiele

3.1. Visual Fox Pro

3.1.1. CREMUL-Konvertierung aus einer Datei

```
x = CREATEOBJECT( "eurobeam.ediconv" )
x.txtKontowortlaut.Value = "Mustermann GmbH."  && Kontowortlaut Empfänger
x.txtBLZ.Value = "60000"                        && BLZ Empfänger
x.txtBankkennung.Value = "ATPSK"               && Bankkennung Empfänger
x.txtKontonummer.Value = "123456789"          && Kontonummer Empfänger
x.txtBelegart.Value = "82"                     && Belegart der Lastschrift
x.txtFrom.Value = "c:\daten\bankdaten.cre"
x.txtTo.Value = "c:\daten\bankdaten.csv"
x.cmdStart.Click
IF x.ErrorNum = 1
    ? "Konvertierung fertig!"
ELSE
    ? "Fehler: " + x.ErrorText
ENDIF
```

EDIFACT-Daten stehen in der Datei c:\daten\bankdaten.csv bereit

3.1.2. DIRDEB-Konvertierung direkt aus Daten

```
x = CREATEOBJECT( "eurobeam.ediconv" )
x.txtKontowortlaut.Value = "Mustermann GmbH."  && Kontowortlaut Empfänger
x.txtBLZ.Value = "60000"                        && BLZ Empfänger
x.txtBankkennung.Value = "ATPSK"               && Bankkennung Empfänger
x.txtKontonummer.Value = "123456789"          && Kontonummer Empfänger
x.txtBelegart.Value = "82"                     && Belegart der Lastschrift
x.cDataFrom = "ktonr,blz,betrag" + CHR(13)+CHR(10) + "724322722,60000,5.2"
x.cDataFrom = x.cDataFrom + CHR(13)+CHR(10) + "32144545,20151,0.2"
x.cmdStart.Click
```

EDIFACT-Daten stehen in x.cDataTo bereit

3.2. ASP (VBScript)

3.2.1. CREMUL-Konvertierung aus einer Datei

```
set x = CREATEOBJECT( "eurobeam.ediconv" )  
x.txtKontowortlaut.Value = "Mustermann GmbH."  
x.txtBLZ.Value = "60000"  
x.txtBankkennung.Value = "ATPSK"  
x.txtKontonummer.Value = "123456789"  
x.txtBelegart.Value = "82"  
x.txtFrom.Value = "c:\daten\bankdaten.cre"  
x.txtTo.Value = "c:\daten\bankdaten.csv"  
x.cmdStart.Click
```

EDIFACT-Daten stehen in der Datei c:\daten\bankdaten.csv bereit

4. Support

Updates erhalten Sie unter:

<http://www.eurobeamer.eu/support.html#updates>

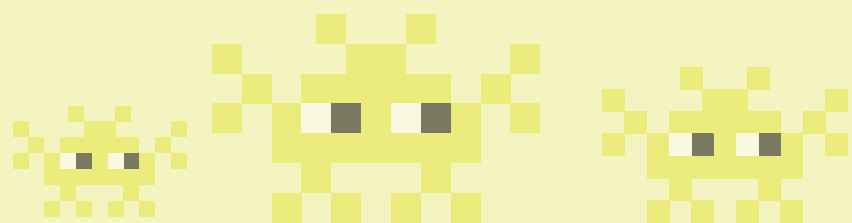
Häufige Support-Anfragen beantworten wir unter:

<http://www.eurobeamer.eu/support.html#anfragen>

Bei Fragen wenden Sie sich bitte an unseren Support:

Telefon: +43-1-2559947

E-Mail: support@ammann.info



EURO-BEAMER