Handbuch

Q3 XML Schnittstelle

6.1.2025

Programm Version 2024 oder höher, ab Windows 10
Handbuch Version 1.2025

Dieses Benutzerhandbuch darf ohne ausdrückliche Genehmigung der Q3 Software AG in keiner Form ganz oder in Auszügen reproduziert oder vervielfältigt werden.

Microsoft Outlook, Microsoft Word, Microsoft Excel und Windows sind eingetragene Marken der Microsoft Corporation. Q3 Zahlungsverkehr, Q3 Buchhaltung, Q3 Fibu, Q3 Auftrag, Q3 Lohn, Q3 Haushalt, Q3 Verein, Q3 Adress und Q3 Software sind Marken der Q3 Software AG. Alle anderen Marken sind Eigentum der jeweiligen Markeninhaber.

Alle in Beispielen verwendeten Namen von Produkten, Vorgängen, Personen und Ereignissen sind frei erfunden. Jede Ähnlichkeit mit bestehenden Firmen, Organisationen, Produkten, Namen oder Ereignissen ist rein zufällig.

© 2016-2025 Q3 Software AG, alle Rechte vorbehalten.

INHALTSVERZEICHNIS

1. Einführung 3
   1.1. Allgemeines ... 3
   1.2. Datenbanken ... 3
   1.3. Tabellen ... 3
   1.4. Datensätze ... 4
   1.5. Felder ... 4
   1.6. XML-Dateien ... 5
   1.7. Datenexport ... 5
   1.8. Datenimport ... 5
2. Wichtige Tabellen in Q3 Programmen 6
   2.1. Adressen ... 6
   2.2. Artikel ... 7
   2.3. Dokumente (z.B. Rechnungen) ... 8
   2.4. Positionen (z.B. von Rechnungen) ... 10
   2.5. Rabattgruppen ... 11
   2.6. Fibu-Konten ... 12
   2.7. Fibu-Journal ... 13
3. Exportauftrag 14
   3.1. Einleitung ... 14
   3.2. Beispiel Adresse ... 14
4. Importauftrag 15
   4.1. Einleitung ... 15
   4.2. Beispiel Debitor-Rechnungen ... 15
   4.3. Beispiel Fibu-Einzelbuchungen ... 16
   4.4. Beispiel Fibu-Sammelbuchung ... 17
   4.5. Vorgaben ... 18
   4.6. Vervollständigen ... 18
   4.7. Logfile ... 18
5. Automatismen (insb. für Shop-Software) 19
   5.1. Allgemeines ... 19
   5.2. Externe Datensatz-Identifikation ... 19
   5.3. XML-Transferordner ... 19
   5.4. Adressen ... 22
   5.5. Artikel ... 24
6. Anhang 25
   6.1. Parameter-Steuerung ... 25

1. EINFÜHRUNG

1.1. ALLGEMEINES

Die Q3 XML-Schnittstelle soll ermöglichen, zwischen Q3 Software und andern Programmen (z.B. Internet Shop, Kassenlösung) Daten auszutauschen.

! Die XML-Schnittstelle wird erreicht über Menü Datei+Datenbank, XML-Datentransfer. Dort kann die Schnittstelle konfiguriert und können Daten via XML-Dateien importiert und exportiert werden.

Das vorliegende Handbuch informiert, welche Möglichkeiten bestehen. Der Austausch geschieht immer über Datensätze, die via Schnittstelle aus Q3 Software Datenbanken gelesen oder in Q3 Software Datenbanken eingefügt oder geändert werden.

Dieses Handbuch enthält explizit KEINE Listen von Felder, die angesprochen werden können. Q3 Software ist ein lebendiges Programm, laufend kommen neue Felder hinzu. Deshalb können Feldlisten für alle Datenbanken (Adressen, Artikel usw.) direkt aus Q3 Software exportiert werden (siehe Kapitel 1.5 Felder).

1.2. DATENBANKEN

Jedes Q3 Programm arbeitet mit Datenbanken im Q3-eigenen Format „QDB". Jede Datenbankdatei enthält eine Haupttabelle und 0..n weitere Tabellen. Die wichtigsten grossen Datenbestände werden in Q3 Programmen in separaten Dateien im Mandantenverzeichnis geführt, so z.B. die Adressen in der Datei SWA_ADR.SQA als Haupttabelle.

1.3. TABELLEN

Jede Datenbankdatei (z.B. SWA_ADR.SQA) kann nebst der Haupttabelle (Name: 'QStdSort') beliebig weitere Tabellen aufnehmen. So gibt es z.B. in der Datenbank SWA_ADR.SQA zusätzlich die Tabelle 'AdrVerbindungen' für Beziehungen innerhalb der Adresstabelle (z.B. Lehrer/Schüler oder Trainer/Teammitglied).

Zudem gibt es in jeder Datenbank eine „Code-Tabelle" (Name: 'QCodeSort'), welche verschiedenste Informationspools in einer einzigen Tabelle aufnimmt (z.B. Merkmale, Codes oder Gruppen bei den Adressen). Jeder Pool erhält dabei eine eigene CodeArt, z.B. 'ME' für Merkmale, 'TG' für Gruppen).

Jeder Tabelle ist eine Formatdefinition zugeordnet. In einer Formatdefinition werden alle Felder eines Datensatzes mit Namen, Typ und einer eindeutigen Nummer definiert. Die Formatdefinition bestimmt darüber hinaus, welche Sortierungen für eine Tabelle ständig geführt werden. Z.B. werden für Adresstabellen ständig die Sortierungen nach Adressnummer, nach Name und nach Ort geführt.

Jede Tabelle ist durch Tabellennamen und Formatdefinition eindeutig definiert.

Formatdefinitionen sind in einer Q3-internen Resource abgelegt. Sie sind über die XML-Schnittstelle nicht zugänglich.

1.4. DATENSÄTZE

Jede Tabelle einer Datenbank enthält ausschliesslich Datensätze von ***einer*** bestimmten Formatdefinition, welche alle Felder pro Datensatz definiert.

Es gibt folgende Feldtypen:

String	Zeichenfolge 0..200, mit 0 abgeschlossen
Number	Zahl (max. 6 Nachkommastellen, +/- 999 Mia)
Boolean	true/false
Text	mehrzeiliger Text, max 60'000 Zeichen
Choice	Auswahl (Combo), Bedeutung individuell pro Feld, als Zahl 0..99 abgelegt
Table	komplette Tabelle in ein Feld gepackt
Byte	binäre Folge von Zeichen

1.5. FELDER

In der Formatdefinition ist festgehalten, welche Felder für den Datenimport und –export freigegeben sind. Dies kann von externen Programmen nicht beeinflusst werden.

! Wird in Q3 Software in einem Fenster (z.B. Adressfenster) der Mauszeiger auf ein Eingabefeld gesetzt, so wird unten links im Programmfenster der Name des Feldes angezeigt. Wird kein Name angezeigt, ist das Feld nicht freigegeben.

Wird beim Datenexport aus Q3 Software (Menü *Datei+Daten exportieren*), eine Export-definition mit dem Datenformat „CSV-Platzhaltertabelle" eingerichtet, so wird beim Export eine CSV-Datei geschrieben, welche sämtliche freigegebenen Felder einer Tabelle mit Feldnummer, Feldtyp und Feldname auflistet.

! Feldnamen können in XML-Dateien mit oder ohne vorangestelltem '#' eingesetzt werden. In diesem Handbuch werden Feldnamen immer mit vorangestelltem '#' geschrieben. In den XML-Beispielen werden die Feldnamen immer ohne '#' geschrieben.

Mögliche Werte für Felder vom Typ 'Choice' werden im Kapitel 2 jeweils speziell aufgelistet.

Zusätzlich zu den Feldern pro Datensatz (Adressfelder im Adressdatensatz, Artikelfelder im Artikeldatensatz usw.) sind für jeden Datensatz folgende Spezialfelder abfragbar:

- #RecDate1 Geburtsdatum des Datensatzes
- #RecDate2 Datum letzte Änderung

Diese Felder können in Felddefinitionen und Exportaufträgen eingesetzt werden. Sie können jedoch nicht gesetzt werden durch Importaufträge.

1.6. XML-Dateien

XML-Dateien benötigen zwingend die Codierung UTF-8. Folgende XML-Dateitypen gibt es:

- **Importaufträge** (enthält Datensätze und zugehörige Felder, die importiert werden sollen)
- **Exportaufträge** (welche Felder für welche Datensätze sollen exportiert werden)
- **Felddefinitionen** (welche Felder sollen für automatisierten Export exportiert werden)
- **Informationsdateien** (welche Aufträge wurden automatisiert abgehandelt – Format analog Importauftrag, Felder anhand Felddefinition)
- **Bestätigungsdateien** (Datensatzbestätigungen, Felder anhand Felddefinitionen)

1.7. Datenexport

In jedem Fenster, welches eine Tabelle anzeigt, können mittels *Datei+Daten exportieren* Exportdefinitionen mit Datenformat «XML-Format» eingerichtet werden für das Exportieren von Datensätzen aus der im Fenster angezeigten Tabelle in eine XML-Datei.

Zusätzlich bietet die Option «Q3 XML» die Möglichkeit, für einige Datenbanken (Adressen, Artikel, Dokumente usw.) **Exportaufträge** in einem speziellen XML-Format zu definieren und ausführen zu lassen (siehe Kapitel 3. Exportauftrag).

! Exportaufträge können Daten aus mehrere Tabellen und auch aus mehrere Datenbanken ansteuern. Exportaufträge können nicht via Menü *Datei+Datenexport* definiert werden, sondern müssen in einem XML-oder einem Text-Editor erstellt werden.

1.8. Datenimport

In jedem Fenster, welches eine Tabelle anzeigt, können mittels *Datei+Daten importieren* Importdefinitionen erstellt und ausgeführt werden. Dies gilt jedoch nicht für das XML-Format.

XML-Dateien können ausschliesslich mit lizenzierter Option «Q3 XML» importiert werden (*Datei+Datenbank, XML-Datentransfer*). Dazu wird keine Importdefinition benötigt, entscheidend ist nur der korrekte Inhalt der zu importierenden Datei (Importauftrag).

Ist eine XML-Schnittstelle aktiviert, so können zusätzlich Importaufträge automatisiert abgearbeitet werden (siehe Kapitel 4. Importauftrag).

! Importaufträge können Daten für mehrere Tabellen und auch mehrere Datenbanken beinhalten. Importaufträge müssen in einem XML- oder einem Text-Editor erstellt werden.

2. WICHTIGE TABELLEN IN Q3 PROGRAMMEN

2.1. ADRESSEN

Allgemeines

Datenbank-Dateiname:	SWA_ADR.SQA
Formatdefinition:	ADF
Identifikation innerhalb Mandant:	AdressDB
Tabellenname (fakultativ):	QStdSort
Eindeutigkeit	#AdressNr

Wichtige Felder

#AdressNr, #Vorname, #Name1, #Name2, #Adresse1, #Adresse2, #Plz, #Ort, #EMail, #Telefon1, #Telefon2, #Natel, #Gruppe, #Zustaendig, #ZuVorname, #ZuAnrede,

Spezielle Felder

#ExternalID	Freies Feld für externe Software
#AdressNr	Verbindung zu Dokumenten (Rechnungen, Quittungen usw.) Wichtig: Wird eine Adressnummer mitgegeben, so prüft das Programm, ob diese Adressnummer bereits vorhanden ist. Wenn ja, wird die bestehende Adresse mit den andern mitgegebenen Feldern aktualisiert.
#Sprache	Sprache (0=deutsch, 1=französisch, 2=englisch, 3=italienisch)
#ECKundenTyp	(0=inaktiv, 1=Endkunde,2=Partner)
#ECUser	Benutzername (z.B. Shop-Login)
#ECPasswort	Passwort (z.B. Shop-Login)

Spezielle Regelung

Firmenadresse	#Anrede="Firma", #Vorname=leer, #Name1=Firmenname Zuständige Person: #Zustaendig, #ZuVorname, #ZuAnrede
Privatadresse	#Anrede, #Vorname, #Name1 Felder #Zustaendig, #ZuVorname, #ZuAnrede: leer lassen

Beispiel Felddefinition ADF.xml (siehe Kapitel 5.3)

```
<?xml version="1.0" encoding="utf-8"?>
<Q3Record name="ADF">
 <AdressNr/>
 <ExternalID/>
 <Anrede/>
 <Vorname/>
 <Sprache/>
 <Name1/>
 <Plz/>
 <Ort/>
 <EMail/>
 <Telefon1/>
 <Zustaendig/>
 <ZuVorname/>
</Q3Record>
```

2.2. ARTIKEL

Allgemeines

Datenbank-Dateiname:	SWA_ART.SQS
Formatdefinition:	ARF
Identifikation innerhalb Mandant:	ArtikelDB
Tabellenname (fakultativ):	QStdSort
Eindeutigkeit	#S_ArtikelNr

Wichtige Felder

#S_ArtikelNr, #S_Produkt, #S_Gruppe, #S_VP, #S_EP, #S_VP2, #S_VP3, #S_Einkaufspreis, #S_EANNr, #S_Hinweis, #S_Beschreibung, #S_Bild

Spezielle Felder

#S_ExternalID	Freies Feld für externe Software
#S_ArtikelNr	Verbindung zu Dokument-Positionen (Rechnungen usw.) Wichtig: Wird eine Artikelnummer mitgegeben, so prüft das Programm, ob diese Artikelnummer bereits vorhanden ist. Wenn ja, wird der bestehende Artikel mit den andern mitgegebenen Feldern aktualisiert.
#S_Produkt	Artikelbezeichnung deutsch
#S_ProduktF	Artikelbezeichnung französisch
#S_BreschreibungD	Artikelbeschrieb deutsch
#S_BreschreibungF	Artikelbeschrieb französisch
#S_KurzBeschriebD	Kurzbeschrieb deutsch
#S_KurzBeschriebF	Kurzbeschrieb französisch
#S_Bestand	Aktueller Bestand

Beispiel Felddefinition ARF.xml (siehe Kapitel 5.3)

```
<?xml version="1.0" encoding="utf-8"?>
<Q3Record name="ARF">
 <S_ArtikelNr/>
 <S_ExternalID/>
 <S_Produkt/>
 <S_VP/>
 <S_Gruppe/>
 <S_Bild/>
 <S_BeschreibungD/>
 <S_KurzbeschriebD/>
</Q3Record>
```

2.3. Dokumente (z.B. Rechnungen)

Allgemeines

Datenbank-Dateiname:	SWA_DEBJ.SQD oder SWA_KREJ.SQK
Formatdefinition:	DKF
Identifikation innerhalb Mandant:	DebiJournal oder KrediJournal
Tabellenname (fakultativ):	QStdSort
Eindeutigkeit	#A_Auftragstyp+#A_DokumentNr

Wichtige Felder

#A_Auftragstyp, #A_DokumentNr, #A_Datum, #A_AdressNr, #A_Status, #A_Positionen, #A_Sprache, #A_Valuta, #A_Totalbetrag, #A_Hinweis, #A_Skonto, #A_Skontofrist, #A_Zahlungsfrist, #A_UnsereReferenz, #A_IhreReferenz, #A_LAdressNr, #A_Fibutext, #A_Sachbearbeiter, #A_Zahlstelle, #A_Zahlkonto, #A_Zuteilung, #A_ShopEsrNr, #A_Gueltig, #A_Termin, #A_Waehrung, #A_WaehrungsKurs, #A_Versandart

Spezielle Felder

#A_Auftragstyp	Art des Dokument (1=Rechnung, 2=Quittung, 5=Gutschrift, 6=Lieferschein, 7=Auftragsbestätigung, 8=Offerte, 9=Zahlung) -> zwingend erstes Feld pro Datensatz bei Importaufträgen (siehe Kapitel 4)
#A_ExternalID	Freies Feld für externe Software
#A_AuftragNr	wird automatisch gesetzt, wenn leer (Empfehlung)
#A_DokumentNr	Rechnungs-, Quittungs-Nr. usw., wird automatisch gesetzt, wenn leer
#A_AdressNr	Verbindung zu Kundenadresse
#A_LAdressNr	Verbindung zu Lieferadresse des Kunden
#A_Fibutext	Text für die Fibubuchung, wird automatisch gesetzt, wenn leer
#A_Status	Status des Dokuments (0=offen, 11=erledigt)
#A_Positionen	Positionen eines Dokuments sind als Tabelle im Feld #A_Positionen abgelegt (nicht zwingend für KrediJournal)
#A_Sprache	(0=deutsch, 1=französisch, 2=englisch, 3=italienisch)
#A_Zahlstelle	Über welches Konto soll die Zahlung erfolgen (0 = Zahlstelle 1, …, 3 = Zahlstelle 4)
#A_Zuteilung	Gruppierungsfeld für Statistiken
#A_OriginalBeleg	Name des Originalbelegs (z.B. PDF-Datei)
#A_BelegArt	Wo ist der Beleg (#A_OriginalBeleg) abgelegt? (0=lokal, 1=web)

Felder nur gültig für DebiJournal

#A_ErtragSteuer	Empfehlung: nicht verwenden, MwSt pro Position setzen
#A_ErtragKonto	Empfehlung: nicht verwenden, Konto pro Position setzen
#A_ECZahlart	1=Rechnung, 5=Kreditkarte, 6=Postfinance,9=Paypal. Ist bei einer Rechnung A_ECZahlart>1 und der Stauts>=11 (11=erledigt), so wird zusätzlich eine Zahlung erzeugt mit dem in der Konfiguration gesetzten Fibu-Konto.

Felder nur gültig für KrediJournal

#A_AufwandSteuer	Empfehlung: nicht verwenden, MwSt pro Position setzen
#A_AufwandKonto	Empfehlung: nicht verwenden, Konto pro Position setzen

Dokumente für neue Adressen importieren

Soll eine neue Adresse gleich mit dem Dokument angelegt werden, gibt es zwei Wege:

1) Das Fremdprodukt (z.B. Shop) verwaltet selber einen Nummernkreis, der vom Nummernkreis von manuell erfassten Adressen nicht berührt wird, und teilt jeder neuen Adresse eine Adressnummer zu.

   Beispiel1:
   - Nummernkreis manuell erfassen: 00001-49999
   - Nummernkreis Shop: 30000-99000

   Beispiel2:
   - Nummernkreis manuell erfassen: M00001-M99999
   - Nummernkreis Shop: S00001-S99999

2) Das Fremdprogramm gibt weder in der Adresse (Feld #AdressNr) noch im Dokument (Feld #A_AdressNr) eine Adressnummer mit, und liefert neue Adressen jeweils direkt vor dem dazugehörigen Dokument oder den dazugehörigen Dokumenten. Werden mehrere Dokumente (z.B. mehrere Rechnungen) in der XML-Datei geliefert, so müssen demnach darin Adressen und Dokumente abwechselnd folgen.

2.4. Positionen (z.B. von Rechnungen)

Allgemeines

Datenbank-Dateiname:	SWA_DEBJ.SQD oder SWA_KREJ.SQK
Formatdefinition:	PZF
Identifikation innerhalb Mandant:	DebiJournal oder KrediJournal
Tabellenname:	- (Feld innerhalb Datensatz DKF)
Eindeutigkeit	- (keine Sortierung)

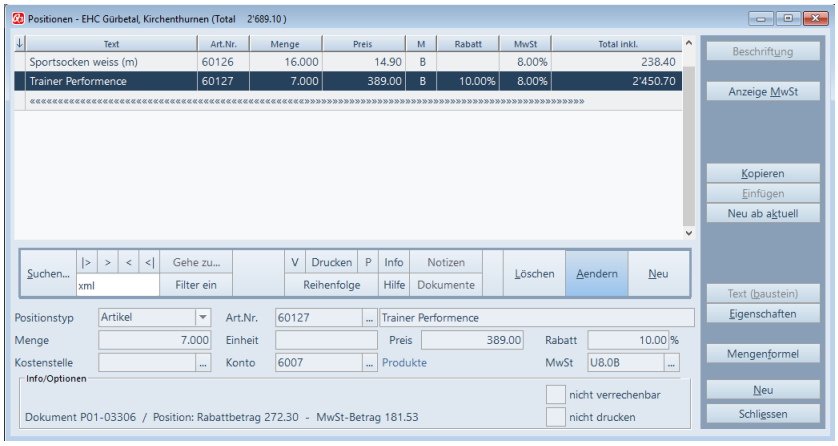
Jedes Dokument (Rechnung, Offerte, Lieferschein usw.) kann 0..n Positionen enthalten. Die Positionstabelle wird als komplexes Feld innerhalb des Dokumentdatensatzes angelegt.

Wichtige Felder

#P_Postyp, #P_ArtikelNr, #P_Text, #P_Menge, #P_Preis, #P_Steuer, #P_RabattP, #P_Fibukonto, #P_Kostenstelle, #P_KSTSeite, #P_Einheit, #P_Einheit2, #P_Hinweis, #P_Kommentar, #P_Beschreibung

Spezielle Felder

#P_Postyp	Positionstyp (10=Zwischentotal, 11=Gruppentotal, 12=Gesamttotal, 14=Gruppe, 15=Text, 16=Textbaustein, 17=Kommentar, 18=Zuschlag, 20=Rabatt, 22=Versandkosten, 24=Betrag, 25=Doppelmenge, 27=Menge, 28=Artikel, 29=Artikel extern, 31=Seitenumbruch)
#P_Steuer	MwSt-Code (z.B. U8.0B, U2.5B, EXPORT)
#P_ArtikelNr	Mit ReadArticle="yes" kann ab Q3 Artikelstamm die Position mit VP, MwSt und Produktbezeichnung komplettiert werden. Dabei wird der ab Import gelieferte Preis überschrieben. Beispiel: <#P_ArtikelNr readarticle="yes">00001</#P_ArtikelNr>



Positionsfenster in Q3 Software

2.5. RABATTGRUPPEN

Allgemeines

Datenbank-Dateiname:	SWA_ARTK.SQK
Formatdefinition:	AKF
Identifikation innerhalb Mandant:	RabattGruppen
Tabellenname:	QStdSort
Eindeutigkeit	-

Wichtige Felder

#R_Name, R_RabattTyp, #R_RabattBasis1, #R_MengeUmsatz1, #R_RabattArt1, #R_RabattPreis1, #R_ArtikekGruppe, #R_ArtikelNr, #R_AdressNr, #R_PeriodeVon, #R_PeriodeBis

Spezielle Felder

#R_Name	Name einer Rabattgruppe, kann 1..n x vorkommen, z.B. pro Artikel einen Eintrag, oder pro Artikelgruppe einen Eintrag oder pro Kunde einen Eintrag
#R_RabattTyp	Rabatttyp (0=Positionsrabatt, 1=Umsatzrabatt, 2=Mengenrabatt)
#R_RabattBasis	Rabattbasis 1..4 (0=Menge, 1=Umsatz)
#R_RabattArt	Rabattart (0=Rabatt in Prozent, 1=Rabattpreis)

Pro Eintrag können bis zu 4 Regeln erfasst werden mit den Feldern #R_RabattBasis1..4, #R_MengeUmsatz1..4, #R_RabattArt1..4, #R_RabattPreis1..4. Zum Einsatz kommt immer diejenige Regel, die den besten Preis ergibt (von allen Regeln, die zutreffen).

Rabattgruppenfenster in Q3 Software

2.6. FIBU-KONTEN

Allgemeines

Datenbank-Dateiname:	SWA_FIBK.SQF
Formatdefinition:	FKF
Identifikation innerhalb Mandant:	FibuKonten
Tabellenname (fakultativ):	QStdSort
Eindeutigkeit	#K_KontoNr+#K_LaufNr

Wichtige Felder

#K_KontoNr, #K_LaufNr, #K_Bezeichnung, #K_Kategorie, #K_Kontotyp, #K_Waehrung, #K_DifferenzKonto

Spezielle Felder

#K_LaufNr	Fortlaufende Nummer, damit nach einem Konto mit der gleichen Kontonummer Leerzeilen eingefügt werden können
#K_Kategorie	Kontokategorie (0=undefiniert, 1=Aktiven, 2=Passiven, 3=Aufwand, 4=Ertrag, 5=Spezialkonto)
#K_Kontotyp	Kontotyp (0=undefiniert, 1=CHF-Konto, 2=Fremdwährungskonto, 3=Hauptsumme, 4=Untersumme, 5=Haupttitel, 6=Untertitel, 7=Leerzeile, 8=Hauptsaldierung, 9=Gruppentitel, 10=gruppensumme, 11=Gruppensaldierung, 12=Klasse, 13=Seitenumbruch
#ECUser	Benutzername (z.B. Shop-Login)
#ECPasswort	Passwort (z.B. Shop-Login)

Beispiel Felddefinition FKF.xml

```
<?xml version="1.0" encoding="utf-8"?>
<Q3Record name="FKF">
 <K_KontoNr/>
 <K_LaufNr/>
 <K_Bezeichnung/>
</Q3Record>
```

2.7. FIBU-JOURNAL

Allgemeines

Datenbank-Dateiname:	SWA_FIBJ.SQF
Formatdefinition:	FJF
Identifikation innerhalb Mandant:	FibuJournal
Tabellenname (fakultativ):	QStdSort
Eindeutigkeit	#J_BelegNr+#J_LaufNr

Wichtige Felder

#J_BelegNr, #J_LaufNr, #J_Buchungsdatum, #J_Buchtyp, #J_Sollkonto, #J_Habenkonto, #J_BetragCHF, #J_Buchungstext1, #J_WaehrungSoll, #J_WaehrungHaben, #J_TagesKurs, #J_BetragFremd, #J_SteuerSoll, #J_SteuerHaben, #J_KST, #J_KST_Kontenseite

Spezielle Felder

#J_LaufNr	0 bei Einzelbuchung und bei Sammel-Hauptbuchung. Fortlaufende Nummer ab 1 bei Sammelgegenbuchungen (Buchtyp 1 und 2)
#J_Buchtyp	Buchungstyp (0=normal, 1=Soll-Sammelbuchung, 2=Haben-Sammelbuchung, 3=Eröffnungsbuchung, 4=Abschlussbuchung)
#J_Sollkonto	Sollkonto, leer bei der Hauptbuchung einer Sammelbuchung mit Haben-Gegenkonti
#J_Habenkonto	Habenkonto, leer bei der Hauptbuchung einer Sammelbuchung mit Soll-Gegenkonti
#J_WaehrungSoll	Währungskürzel (z.B. EUR) bei Buchungen mit Sollkonto gleich Fremdwährungskonto
#J_WaehrungHaben	Währungskürzel (z.B. USD) bei Buchungen mit Habenkonto gleich Fremdwährungskonto
#J_BetragFremd	Fremdwährungsbetrag bei Buchung
#J_Tageskurs	Währungskurs zum Zeitpunkt der Buchung

Hinweise

- Es gibt nur entweder ein Soll-Fremdwährungskonto oder ein Haben-Fremdwährungskonto, nicht beides gleichzeitig.
- Bei Sammelunterbuchungen muss immer auch das Gegenkonto aus der Hauptbuchung mitangegeben werden
- Bei Import in Q3 Fibujournal: J_TagesKurs*J_BetragFremd muss = J_BetragCHF sein
- Empfehlung bei Import: J_BelegNr weglassen, damit die XML-Schnittstelle die Belegnummer automatisch vergibt
- Wird ein Feld nicht mitgegeben, so bleibt es leer (entspricht bei Zahlenwert=0). Ausnahmen:
  - J_WaehrungSoll und J_WaehrungHaben werden automatisch auf CHF gesetzt
  - J_BelegNr wird automatisch vergeben

3. EXPORTAUFTRAG

3.1. EINLEITUNG

Eine XML-Datei für den Datenexport (= 'Exportauftrag') aus Q3 Datenbanken beinhaltet die Definition von 1..n Datenbanken, pro Datenbank 1..n Tabellen, sowie pro Tabelle ein Filter (entspricht dem Where-Statement in SQL) und 1..n Feldnamen der Felder, die pro Datensatz exportiert werden sollen.

Mit <Q3DB name=…" kann der Dateiname (z.B. SWA_ADR.SQA) oder die Identifikation (z.B. AdressDB) der Datenbank angegeben werden. Fehlt beim Dateinamen der Pfad, so wird die Datenbankdatei aus dem aktuellen Mandantenverzeichnis angesprochen.

Eine Tabelle kann 1..n mal pro Datenbank oder für ein Table-Feld deklariert werden (z.B. für das Feld #A_Positionen im Datensatz 'DKF'). Die Syntax des Where-Statements entspricht derjenigen in Formularen (siehe F1-Hilfe, Formeln). Wichtig: Texte innerhalb der Formel müssen in einfachen, die Formel als Ganzes muss in doppelten Hochkommas gesetzt werden (siehe Beispiel). Feldnamen müssen mit '#' beginnen.

Im Parameter <Q3Export> kann angegeben werden, in welche XML-Datei das Resultat geschrieben werden soll. Wird kein Name (dest=) angegeben, so wird ein Dateiname mit Datum und Zeit erzeugt. Wird kein Name oder ein Name ohne Pfad angegeben, so wird die Exportdatei im Ordner „Q3Result" des aktuellen Mandanten, oder, falls festgelegt, ins XML-Transferverzeichnis abgelegt (siehe Kapitel 5.3).

3.2. BEISPIEL ADRESSE

```
<?xml version="1.0" encoding="utf-8"?>
<Q3Export dest="swa_adr.xml">
 <Q3DB name="SWA_ADR.SQA">
  <Q3Table name="QStdSort" Where="(#RecDate2>6.5.2016)AND(#Ort='Bern')">
   <Q3Record>
    <AdressNr/>
    <ExternalID/>
    <Anrede/>
    <Vorname/>
    <Sprache/>
    <Name1/>
    <Plz/>
    <Ort/>
    <EMail/>
    <Zustaendig/>
    <ZuVorname/>
    <ZuAnrede/>
   </Q3Record>
  </Q3Table>
 </Q3DB>
</Q3Export>
```

Der Parameter «dest=» bei «Q3Export» kann auch einen Platzhalter beinhalten, z.B. wenn die Exportdatei wie die Kundennummer heissen soll, z.B.:
<Q3Export dest="ADR #AdressNr.xml">

4. IMPORTAUFTRAG

4.1. EINLEITUNG

Eine XML-Datei für den Datenimport (= 'Importauftrag') in Q3 Datenbanken beinhaltet die Definition von 1..n Datenbanken, pro Datenbank 1..n Tabellen, sowie pro Tabelle 1..n Datensätze mit jeweils 1..n Feldern mit Feldinhalt.

Eine Tabelle kann 1..n mal pro Datenbank oder für ein Table-Feld deklariert werden (z.B. Feld A_Positionen im Datensatz 'DKF'.

! Als Format wird UTF-8 erwartet.

4.2. BEISPIEL DEBITOR-RECHNUNGEN

```
<?xml version="1.0" encoding="utf-8"?>
<Q3Import>
 <Q3DB name="D:\QDaten\Standard\SWA_DEBJ.SQD">
  <Q3Table Name="QStdSort">
   <Q3Record>
    <A_Auftragstyp>1</A_Auftragstyp>
    <A_AdressNr>00161</A_AdressNr>
    <A_Positionen>
     <Q3Record>
      <P_Text>UHT-Milch</P_Text>
      <P_Preis>2.00</P_Preis>
      <P_Rabatt>11.00</P_Rabatt>
      <P_Steuer>EXPORT</P_Steuer>
     </Q3Record>
     <Q3Record>
      <P_Text>UHT-Milch</P_Text>
      <P_Preis>2.000</P_Preis>
      <P_Steuer>EXPORT</P_Steuer>
     </Q3Record>
    </A_Positionen>
   </Q3Record>
   <Q3Record>
    <A_Auftragstyp>1</A_Auftragstyp>
    <A_AdressNr>30358</A_AdressNr>
    <A_Positionen>
     <Q3Record>
      <P_Text>Programmierung Akonto 2008</P_Text>
      <P_Preis>77980.00</P_Preis>
      <P_Steuer>U7.6B</P_Steuer>
     </Q3Record>
    </A_Positionen>
   </Q3Record>
  </Q3Table>
 </Q3DB>
</Q3Import>
```

4.3. Beispiel Fibu-Einzelbuchungen

```
<?xml version="1.0" encoding="utf-8"?>
<Q3Import>
  <Q3DB name="FibuJournal">
    <Q3Table Name="QStdSort">
<!-- Einzelbuchung CHF-->
      <Q3Record>
        <J _FLaufNr>0</J_FLaufNr>
        <J _Buchtyp>0</J_Buchtyp>
        <J_Buchungsdatum>10.03.2017</J_Buchungsdatum>
        <J _Sollkonto>4070</J_Sollkonto>
        <J _Habenkonto>1010</J_Habenkonto>
        <J _SteuerSoll>V7.7B1</ J _SteuerSoll>
        <J _SteuerHaben></ J _SteuerHaben>
        <J _BetragCHF>80.50</ J _BetragCHF>
        <J _Buchungstext>Swisscom-Rechnung Februar 2017</J_Buchungstext>
      </Q3Record>
<!-- Einzelbuchung EUR-->
      <Q3Record>
        <J _FLaufNr>0</J_FLaufNr>
        <J _Buchtyp>0</J_Buchtyp>
        <J_Buchungsdatum>12.03.2017</J_Buchungsdatum>
        <J _Sollkonto>1021</J_Sollkonto>
        <J _Habenkonto>6001</J_Habenkonto>
        <J _SteuerSoll></ J _SteuerSoll>
        <J _SteuerHaben>U7.7B</ J _SteuerHaben>
        <J _BetragCHF>450.00</ J _BetragCHF >
        <J _WaehrungSoll>EUR</ J _WaehrungSoll>
        <J _WaehrungHaben>CHF</ J _WaehrungHaben>
        <J _Tageskurs>1.5000</ J _Tagskurs>
        <J _BetragFremd>300</ J _BetragFremd >
        <J _Buchungstext>Provision Januar 2017</J_Buchungstext>
      </Q3Record>
```

4.4. Beispiel Fibu-Sammelbuchung

```
<?xml version="1.0" encoding="utf-8"?>
<Q3Import>
 <Q3DB name="FibuJournal">
  <Q3Table Name="QStdSort">
<!—Sammel-Hauptbuchung -->
    <Q3Record>
     <J _FLaufNr>0</J_FLaufNr>
     <J _Buchtyp>1</J_Buchtyp>
     <J_Buchungsdatum>20.03.2017</J_Buchungsdatum>
     <J _Sollkonto>1010</J_Sollkonto>
     <J _BetragCHF>1000.00</ J _BetragCHF>
     <J _Buchungstext>Verkauf Produkt A</J_Buchungstext>
    </Q3Record>
<!— 1. Sammel-Gegenbuchung -->
    <Q3Record>
     <J _FLaufNr>1</J_FLaufNr>
     <J _Buchtyp>2</J_Buchtyp>
     <J_Buchungsdatum>20.03.2017</J_Buchungsdatum>
     <J _Sollkonto>1010</J_Sollkonto>
     <J _Habenkonto>6001</J_Habenkonto>
     <J _SteuerHaben>U7.7B</ J _SteuerHaben>
     <J _BetragCHF>950.00</ J _BetragCHF>
     <J _Buchungstext>Verkauf Produkt A</J_Buchungstext>
    </Q3Record>
<!— 2. Sammel-Gegenbuchung -->
    <Q3Record>
     <J _FLaufNr>2</J_FLaufNr>
     <J _Buchtyp>2</J_Buchtyp>
     <J_Buchungsdatum>20.03.2017</J_Buchungsdatum>
     <J _Sollkonto>1010</J_Sollkonto>
     <J _Habenkonto>6020</J_Habenkonto>
     <J _SteuerHaben>U8.0B</ J _SteuerHaben>
     <J _BetragCHF>50.00</ J _BetragCHF>
     <J _Buchungstext>Versandeinnahmen</J_Buchungstext>
    </Q3Record>
  </Q3Table>
 </Q3DB>
</Q3Import>
```

4.5. Vorgaben

In Q3 Software können für jeden Datensatztyp (insb. Adressen, Artikel, Dokumente) mittels Menü *Datei+Vorgaben* Vorgabewerte für Felder gesetzt werden (z.B. Feld Zahlungsfrist=30 für Dokument Rechnung).

Bei jeder Tabelle, die im Importauftrag deklariert wird, kann mittels new="default" mitgegeben werden, dass diese Vorgaben übernommen werden sollen.

Beispiel: <Q3Table Name="QStdSort" New="Default">

4.6. Vervollständigen

Q3 Software prüft beim Sichern verschiedener Datensätze (insb. Adressen, Artikel, Dokumente), ob alle Werte richtig gesetzt sind und vervollständigt bei Bedarf.

Das automatische Vervollständigen kann ausgeschaltet werden (nicht empfehlenswert).

Beispiel: <Q3Table Name="QStdSort" Complete="no">

4.7. Logfile

Mit dem Namen des Import- oder des Exportauftrags, jedoch mit Erweiterung „.log" wird ein Logfile geschrieben mit allen Fehlsituationen (ins gleiche Verzeichnis). Wird kein Logfile geschrieben, hat der Import / Export ohne Probleme geklappt.

5. AUTOMATISMEN (INSB. FÜR SHOP-SOFTWARE)

5.1. ALLGEMEINES

Im XML Datentransfercenter (Datei+Datenbank, XML-Datentransfer) kann mit «Konfigurieren» ein XML-Transferordner festgelegt werden, in welchem XML-Dateien zum Einsatz kommen für den automatisierten Datenaustausch zwischen externen Programmen und Q3 Software.

Zudem kann für verschiedene Datentypen (Adressen, Artikel usw.) festgelegt werden, dass Datenmanipulationen (Datensatz neu einfügen, ändern oder löschen) in Exportdateien festgehalten werden.

5.2. EXTERNE DATENSATZ-IDENTIFIKATION

Die meisten Datensätze sind in Q3 Software eindeutig identifiziert (z.B. eine Adresse durch die Adressnummer, ein Artikel durch die Artikelnummer). In externen Programmen (z.B. Internet-Shop) müssen oft eindeutige Datensatz-Identifikationen erzeugt werden, bevor ein Datensatz zu Q3 Software transferiert wird.

Die Datensätze der wichtigsten Tabelle verfügen deshalb über ein ExternalID-Feld. Hier kann das externe Programm eine eindeutige Referenz mitgeben, welche im Q3 Datensatz abgelegt wird.

! Das ExternalID-Feld wird nie von Q3 Software selber vergeben. Es wird aber im Eigenschaften-Fenster des Datensatzes (z.B. Adress-Eigenschaften) angezeigt.

Für folgende Tabellen ist ein ExternalID-Feld verfügbar:

Adressen	#ExternalID	
Artikel	#S_ExternalID	
Dokumente	#A_ExternalID	Rechnung, Quittung, Lieferschein usw.

! Wichtig ist die ExternalID insbesondere bei Adressen und Artikeln.

5.3. XML-TRANSFERORDNER

Ist ein XML-Transferordner gesetzt (XML-Konfiguration), werden in Q3 Software automatische Abläufe ausgeführt:

- Neue, geänderte und gelöschte Datensätze werden in Export-Dateien geschrieben (Q3Changes), sofern dies für einen Datentyp eingestellt ist, und für diesen Datentyp auch eine Felddefinition im Ordner „Q3Definition“ abgelegt ist.
- Import- und Exportaufträge werden abgearbeitet (manuell ausgelöst im XML-Datentransfercenter oder automatisiert via Taskmanager)
- Resultate von Exportaufträgen werden in Resultatdateien geschrieben
- Ein Taskmanager kann angewiesen werden, automatisiert Import- und Exportaufträge abzuarbeiten

Dazu werden folgende Ordner im Transferordner geführt:

Q3Export	Exportaufträge (Dateien mit Namen *.xml)
Q3Import	Importaufträge (Dateien mit Namen *.xml)
Q3ExportDone	abgearbeitete Exportaufträge (Dateien mit Namen *.xml)
Q3ImportDone	abgearbeitete Importaufträge (Dateien mit Namen *.xml)
Q3Result	Resultatdateien von Exportaufträgen (mit Namen *.xml)
Q3Definition	Felddefinitionen (Typ Exportauftrag mit Namen ???.xml)
Q3Changes	Informationsdateien mit Namen *.xml
Q3Confirm	Informationsdateien mit Namen *.xml

Im Transferordner selber wird die Datenbank 'Q3XML.QRR' geführt. Darin werden alle abgearbeiteten Import- und Exportaufträge eingetragen. Die Einträge dieser Datenbank werden im XML-Datentransfercenter angezeigt.

! Resultatdateien (Ordner Q3Result) und Informationsdateien (Ordner Q3Changes) sind vom gleichen Format wie Importaufträge (siehe Kapitel 4).

Exportaufträge

Ist in Q3 Software ein XML-Datentransferverzeichnis definiert, so kann im XML-Datentransfercenter das Programm angewiesen werden, alle Exportaufträge im Ordner 'Q3Export' abzuarbeiten. Dabei erzeugt das Programm im Ordner 'Q3Result' pro Exportauftrag eine Resultatdatei. Der Name der Resultatdatei kann unter „<Q3Export dest=""…>" im Exportauftrag festgelegt werden. Enthält diese Zielangabe keinen Pfad, so wird die Resultatdatei im Ordner 'Q3Result' unter dem angegebenen Namen abgelegt. Wird der dest="-Parameter weggelassen, so erzeugt Q3 Software die Resultatdatei mit gleichem Dateinamen wie beim Exportauftrag. Existiert unter dem gleichen Namen bereits eine Resultatdatei, so wird diese um die neuen Resultate ergänzt.

Abgearbeitete Exportaufträge werden in den Ordner Q3ExportDone verschoben. Existiert im Zielordner bereits eine Datei mit gleichem Namen, wird eine Laufnummer an den Dateinamen angehängt.

Felddefinitionsdateien

Für jeden Datenbanktyp (Adressen, Artikel usw.) kann im XML-Unterordner «Q3Definition» eine Felddefinitionsdatei mit Namen des Datensatztyps abgelegt werden, um automatisierte Exporte und Bestätigungen zu ermöglichen.

Folgende Felddefinitionsdateien sind aktuell zulässig:

ADF.XML	Adressfelder
ARF.XML	Artikelfelder
DKF.XML	Dokumentfelder (Debi und Kredi – Rechnungen, Offerten usw.)
FKF.XML	Fibu-Kontenfelder
FJF.XML	Fibu-Buchungsfelder
AKF.XML	Rabattgruppenfelder

Informationsdateien

Informationsdateien nach Q3Changes werden geschrieben, wenn im Verzeichnis Q3Definition für einen Datensatz-Typ eine Felddefinition (Format = Exportauftrag) mit dem Namen der Formatdefinition vorhanden ist, z.B. Felddefinition «ADF.xml» für Adressen und Felddefinition «ARF.xml» für Artikel). In diesen Felddefinitionen müssen anders als in Exportaufträgen die Elemente <Q3DB> und <Q3Table> weggelassen werden.

! Informationsdateien haben immer das Format von Importdateien.

Für jeden neu angelegten, jeden geänderten und jeden gelöschten Datensatz von einem in der Konfiguration aktivierten Typ erzeugt Q3 Software einen Eintrag in Q3Changes. Dabei gilt:

- Wird in einer Felddefinition im Element <Q3Record> mit dest='"…"' ein Dateiname angegeben, so erzeugt das Programm unter diesem Namen eine Informationsdatei (z.B. <Q3Record name=ADF" dest="ADF-Changes.xml">).
- Fehlt der 'dest='-Parameter im <Q3Record>-Element, so wird als Name die Formatdefinition verwendet (also z.B ADF.xml für geänderte Adressen).
- Im Element <Q3Record> wird in der Informationsdatei mittels Parameter 'Action=' mitgegeben, ob der Datensatz neu erfasst („new"), geändert („update") oder gelöscht („delete") wurde. Beispiel: <Q3Record Action=update">

Ist noch keine solche Informationsdatei vorhanden, wird sie neu angelegt, ist sie vorhanden, werden Änderungen angehängt.

Bestätigungsdateien

Ist für einen Datenbanktyp eine Felddefinition im Ordner «Q3Definition» vorhanden, so können Bestätigungsdateien anhand der Definitionsdatei ausgelöst werden. Der Inhalt von Bestätigungsdateien entspricht dem Inhalt von Informationsdateien, das Format entspricht jenem von Importdateien.

Bestätigungsdateien können jederzeit manuell ausgelöst werden: Menü *Bearbeiten+Info*, Schaltfläche «XML-Confirmation».

Zusätzlich kann in der XML-Konfiguration für den Debitor- und den Kreditor-Übertrag in die Buchhaltung gesetzt werden, dass pro übertragenes Dokument automatisch eine Bestätigungsdatei erstellt wird. Das Prinzip ist das gleiche wie bei Informationsdateien: Ist noch keine solche Bestätigungsdatei vorhanden, wird sie neu angelegt, ist sie vorhanden, werden Änderungen angehängt.

! Es kann sinnvoll sein, pro Rechnung eine eigene Bestätigungsdatei zu erzeugen. Möchten Sie dies erreichen, können Sie im «dest»-Parameter des <Q3 Record>-Elements Platzhalter einsetzen, z.B. für die Adressnummer und die Dokumentnummer:
<Q3Record name="DKF" dest="#A_AdressNr #A_DokumentNr.xml">

Importaufträge

Ist in Q3 Software ein XML-Datentransferordner definiert, so kann im XML-Datentransfercenter das Programm angewiesen werden, alle Importaufträge im Ordner 'Q3Import' abzuarbeiten.

Abgearbeitete Importaufträge werden in den Ordner Q3ImportDone verschoben. Existiert im Zielordner bereits eine Datei mit gleichem Namen, wird eine Laufnummer an den Dateinamen angehängt.

Taskmanager

Q3 Software kann in zweiter Instanz auch als Taskmanager gestartet werden. Dabei kommt die gleiche Verknüpfung zum Einsatz wie beim normalen Programmstart, jedoch ergänzt um den Parameter /tm. Idealerweise sollten auch Benutzer und Mandant als Parameter mitgegeben werden, so dass der Taskmanager ohne manuelle Eingabe gestartet werden kann.

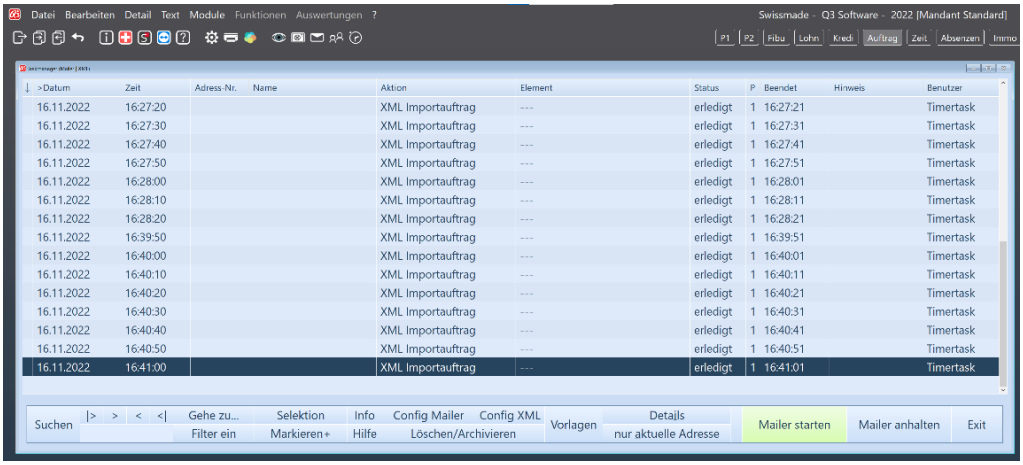
Beispiel C:\...Q3Software\Q3Software.exe … /tm /m ProbeAG /u Supervisor

> ! Der Taskmanager muss separat lizenziert werden! Bitte fragen Sie bei Bedarf die Konditionen an.

Dem Taskmanager kann die Aufgabe erteilt werden, automatisch, z.B. alle 15 Minuten alle Exportaufträge im Ordner 'Q3Export' abzuarbeiten. Das externe Programm muss die Ordner Q3Result und Q3Changes regelmässig auf enthaltene Dateien *.xml prüfen, diese Dateien abarbeiten und anschliessend löschen oder in einen andern Ordner verschieben.

Dem Taskmanager kann analog die Aufgabe erteilt werden, automatisch alle Importaufträge im Ordner 'Q3Import' abzuarbeiten.

Die Konfiguration des Taskmanagers erfolgt nicht im Taskmanager selber sondern über das XML-Datentransfercenter (Schaltfläche *Konfiguration Taskmanager*).



Fenster Taskmanager XML

5.4. ADRESSEN

Sollen Adressen zwischen einer externen Software (insb. Shop-Lösung, nachfolgende Beispiele beziehen sich auf eine Shop-Lösung) und Q3 Software ausgetauscht werden, so sind erweiterte Funktionalitäten im Importauftrag verfügbar.

Neue Adresse ab Shop in Q3 Software einfügen

Prämisse: Neuer Kunde registriert sich im Shop. Adresse existiert noch nicht in Q3 Software.

Im Importauftrag kann im Element <Q3Record> oder im Feld #ExternalID gesteuert werden, ob und wie mit einer allfällig bereits in Q3 Software vorhandenen Adresse umgegangen werden soll:

action="insert"	Neue Adresse einfügen ohne Prüfung
action="update"	Duplikat suchen und ergänzen, sonst neu einfügen (Wichtig: #AdressNr darf nicht übergeben werden)

Beispiel: <#ExternalID action="update">123456</#ExternalID>

Beispiel: <Q3Record action="update">

Da für jede Adresse, die importiert wird, ein Eintrag in 'Q3Changes' erfolgt, ist es der Shop-Software in einem nachfolgenden Lauf möglich, Shop-Adressen um die Kundennummer aus Q3 Software zu ergänzen.

! Wenn das Feld #AdressNr ab der Shop-Software gesetzt werden soll, muss in Q3 Software dafür ein Nummernkreis reserviert werden. Werden Adressen direkt in Q3 Software erfasst, müssen Adressnummern oberhalb dem Shop-Nummernkreis vergeben werden.

! Wichtig: Alle unter *Datei+Vorgaben* definierten Vorgabewerte werden beim Einfügen einer neuen Adresse gesetzt, und müssen nicht von der externen Software übergeben werden, wenn der Parameter new="default" gesetzt ist (siehe Kapitel 4.3)!

Adresse ab Shop in Q3 Software ändern

Prämisse: Kunde ist im Shop und in Q3 Software vorhanden. Kunde ändert Adresse.

Wird im Importauftrag das Feld #AdressNr mitgegeben, so kann mit Parameter handle="update" Q3 Software angewiesen werden, die mitgegebenen Felder in der Q3 Adresse zu ändern. Wird das Feld #ExternalID ebenfalls mitgegeben, so wird die Anweisung nur akzeptiert, wenn der Wert mit dem in Q3 Software gespeicherten Wert übereinstimmt.

Beispiel: <#AdressNr action="update">123456</#AdressNr>

Beispiel: <Q3Record action="insert">

Adresse wird in Q3 Software geändert

Prämisse: Kundenadresse wird in Q3 Software geändert.

Für jede Adresse, welche in Q3 Software geändert wird, und bei der das Feld #ExternalID gesetzt ist, schreibt Q3 Software einen Eintrag in 'Q3Changes'. Dabei wird das Element Q3Record um den Parameter action="update"' ergänzt.

Beispiel: <Q3Record action="update">

Adresse wird in Q3 Software gelöscht

Prämisse: Adresse wird in Q3 Software gelöscht, und soll auch im Shop gelöscht werden.

Für jede Adresse, welche in Q3 Software gelöscht wird, und bei der das Feld #ExternalID gesetzt ist, schreibt Q3 Software einen Eintrag in 'Q3Changes'. Dabei wird das Element Q3Record um den Parameter action="delete"' ergänzt.

5.5. ARTIKEL

Sollen Artikel zwischen einer externen Software (insb. Shop-Lösung, nachfolgende Beispiele beziehen sich auf eine Shop-Lösung) und Q3 Software ausgetauscht werden, so sind erweiterte Funktionalitäten im Importauftrag verfügbar.

Neuer Artikel ab Shop in Q3 Software einfügen (Datenhoheit Q3 Software)

Im Importauftrag kann das Feld #S_ExternalID mitgegeben werden. Da für jeden Artikel, der importiert wird, ein Eintrag in 'Q3Changes' erfolgt, ist es der Shop-Software in einem nachfolgenden Lauf möglich, Shop-Artikel um die Artikelnummer aus Q3 Software zu ergänzen. Das Feld #S_ArtikelNr sollte für neue Artikel nicht oder leer übergeben werden, es sei denn, für Artikel ab Shop ist ein eigener Nummernkreis vorgesehen oder Artikelnummern werden komplett vom Shop verwaltet.

Neuer Artikel ab Shop in Q3 Software einfügen (Datenhoheit Shop)

Im Importauftrag kann das Feld #S_ArtikelNr mitgegeben werden. Die Shop-Software ist dafür verantwortlich, dass keine bereits vorhandene Artikelnummer mitgegeben wird.

! Alle unter *Datei+Vorgaben* für Artikel definierten Vorgabewerte werden beim Einfügen eines neuen Artikels gesetzt, und müssen nicht von der externen Software übergeben werden, wenn der Parameter new="default" gesetzt ist (siehe Kapitel 4.3)!

Artikel ab Shop in Q3 Software ändern

Wird in der XML-Importdatei das Feld #S_ArtikelNr mitgegeben, so kann mit Parameter handle="update" Q3 Software angewiesen werden, die mitgegebenen Felder im Q3 Artikel zu ändern. Existiert noch kein Artikel, mit dieser Artikelnummer, so wird ein neure Artikel angelegt. Der Parameter kann auch im Element Q3Record mitgegeben werden.

Beispiel: <#S_ArtikelNr action="update">12345</#S_ArtikelNr>

Beispiel: <Q3Record action="update">

Artikel wird in Q3 Software geändert

Für jeden Artikel, welcher in Q3 Software geändert wird, und bei dem das Feld #ExternalID gesetzt ist, schreibt Q3 Software einen Eintrag in 'Q3Changes'. Dabei wird das Feld #ArtikelNr oder das Element <Q3Record> mit dem Parameter 'action="update"' ergänzt.

Artikel wird in Q3 Software gelöscht

Prämisse: Adresse wird in Q3 Software gelöscht, und soll auch im Shop gelöscht werden.

Für jeden Artikel, welcher in Q3 Software gelöscht wird, und bei dem das Feld #ExternalID gesetzt ist, schreibt Q3 Software einen Eintrag in 'Q3Changes'. Dabei wird das Feld #ArtikelNr oder das Element <Q3Record> mit dem Parameter 'action="delete"' ergänzt.

6. ANHANG

6.1. PARAMETER-STEUERUNG

Q3DB

Name="…"

Physischer Name der Datenbankdatei oder Bezug auf Datenbank in Mandantenverzeichnis:

AdressDB	Adressdatenbank SWA_ADR.SQA
ArtikelDB	Artikeldatenbank SWA_ART.SQS
DebiJournal	Debitorjournal SWA_DEBJ.SQD
KrediJournal	Kreditorjournal SWA_KREJ.SQK
LagerJournal	Lagerjournal SWA_ARTJ.SQJ
FibuKonten	Kontentabelle SWA_FIBK.SQF
FibuJournal	Fibujournal SWA_FIBJ.SQF
RabattGruppen	Rabattgruppentabelle SWA_ARTK.SQK

Beispiel <Q3DB name="AdressDB">

Q3Table

New="default"

Vorgabewerte aus Datei+Vorgaben werden übernommen.
Wenn nicht angegeben, ist new="empty".

Complete="no"

Datensätze werden nicht automatisch komplettiert (z.B. Kurzadresse anhand AdressNr setzen in Dokumenten). Nicht zu empfehlen!
Wenn nicht angegeben, ist complete="yes".

Name="xxx"

Name der Tabelle innerhalb der Datenbank.
Wenn nicht angegeben, ist Name="QStdSort".

Beispiel <Q3Table name="AdrVerbindungen">

Q3Export

Dest=".."

Name der Resultatdatei. Fehlt dieser Parameter, so wird im Ordner Q3Result unter gleichen Namen wie des Exportauftrages die Resultatdatei erstellt. Wird lediglich ein Dateiname (ohne Pfadangabe) angegeben, so wird die Datei im Ordner Q3Result abgelegt.

Beides gilt auch für ab dem Ordner Q3Changes abgehandelte Exportaufträge aufgrund von Datensatz-Manipulationen.

Q3Record

Action=".."

Wird gesetzt in im Ordner Q3Changes abgelegten Informationsdateien und in im Ordner Q3Confirm abgelegten Bestätigungsdateien.

action="insert"	Datensatz wurde neu eingefügt
action="update"	Datensatz wurde geändert
action="delete"	Datensatz wurde gelöscht
Action="confirm"	Datensatz wurde bestätigt

Beispiel: <Q3Record action="update">

Dest=".."

Name der Bestätigungsdatei. Der Name kann Platzhalter enthalten, z.B. #AdressNr für die Adressen oder #A_DokumentNr für Dokumente.