

INFONIQA

BeamNT-Schnittstelle

ONE 50 Finanz

Infoniqa Schweiz AG

Inhalt

1. Das Tool BeamNT 3
   1.1 Einleitung zum BeamNT Tool 3
   1.2 Aufruf des BeamNT Tools 4
   1.3 Datenformat 7
   1.4 Datentypen 9
2. Adressen im TAF-Format 10
   2.1 Daten-Struktur 10
3. Personenkonto im TAF-Format 11
   3.1 Daten-Struktur 11
4. Offene Posten im TAF-Format 14
   4.1 .Daten-Struktur 14
5. Buchungen im TAF-Format 16
   5.1 Daten-Struktur 16
   5.2 Automatische Steuerbuchung 17
   5.3 Automatische Kostenstellenbuchung 19
   5.4 Valutaausgleichsbuchung bei Fremdwährungskonten 19
   5.5 Tag „Payit“ – Elektronischer Zahlungsverkehr 20
6. Vorgehen zur Implementierung von TAF und CSV 21
7. Kompatibilität zum Beam-Format 21
   7.1 Beispiel Beam-Datei 21
   7.2 Anmerkung zu Block Sys im TAF-Format 21
8. Fehlercodes 22
   8.1 Allgemein 22
   8.2 BeamNT Automat 25

1. Das Tool BeamNT

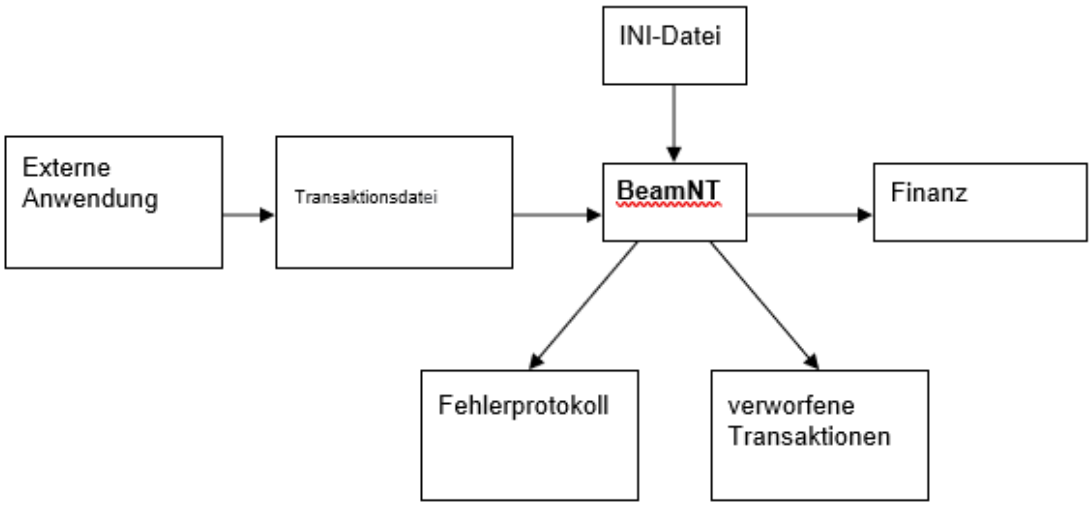
1.1 Einleitung zum BeamNT Tool

Die ASCII-Schnittstelle beim Finanz von ONE 50 heisst BeamNT. BeamNT wird über eine Transaktionsdatei realisiert, d. h. die Branchenlösung schreibt alle Aktionen sequenziell in eine Datei. Startet man BeamNT, wird diese Datei kopiert (Backup-Datei) und umbenannt (Arbeits-Datei). So entstehen zwei Dateien, welche BEAMNT.WRK und BEAMNT.BAK heissen und im Mandanten-Verzeichnis stehen. Dies wird so gehandhabt, damit die externe Branchenlösung sofort wieder Transaktionen produzieren kann. Die Datei BEAMNT.BAK wird nun nicht mehr gebraucht, sie dient als Sicherheitskopie. Die Datei BEAMNT.WRK wird von BeamNT eingelesen und verarbeitet.

Treten beim Importieren Fehler auf, werden diese in einem Fehlerprotokoll festgehalten (Datei Beam.log im Mandantenverzeichnis). Die fehlerhaften Transaktionen werden in einem File (verworfene Transaktionen) festgehalten, damit man diese bei Bedarf nochmals einlesen kann (Datei second.taf im Mandantenverzeichnis). Weil die Transaktionsdatei sequenziell verarbeitet wird, müssen Adressen vor Personenkonten und diese vor den OP kreiert werden (natürlich nur, wenn Adresse oder Personenkonto noch nicht existieren).

BeamNT wurde bewusst sehr einfach gehalten, so dass Entwickler von Branchenlösungen ohne grosse Schwierigkeiten damit arbeiten können.

Der Datenfluss von BeamNT läuft nur in eine Richtung, nämlich von der externen Anwendung zum Finanz. Damit die externe Anwendung auch Daten vom ONE 50 Finanz lesen und bearbeiten kann, wird der ONE50 OLE Automation Server Kit (SOK – OLE-Programmier-Schnittstelle) verwendet.



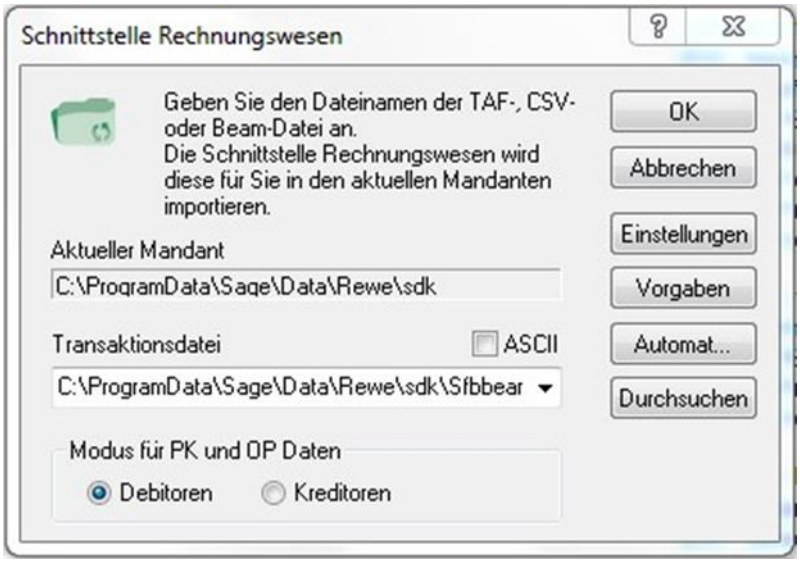
1.2 Aufruf des BeamNT Tools

Das Tool BeamNT bietet drei verschiedene Möglichkeiten für den Datenimport:

1. Menü Nebenbücher, Befehl „Offene Posten Einlesen“ aus dem ONE 50 Finanz
2. BeamNT Automat.
3. BeamNT über Kommandozeile.

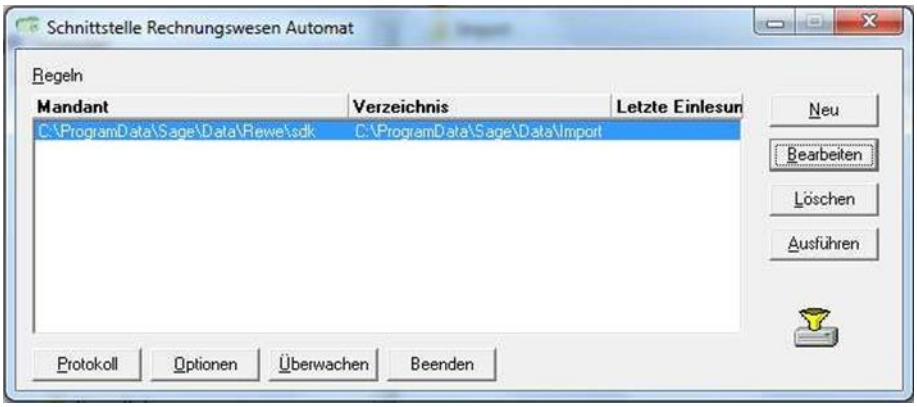
1.2.1 Aus ONE50 Finanz

Menü Nebenbücher, Befehl „Offene Posten Einlesen“ aus dem ONE 50 Finanz

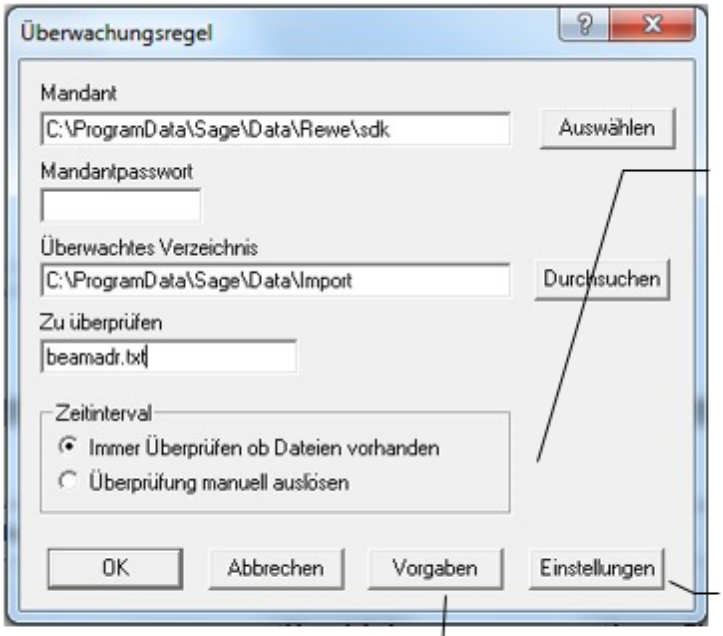


1.2.2 Schnittstelle Finanz Automat (Der „Staubsauger“!)

- Der BeamNT Automat kann beim automatischen Einlesen im Hintergrund laufen.
- Kann als Status-Icon verkleinert werden.
- Mehrere Regeln für verschiedene Mandanten können eingerichtet werden (siehe Dialog).
- **ACHTUNG**: Der Button Protokoll ist nur für allgemeine Fehler. Die Fehler, die beim Einlesen auftreten, müssen über Bearbeiten, Einstellungen und Protokoll angezeigt werden!



Durch “Bearbeiten” kommt man in die entsprechende Überwachungsregel.



Einstellung, ob BeamNT immer wenn es läuft, nach einzulesenden Dateien suchen soll (und sie natürlich auch einlesen...).

Bei “Überprüfung manuell...” muss man im BeamNT Automat die Regel anwählen und “Ausführen” drücken, damit BeamNT die Suche aufnimmt.

Allgemeine Einstellungen und
+ Ansicht von EinlesepProtokoll!
+ Ansicht und Re-Import von den verworfenen Transaktionen!

Mandantenspezifisch!

Vorgaben für:
+ Adressen
+ PK’s und OP’s
+ Buchungen

1.2.3 Kommandozeilenaufruf

BeamNT kann mit folgenden Parametern aufgerufen werden.

Beamnt.exe [Mandant, Passwort, Überwachung, Filter, Log-Datei, Verworfene-Transaktions-Datei, Format, Origin, Interval] [/hide] [/dontadd]

Mandant **Verzeichnis des Mandanten**
(z.B. C:\PROGRAMDATA\SAGE\DATA\REWE\DemoAG2024).

Passwort Passwort des Mandanten.

Überwachung Verzeichnis, in welchem die einzulesenden Daten erwartet werden.

Filter Gibt die Dateitypen an, welche eingelesen werden sollen. Als Beispiel könnte "a*.xxx" stehen, dann würden alle Dateien, welche mit dem a beginnen und die Endung XXX haben eingelesen. Standard ist ein "*", welcher alle Dateien mit TAF-Endung einliest.

Log-Datei Gibt die Protokoll-Datei an. Wenn diese nicht existiert, wird sie erstellt, andernfalls werden die Einträge am Ende der Datei angefügt. Ist dieses Feld leer, dann wird als Standard <Mandantenpfad>\Beam.log verwendet.

Verworfene-Transaktions-Datei Gibt die Verworfene-Transaktions-Datei an. Wenn diese nicht existiert, wird sie erstellt, andernfalls werden die Einträge am Ende der Datei angefügt. Ist dieses Feld leer, dann wird als Standard <Mandantenpfad>\Second.TAF verwendet.

Format Ist die Zeichenfolge "ASCII" gesetzt, dann wird die Transaktions-Datei im ASCII-Format erwartet. Ist nichts gesetzt, dann wird die Datei im ANSI-Format erwartet. Der Default-Wert ist ANSI.

Origin Dieser Eintrag ist nur notwendig, wenn Dateien im Beam- oder CSV-Format eingelesen werden. Er gibt an, ob die einzulesende Datei für die Debitoren- oder Kreditorenbuchhaltung bestimmt ist (insbesondere für Offene Posten und Personenkonten wichtig). Ist die Zeichenfolge "KREDI" gesetzt, wird die Transaktions-Datei in die Kreditorenbuchhaltung eingelesen. Ist nichts gesetzt, wird die Datei in die Debitorenbuchhaltung eingelesen. Der Default-Wert ist auf die Debitorenbuchhaltung gesetzt.

Interval Ist die Zeichenfolge "EVER" gesetzt, dann wird das Verzeichnis immer durchsucht. Bei "NEVER" muss man die Überprüfung manuell auslösen. Der Default-Wert ist "NEVER".

/hide Unterdrückt das Anzeigen von BeamNT, d. h. BeamNT wird nur als Status-Icon dargestellt und der Haupt-Dialog ist nicht sichtbar.

/dontadd Wenn eine Regel über Parameter angegeben wird (d. h. Mandant, Passwort,...) und dieses Flag gesetzt ist, so wird die Regel nicht in die Regel-Liste im Hauptdialog von BeamNT aufgenommen, sondern nur einmal ausgeführt.

Beispiel für eine Anweisung, welche die Überprüfung des Verzeichnisses "C:\Import" nach *.taf Dateien explizit auslöst:

C:\PROGRAMME(X86)\SAGE\SAGE50\PROG\Beamnt.exe
C:\PROGRAMDATA\SAGE\DATA\REWE\DemoAG2024.
,,C:\Import,"*.taf",,,, ,EVER /hide /dontadd

1.3 Datenformat

Grundsätzlich kommt BeamNT mit drei verschiedene Datenformaten zurecht. Es ist das TAF-, Beam- und CSV-Format, wobei das Erstgenannte zur Verwendung empfohlen ist und die anderen beiden aus Kompatibilitätsgründen weiterhin unterstützt werden.

1.3.1 TAF-Format (Tagged Format)

Es ist ein objektorientiertes Format.
• Es können mehrere Objekte in der gleichen Datei sein.

TAF-Dateien sind durch Blöcke aufgebaut, welche durch ein Block-Feld identifiziert sind. In diesen Blöcken sind eine oder mehrere Felder (sog. Tags) mit einem Wert gesetzt. Die Blöcke sind durch geschweifte Klammern '{…}' getrennt.
{Block-Tag Tag=Wert [Tag=Wert] […]}

Block-Tag Identifiziert den Block. Z. B. "Adr" für ein Adressen-Objekt

Tag Identifiziert den Wert, welchen man setzen will. Z. B. "AdrId" für die Adress-Nummer.

Wert Text oder Nummer des Tags.

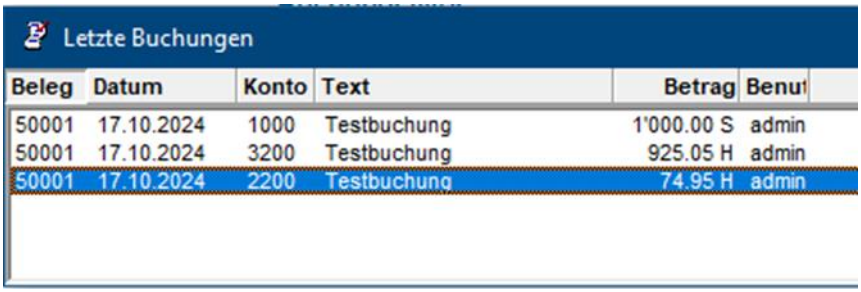
{Blg BlgNr=47 Date=19.11.24 Orig=0 MType=2 {Bk AccId=1000 ValNt=100.00 Text=Testbuchung}
{Bk AccId=3200 ValNt=92.50 ValTx=7.50 TaxId=USt81 Text=Testbuchung}
{Bk AccId=2200 ValNt=7.40 ValTx=92.50 Text="8.1% von 92.60"}
}

In der Dokumentation TAF-Format.pdf sind weitere Details beschrieben.

Spezialfälle sind weiter hinten im Kapitel 3 ff beschrieben.

1.3.2 CSV-Format (Kommagetrenntes Format)

- Die erste Zeile enthält die Reihenfolge der Felder, die eingelesen werden.
- Die zweite und jede weitere Zeile enthält die Daten.
- Es gilt: eine Zeile ist ein Datensatz.



Wird der Beleg 50001 mit ONE50 Administration als CSV-Datei exportiert, enthält die entstandene Datei sfbbuch.csv folgende Zeilen (gekürzt):

```
BlgNr,Date,AccId,Grp,Orig,MType,Type,CAcc,TaxId,TIdx,CIdx,BType,Code,ValNt,ValTx,ValFW,Text
50001,17.10.24,1000,"",0,1,0,3200,"",0,0,0,"",1000.00,0.00,0.00,Testbuchung
50001,17.10.24,3200,"",0,1,1,1000,USt81,3,0,0,"",925.05,74.95,0.00,Testbuchung
50001,17.10.24,2200,"",0,1,1,1000,"",0,0,2,"",74.95,925.05,0.00,Testbuchung
```

In der Dokumentation CSV-Format.pdf sind genauere Angaben über den Aufbau des CSV-Formats zu finden. Achtung: die genauen Feldbezeichnungen mit der Beschreibung dazu finden Sie in der Dokumentation TAF-Format.pdf.

1.3.3 Beam-Format

- CSV ähnlich, jedoch können verschiedene Daten in einer Datei übergeben werden.
- Es sind nur die Daten Adressen, PK, OP und Buchungen zugelassen.
- Buchungen können nur für die Debitoren- oder Kreditorenbuchhaltung erstellt werden.
- Dieses Format sollte längerfristig durch TAF abgelöst werden.

Beispiel aus sfbbeam.txt

```
ADRESSE,WRITE,30,SAGE,"Herr","Infoniqa Schweiz AG",Zusatz1,Zusatz2,"D4 Platz 10",6039,"Root","CH"
PKONTO,WRITE,30,,30,,01,"Infoniqa Schweiz","041 111 11 11",,,,,,,,,,
OP,WRITE,OP999,30,30,,"20.01.2024",R,,"Rechnung für Beam-File",01,,,,,,
DETAILOP,WRITE,OP999,3200,,100.00,,"USt81",TRUE
```

In der Dokumentation Beam-Format.pdf sind genauere Angaben zum Beam-Format zu finden.

1.4 Datentypen

Bei der Beschreibung der Datenstrukturen wurden einige Datentypen verwendet, die hier etwas genauer erklärt werden.

Datentyp	*Beschreibung*	*Wertebereich*
STRING[x]	String, der x Zeichen enthält	
BYTE	Vorzeichenloser 8-Bit Wert	0 - 255
LONG	Vorzeichenloser 32-Bit Wert	0 - 4'294'967'295
INTEGER	Ganzzahl 16-Bit Wert	- 32768 bis 32767
BOOLEAN	1-Bit Wert	0 oder 1
CURRENCY	64-Bit Wert	922'337'203'685'477.5807 -22'337'203'685'477.5807
DATE	Datum im Format tt.mm.jj/jjjj	1.1.1970 .. 1.1.2037

2. Adressen im TAF-Format

2.1 Daten-Struktur

{Adr <Identifikation> [<Optionale Felder>]}

2.1.1 <Identifikation>

Bezeichnung

AdrId Eindeutige Identifikation der Adresse.

Wertebereich STRING[13]

Bedeutung Defaultwert

2.1.2 <Optionale Felder>

Es sind alle übrigen Tags vom TAF-Format erlaubt (siehe TAF-Format.pdf)

2.1.3 Defaults

Für die Tags Lang, CCode, Code und PCode werden die Defaultwerte der BeamNT-Vorgaben genommen.

2.1.4 Beispiel

Kreiert oder ändert die Adresse 30 von Muster AG.

{Adr AdrId=30 Corp="Muster AG" Line1="Herr Muster" Road="Musterstr." ACode=6340 City="Baar" TelB="041 769 68 68"}

3. Personenkonto im TAF-Format

3.1 Daten-Struktur

{PK[d|k] <Identifikation> [<Optionale Felder>]}

3.1.1 Objektbezeichnung

{PKd … }steht für ein Debitoren-Personenkonto.
{PKk … }steht für ein Kreditoren-Personenkonto.

3.1.2 Identifikation

Es gibt 3 Möglichkeiten ein Personenkonto zu identifizieren. Werden mehr als die nötigen Identifikationsfelder angegeben, so werden diese überprüft.

Identifikation über PkKey und AdrId

Bezeichnung	*Wertebereich*	*Bedeutung*
PkKey	LONG	Eindeutige Identifikation des Personenkontos
AdrId	STRING[13]	Identifikation der dazugehörenden Adresse.

Beispiel: Kreiert oder ändert das Debitoren-Personenkonto 30 mit der Adress-Identifikation 3000.

{PKd PkKey=30 AdrId=3000 …}

Identifikation über Ref

Bezeichnung	*Wertebereich*	*Bedeutung*
Ref	STRING[17]	Zweite eindeutige Id des Personenkontos.

Beispiel: Kreiert oder ändert das Debitoren-Personenkonto mit der Referenznummer 123456789.

{PKd Ref="123456789" …}

Identifikation über AdrId (Optional mit AccId)

Bezeichnung	*Wertebereich*	*Bedeutung*
AccId	STRING[13]	Id des Debi/Kredi-Kontos
AdrId	STRING[13]	Id der Adresse des Personenkontos.

Beispiel 1: Kreiert oder ändert das Debitoren-Personenkonto mit der Adress-Nr 30.

{PKd PkKey="[30,]" …}

Beispiel 2: Kreiert oder ändert das Debitoren-Personenkonto mit der Adress-Nr 30 und dem Debitoren-Konto 1101.

{PKd PkKey="[30,1101]" …}

3.1.3 <Optionale Felder>

Es sind alle übrigen Tags vom TAF-Format erlaubt (siehe TAF-Format.pdf)

3.1.4 Default-Personenkonto

Wird das Personenkonto über AdrId identifiziert, so wird bei einem noch nicht existierenden Personenkonto zu der Adresse ein sog. Default-Personenkonto kreiert. Der PkKey wird versucht aus der AdrId zu generieren, d. h. die gleiche Nummer zu verwenden. Gelingt dies nicht, so wird die Nummer von den BeamNT-Einstellungen "Letztes Personenkonto" genommen und hinaufgezählt.
Dieses Default-Verhalten kann ebenfalls über die BeamNT-Einstellungen verhindert ausgeschaltet werden.

3.1.5 Defaults

Für die Tags AccId, ZabId, BnkNr, Payit, Code und WrnLk werden die Defaultwerte der BeamNT-Vorgaben genommen.

Folgende Tags werden von der Adresse übernommen, falls diese nicht übergeben werden:

Personenkonto	*Adresse*
CallN	Line1
CallT	TelB
Code	Code
Text	Corp

Das Feld Code wird nur dann von der Adresse übernommen, wenn es in den BeamNT-Vorgaben nicht definiert ist.

Beispiel

Kreiert oder ändert das Debitoren-Personenkonto 30 für Adresse-Nr 30.

{PKd PkKey=30 AdrId=30 AccId=1101 GAcc=3200 Code="123" Limit=20000}

4. Offene Posten im TAF-Format

4.1 .Daten-Struktur

{OP[d|k] <Identifikation> [<Optionale Felder>]}

4.1.1 Objektbezeichnung

{PKd … }steht für ein Debitoren-OP.
{PKk … }steht für ein Kreditoren-OP.

4.1.2 Identifikation

Bezeichnung	*Wertebereichung*	*Bedeutung*
OpId	STRING[13]	Nummer des Offenen Postens

Damit in einem OP die Defaults des dazugehörigen Personenkontos übernommen werden können, muss vor der OP-Identifikation das Personenkonto identifiziert werden. Da ein Personenkonto auf 3 Arten identifiziert werden kann (siehe Personenkonto), geht das natürlich auch beim OP.

Beispiel: Personenkonto über PkKey identifiziert.

{OPd PkKey="30" OpId=97001 …}

Beispiel: Personenkonto über Ref identifiziert.

{OPd Ref="123456789" OpId=97001 …}

Beispiel: Personenkonto über AdrId (optional mit AccId) identifiziert.

{OPd PkKey="[30,]" OpId=97001…}

4.1.3 Optionale Felder

Es sind alle übrigen Tags vom TAF-Format erlaubt (siehe TAF-Format.pdf)

4.1.4 Defaults

Für die Tags ZabId, BnkNr, Payit, Code und WrnLk werden die Defaultwerte der BeamNT-Vorgaben übernommen.
Folgende Tags werden vom PK übernommen, falls diese nicht übergeben werden und in den Defaultwerten der BeamNT-Vorgaben leer sind.

OP	*Personenkonto*
BnkId	BnkId
BnkAc	BnkAc
BnkNr	BnkNr
ZabId	ZabId
Payit	Payit

Das Feld Code wird nur dann vom Personenkonto übernommen, wenn es in den BeamNT-Vorgaben nicht definiert ist.

Beispiel

Kreiert oder ändert den Debitoren-OffenenPosten OP9000 für das Personenkonto 30.

{OPd OpId=OP9000 PkKey=30 ZabId=30T}

5. Buchungen im TAF-Format

5.1 Daten-Struktur

{Blg [<Beleg-Identifikation>] [<Beleg-Optionale Felder>] {Bk [<Bk-Optionale Felder>]} [Weitere Bk-Objekte]}

5.1.1 Beleg-Identifikation

Bezeichnung	*Wertebereich*	*Bedeutung*	*Defaultwert*
BlgNr	LONG	Beleg-Nummer	Aus Einstellungen

Belege können nur erstellt und nicht geändert werden!

Falls eine Beleg-Nummer angegeben wird, so wird der Fehler 1322 ausgegeben, falls die Beleg-Nummer schon belegt ist.

5.1.2 Beleg-Optionale Felder

Es sind alle übrigen Tags vom TAF-Format erlaubt (siehe TAF-Format).

5.1.3 Bk-Optionale Felder

Es sind alle übrigen Tags vom TAF-Format erlaubt (siehe TAF-Format).

5.1.4 Beispiel

Erstellt einen Sammelbeleg mit 3 Buchungen auf Ertragskonten mit MwSt. (für einen Debitoren-OP).

```
{Blg Date=02.03.2024 Orig=1
{Bk Type=0 CAcc=div ValNt=2000.00 Text="Sammelbuchung" OpId=R9001 PkKey=10001 Flags=1}
{Bk Type=1 AccId=3200 TaxId=USt81 ValBt=1200.00 MkTxB=1 Text="Sammelbuchung Schliessanlagen"}
{Bk Type=1 AccId=3201 TaxId=USt81 ValBt=500.00 MkTxB=1 Text="Sammelbuchung Kundenarbeit"}
{Bk Type=1 AccId=3202 TaxId=USt81 ValBt=300.00 MkTxB=1 Text="Sammelbuchung Beratung"}}
```

Achtung: ValNt in der Entstehungsbuchung muss den Bruttobetrag enthalten und nicht wie es in der Beschreibung des Feldes heisst, den Nettobetrag!

Beschreibung: die 1. Bk-Zeile enthält die Entstehungsbuchung mit Flags=1 und der Verknüpfung zum entsprechenden offenen Posten und Personenkonto. Ebenfalls wird hier der Gesamtbetrag des Belegs (Brutto) angegeben. Die 2.-n. Bk-Zeilen enthalten die einzelnen Buchungen.

5.2 Automatische Steuerbuchung

Tag	*Datentyp*	*Bedeutung*	*Defaultwert*
MkTxB	BOOLEAN	Wenn dieser Tag auf 1 gesetzt ist, dann wird eine automatische Steuerbuchung vorgenommen. Jedoch muss zwingend der Tag TaxId gesetzt sein, welcher das Kürzel des Steuerschlüssels angibt.	0
TaxId	STRING[5]	Kürzel des Steuerschlüssels.	leer
ValBt	CURRENCY	Bruttobetrag der Buchung (entweder ValBt oder ValNt nicht beide!). Darf in Buchungen, welche MkTxB nicht gesetzt haben, nicht verwendet werden!	
ValNt	CURRENCY	Nettobetrag der Buchung (entweder ValBt oder ValNt nicht beide!).	
ValBtFW	CURRENCY	Bruttobetrag in Fremdwährung (siehe ValBt)	
ValFW	CURRENCY	Nettobetrag in Fremdwährung (siehe ValNt)	

Beispiel:
Folgendes Beispiel verwendet die automatische MwSt.-Steuerbuchung.

```
{Blg Date=12.08.2024 Orig=1
{Bk Type=0 CAcc=3200 Text="Test" ValNt=108.10 OpId=41000 PkKey=4001 Flags=1}
{BK Type=1 AccId=3200 Text="Test" ValNt=100.00 TaxId="USt81" MkTxB=1}}
```

5.2.1 Validierung

Folgende Validierungsfelder können angegeben werden, damit man den mit TaxId übergebenen Steuerschlüssel genauer prüfen kann. Falls die Angaben nicht mit dem Steuerschlüssel übereinstimmen, wird eine Fehlermeldung ausgegeben.

Tag	*Datentyp*	*Bedeutung*	*Defaultwert*
NetIn	BOOLEAN	Flag für Netto- oder Bruttosteuerschlüssel. 1=Brutto, 0=Netto	leer
Rate	LONG	Prozentsatz des Steuerschlüssels. (Beispiel bei 8.1% -> 0.81)	Leer
Quote	LONG	Quote des Steuerschlüssels (Beispiel bei 100% -> 0 oder 1, bei 50% -> 0.5, bei 20% -> 0.2)	leer

5.2.2 Fehlermeldungen:

Fehlernummer	*Fehler-Text*
1316	Der Tag "TaxId" muss vor den Validierungs-Tags "NetIn", "Quote" oder "Rate" kommen!
1317	Der Prozentsatz des Steuerschlüssels 'x' stimmt nicht überein.
1318	Die Quote des Steuerschlüssels 'x' stimmt nicht überein.
1319	Das Brutto-/Netto-Flag des Steuerschlüssels 'x' stimmt nicht überein.

5.3 Automatische Kostenstellenbuchung

Tag	*Datentyp*	*Bedeutung*	*Defaultwert*
MkKstB	STRING[13]	Konto-Nummer der Kostenstelle, auf welche der ganze Betrag der Buchung zugeschrieben wird. Falls dieser Tag nicht gesetzt wird, wird auch keine Kostenstellen-Buchung vorgenommen.	Leer

Beispiel:

Folgendes Beispiel bucht CHF 100.- auf die Kostenstelle 9050.

{Blg Date=12.08.2024 Orig=1
{Bk Type=0 CAcc=3200 Text="Test KST-Buchung" ValNt=100.00 OpId=41000 PkKey=4001 Flags=1}
{BK Type=1 AccId=3200 Text="KST-Buchung" ValNt=100.00 MkKstB=9050}}

5.3.1 Fehlermeldungen

Fehlernummer	*Fehler-Text*
1314	Kostenstelle 'x' existiert nicht in der Fibu.
1315	Kostenstelle 'x' ist kein Kostenstellen-Konto in Fibu.

5.4 Valutaausgleichsbuchung bei Fremdwährungskonten

Bei Buchungen auf Fremdwährungskonten, die jedoch keine FW-Buchung sondern nur eine LW-Buchung sein soll, muss der Tag OnlyFW auf 1 gesetzt werden. Dies wird für Valutaausgleichsbuchungen benötigt.

Tag	*Datentyp*	*Bedeutung*	*Defaultwert*
OnlyLW	BOOLEAN	Wenn dieser Tag gesetzt ist, dann wird keine Fremdwährungsbuchung vorgenommen.	0

5.5 Tag „Payit“ - Elektronischer Zahlungsverkehr

5.5.1 Debitoren

Payit kann folgende Werte annehmen:

Wert	*Bedeutung*
0	Manuell
1	LSV (veraltet)
2	IBAN/IPI
3	LSV+

5.5.2 Kreditoren

Payit kann folgende Werte annehmen:

Wert	*Bedeutung*
0	Manuell
1	ESR 15-stellig (veraltet)
2	ESR 16-stellig (veraltet)
3	ESR 27-stellig (veraltet)
4	“Roter” Bank (veraltet)
5	“Roter” Post (veraltet)
6	“Roter” Treuhand (veraltet)
7	Postmandat (veraltet)
8	Fremdwährung
9	IBAN/IPI
10	QR mit Referenz
11	QR ohne Referenz
12	QR mit Kreditor-Referenz

6. Vorgehen zur Implementierung von TAF und CSV

Wenn Sie schnell einen Überblick über das TAF- und CSV-Format erhalten wollen, dann können Sie im ONE 50 Finanz den ganzen Mandanten, oder gewisse Teile, im TAF- oder CSV-Format exportieren! Dies geschieht im Menü Extras – Administration – Daten exportieren….

7. Kompatibilität zum Beam-Format

Die Beam-Datei wird in einem ersten Schritt in das TAF-Format konvertiert und anschliessend eingelesen. Falls Fehler bei der Konvertierung auftreten, wird die Verarbeitung abgebrochen.

In der Beam.Log-Datei ist ersichtlich wie die Konvertierungs-Datei heisst.

7.1 Beispiel Beam-Datei

ADRESSE,WRITE,00001,ABEGGZUERI,,Abegg Immobilien AG,Herr Müller,Abteilung Unterhalt,Münsterstrasse 5,8800,Zürich,,Sehr geehrter Herr Müller,,,,,01 222 23 33,01 222 23 25,,,
OP,WRITE,97111,00001,,,15.1.24,R,007008460500000000009711103,Lieferung Parkhotel,30T,,,,,E N
DETAILOP,WRITE,97111,3200,,111.00,,USt81,TRUE

7.1.1 Analoge Funktionalität im TAF-Format

{Adr AdrId="00001" Sort="ABEGGZUERI" Corp="Abegg Immobilien AG"
Line1="Herr Müller" Line2="Abteilung Unterhalt" Road="Münsterstrasse 5" ACode="8800" City="Zürich"
Salut="Sehr geehrter Herr Müller"
TelB="01 222 23 33" TelP="01 222 23 25" }

{OPd PkKey="[00001,]" OpId="971110" DateO="15.1.24" Ref="007008460500000000009711103"
Text="Lieferung Park Hotel" ZabId="30T" Code="E N" }

{Blg Date=15.1.12 Orig=1
{Bk Type="0" CAcc="3200" ValNt="111.00" Text="Lieferung Park Hotel"
OpId="97111" PkKey="[00001,]" Flags="1" } {Bk Type="1" AccId="3200" ValBt="111.00"
Text="Lieferung Park Hotel" TaxId="USt81" MkTxB="1" }}

7.2 Anmerkung zu Block Sys im TAF-Format

Der Tag MName (z. B. {Sys MName =D:\PROGRAMDATA\SAGE\DATA\REWE\DemoAG2024} wird nur geprüft, wenn in der Registry folgender Eintrag definiert ist:

In der Sektion
„Computer\HKEY_CURRENT_USER\SOFTWARE\SESAM Software\5.0\Beamnt\Settings“
der Eintrag BeamSys = "1" (Text!) setzen.

Ansonsten wird nichts überprüft.

8. Fehlercodes

8.1 Allgemein

>0 Wenn der Fehlerstatus positiv ist, deutet es auf einen Datenbankfehler hin.
- 1001: Zu wenig Speicherplatz.
- 1006: Die externe Adressverwaltung '%s' konnte nicht geladen werden.
1100: String zu lang (wenn Schalter für Strings abschneiden abgeschaltet).
- 1101: Ungültige Zeichen in Zahl.
- 1102: Datentyp (Objekt) nicht erkannt. - 1103: Operation nicht erkannt.
- 1104: Operation mit diesem Objekt nicht erlaubt (DELETE). - 1105: kein Boolean-Wert.
- 1106: Range Error.
- 1107: DetailOP darf nur unmittelbar nach einem OP auftreten. - 1108: Konnte OP nicht stornieren, weil er nicht existiert.
- 1109: Anführungszeichen nach dem Feld.
- 1110: Einzelnes Anführungszeichen im Feld. - 1111: Zeichen nach dem Feld.
- 1112: Falscher Wert für Rechnungsart.
- 1113: Eine Transaktion darf maximal 1000 Zeichen lang sein. - 1114: Ein Feld darf maximal 100 Zeichen lang sein.
- 1115: Das Sicherheitsfeld enthält nicht den Wert 'EXT'. - 1116: Kann Datum nicht bestimmen.
- 1117: OP-Nr wurde nicht eingelesen, da das Valutadatum ausserhalb des Finanzjahres liegt.
- 1118: Op-Nr enthält Valutadatum ausserhalb der Periode.
- 1200: Die Adresse konnte nicht in Datenbank gespeichert werden. - 1201: Das Personenkonto konnte nicht in Datenbank gespeichert werden.
- 1202: Der OP konnte nicht in Datenbank gespeichert werden.
- 1207: Der DetailOP konnte nicht in Datenbank gespeichert werden. - 1217:Die externe Bankverbindung existiert nicht in Debi/KrediNT.
- 1218: DTA-Zahlungsart ist ungültig.
- 1219: <HauptPKonto> gehört nicht zu dieser Adresse. - 1220: <HauptPKonto> existiert nicht in Debi/KrediNT.
- 1221: Die Adresse existiert nicht in Debi/KrediNT (nur bei Delete). - 1221: Adresse zu PKonto existiert nicht.
- 1222: Personenkonto existiert nicht.
- 1223: Überdefinierter Key nicht konsistent. - 1224: Fibukonto existiert nicht in FibuNT.
- 1225: PKonto zu dieser Adresse mit diesem Fibukonto existiert bereits.

- 1226: <AId> in Beam-Transaktion nicht definiert <SprachId> existiert nicht in Debi/KrediNT.
- 1226: Keine Id definiert.
- 1226: Keine oder ungenügende Identifikation des Personenkontos. - 1227: OP existiert bereits in Debi/KrediNT.
- 1228: Kann Personenkonto nicht aus Adresse bestimmen (wenn nur AId definiert).
- 1229: OPId ist falsch.
- 1230: Betrag darf nicht 0 sein.
- 1231: Steuersatz existiert nicht in FibuNT.
- 1232: Bei Fremdwährungskonto muss Fremdwährung angegeben sein.
- 1233: Nur bei Fremdwährungskonto darf Fremdwährung angegeben sein.
- 1234: Angegebenes Konto ist keine Kostenstelle.
- 1235: Angegebenes Konto muss eine Kostenart sein.
- 1236: Keine Kostenart definiert (nur bei reiner Kostenstellen-Buchung).
- 1237: Fibukonto darf keine Kostenstelle sein. - 1238: <Land> existiert nicht in Debi/KrediNT.
- 1239: Zahlungsbedingung existiert nicht in Debi/KrediNT. - 1240: <PkIndex> existiert bereits in Debi/KrediNT (nur bei Kreieren).
- 1241: Zu Kreierendes PKonto ist keiner Adresse zugewiesen.
- 1242: Neuer <PkIndex> existiert bereits bei einem anderen PKonto (nur bei Modifizieren).
- 1243: Default-Personenkonto erzeugt (mit Fibukonto aus INI-Datei).
- 1244: Kein Valuta-Datum für OP definiert.
- 1245: Keine Rechnungsart für OP definiert.
- 1246: Default Erfolgs-Konto X von Einstellungen verwendet. - 1247: OP existiert nicht.
- 1248: Fremdwährung zu Debi/Kredi-Konto existiert nicht. - 1249: Es gibt bereits ein OP mit dieser ESR-Nr.
- 1252: <Fibukonto> ist kein Aufwand-/Erfolgskonto.
- 1253: <Fibukonto> ist kein Aktivkonto (nur wenn von DebiNT gestartet).
- 1254: <Fibukonto> ist kein Passivkonto (nur wenn von KrediNT gestartet).
- 1255: Eintrag DefaultErtragsKonto in BeamNT Einstellungen nicht gesetzt.
- 1256: Schalter PKGegenKonto gesetzt, aber Ertragskonto (Gegenkonto) bei Personenkonto nicht vorhanden.
- 1257: Ertragskonto ist nicht in der Leitwährung.
- 1261: Default Erfolgs-Konto X von PK verwendet.
- 1300: Unbekannter Feldbezeichner.
- 1301: Syntax-Fehler - Die Zeile enthält ein unerwartetes Zeichen. - 1302: Ungültiger

Wert für das Feld '%s = %s'.
- 1303: Der Datensatz konnte nicht geschrieben werden. Status = %d.
- 1304: Datei '%s' kann nicht geöffnet werden. Status = %d. - 1305:Der Beleg %ld ist unvollständig (nicht ausgeglichen).
- 1306: Beleg %ld kann nicht gespeichert werden. Status = %d.
- 1307: Der Kontenplan-Eintrag '%s' konnte nicht eingefügt werden. - 1308: Der Referenz-Eintrag '%s' im Kontenplan kann nicht gelesen werden, oder ist nicht vorhanden.
- 1309: Mandant '%s' kann nicht geöffnet werden. Status: %d. - 1310: Objekt existiert bereits.
- 1311: Ausnahme aufgetreten:
- 1312: %d: Mandant '%s' kann nicht geöffnet werden. Status: %d - 1313: BelegNr %ld existiert bereits in Debi/KrediNT.
- 1314: Kostenstelle '%s' existiert nicht in der FibuNT.
- 1315: Kostenstellen '%s' ist keine Kostenstelle-Konto in FibuNT.
- 1316: Der Tag ""TaxId"" muss vor den Validierungs-Tags ""NetIn"", ""Quote"" oder ""Rate"" kommen!
- 1317: Der Prozentsatz des Steuerschlüssels '%s' stimmt nicht überein.
- 1318: Die Quote des Steuerschlüssels '%s' stimmt nicht überein. - 1319: Das Brutto-/Netto-Flag des Steuerschlüssels '%s' stimmt nicht überein.
- 1320: Personenkonto '%ld' existiert nicht in Debi/KrediNT. - 1321: OP '%s' existiert nicht in Debi/KrediNT.
- 1322: Es existiert bereits ein Beleg mit der Nummer '%ld'. - 1323: Personenkonto '%ld' enthält keine Adress-Nr.!
- 1324: OP '%s' ist vom PK '%ld' und nicht '%ld', wie es in der Buchung angegeben wurde!
- 1325: OP '%s' ist schon vorhanden.
- 1326: Buchungen auf OP '%s' werden nicht gemacht, da OP schon vorhanden war.
- 1327: Max. Fremdwährungsabweichung überschritten (FW = %s %s, LW = %s).
- 1328: Das PK '%ld' ist bereits bebucht und sein Fibukonto '%s' kann nicht mehr geändert werden.
- 1329: Die Importdatei ist für Mandant '%s' geschrieben worden (siehe SYS Tag).
- 1330: Das PK '%ld' (Fibukonto = %s) wurde versucht mit %s an %s zu bebuchen.
- 1399: Ausnahmebehandlung ""SetBuchung""
- 1400: Konnte Objekt '%s' nicht erkennen.
- 1401: Konnte Operation '%s' nicht erkennen. - 1402: 'DETAILOP' muss nach 'OP' folgen.
- 1403: Konnte keinen Arbeitsnamen in Verzeichnis '%s' für Konvertierungs-Datei ermitteln.
- 1404: Kein Operator definiert.
- 1405: 'DELETE'-Operation wird nicht unterstützt.

- 1406: 'OP'-OpId='%s' ist nicht identisch mit 'DETAILOP'-OpId='%s'. - 1407: Kein Objekt definiert.
- 1408: Es traten Fehler beim Konvertieren auf! - 1409: OP '%s' konnte nicht gelesen werden.
- 1410: Datei '%s' kann nicht geöffnet werden. Status = %d. - 1411:Ausnahmebehandlung aufgetreten!
- 1412: OP braucht eine Adress-Nr., sonst kann kein Default-Personenkonto erstellt werden!
- 1413: Konnte keinen Abschluss der Anführungszeichen finden.

8.2 BeamNT Automat

- 5000: Fehler beim Setzen der Überwachung auf das Verzeichnis '%s'. Status: %lu.
- 5001: Überwachung fehlgeschlagen. Status: %lu.
- 5002: Fehler beim Initialisieren der Überwachung. Status: %d. - 5003: Fehler beim Initialisieren von BeamNT.
- 5004: Konnte die Überprüfung auf neue Dateien nicht starten. Status: %lu.
- 5005: Fehler beim Überprüfen von neuen Dateien. Status: %lu. - 5006: Konnte Datei '%s' nicht öffnen. Status: %d.
- 5007: Wählen Sie ein Mandantenverzeichnis aus.
- 5010: Konnte keinen Arbeitsnamen für Datei '%s' ermitteln. Status: %lu.
- 5011: Fehler beim Lesen von der Datei '%s'. Status: %d.
- 5012: Fehler beim Schreiben in die Datei '%s'. Status: %d. - 5013:Konnte die Datei '%s' nicht löschen. Status: %d.
- 5014: Vorgaben
- 5015: Datei '%s' existiert nicht.
- 5016: Mandant '%s' kann nicht geöffnet werden.\n\nStatus: %d. - 5017: Verarbeitung wurde abgebrochen.
- 5018: Einstellungen
- 5019: Protocol Dateien (*.log)|*.log|Alle Dateien (*.*)|*.*||. - 5020: Mandant, Verzeichnis, Letzte Einlesung,……
- 5021: Datei '%s' kann nicht gelöscht werden.\n\nStatus = %d. - 5022: Das Verzeichnis '%s' enthält keinen gültigen Mandanten oder der Mandant kann nicht geöffnet werden.\nStatus = %d.
- 5023: Möchten Sie die Datei wirklich löschen? - 5024: \\Beam.log
- 5025: \\Second.taf
- 5026: Beam Dateien (*.taf; *.csv; *.txt)|*.taf; *.csv; *.txt|Alle Dateien (*.*)|*.*||.

- 5027: Konnte Datei '%s' nicht öffnen.\n\nStatus: %d.
- 5028: Beam Dateien (*.taf; *.csv; *.txt)|*.taf; *.csv; *.txt|Alle Dateien (*.*)|*.*||.
- 5029: Konnte Pfad '%s' nicht parsen.
- 5030: Die Datei mit den Verworfenen Transaktionen '%s' wird nach '%s' kopiert und anschliessend nochmals eingelesen. \n\nWollen Sie weiterfahren?
- 5031: Datei '%s' kann nicht nach '%s' kopiert werden. Stellen Sie sicher, dass diese Datei nicht schon existiert!
- 5032: Datei '%s' kann nicht gelöscht werden.\n\nStatus: %d. - 5033: ReImport.taf
- 5034: Datei einlesen vorbereiten...
- 5035: Datei '%s' existiert bereits.\n\nWollen Sie die Datei löschen? - 5036: Version
- 5037: DebiNT - 5038: KrediNT
- 5039: %s (%d Fehler und %d Warnungen). - 5040: (wird durchgeführt...)
- 5041: Leider funktioniert das Überwachen auf einem Laufwerk nicht.\n\nBitte geben Sie ein Verzeichnis an.