

# Benutzerhandbuch

Autor: delight software gmbh  
Erstellt am: 10.10.2009  
Version: 4.0

---

## MLXML Business Integration

© 2001-2009 by delight software gmbh

# MLXML Business Integration

Basierend auf Multilevel 4.0

*by delight software gmbh*



**Multilevel Application Framework**

# **MLXML Business Integration**

© 2001-2009 by delight software gmbh

Es gelten die Allgemeinen Geschäftsbedingungen und die Allgemeinen Lizenzvereinbarungen der delight software gmbh

# Inhaltsverzeichnis

<b>Kapitel 1 - MLXML Business Integration</b>	<b>3</b>
<b>Kapitel 2 - Import</b>	<b>5</b>
<b>Kapitel 3 - Transport</b>	<b>8</b>
<b>Kapitel 4 - Konverter</b>	<b>10</b>
4.1 Konfigurations-Datei .....	10
<b>Kapitel 5 - MLXML-Format</b>	<b>14</b>
5.1 Datenabgleich .....	14

# Kapitel

MLXML Business Integration



1

# 1 MLXML Business Integration

Mit der *MLXML Business Integration* können Daten automatisiert mit anderen Anwendungen ausgetauscht werden. Die Erweiterung beinhaltet verschiedene [Konverter](#) und [Transportprotokolle](#) zum ausliefern und importieren von Daten.

[MLXML](#) beschreibt ein XML-Schema, das alle (durch MLXML unterstützen) Daten der Software in einem einheitlichen XML-Format notieren kann. Eine MLXML-Datei kann verschiedene Daten (z.B. Adressen, Journal-Einträge, Skripte, Stammdaten wie Postleitzahlenstamm oder Bankenstamm u.s.w.) in einer Datei beinhalten. MLXML-Dateien können an einem zentralen Punkt in der Software [importiert](#) werden, egal welche Daten die Datei beinhaltet.

Mit geeigneten [Konvertern](#) kann praktische jede Datenquelle in ein MLXML-Dokument konvertiert werden. Optional merkt sich der Import zu jedem importieren Datensatz die Herkunft und eine eindeutige Datensatznummer (z.B. Primärschlüssel des Datensatzes in der Quell-Datenbank). Somit können verschiedene Daten aus verschiedenen Datenquellen automatisiert zusammengezogen und aktualisiert werden.

**siehe auch:**

[Import](#)

[Transport](#)

[Konverter](#)

[MLXML-Format](#)

# Kapitel

---

Import

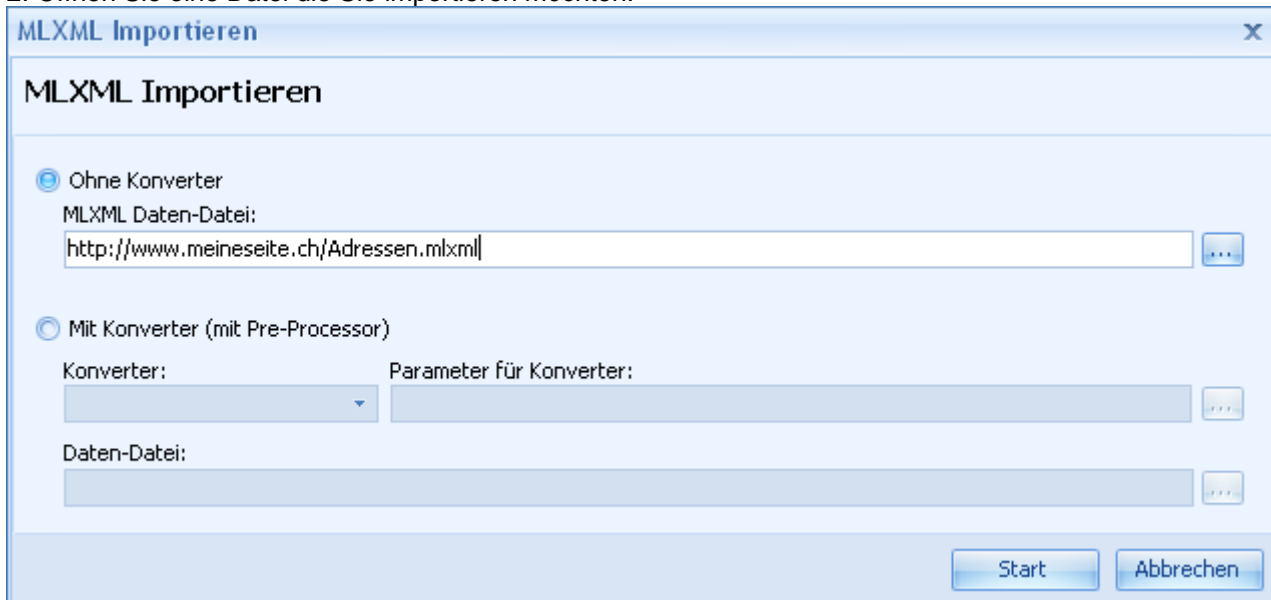


2

## 2 Import

1. Klicken Sie im Menü auf Import => *MLXML Datei* Importieren

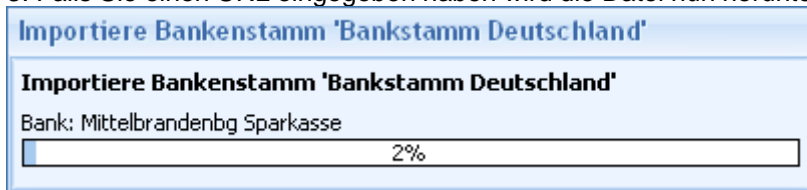
2. Öffnen Sie eine Datei die Sie importieren möchten.



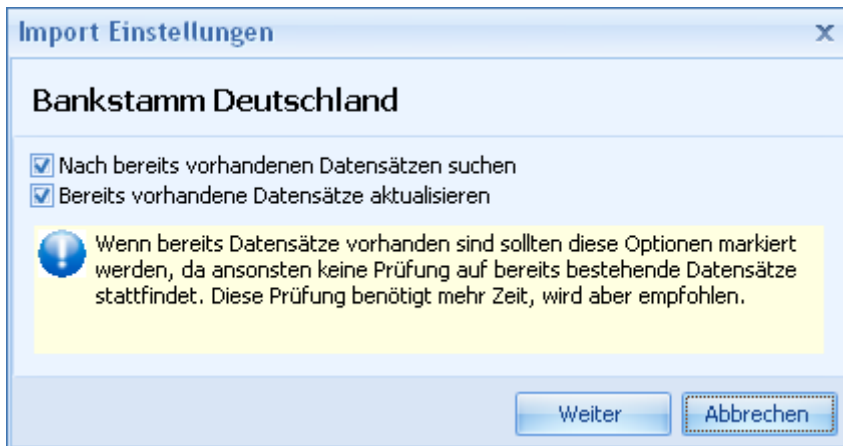
**Datei:** Wenn Sie die Datei lokal auf Ihrem Computer gespeichert haben, klicken Sie auf die Schaltfläche [...] am Ende des Eingabefeldes MLXML-Datei und wählen Sie die gewünschte Datei.

**URL:** Wenn Sie eine URL (z.B. [http://mirror01.delight.ch/banks/german\\_blz\\_20060605.mlxml](http://mirror01.delight.ch/banks/german_blz_20060605.mlxml)) importieren möchten, geben Sie den URL im Eingabefeld MLXML-Datei ein. Klicken Sie anschliessend auf Start um den Import zu starten.

3. Falls Sie einen URL eingegeben haben wird die Datei nun heruntergeladen.



4. Wenn Sie die Datei zum ersten Mal importieren, können Sie den Update-Modus deaktivieren um den Import zu beschleunigen.



Um den Update-Modus zu deaktivieren, entfernen Sie die Markierung bei *Bereits vorhandene Datensätze suchen*.

Klicken Sie anschliessend auf Weiter um den Import zu starten.

**Hinweis:**

Liegen sehr viele Daten vor, kann der Import einige Zeit in Anspruch nehmen. (z.B. beim Import von über 39'000 französischen Postleitzahlen).

Sollten Sie den Import zwischendurch abbrechen (nicht empfohlen!) müssen Sie bei einem zweiten Import den Update-Modus für den Import verwenden!

**siehe auch:**

[Transport](#)  
[Konverter](#)

# Kapitel

Transport



3

### 3 Transport

Der MLXML-Import und Export unterstützt verschiedene Transport-Protokolle um die Daten zu übertragen. So ist es z.B. möglich, Daten direkt ab einem FTP-Server oder Web-Server (HTTP) zu importieren. Analog können die Daten beim Export entsprechend auch auf eine FTP-Server hochgeladen werden.

Protokoll	Beschreibung	Download	Upload
HTTP	Import direkt ab einem Internet-Link.  Beispiel: <a href="http://www.meineseite.ch/Adressen.mlxml">http://www.meineseite.ch/Adressen.mlxml</a>	Ja	Nein
FTP	Import/Export ab einem FTP-Server.  Beispiel: <a href="ftp://benutzer:passwort@ftp.meineseite.ch/Adressen.mlxml">ftp://benutzer:passwort@ftp.meineseite.ch/Adressen.mlxml</a>	Ja	Ja

# Kapitel

Konverter



4

## 4 Konverter

Mit dem entsprechenden Konverter kann praktisch jede Datenquelle in ein MLXML-Dokument konvertiert und importiert werden.

Konverter	Beschreibung	Parameter	Kurzbezeichnung
ADODB to MLXML Converter	Kann ADO/ODBC Datenbanken nach MLXML konvertieren.	INI-Datei mit der Konfiguration für den Konverter.	ado-c
CSV to MLXML Converter	Kann CSV (Komma-Getrennte Textdateien) nach MLXML konvertieren.	INI-Datei mit der <a href="#">Konfiguration</a> für den Konverter.	csv-c
XSLT Transformation	Kann ein beliebiges XML-Dokument anhand eines XSL-Stylesheets nach MLXML transformieren.	XSL-Stylesheet-Datei für die Transformierung.	xslt-c

### 4.1 Konfigurations-Datei

Ein paar [Konverter](#) benötigen eine Konfigurations-Datei mit Einstellungen. In der Konfiguration werden Feldzuweisungen, MLXML-Einstellungen sowie individuelle Einstellungen für den jeweiligen Konverter konfiguriert. Die Konfiguration wird in einer INI-Datei gespeichert.

#### Bereich [Settings]

Name	Beispiel-Wert	Beschreibung	Konverter
SkipRows	0	Anzahl der Zeilen die zu Beginn des Dokumentes übersprungen werden sollen	csv-c
UseHeader	1	Muss auf 1 gesetzt werden, wenn die CSV-Datei Spaltenüberschriften in der 1. Zeile enthält	csv-c
Separator	;	Zeichen das als Spalten-Trenner verwendet wird	csv-c
QuoteChar	"	Zeichen das den Inhalt der Spalten abgrenzt	csv-c
DateFormat	dd-mm-yyyy	Format der Datumswerte	csv-c
TimeFormat	hh:mm:ss	Format der Zeitwerte	csv-c
DateSeparator	.	Trennzeichen für Datumswerte	csv-c
TimeSeparator	:	Trennzeichen für Zeitwerte	csv-c
DecimalSeparator	.	Dezimal-Trennzeichen für Kommazahlen	csv-c
ThousandSeparator	'	1000er-Trennzeichen für Zahlenwerte	csv-c
ConnectionString		ADO-Verbindungszeichenfolge für den Verbindungsaufbau zur Datenbank	ado-c
TableName	addresses	Name der SQL-Tabelle*, alternativ können die Daten auch per SQL-Query selektiert werden	ado-c
Query	SELECT [id], [name], [surname], [adress], [zip], [city], [phone1] FROM [addresses]	SQL-Query*, alternativ kann auch die komplette Tabelle über TableName definiert werden	ado-c

\* Es sollte jeweils TableName oder Query verwendet werden, nicht aber beides gleichzeitig.

#### Bereich [MLXML]

Name	Beispiel-Wert	Beschreibung	Konverter
Application	Datenbank XY	Name der Anwendung (Freitext)	csv-c, ado-c
PartName	crm_address	MLXML-Part, bezeichnet die Art der Daten. Für Adressen muss <i>crm_address</i> angegeben werden.	csv-c, ado-c
ItemName		MLXML-ItemPart, bezeichnet die einzelnen Daten. Wird kein Wert angegeben, wird der Wert automatisch bestimmt	csv-c, ado-c
PreProcessor		Nur zur internen Verwendung bestimmt, muss nicht definiert werden.	
FileExtension		Nur zur internen Verwendung bestimmt, muss nicht definiert werden.	
UserCaption		Nur zur internen Verwendung bestimmt, muss nicht definiert werden.	
MXCIdentifier		Nur zur internen Verwendung bestimmt, muss nicht definiert werden.	
Parameter		Nur zur internen Verwendung bestimmt, muss nicht definiert werden.	

### Bereich [MLXML Options]

Die verfügbaren Optionen unter [MLXML-Options] hängen von der Anwendung und der zu importierenden Datenart (z.B. Adressen, Postleitzahlen, Banken u.s.w.) ab.

Name	Beispiel-Wert	Beschreibung	Datenart/PartName
journalnotemode	2	0: Journal-Eintrag für neue Adressen hinzufügen 1: Journal-Eintrag für neue und aktualisierte Adressen hinzufügen 2: Keinen Journal-Eintrag erstellen  Diese Option ist nur für delight crm verfügbar.	crm_address
updatecheck	1	Definiert ob beim Importieren auf bereits importierte Daten geprüft werden soll. 1: Prüfen 2: Nicht prüfen	(alle)
modifyexisting	1	Definiert ob bereits vorhanden Datensätze aktualisiert werden sollen. Ist nur relevant bei modifyexisting=1.  0: Bereits importierte Datensätze ignorieren. 1: Bereits importierte Datensätze aktualisieren.	(alle)
attachmentpath		Pfad für den Import von externen Dateien auf einen Journal-Eintrag.  Diese Option ist nur für delight crm verfügbar.	crm_address

### Bereich [Mapping]

In diesem Bereich werden die Felder der Datenquelle den Feldern der Anwendung zugeordnet. Auf der linken Seite (Name) steht jeweils der Feldname aus der Datenquelle, auf der rechten Seite (Wert) der Feldname der Anwendung.

#### Beispiel:

```
[Mapping]
Name=name
Vorname=surname
Firma=companyname
Strasse=adress
PLZ=zip
```

```
Ort=city
Tel_P=phone1
autonummer=crm_address_ident
```

### Bereich [ConstantValues]

In diesem Bereich können für gewisse Felder konstante Werte definiert werden. Auf der linken Seite (Name) steht der Feldname der Anwendung, auf der rechten Seite (Wert) der konstante Wert.

#### Beispiel:

```
[ConstantValues]
acomment=ADO MLXML Import Test
crm_address_source=Demo.mdb
```

### Bereich [DataFormats]

Dieser Bereich wird nur für den csv-c Konverter benötigt. Er definiert, in welchem Format die einzelnen Spalten in der CSV-Datei gespeichert sind. Es werden die Feldnamen nach dem Mapping verwendet (Feldname der Anwendung). Auf der linken Seite (Name) steht der Feldname, auf der rechten Seite (Wert) das Format. Der Konverter ado-c kann die Datentypen der einzelnen Spalten automatisch bestimmen.

#### Beispiel:

```
[DataFormats]
crm_journal_note=Blob
level=Integer
amount=Float
birthday=Date
creadate=DateTime
```

### Bereiche [Replacements\_XY]

Um Zeichen oder ganze Zeichenfolgen vor dem Importieren durch Andere zu ersetzen, kann für ein Feld, in dem die Werte ersetzt werden sollen, ein eigener Replacement-Bereich erstellt werden. Auf der linken Seite (Name) steht der Wert der ersetzt werden soll, auf der rechten Seite der neuen Wert.

#### Beispiel:

```
[Replacements_countryid]
CH=Schweiz
FR=Frankreich
IT=Italien
```

### Bereich [DeleteSources]

In diesem Bereich können Daten mit einem bestimmten Wert im [Quellen-Feld](#) vor dem Import gelöscht werden. Dieser Bereich kann verwendet werden, wenn mit jedem Importlauf andere Daten importiert werden müssen und die vorgängig importierten Daten nicht mehr benötigt werden. Auf der linken Seite (Name) werden die einzelnen Einträge nummeriert, auf der rechten Seite (Wert) steht das [Quellen-Feld](#) mit dem jeweiligen Wert.

#### Beispiel:

```
[DeleteSources]
1=crm_address_source=Demo.mdb
```

#### siehe auch:

[Datenabgleich](#)

# Kapitel

MLXML-Format



5

## 5 MLXML-Format

MLXML beschreibt ein XML-Schema, das alle (durch MLXML unterstützen) Daten der Software in einem einheitlichen XML-Format notieren kann. Eine MLXML-Datei kann verschiedene Daten (z.B. Adressen, Journal-Einträge, Skripte, Stammdaten wie Postleitzahlenstamm oder Bankenstamm u.s.w.) in einer Datei beinhalten.

### Aufbau:

```
<?xml version="1.0" encoding="utf-8"?>
<mlxmldata application="Quelle XY" version="1">
  <part_name art1="1" atrn="2">
    <part_item_name value1="a" value2="" valuen="" />
    <part_item_name value1="b" value2="" valuen="" />
    <part_item_name value1="c" value2="" valuen="" />
    <part_item_name value1="d" value2="" valuen="" />
  </part_name>
</mlxmldata>
```

### Beispiel:

```
<?xml version="1.0" encoding="utf-8"?>
<mlxmldata application="Quelle XY" version="1">
  <crm_address count="1">
    <crm_address_item crm_journal_type="" countryid="Schweiz" name="Elias"
surname="Zurschmiede" adress="Unterstrasse 39" zip="9000" city="St. Gallen"
crm_journal_note="Text für den Eintrag im Journal der beim Importieren erstellt wird."/>
  </crm_address>
</mlxmldata>
```

## 5.1 Datenabgleich

Damit das Abgleichen von Daten (Prüfung ob ein Datensatz aus einer bestimmten Quelle bereits importiert wurde) funktioniert, muss jeder Datensatz mit einer Quelle und einem eindeutigen Schlüssel bestückt werden. Als eindeutigen Schlüssel bietet sich der Primär-Key aus der Tabelle der Quell-Datenbank an.

Da es theoretisch möglich ist, in einer MLXML-Datei Daten aus verschiedenen Quellen zu speichern, muss Quelle und Schlüssel auf jedem Eintrag definiert werden. Die Name dieser beiden Felder in MLXML hängen von der Datenart, die man importieren möchte, ab.

PartName	PartItemName	Quellen-Feld	Schlüssel-Feld	Bemerkungen
crm_address	crm_address_item	crm_address_source	crm_address_ident	
crm_journal	crm_journal_item	crm_address_source	crm_journal_ident	Nur delight crm
crm_address	crm_contactperson	crm_address_source	crm_contactperson_ident	
address_zip	zip	address_zip_source	address_zip_ident	
address_zip	canton	address_zip_source	address_zip_ident	
address_bank	bank	address_bank_source	address_bank_ident	
script	script_item	script_source	script_ident	

### Beispiel für ado-c und csv-c Konverter:

Diese Beispiel zeigt einen Auszug aus der [Konfigurations-Datei](#) für die [Konverter](#) ado-c und csv-c. Mit dieser Konfiguration werden Adressen (*PartName=crm\_address*) aus einer Datei oder Datenbank (je nach Konverter) importiert.

Im Bereich *[Mapping]* wird die Spalte mit der Datensatznummer (*auto\_nummer*), aus der Tabelle die Importiert werden soll, dem entsprechenden Schlüssel-Feld (*crm\_address\_ident*) zugewiesen. Im Bereich *[ConstantValues]* wird dem Quellen-Feld ein fixer Wert (*Demo.mdb*) zugewiesen.

```
[MLXML]
Application=Demo.mdb
PartName=crm_address

[Mapping]
Name=name
Vorname=surname
...
auto_nummer=crm_address_ident

[ConstantValues]
crm_address_source=Demo.mdb
```

**Anmerkung:**

Der Wert für *crm\_address\_ident* muss nicht zwingend eine Nummer sein. Zeichenfolgen (z.B. wenn als Primär-Schlüssel eine GUID verwendet wird) sind ebenfalls erlaubt.