

# Integrare ERP - InterfacesWS

(WebService Interfete 3.1 - IInterfacesBO3.asmx – din 05.03.2021)

I.	Descriere Web Service	2
1.	LogIn	2
2.	Setare optiune	3
3.	ExportEx	3
4.	Import	4
5.	SalvareRaportDocumentEx	5
6.	LogOut	5
II.	Entitati predefinite pentru import	6
1.	Banca	6
2.	CategorieProdus	6
3.	Ramura	6
4.	GrupaProdus	6
5.	Produs	7
6.	UMProdus	9
7.	PretPerGestiune	9
8.	InfoAprovizionare	10
9.	CodBara	10
10.	Client	11
11.	PersoanaContact	14
12.	ContBancar	14
13.	Comanda	14
14.	Factura	18
15.	BonFiscal	22
16.	FacturaSpeciala	23
17.	Incasare	26
18.	Plata	29
19.	DocumentTransformareStoc	30
20.	Transfer	32
21.	Consum	34
22.	FacturaStorno	35
23.	Aviz	38
24.	AvizIncarcare	42
25.	ReturCustodie	44
26.	ComandaFurnizor	45
27.	FacturaFurnizor	47
28.	Intrare	49
29.	NotaContabila	51

# I. Descriere Web Service

WebService-ul de Interfete are implementat modul de comunicare HTTPSOAP, deci trebuie sa se foloseasca un client de SOAP pentru apelul programatic.

Majoritatea metodelor de acces puse la dispozitie prin Web Service-ul de Interfete necesita existenta unui Tip de Sursa si a unei Surse definite in back-office-ul Interfete (modulul Interfete). Tipul de Sursa defineste entitatile disponibile pentru export (reunite in Grupuri) iar Sursa defineste un context distinct de integrare (valori de parametri, context incremental etc.).

Intr-o sesiune de lucru, se apeleaza metoda *LogIn* o singura data si apoi se foloseste *ClientToken*-ul returnat pentru toate apelurile urmatoare. Daca la apelul unei metode se primeste eroare in legatura cu expirarea sesiunii de lucru (... <ExceptionInfo Code="invalid\_token"> ...), se apeleaza din nou metoda *LogIn* pentru obtinerea unui *ClientToken* valid (cu care se vor face apelurile urmatoare). La sfarsitul sesiunii de lucru trebuie apelata metoda *LogOut*.

In caz de eroare, se semnaleaza o eroare de tip SoapException. Pe campul Message (nodul <faultstring>) se gaseste eroarea care poate fi afisata utilizatorului final. Pentru erorile care nu pot fi corectate de utilizator (erori de integrare/programare) campul Message contine textul "Eroare interna !" (hardcodat) concatenat cu mesajul complet (pentru depanari/debug ar trebui afisat tot mesajul iar pentru productie ar trebui afisat doar textul "Eroare interna !"). Mesajul complet de eroare se gaseste in SoapException.Detail.InnerText/InnerXML (nodul <ExceptionInfo>) (in functie de limbajul de programare). Nodul <ExceptionInfo> contine si atributul Code cu una dintre cele 3 valori: *invalid\_credentials* (eroare la autentificare), *invalid\_token* (apel cu token invalid sau a expirat sesiunea de lucru) si *internal\_error* (pentru restul cazurilor de eroare).

De exemplu, in caz de eroare XML-ul SOAP arata ca si mai jos:

```
<?xml version="1.0" encoding="utf-16"?>
<soap:Envelope [ ... ] >
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>Eroare la identificarea utilizatorului sau a parolei !</faultstring>
      <faultactor>http://localhost/BGInterfaces WS/IIInterfacesBO3.asmx</faultactor>
      <detail>
        <ExceptionInfo Code="invalid_credentials">·55300·Eroare la
CInterfacesBO.Initializare !·BGInterfaces.CInterfacesBO.Initializare(penmVersiune = 1,
pstrNumeFirma = Firma)···0··0··59055·Eroare la identificarea utilizatorului sau a parolei
!·BGInterfaces.CInterfacesBO.Initializare···0··0··</ExceptionInfo>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## 1. LogIn

Inainte de a face orice operatie de import sau export, este necesara autentificarea cu metoda *LogIn*. Acesta metoda permite autentificarea si crearea unei sesiuni de lucru pe firma specificata. Daca s-a realizat cu succes atunci returneaza un *ClientToken* folosit pentru toate apelurile ulterioare.

### A. Declaratie

```
public string LogIn(int plngVersiune, string pstrNumeFirma, string pstrUtilizator, string
pstrParolaSecurizata)
```

### B. IN

*plngVersiune*: Versiunea de interfata care se doreste a fi utilizata, adica valoarea 1.

*pstrNumeFirma*: Numele firmei pe care se vor face operatiunile de Import-Export.

*pstrUtilizator*: Numele utilizatorului cu care se face autentificarea.

*pstrParolaSecurizata*: Parola utilizatorului securizata cu algoritmul MD5.

### C. OUT

Se returneaza un string care contine elementul *ClientToken*, folosit pentru apelurile ulterioare.

Daca firma nu exista sau nu este corect configurata, daca autentificarea cu Utilizatorul si ParolaSecurizata nu se finalizeaza cu succes, pot fi semnalate erori.

## 2. Setare optiune

Seteaza una dintre optiunile de functionare ale integrarii.

### A. Declaratie

```
public void SetareOptiune(string ClientToken, string pstrNumeOptiune, object pvntValue)
```

### B. IN

*ClientToken*: Token-ul obtinut in urma apelului functiei LogIn

*pstrNumeOptiune*: Numele optiunii care se doreste a fi setata.

*pvntValue*: Valoarea care se doreste a fi setata.

### C. OUT

Nimic.

Optiuni existente si valori default:

- *NoSessionForExports* – default false; se seteaza la true daca se doreste sa nu se inregistreze sesiuni si rezultate pentru apelul metodei *ExportEx* (pentru procese online ce utilizeaza intensiv metoda *ExportEx*)

## 3. ExportEx

Exporta date specifice prin intermediul entitatilor de export definite intr-un tip de sursa personalizat.

### A. Declaratie

```
public string ExportEx(string ClientToken, string pstrDenumireSursa, string pstrGrup, string pstrEntitate, string pstrFiltersAsXML)
```

### B. IN

*ClientToken*: Token-ul obtinut in urma apelului functiei LogIn

*pstrDenumireSursa*: Numele sursei.

*pstrGrup*: Numele grupului.

*pstrEntitate*: Numele entitatii.

*pstrFiltersAsXML*: Contine filtrele aplicate interogarii care sta la baza exportului, in format XML.

Parametrul *pstrFiltersAsXML* contine toate filtrele (nume si valori) care se doresc a fi aplicate interogarii si este de forma unui XML cu urmatoarea sintaxa:

```
<Filters>
  <Filter>
    <Name>NUME_FILTRU_1</Name>
    <Values>
      <Value>VALOARE_1</Value>
      ...
      <Value>VALOARE_n</Value>
    </Values>
  </Filter>
  ...
  <Filter>
    <Name>NUME_FILTRU_2</Name>
    <Values>
      <Value>VALOARE_1</Value>
      ...
      <Value>VALOARE_m</Value>
    </Values>
  </Filter>
</Filters>
```

Pentru a dezactiva filtrele optionale se completeaza valoarea cu cuvantul rezervat DISABLED. Filtrele obligatorii nu pot fi dezactivate.

Pentru a filtra o lista de valori complementara, se poate folosi atributul MOD="NOT" in tag-ul VALUES. Acest nu poate fi folosit in cazul filtrelor de tip Data si Logic.

```
<Values MOD="NOT">
  <Value>VALOARE_1</Value>
  ...
</Values>
```

### C. OUT

Se returneaza un string continand rezultatul interogarii de export. Formatul este definit in tipul de sursa (cel mai comun este Tab Delimited, liniile fiind delimitate cu CRLF).

Campurile returnate la apelurile de export pot avea urmatoarele tipuri:

- Int – numar intreg
- Decimal – numere reale
- VarChar – valori de tip text
- DateTime – valori de tip data, de forma “DD/MM/YYYY”
- Bit – valori de tip logic, pot fi atat 0/1 cat si False/True

## 4. Import

Permite importul unor entitati predefinite.

### A. Declaratie

*public string Import(string ClientToken, string pstrEntitate, string pstrDateAsXML)*

### B. IN

*ClientToken*: Token-ul obtinut in urma apelului functiei LogIn

*pstrEntitate*: Numele entitatii.

*pstrDateAsXML*: Datele din care se importa entitatea, in format XML.

Parametrul *pstrDateAsXML* este de forma:

- pentru entitatile care nu au detaliu (ex. Produs, Client etc.):

```
<NUME_ENTITATE>
  <Antet>
    <NUME_CAMP_1>VALOARE_1</NUME_CAMP_1>
    ...
    <NUME_CAMP_n>VALOARE_n</NUME_CAMP_n>
  </Antet>
</NUME_ENTITATE>
```

- pentru entitatile care au si detaliu (ex. Comanda, Factura etc.):

```
<NUME_ENTITATE>
  <Antet>
    <NUME_CAMP_ANTET_1>VALOARE_1</NUME_CAMP_ANTET_1>
    ...
    <NUME_CAMP_ANTET_n>VALOARE_n</NUME_CAMP_ANTET_n>
  </Antet>
  <Detaliu>
    <Linie>
      <NUME_CAMP_DETALIU_1>VALOARE_LINIE_1</NUME_CAMP_DETALIU_1>
      ...
      <NUME_CAMP_DETALIU_m>VALOARE_LINIE_m</NUME_CAMP_DETALIU_m>
    </Linie>
    ...
    <Linie>
      <NUME_CAMP_DETALIU_1>VALOARE_LINIE_1</NUME_CAMP_DETALIU_1>
      ...
    </Linie>
  </Detaliu>
</NUME_ENTITATE>
```

```

        <NUME_CAMP_DETALIU_m>VALOARE_LINIE_m</NUME_CAMP_DETALIU_m>
    </Linie>
</Detaliu>
</ NUME_ENTITATE>

```

Pentru campurile de tip DateTime, formatul de data acceptat este “DD/MM/YYYY hh:mm:ss”.

### C. OUT

Se returneaza un string care contine rezultatele importului in urmatorul format XML:

```

<NUME_ENTITATE>
  <Antet>
    <ID_ENTITATE>valoarea ID-ului de entitate</ID_ENTITATE>
    <NUME_ATRIBUT_1>VALOARE_1</NUME_ATRIBUT_1>
    ...
    <NUME_ATRIBUT_n>VALOARE_n</NUME_ATRIBUT_n>
  </Antet>
</NUME_ENTITATE>

```

ID-ul entitatii salvate in baza de date sau ID-ul entitatii parinte se returneaza tot timpul. In unele cazuri se returneaza si alte attribute, specificate in sectiunile dedicate importului fiecarei entitati.

## 5. *SalvareRaportDocumentEx*

Permite generarea unui raport PDF aferent listarii unui document, prin intermediul entitatilor de export definite intr-un tip de sursa. Obtinerea fisierului PDF presupune un mecanism in doi pasi (vezi mai jos).

### A. Declaratie

```
public string SalvareRaportDocumentEx(string ClientToken, string pstrDenumireSursa, string
pstrGrup, string pstrEntitate, int plngID_Document)
```

### B. IN

*ClientToken*: Token-ul obtinut in urma apelului functiei LogIn

*pstrDenumireSursa*: Numele sursei.

*pstrGrup*: Numele grupului.

*pstrEntitate*: Numele entitatii. Este permisa si completarea cu valori de forma “<Nume\_entitate>.<Nume\_raport>” astfel incat sa se specifice in apel si numele raportului folosit pentru exportul documentului.

*plngID\_Document*: ID-ul documentului.

### C. OUT

Returneaza un *FileToken* care se foloseste, impreuna cu *ClientToken*, pentru download-ul ulterior al fisierului. Download-ul se solicita folosind un URL de forma:

http://AdresaWS/GetFile.ashx?InterfaceNumber=3&ClientToken=1234&FileToken=76543

Se permite download-ul o singura data iar URL-ul este valabil 5 minute.

## 6. *Logout*

Inchide sesiunea de lucru.

### A. Declaratie

```
public void Logout(string ClientToken)
```

### B. IN

*ClientToken*: Token-ul obtinut in urma apelului functiei LogIn

### C. OUT

Nimic.

## II. Entitati predefinite pentru import

### 1. Banca

Importa banca primita ca si parametru.

*Nume entitate:* Banca.

*UIName:* G.Banca (Meniul Parametri -> Banci).

#### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
Simbol	VarChar(25)	Simbol (Banca)
Denumire	VarChar(50)	Denumire (Nume)
CodSWIFT	VarChar(11)	CodSWIFT

#### B. OUT

Se returneaza ID-ul de Banca.

### 2. CategorieProdus

Importa categoria de produs primita ca si parametru.

*Nume entitate:* CategorieProdus.

*UIName:* G.CategoriiProduse (Meniul Parametri -> CategoriiProduse).

#### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
Denumire	VarChar(50)	Denumire
Descriere	VarChar(250)	Descriere
Observatii	VarChar(250)	Observatii

#### B. OUT

Se returneaza ID-ul de CategorieProdus.

### 3. Ramura

Importa ramura de produs primita ca si parametru.

*Nume entitate:* Ramura.

*UIName:* G.Ramuri (Meniul Parametri -> Ramuri).

#### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
Denumire	VarChar(50)	Denumire
Descriere	VarChar(250)	Descriere
CategorieProdus_ID	Int	CategorieProdus_ID
Observatii	VarChar(250)	Observatii

#### B. OUT

Se returneaza ID-ul de Ramura.

### 4. GrupaProdus

Importa grupa de produs primita ca si parametru.

*Nume entitate:* GrupaProdus.

UIName: G.GrupaProdus (Meniul Parametri -> Grupe Produse).

#### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
Denumire	VarChar(50)	Denumire
Prefix	VarChar(20)	Prefix
Sufix	VarChar(120)	Sufix
TipProdusAccizat_ID	Int	TipProdusAccizat_ID
Descriere	VarChar(255)	Descriere
Ramura_ID	Int	Ramura_ID
TipGrupa (0 = Normala, -1 = Serii, 1 = Speciala)	Int	TipGrupa
TaxaVP	Decimal(19, 6)	TaxaVP (Procent Taxa Vamala)
ComisionVP	Decimal(19, 6)	Se ignora acest camp
NrZileStocOptim	Int	NrZileStocOptim
Garantie	Int	Garantie
SerieLot (0 = NU, 1 = DA)	Bit	SerieLot
DataExpirare (0 = NU, 1 = DA)	Bit	DataExpirare (Se ia in considerare doar daca SerieLot = 1)
DeclaratieConformitate (0 = NU, 1 = DA)	Bit	DeclaratieConformitate (Se ia in considerare doar daca SerieLot = 1)
ZileExpirare	Int	ZileExpirare (se ia in considerare doar daca DataExpirare = 1)

#### B. OUT

Se returneaza ID-ul de GrupaProdus.

### 5. Produs

Importa produsul primit ca si parametru.

Nume entitate: Produs.

UIName: G.Nomenclator (Meniul Parametri -> Nomenclator).

#### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
Denumire	VarChar(250)	Denumire
Cod	VarChar(50)	Cod
DenumireScurta	VarChar(100)	DenumireScurta
Grupa_ID	Int	Grupa_ID (Grupa de produs)
Categorie_ID	Int	Categorie_ID (Categorie contabila)
CodVamal_ID	Int	CodVamal_ID (poate sa vina NULL)
Web	VarChar(255)	Web (poate sa vina NULL)
AparePeWeb (0 = NU, 1 = DA)	Bit	AparePeWeb
AUF	Decimal(19, 6)	AUF
AUP	Decimal(19, 6)	AUP
PVAMaxim	Decimal(19, 6)	PVA (PVA maxim)
UM_ID	Int	... UM_ID Propriu
SerieLot (0 = NU, 1 = DA)	Bit	SerieLot
DataExpirare (0 = NU, 1 = DA)	Bit	DataExpirare (Se ia in considerare doar daca SerieLot = 1)

DeclaratieConformitate (0 = NU, 1 = DA)	Bit	DeclaratieConformitate (Se ia in considerare doar daca SerieLot = 1)
ZileExpirare	Int	ZileExpirare (Se ia in considerare doar daca DataExpirare = 1)
ClasificareProdus_ID	Int	ClasificareProdus_ID
Observatii	VarChar(2000)	Observatii
Producator_ID	Int	Producator_ID (este un Tert Furnizor)
NrZileDinGrupe (0 = NU, 1 = DA)	Bit	NrZileDinGrupe
NrZileStocOptim	Int	NrZileStocOptim (Se ia in considerare doar daca NrZileDinGrupe = 0)
Text1	VarChar(250)	Text1 (poate sa vina NULL)
Text2	VarChar(250)	Text2 (poate sa vina NULL)
Text3	VarChar(250)	Text3 (poate sa vina NULL)
Text4	VarChar(250)	Text4 (poate sa vina NULL)
Double1	Decimal(19, 6)	Double1 (poate sa vina NULL)
Double2	Decimal(19, 6)	Double2 (poate sa vina NULL)
Double3	Decimal(19, 6)	Double3 (poate sa vina NULL)
Long1	Int	Long1 (poate sa vina NULL)
Long2	Int	Long2 (poate sa vina NULL)
Long3	Int	Long3 (poate sa vina NULL)
Bit1 (0 = NU, 1 = DA)	Bit	Bit1 (poate sa vina NULL)
Bit2 (0 = NU, 1 = DA)	Bit	Bit2 (poate sa vina NULL)
Bit3 (0 = NU, 1 = DA)	Bit	Bit3 (poate sa vina NULL)
Date1	DateTime	Date1 (se elimina componenta de Timp, daca exista)
Date2	DateTime	Date2 (se elimina componenta de Timp, daca exista)
Date3	DateTime	Date3 (se elimina componenta de Timp, daca exista)
Lista1_ID	Int	Lista1_ID (poate sa vina NULL)
Lista2_ID	Int	Lista2_ID (poate sa vina NULL)
Lista3_ID	Int	Lista3_ID (poate sa vina NULL)
Lista4_ID	Int	Lista4_ID (poate sa vina NULL)
Ierarhie1_ID	Int	Ierarhie1_ID (poate sa vina NULL)
Gestiune_ID	Int	... Gestiune_ID (trebuie sa fie o gestiune de tip Depozit sau Magazin; se importa informatiile de gestiuni doar daca Gestiune_ID <> 0)
PretVanzare	Decimal(19, 6)	... PretVanzare (Se ia in considerare doar daca Gestiune_ID <> 0; Daca Valuta_ID = 3 atunci trebuie sa aiba numarul de zecimale semnificative maxim ZP; Daca Valuta_ID <> 3 atunci trebuie sa aiba numarul de zecimale semnificative maxim ZV; Campul are sens pentru Gestiune de tip Depozit, din el calculandu-se PVA)
Valuta_ID	Int	... Valuta_ID (Se ia in considerare doar daca Gestiune_ID <> 0; pentru Gestiune de tip Magazin trebuie sa vina Valuta_ID = 3)
Adaos	Decimal(19, 6)	... Adaos (Se ia in considerare doar daca Gestiune_ID <> 0)



SL	Decimal(19, 6)	... SL (Stoc limita) (Se ia in considerare doar daca Gestione_ID <> 0)
Garantie	Int	... Garantie (Se ia in considerare doar daca Gestione_ID <> 0)
TVAP	Decimal(19, 6)	... TVAP (Se ia in considerare doar daca Gestione_ID <> 0)
PVA	Decimal(19, 6)	... PVA (Se ia in considerare doar daca Gestione_ID <> 0; Campul are sens pentru Gestione de tip Magazin, din el calculandu-se PretVanzare)
CodBara	VarChar(50)	... CodBara (Daca este valoarea AUTO atunci se considera Cod de bare Auto si se genereaza un cod automat conform definitiei; daca este completat de importa ca si codul de bara implicit)

#### B. OUT

Se returneaza ID-ul de Produs.

### 6. *UMProdus*

Importa unitatile de masura alternative ale produsului primite ca si parametru cu UMBaza=False.

*Nume entitate:* UMProdus.

*UIName:* G.Nomenclator (Meniul Parametri -> Nomenclator).

#### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	NomGeneral_ID
UM_ID	Int	UM_ID
C1	Int	C1 (Trebuie sa fie <> 0)
C2	Int	C2 (Trebuie sa fie <> 0)
Ambalaj_ID	Int	Ambalaj_ID (Se importa doar daca Ambalaj_ID <> 0; trebuie sa fie un Produs cu Categorie_ID = 26)

#### B. OUT

Se returneaza ID-ul de Produs.

### 7. *PretPerGestione*

Importa preturile per gestiuni ale produsului primite ca si parametru.

*Nume entitate:* PretPerGestione.

*UIName:* G.Nomenclator (Meniul Parametri -> Nomenclator).

#### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	NomGeneral_ID
Gestione_ID	Int	Gestione_ID (Trebuie sa fie o gestiune de tip Depozit sau Magazin)

PretVanzare	Decimal(19, 6)	PretVanzare (Daca Valuta_ID = 3 atunci trebuie sa aiba numarul de zecimale semnificative maxim ZP; Daca Valuta_ID <> 3 atunci trebuie sa aiba numarul de zecimale semnificative maxim ZV; Campul are sens pentru Gestiune de tip Depozit, din el calculandu-se PVA)
Valuta_ID	Int	Valuta_ID (Pentru Gestiune de tip Magazin trebuie sa vina Valuta_ID = 3)
Adaos	Decimal(19, 6)	Adaos
SL	Decimal(19, 6)	SL (Stoc limita)
Garantie	Int	Garantie
TVAP	Decimal(19, 6)	TVAP
PVA	Decimal(19, 6)	PVA (Campul are sens pentru Gestiune de tip Magazin, din el calculandu-se PretVanzare)

## B. OUT

Se returneaza ID-ul de Produs.

## 8. InfoAprovizionare

Importa informatiile de aprovizionare ale produsului primite ca si parametru.

*Nume entitate:* InfoAprovizionare.

*UIName:* G.Nomenclator (Meniul Parametri -> Nomenclator).

## A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	NomGeneral_ID
Tert_ID	Int	Tert_ID (este un Tert Furnizor)
Cod	VarChar(250)	Cod
Pret	Decimal(19, 6)	Pret (Pret Intreg)
DiscountP	Decimal(19, 6)	DiscountP
Valuta_ID	Int	Valuta_ID
ZileLivrare	Int	ZileLivrare
ZilePlata	Int	ZilePlata
ModLivrare	VarChar(50)	ModLivrare
TaraOrigine_ID	INT	TaraOrigine_ID (poate sa vina NULL)
Implicit (0 = NU, 1 = DA)	Bit	Implicit (La prima informatie de aprovizionare a unui produs se va considera Implicit = 1 indiferent de valoarea primita.)

## B. OUT

Se returneaza ID-ul de Produs.

## 9. CodBara

Importa codurile de bare ale produsului primite ca si parametru.

*Nume entitate:* CodBara.

*UIName:* G.Nomenclator (Meniul Parametri -> Nomenclator).

## A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
------	-----	----------

NomGeneral_ID	Int	NomGeneral_ID
Cod	VarChar(50)	Cod (Daca este valoarea AUTO atunci se considera Cod de bare Auto si se genereaza un cod automat conform definitiei)
UM_ID	Int	UM_ID (poate sa vina NULL)
Implicit (0 = NU, 1 = DA)	Bit	Implicit (La primul cod de bara a unui produs se va considera Implicit = 1 indiferent de valoarea primita.)

## B. OUT

Se returneaza ID-ul de Produs.

## 10. Client

Importa tertul primit ca si parametru.

*Nume entitate:* Client.

*UIName:* G.Tert (Meniul Parametri -> Terti).

## A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
TertMaster_ID	Int	TertMaster_ID (daca este NULL atunci adaugam un TertMaster, altfel adaugam un PL)
Denumire	VarChar(200)	Denumire (daca TertMaster_ID este completat campul se trimite fara prefixarea cu denumirea Tertului master)
Cod	VarChar(50)	Cod (daca este completat se importa, daca nu este completat se genereaza un cod; se verifica faptul ca acest cod este unic pentru clienti)
CodGLN	VarChar(13)	CodGLN (daca este completat se verifica sa fie unic pentru clienti)
Tara_ID	Int	Tara_ID (daca este NULL se lasa ce propune obiectul, adica Romania; daca TertMaster_ID este completat campul trebuie sa vina NULL)
Valuta_ID	Int	Valuta_ID (daca este NULL se lasa ce propune obiectul, adica Lei pentru Romania si Euro pentru Tara diferita de Romania; daca TertMaster_ID este completat campul trebuie sa vina NULL)
TipTert	Int	1 → Client 2 → Furnizor 3 → Client si Furnizor (daca TertMaster_ID este completat campul trebuie sa vina NULL)
Localitate_ID	Int	Localitate_ID
CodPostal	VarChar(15)	CodPostal
Adresa	VarChar(100)	Adresa
Latitudine	Decimal(19, 6)	Latitudine
Longitudine	Decimal(19, 6)	Longitudine
Telefon	VarChar(50)	Telefon
Fax	VarChar(50)	Fax
Email	VarChar(255)	Email

Web	VarChar(255)	Web
RegCom	VarChar(25)	RegCom (poate sa vina NULL daca TertMaster_ID este completat)
CodFiscal	VarChar(30)	CodFiscal (poate sa vina NULL daca PersoanaFizica = 1, altfel trebuie sa fie un CIF/CUI valid; daca TertMaster_ID este completat campul trebuie sa vina NULL)
PersoanaFizica (0 sau 1)	Bit	PersoanaFizica (daca TertMaster_ID este completat campul trebuie sa vina NULL)
Observatii	VarChar(255)	Observatii
Banca_ID	Int	ConturiBancare.Banca (se adauga un Cont doar daca este completat si Contul; daca TertMaster_ID este completat atunci se pastreaza doar acest cont bancar)
ContBancar	VarChar(50)	ConturiBancare.Denumire (se adauga un Cont doar daca este completata si Banca; daca TertMaster_ID este completat atunci se pastreaza doar acest cont bancar)
GrupaTert_ID	Int	GrupaTert_ID (daca este NULL se lasa ce propune obiectul, adica valoarea 1 <NORMAL>; daca TertMaster_ID este completat campul trebuie sa vina NULL)
ZileScadenta	Int	ZileScadenta (daca este NULL se lasa ce propune obiectul, adica GrupaTert.ZileScadenta)
Agent_ID	Int	Agent_ID (daca este NULL se lasa ce propune obiectul, adica agentul implicit al firmei)
Delegat_ID	Int	Delegat_ID (daca este NULL se lasa ce propune obiectul, adica valoarea 1 <NECUNOSCU>)
CategorieTert_ID	Int	CategorieTert_ID (daca este NULL se lasa ce propune obiectul, adica valoarea 1 <NECUNOSCU>)
LocatieTert_ID	Int	LocatieTert_ID (daca este NULL se lasa ce propune obiectul, adica valoarea 1 <NECUNOSCU>)
Exceptat1	Bit	Exceptat1
Exceptat2	Bit	Exceptat2
Text1	VarChar(250)	Text1 (poate sa vina NULL)
Text2	VarChar(250)	Text2 (poate sa vina NULL)
Text3	VarChar(250)	Text3 (poate sa vina NULL)
Text4	VarChar(250)	Text4 (poate sa vina NULL)
Double1	Decimal(19, 6)	Double1 (poate sa vina NULL)
Double2	Decimal(19, 6)	Double2 (poate sa vina NULL)
Double3	Decimal(19, 6)	Double3 (poate sa vina NULL)
Long1	Int	Long1 (poate sa vina NULL)
Long2	Int	Long2 (poate sa vina NULL)
Long3	Int	Long3 (poate sa vina NULL)
Bit1 (0 = NU, 1 = DA)	Bit	Bit1 (poate sa vina NULL)
Bit2 (0 = NU, 1 = DA)	Bit	Bit2 (poate sa vina NULL)

Bit3 (0 = NU, 1 = DA)	Bit	Bit3 (poate sa vina NULL)
Date1	DateTime	Date1 (se elimina componenta de Timp, daca exista)
Date2	DateTime	Date2 (se elimina componenta de Timp, daca exista)
Lista1_ID	Int	Lista1_ID (poate sa vina NULL)
Lista2_ID	Int	Lista2_ID (poate sa vina NULL)
Lista3_ID	Int	Lista3_ID (poate sa vina NULL)
Ierarhie1_ID	Int	Ierarhie1_ID (poate sa vina NULL)
Exceptat	Bit	Exceptat (poate sa vina NULL)
DataMaxExceptare	DateTime	DataMaxExceptare (se elimina componenta de Timp, daca exista; poate sa vina NULL; daca este completat trebuie sa fie completat si campul Exceptat cu valoarea 1; trebuie completat daca Exceptat = 1)
Blocat	Bit	Blocat (poate sa vina NULL)
ValidareCredit	Bit	ValidareCredit (poate sa vina NULL; campurile ValidareCredit, CreditMaxim si ValutaCredit_ID trebuie sa fie toate completate sau toate necompletate; pentru compatibilitate cu versiunile anterioare, daca campurile ValidareCredit si ValutaCredit_ID sunt NULL iar campul CreditMaxim este completat atunci se considera ValutaCredit_ID = LEI, iar valoarea -1 a campului CreditMaxim dezactiveaza validarea de credit maxim)
CreditMaxim	Decimal(19, 6)	CreditMaxim (poate sa vina NULL)
ValutaCredit_ID	Int	ValutaCredit_ID (poate sa vina NULL)
ValidareZileScadenta	Bit	ValidareZileScadenta (poate sa vina NULL; campurile ValidareZileScadenta, ZileScadentaA si ZileScadentaB trebuie sa fie toate completate sau toate necompletate)
ZileScadentaA	Int	ZileScadentaA (poate sa vina NULL)
ZileScadentaB	Int	ZileScadentaB (poate sa vina NULL)
ValidareZileVechime	Bit	ValidareZileVechime (poate sa vina NULL; campurile ValidareZileVechime, ZileVechimeA si ZileVechimeB trebuie sa fie toate completate sau toate necompletate)
ZileVechimeA	Int	ZileVechimeA (poate sa vina NULL)
ZileVechimeB	Int	ZileVechimeB (poate sa vina NULL)
ValidareNrFacturi	Bit	ValidareNrFacturi (poate sa vina NULL; campurile ValidareNrFacturi, NrFacturiA si NrFacturiB trebuie sa fie toate completate sau toate necompletate)
NrFacturiA	Int	NrFacturiA (poate sa vina NULL)
NrFacturiB	Int	NrFacturiB (poate sa vina NULL)
ObservatiiCredit	VarChar(255)	ObservatiiCredit

## B. OUT

Se returneaza ID-ul de Tert.

Observatii:

Pentru PL-uri nu se preiau Persoanele de contact de la TertMaster.

## 11. PersoanaContact

Importa persoana de contact primita ca si parametru ca si Persoana de Contact asociata a tertului cu ID-ul primit.

*Nume entitate:* PersoanaContact.

*UIName:* G.Tert (Meniul Parametri -> Terti).

### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
Tert_ID	Int	Tert_ID
Cod	VarChar(15)	Cod
Denumire	VarChar(50)	Denumire
Functia	VarChar(50)	Functia
Telefon	VarChar(15)	Telefon
Email	VarChar(255)	Email
Departament_ID	Int	Departament_ID (daca este NULL se lasa ce propune obiectul, adica valoarea 1 <NECUNOSCU>)

### B. OUT

Se returneaza ID-ul de Tert.

## 12. ContBancar

Importa contul bancar primit ca si parametru ca si cont bancar asociat tertului cu ID-ul primit.

*Nume entitate:* ContBancar.

*UIName:* G.Tert (Meniul Parametri -> Terti).

### A. IN

In *pstrDateAsXML* apare doar sectiunea <Antet>, o singura data, cu urmatoarele campuri:

Camp	Tip	Camp ERP
Tert_ID	Int	Tert_ID
Banca_ID	Int	Banca_ID
Denumire	VarChar(50)	Denumire
Implicit (0 = NU, 1 = DA)	Bit	Implicit (La primul cont bancar al unui tert se va considera Implicit = 1 indiferent de valoarea primita.)

### B. OUT

Se returneaza ID-ul de Tert.

## 13. Comanda

Importa o comanda primita ca si parametru ca si FP in orice Valuta sau FPM cu Valuta = LEI.

*Nume entitate:* Comanda.

*UIName:* G.FP (Meniul Vanzari -> Proforma).

### A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de comanda
- <Detaliu> reprezinta detaliul de comanda; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → FP 11 → FPM
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de FP-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit)
Referinta	VarChar(30)	Referinta
Tert_ID	Int	Tert_ID
Agent_ID	Int	Agent_ID (daca este NULL se lasa ce propune obiectul, adica Agentul Tertului)
Valuta_ID	Int	Valuta_ID (daca TipDoc = 1 si campul este NULL se considera valoarea 3 = LEI; daca TipDoc = 11 atunci campul trebuie sa fie NULL sau 3 = LEI)
TVA_ID	Int	TVA_ID (permite alegerea Regimului de TVA; daca este NULL se lasa ce propune obiectul; daca TipDoc = 11 atunci campul trebuie sa fie NULL, 0 = Taxabil sau 35 = Taxabila – TVA la incasare) 0 → Taxabila 35 → Taxabila – TVA la incasare 30 → Scutita cu drept de deducere - livrare IC art.294(2-a,d) (forma de 0) 5 → Scutita cu drept de deducere - export (forma de 0) 2 → Taxabila - cota TVA externa etc.
EvidentiereDiscount (0 = fara, 1 = cu)	Bit	EvidentiereDiscount
BancaFacturare_ID	Int	BancaFacturare_ID (daca este NULL se lasa ce propune obiectul, adica Banca implicita la iesiri a gestiunii facturii)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
DataLivrare	DateTime	DataLivrare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul)
NrZileScadenta	Int	ZileScadenta (daca este NULL se lasa ce propune obiectul, adica zilele de scadenta ale tertului)
RecalculeazaPretUnitar (0 = NU, 1 = DA)	Bit	RecalculeazaPretUnitar (daca este NULL se considera valoarea 0 = NU; )
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
ContTert_ID	Int	ContTert_ID (Contul bancar al tertului; daca este NULL se lasa ce propune obiectul, adica Contul bancar implicit al tertului)

ProcentAgent	Decimal(19, 6)	ProcentAgent (Procent Comision agent; daca este NULL se lasa ce propune obiectul, adica comisionul agentului)
TipPlataFC_ID	Int	TipPlataFC_ID (daca este NULL se lasa ce propune obiectul) 0 → La termen 1 → Pe loc 2 → Ramburs 3 → Card 4 → Ordin de plata 5 → Efect comercial 6 → Exceptat
ObservatiiTipPlataFC	VarChar(255)	ObservatiiTipPlataFC (daca este NULL se lasa ce propune obiectul)
Delegat	VarChar(50)	Delegat (daca este NULL se lasa ce propune obiectul)
Transport	VarChar(30)	Transport (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NrMasina	VarChar(10)	NrMasina (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
SerieCI	VarChar(10)	SerieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NumarCI	VarChar(20)	NumarCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
PolitieCI	VarChar(50)	PolitieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

<b>Camp</b>	<b>Tip</b>	<b>Camp ERP</b>
GrupID	Int	GrupID
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
PretUnitar	Decimal(19, 6)	Pret (daca FP.Valuta <> LEI atunci campul trebuie sa fie NULL; daca TipDoc = 11 atunci campul are sensul de PretAmanunt adica cu TVA inclus)
PretValuta	Decimal(19, 6)	PretValuta (daca TipDoc = 11 atunci campul trebuie sa fie NULL)
Valuta_ID	Int	Valuta_ID (daca FP.Valuta <> LEI atunci campul trebuie sa aiba valorile NULL sau FP.Valuta; daca TipDoc = 11 atunci campul trebuie sa fie NULL))
DiscountP	Decimal(19, 6)	DiscountP (daca este NULL se lasa ce propune obiectul)



TVAP	Decimal(19, 6)	TVAP (daca este NULL se lasa ce propune obiectul; daca TVA_ID din antet corespunde unei forme de 0 atunci campul trebuie sa fie cu valoarea NULL sau 0)
Garantie	Int	Garantie (Se ia pentru toate liniile mai putin pentru linia de discount de grup; daca este NULL se lasa ce propune obiectul)
TipOperatieStoc	Int	1 = Rezerva tot 2 = Comanda tot 3 = Confirma cat poti, restul rezerva 4 = Confirma cat poti, restul comanda 5 = Confirma tot (daca nu se poate bloca toata catitatea se da eroare)
Observatii	VarChar(500)	Observatii

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (FP), Suma si TVA.

Observatii:

Pentru cazurile de "Confirma cat poti", se confirma (blocheaza) cat se poate de pe fiecare linie, dar nu mai mult de SD. Pentru produsele speciale si discounturi campul TipOperatieStoc se ignora.

(\*) Transmiterea discountului la nivel de detaliu depinde de modul de evidentiare al discountului EvidentiazaDiscount

= 0 → se transmite discountul inclus în pret (pret e pretul discountat)

= 1 → se transmite discountul evidentiat (pret e pretul fara discountul evidentiat aplicat)

Ambele variante preiau pretul si discountul si il inregistreaza nemodificat (deci fara sa-l mai discounteze).

Pentru varianta cu evidentiare putem avea mai multe grupuri si mai multe linii libere. Un grup se identifica prin acelasi GrupID  $\neq 0$  si contine una sau mai multe linii de produse normale si o singura linie de discount de grup, care trebuie sa fie ultima linie in grup, cu Cantitate = 1 si Pret = suma discounturilor de pe liniile cu produse normale. O linie libera are GrupID = 0 si poate contine un produs de tip discount sau un produs special. Atentie: In cadrul unui Grup toate produsele (inclusiv cel de discount) trebuie sa aiba acelasi TVAP. In caz contrar se da eroare.

Pentru varianta fara evidentiare putem avea numai linii libere. O linie libera are GrupID = 0 si poate contine orice tip de produs.

Daca FP.Valuta = LEI, in functie de valorile lui Pret si PretValuta avem urmatoarele variante de utilizare:

- Pivotal este Pret: Pret  $\neq$  NULL si PretValuta = orice; daca PretValuta  $\neq$  NULL si Valuta\_ID  $\neq$  NULL se pastreaza acest PretValuta, altfel PretValuta = Pret / Cursul bancii de facturare
- Pivotal este PretValuta: PretValuta  $\neq$  NULL si Pret = NULL; Pret se calculeaza din PretValuta la cursul bancii de facturare
- Pivotal este Nomenclatorul: Pret = NULL, PretValuta = NULL, Valuta\_ID = NULL si DiscountP = orice; se aduc din nomenclator valorile pentru PretValuta, Valuta\_ID si DiscountP daca trebuie; Pret se calculeaza din PretValuta la cursul bancii de facturare

Daca FP.Valuta  $\neq$  LEI, in functie de valorile lui PretValuta avem urmatoarele variante de utilizare:

- Pivotal este PretValuta: PretValuta  $\neq$  NULL; Pret se calculeaza din PretValuta la cursul BNR
- Pivotal este Nomenclatorul: PretValuta = NULL, Valuta\_ID = NULL si DiscountP = orice; se aduc din nomenclator valorile pentru PretValuta, Valuta\_ID si DiscountP daca trebuie; Pret se calculeaza din PretValuta la cursul BNR

Pentru linia de Discount de Grup trebuie sa vina toate campurile dar se copiaza doar Stoc\_ID-ul si se verifica GrupID-ul sa fie corect.

Nu se verifica numarul de linii.

## 14. Factura

Importa o factura primita ca si parametru ca si FCD, FCD din FP sau FCA din FP in orice Valuta sau FCDM sau FCDM din FPM cu Valuta = Lei.

Nume entitate: Factura.

UIName: G.FCD (Meniul Vanzari -> Factura si Aviz).

### A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de factura
- <Detaliu> reprezinta detaliul de factura; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → FCD 2 → FCD (din FP), FCDM (din FPM) 3 → FCA (din FP), FCAM (din FPM) 11 → FCDM
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de FC-uri din baza de date)
FP_ID	Int	FP_ID (daca TipDoc IN (1, 11) atunci campul trebuie sa fie NULL)
InchideFP (0 = NU, 1 = DA)	Bit	InchideFP (daca TipDoc IN (1, 3, 11) atunci campul trebuie sa fie NULL; daca TipDoc = 2 si campul este NULL atunci se lasa ce propune obiectul, adica optiunea pe firma "Inchidere FP dupa avizare")
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit sau Van-sales; daca TipDoc IN (2, 3) atunci validam ca Gestiune_ID = FP.Gestiune_ID)
Prefix.Numar.Sufix Serie	As VarChar(22)	Serie (Daca este valoarea AUTO atunci se propune o serie din plaja, altfel se salveaza cu valoarea trimisa; se permite ca Seria sa fie si dintr-o plaja definita la FC)
Tert_ID	Int	Tert_ID (Trebuie sa fie Client; daca TipDoc IN (2, 3) atunci validam ca Tert.TertMaster_ID = FP.Tert.TertMaster_ID daca campul este completat)
ContTert_ID	Int	ContTert_ID (Contul bancar al tertului; daca este NULL se lasa ce propune obiectul, adica Contul bancar implicit al tertului sau FP.ContTert_ID)
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca TipDoc IN (1, 2, 11) si campul este NULL se lasa ce propune obiectul, adica Agentul Tertului sau FP.Agent_ID; daca TipDoc = 3 atunci campul trebuie sa fie NULL)

ProcentAgent	Decimal(19, 6)	ProcentAgent (Procent Comision agent; daca este NULL se lasa ce propune obiectul, adica comisionul agentului sau FP.ProcentAgent)
Valuta_ID	Int	Valuta_ID (daca TipDoc = 1 si campul este NULL se considera valoarea 3 = LEI; daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL; daca TipDoc = 11 atunci campul trebuie sa fie NULL sau 3 = LEI)
TVA_ID	Int	TVA_ID (permite alegerea Regimului de TVA; daca TipDoc = 1 si campul este NULL se lasa ce propune obiectul; daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL; daca TipDoc = 11 atunci campul trebuie sa fie NULL, 0 = Taxabil sau 35 = Taxabila – TVA la incasare) 0 → Taxabila 35 → Taxabila – TVA la incasare 30 → Scutita cu drept de deducere - livrare IC art.294(2-a,d) (forma de 0) 5 → Scutita cu drept de deducere - export (forma de 0) 2 → Taxabila - cota TVA externa etc.
EvidentiereDiscount (0 = fara, 1 = cu)	Bit	EvidentiereDiscount (daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL)
BancaFacturare_ID	Int	BancaFacturare_ID (daca este NULL se lasa ce propune obiectul, adica Banca implicita la iesiri a gestiunii facturii; daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
DataScadenta	DateTime	DataScadenta (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica se tine cont de zilele de scadenta ale tertului sau se tine cont de FP.ZileScadenta)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
PersoanaLivrare_ID	Int	PersoanaLivrare_ID (daca este NULL se lasa ce propune obiectul, adica o persoana de livrare identica cu utilizatorul, sau cea implicita de pe gestiune)
Delegat	VarChar(50)	Delegat (daca este NULL se lasa ce propune obiectul, adica Delegatul Tertului)
Transport	VarChar(30)	Transport (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NrMasina	VarChar(10)	NrMasina (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de

		NULL)
SerieCI	VarChar(10)	SerieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NumarCI	VarChar(20)	NumarCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
PolitieCI	VarChar(50)	PolitieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
DataExpeditie	DateTime	DataLivrare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica Data documentului)
		OraLivrare (se elimina componenta de Data, daca exista; daca este NULL se lasa ce propune obiectul, adica Ora curenta)
TipPlataFC_ID	Int	TipPlataFC_ID (daca este NULL se lasa ce propune obiectul) 0 → La termen 1 → Pe loc 2 → Ramburs 3 → Card 4 → Ordin de plata 5 → Efect comercial 6 → Exceptat
ObservatiiTipPlataFC	VarChar(255)	ObservatiiTipPlataFC (daca este NULL se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
GrupID	Int	GrupID (daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL)
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
Pret	Decimal(19, 6)	Pret (daca TipDoc = 1 si FCD.Valuta <> LEI atunci campul trebuie sa fie NULL; daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL; daca TipDoc = 11 atunci campul are sensul de PretAmanunt)
PretValuta	Decimal(19, 6)	PretValuta (TipDoc IN (2, 3, 11) atunci campul trebuie sa fie NULL)
Valuta_ID	Int	Valuta_ID (daca TipDoc = 1 si FCD.Valuta <> LEI atunci campul trebuie sa aiba valorile NULL sau FCD.Valuta; daca TipDoc IN (2, 3, 11) atunci campul trebuie sa fie NULL)
DiscountP	Decimal(19, 6)	DiscountP (daca este NULL se lasa ce propune obiectul; daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL)

TVAP	Decimal(19, 6)	TVAP (daca este NULL se lasa ce propune obiectul; daca TipDoc = 1 si TVA_ID din antet corespunde unei forme de 0 atunci campul trebuie sa fie cu valoarea NULL sau 0; daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL)
Garantie	Int	Garantie (Se ia pentru toate liniile mai putin pentru linia de discount de grup; daca este NULL se lasa ce propune obiectul; daca TipDoc IN (2, 3) atunci campul trebuie sa fie NULL)
Observatii	VarChar(500)	Observatii (daca TipDoc = 2 si campul este NULL se lasa ce propune obiectul, adica FPItem.Observatii; daca TipDoc = 3 atunci campul trebuie sa fie NULL)
Serie	VarChar(100)	Serie (poate sa fie NULL; daca este completata atunci trebuie sa avem Cantitate = 1 si SerieLot = NULL)
SerieLot	VarChar(25)	SerieLot (poate sa fie NULL)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca SerieLot este NULL atunci DataExpirare trebuie sa fie NULL)

Data TipDoc = 3 atunci nu trebuie sa se transmita informatiile de detaliu (FCA are detaliul identic cu FP).

Criteriu de grupare: Liniile din detaliu care au completat campul SerieLot se grupeaza dupa Stoc\_ID si Pret (sau PretValuta pentru FCD.Valuta  $\diamond$  LEI). Da eroare daca liniile din grupare difera prin unul dintre campurile GrupID, DiscountP, TVAP, Garantie sau Observatii.

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (FCA sau FCD), Suma si TVA.

Observatii:

(\*) Transmiterea discountului la nivel de detaliu depinde de modul de evidentiare al discountului EvidentiazaDiscount

= 0  $\rightarrow$  se transmite discountul inclus în pret (pret e pretul discountat)

= 1  $\rightarrow$  se transmite discountul evidentiat (pret e pretul fara discountul evidentiat aplicat)

Ambele variante preiau pretul si discountul si il inregistreaza nemodificat (deci fara sa-l mai discounteze).

Pentru varianta cu evidentiare putem avea mai multe grupuri si mai multe linii libere. Un grup se identifica prin acelasi GrupID  $\diamond$  0 si contine una sau mai multe linii de produse normale si o singura linie de discount de grup, care trebuie sa fie ultima linie in grup, cu Cantitate = 1 si Pret = suma discounturilor de pe liniile cu produse normale. O linie libera are GrupID = 0 si poate contine un produs de tip discount sau un produs special. Atentie: In cadrul unui Grup toate produsele (inclusiv cel de discount) trebuie sa aiba acelasi TVAP. In caz contrar se da eroare.

Pentru varianta fara evidentiare putem avea numai linii libere. O linie libera are GrupID = 0 si poate contine orice tip de produs.

Daca FCD.Valuta = LEI, in functie de valorile lui Pret si PretValuta avem urmatoarele variante de utilizare:

- Pivotal este Pret: Pret  $\diamond$  NULL si PretValuta = orice; daca PretValuta  $\diamond$  NULL si Valuta\_ID  $\diamond$  NULL se pastreaza acest PretValuta, altfel PretValuta = Pret / Cursul bancii de facturare
- Pivotal este PretValuta: PretValuta  $\diamond$  NULL si Pret = NULL; Pret se calculeaza din PretValuta la cursul bancii de facturare

- Pivotal este Nomenclatorul: Pret = NULL, PretValuta = NULL, Valuta\_ID = NULL si DiscountP = orice; se aduc din nomenclator valorile pentru PretValuta, Valuta\_ID si DiscountP daca trebuie; Pret se calculeaza din PretValuta la cursul bancii de facturare

Daca FCD.Valuta <> LEI, in functie de valorile lui PretValuta avem urmatoarele variante de utilizare:

- Pivotal este PretValuta: PretValuta <> NULL; Pret se calculeaza din PretValuta la cursul BNR
- Pivotal este Nomenclatorul: PretValuta = NULL, Valuta\_ID = NULL si DiscountP = orice; se aduc din nomenclator valorile pentru PretValuta, Valuta\_ID si DiscountP daca trebuie; Pret se calculeaza din PretValuta la cursul BNR

Pentru linia de Discount de Grup trebuie sa vina toate campurile dar se copiaza doar Stoc\_ID-ul si se verifica GrupID-ul sa fie corect.

Nu se verifica numarul de linii si creditul tertului.

## 15. BonFiscal

Importa un bon fiscal primit ca si parametru ca si BF.

*Nume entitate:* BonFiscal.

*UIName:* G.BF (Meniul Vanzari -> Bon Fiscal).

### A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de bon fiscal
- <Detaliu> reprezinta detaliul de bon fiscal; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
Numar	Int	Trebuie sa fie <> NULL
CasaMarcat_ID	Int	CasaMarcat_ID (din casa de marcat se obtine si Gestione_ID care trebuie sa fie de tip Magazin)
Casier_ID	Int	Casier_ID (Agent de tip Casier)
Tert_ID	Int	Tert_ID (daca este NULL se lasa ce propune obiectul, adica Tertul gestiunii)
Agent_ID	Int	Agent_ID (daca este NULL se lasa ce propune obiectul, adica Agentul tertului)
CodFiscal	VarChar(30)	CodFiscal
Data	DateTime	Data (se elimina componenta de Timp) Ora (se elimina componenta de Data; ora la care s-a listat bonul fiscal pe casa de marcat)
TotalCHS	Decimal(19, 6)	TotalCHS - Numerar (Campurile TotalCHS, TotalTV si TotalCI pot fi toate completate sau toate NULL caz in care se considera incasare integrala cu numerar)
TotalTV	Decimal(19, 6)	TotalTV - Tichet Valoric (Campurile TotalCHS, TotalTV si TotalCI pot fi toate completate sau toate NULL caz in care se considera incasare integrala cu numerar)

TotalCI	Decimal(19, 6)	TotalCI - Card (Campurile TotalCHS, TotalTV si TotalCI pot fi toate completate sau toate NULL caz in care se considera incasare integrala cu numerar)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	... Stoc ID
Cantitate	Decimal(19, 6)	Cantitate
PVA	Decimal(19, 6)	PVA
TVAP	Decimal(19, 6)	TVAP (se foloseste doar pentru verificare)
Serie	VarChar(100)	Serie (poate sa fie NULL)
SerieLot	VarChar(25)	SerieLot (poate sa fie NULL)

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de BonFiscal (BF).

Observatii:

BF-ul va avea Sursa Import. Daca campurile TotalCHS, TotalTV si TotalCI sunt completate atunci Tipariri = 1 (si nu se va putea sterge), altfel Tipariri = 0. Numarul BF-ului trebuie sa fie unic dupa cheia Numar, Data, Ora, CasaMarcat\_ID, Casier\_ID. Totalul bonului SUM(Cantitate \* PVA) trebuie sa fie egal cu TotalCHS + TotalTV + TotalCI si toate cele 4 totaluri trebuie sa fie pozitive.

## 16. FacturaSpeciala

Importa o factura primita ca si parametru ca si FCE in orice Valuta sau FCEBF, FCP sau FCL cu Valuta = LEI.

Nume entitate: FacturaSpeciala.

UIName: G.FCE (Meniul Vanzari -> Factura Speciala).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de factura speciala
- <Detaliu> reprezinta detaliul de factura speciala; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → FCE 2 → FCEBF 3 → FCP Servicii 4 → FCP Diferente 5 → FCP Chirii 6 → FCL Lucrari 7 → FCL Configurari

Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False
		Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de FC-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit pentru TipDoc IN (1, 3, 4, 5, 6, 7) sau Magazin pentru TipDoc = 2)
Prefix.Numar.Suffix Serie	As VarChar(22)	Serie (Daca este valoarea AUTO atunci se propune o serie din plaja, altfel se salveaza cu valoarea trimisa; se permite ca Seria sa fie si dintr-o plaja definita la FC)
Tert_ID	Int	Tert_ID (Trebuie sa fie Client)
ContTert_ID	Int	ContTert_ID (Contul bancar al tertului; daca este NULL se lasa ce propune obiectul, adica Contul bancar implicit al tertului)
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului)
ProcentAgent	Decimal(19, 6)	ProcentAgent (Procent Comision agent, daca este NULL se lasa ce propune obiectul, adica comisionul agentului; daca TipDoc = 2 atunci campul trebuie sa fie NULL)
Valuta_ID	Int	Valuta_ID (daca TipDoc = 1 si campul este NULL se considera valoarea 3 = LEI; daca TipDoc IN (2, 3, 4, 5, 6, 7) atunci campul trebuie sa fie NULL sau 3 = LEI)
TVA_ID	Int	TVA_ID (permite alegerea Regimului de TVA; daca este NULL se lasa ce propune obiectul; daca TipDoc = 2 atunci campul trebuie sa fie NULL sau 0 = Taxabil) 0 → Taxabila 35 → Taxabila – TVA la incasare 30 → Scutita cu drept de deducere - livrare IC art.294(2-a,d) (forma de 0) 5 → Scutita cu drept de deducere - export (forma de 0) 2 → Taxabila - cota TVA externa etc.
BancaFacturare_ID	Int	BancaFacturare_ID (daca este NULL se lasa ce propune obiectul, adica Banca implicita la iesiri a gestiunii facturii)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
DataScadenta	DateTime	DataScadenta (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica se tine cont de zilele de scadenta ale tertului)



Legatura_BF	Int	Legatura_BF (Este ID-ul de Bon Fiscal pe baza caruia s-a facut Factura; pentru TipDoc = 2 trebuie sa vina completat si nu mai trebuie sa se trimita linii in detaliu pentru ca se iau detaliile BF-ului; daca TipDoc IN (1, 3, 4, 5, 6 7) atunci campul trebuie sa fie NULL)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
PersoanaLivrare_ID	Int	PersoanaLivrare_ID (daca este NULL se lasa ce propune obiectul, adica o persoana de livrare identica cu utilizatorul sau cea implicita de pe gestiune)
Delegat	VarChar(50)	Delegat (daca este NULL se lasa ce propune obiectul, adica Delegatul Tertului)
Transport	VarChar(30)	Transport (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NrMasina	VarChar(10)	NrMasina (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
SerieCI	VarChar(10)	SerieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NumarCI	VarChar(20)	NumarCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
PolitieCI	VarChar(50)	PolitieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
DataExpeditie	DateTime	DataLivrare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica Data documentului)
		OraLivrare (se elimina componenta de Data, daca exista; daca este NULL se lasa ce propune obiectul, adica Ora curenta)
TipPlataFC_ID	Int	TipPlataFC_ID (daca este NULL se lasa ce propune obiectul; daca TipDoc = 2 atunci campul trebuie sa fie NULL) 0 → La termen 1 → Pe loc 2 → Ramburs 3 → Card 4 → Ordin de plata 5 → Efect comercial 6 → Exceptat
ObservatiiTipPlataFC	VarChar(255)	ObservatiiTipPlataFC (daca este NULL se lasa ce propune obiectul; daca TipDoc = 2 atunci campul trebuie sa fie NULL)

Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)
------------	-------------	--

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
Denumire	VarChar(2000)	Denumire
UM_ID	Int	UM_ID
Cantitate	Decimal(19, 6)	Cantitate
Pret	Decimal(19, 6)	Pret (daca TipDoc = 1 si FCE.Valuta <> LEI atunci campul trebuie sa fie NULL)
PretValuta	Decimal(19, 6)	PretValuta
Valuta_ID	Int	Valuta_ID (daca TipDoc = 1 si FCE.Valuta <> LEI atunci campul trebuie sa aiba valoarea FCE.Valuta; daca TipDoc = 2 atunci campul trebuie sa aiba valoarea 3 = LEI)
DiscountP	Decimal(19, 6)	DiscountP (daca este NULL se lasa ce propune obiectul)
TVAP	Decimal(19, 6)	TVAP (daca TVA_ID din antet corespunde unei forme de 0 atunci campul trebuie sa fie cu valoarea 0)
Garantie	Int	Garantie

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (FCE/FCEBF/FCP/FCL), Suma si TVA.

Observatii:

Nu se verifica numarul de linii si creditul tertului.

## 17. Incasare

Importa o incasare primita ca si parametru ca si CHS sau EC in orice Valuta sau EC din Efect cu Valuta = Lei.

Nume entitate: Incasare.

UIName: G.CHS (Meniul Incasari/Plati -> Inregistrare Incasari -> Chitanta).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de incasare
- <Detaliu> reprezinta detaliul de incasare; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → CHS 2 → EC 12 → EC din Efect
TipIncasare (1 = Normal, 2 = CuAvans)	Int	AreAvans (Daca TipDoc = 12 atunci campul trebuie sa fie NULL)
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False

		Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de CHS/EC-uri din baza de date)
TipRezolvareFacturi	Int	Folosit pentru rezolvarea automata a facturilor din detaliu: NULL = Fara rezolvare 1 = Rezolvare prin determinare suma exacta, altfel FIFO dupa DataScadenta (inclusiv facturi negative); nu trebuie transmis detaliu Daca TipDoc = 12 atunci campul trebuie sa fie NULL
Efect_ID	Int	Efect_ID (daca TipDoc IN (1, 2) atunci campul trebuie sa fie NULL)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit sau Magazin)
Prefix.Numar.Suffix Serie	As VarChar(22)	Serie (Se considera doar pentru TipDoc = 1, altfel trebuie sa vina NULL. Daca este valoarea AUTO atunci se propune o serie din plaja, altfel se salveaza cu valoarea trimisa; se permite ca Seria sa fie si dintr-o plaja definita la CHS)
Tert_ID	Int	Tert_ID (Trebuie sa fie Client; se permite NULL daca sunt transmise facturi in detaliu si se propune Tert_ID-ul primei facturi din detaliu)
ContTert_ID	Int	ContTert_ID (Contul bancar al tertului; se considera doar pentru TipDoc = 2, altfel trebuie sa vina NULL. Daca este NULL se lasa ce propune obiectul, adica Contul bancar implicit al tertului)
Agent_ID	Int	Agent_ID (se considera doar pentru TipDoc IN (1, 2), altfel trebuie sa vina NULL; poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului)
Valuta_ID	Int	Valuta_ID (daca este NULL se considera valoarea 3 = LEI; pentru TipDoc = 12 trebuie sa aiba valoarea NULL sau 3)
Cont_ID	Int	Cont_ID (daca este NULL se lasa ce propune obiectul, adica contul de casa/banca implicita a gestiunii incasarii; pentru TipDoc = 1 trebuie sa vina un cont de casa si pentru TipDoc IN (2, 12) trebuie sa vina un cont de banca; contul trebuie sa aiba Valuta = Valuta_ID)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
NumarDoc	Int	Referinta (Numar EC; Se considera doar pentru TipDoc IN (2, 12), altfel trebuie sa vina NULL)
NumarDocPlata	VarChar(25)	NumarDocPlata (Numar OP; Se considera doar pentru TipDoc IN (2, 12), altfel trebuie sa vina NULL)

DataDocPlata	DateTime	DataDocPlata (Data OP; se elimina componenta de Timp, daca exista; se considera doar pentru TipDoc IN (2, 12), altfel trebuie sa vina NULL)
Casier	VarChar(30)	Casier (Se considera doar pentru TipDoc = 1, altfel trebuie sa vina NULL)
TVAPAvans_ID	Int	TVAPAvans_ID (Se considera doar pentru TipIncasare = 2, altfel trebuie sa vina NULL; daca este NULL se lasa ce propune obiectul, adica Procent TVA avans din Optiuni Contari)
TipAvans_ID	Int	TipAvans_ID (Se considera doar pentru TipIncasare = 2, altfel trebuie sa vina NULL; daca este NULL se lasa ce propune obiectul, adica Tip Avans Incasari din Optiuni Contari)
Suma	Decimal(19, 6)	Suma (Daca TipDoc = 12 atunci se verifica faptul ca Suma = Efect.Suma)
Comision	Decimal(19, 6)	Comision (Se considera doar pentru TipDoc IN (2, 12), altfel trebuie sa vina NULL)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
FC_ID	Int	FC_ID
Valoare	Decimal(19, 6)	Valoare

Daca TipDoc = 12 atunci nu trebuie sa se transmita informatiile de detaliu (EC-ul are detaliul identic cu Efectul).

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (CHS/EC).

## 18. Plata

Importa o plata primita ca si parametru ca si DP sau OPM in orice Valuta sau OPM din Efect cu Valuta = Lei.

*Nume entitate:* Plata.

*UIName:* G.DP (Meniul Incasari/Plati -> Inregistrare Plati -> Dispozitie de Plata).

### A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de plata
- <Detaliu> reprezinta detaliul de plata; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
------	-----	----------

TipDoc	Int	1 → DP 2 → OPM 12 → OPM din Efect
TipPlata (1 = Normal, 2 = CuAvans)	Int	AreAvans (Daca TipDoc = 12 atunci campul trebuie sa fie NULL)
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False
		Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de DP/OPM-uri din baza de date)
TipRezolvareFacturi	Int	Folosit pentru rezolvarea automata a facturilor din detaliu: NULL = Fara rezolvare 1 = Rezolvare prin determinare suma exacta, altfel FIFO dupa DataScadenta (inclusiv facturi negative); nu trebuie transmis detaliu Daca TipDoc = 12 atunci campul trebuie sa fie NULL
Efect_ID	Int	Efect_ID (daca TipDoc IN (1, 2) atunci campul trebuie sa fie NULL)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit sau Magazin)
Serie	VarChar(30)	Referinta (Se considera doar pentru TipDoc = 1, altfel trebuie sa vina NULL)
Tert_ID	Int	Tert_ID (Trebuie sa fie Furnizor; se permite NULL daca sunt transmise facturi in detaliu si se propune Tert_ID-ul primei facturi din detaliu)
ContTert_ID	Int	ContTert_ID (Contul bancar al tertului; se considera doar pentru TipDoc = 2, altfel trebuie sa vina NULL. Daca este NULL se lasa ce propune obiectul, adica Contul bancar implicit al tertului)
Valuta_ID	Int	Valuta_ID (daca este NULL se considera valoarea 3 = LEI; pentru TipDoc = 12 trebuie sa aiba valoarea NULL sau 3)
Cont_ID	Int	Cont_ID (daca este NULL se lasa ce propune obiectul, adica contul de casa/banca implicita a gestiunii platii; pentru TipDoc = 1 trebuie sa vina un cont de casa si pentru TipDoc IN (2, 12) trebuie sa vina un cont de banca; contul trebuie sa aiba Valuta = Valuta_ID)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
Casier	VarChar(30)	Casier (Se considera doar pentru TipDoc = 1, altfel trebuie sa vina NULL)
TipAvans_ID	Int	TipAvans_ID (Se considera doar pentru TipPlata = 2, altfel trebuie sa vina NULL; daca este NULL se lasa ce propune obiectul, adica Tip Avans Plati din Optiuni Contari)
Suma	Decimal(19, 6)	Suma (Daca TipDoc = 12 atunci se verifica faptul ca Suma = Efect.Suma)

Comision	Decimal(19, 6)	Comision (Se considera doar pentru TipDoc IN (2, 12), altfel trebuie sa vina NULL)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
FF_ID	Int	FF_ID
Valoare	Decimal(19, 6)	Valoare

Data TipDoc = 12 atunci nu trebuie sa se transmita informatiile de detaliu (OPM-ul are detaliul identic cu Efectul).

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (DP/OPM).

## 19. DocumentTransformareStoc

Importa un document de transformare stoc primit ca si parametru ca si DTS.

Nume entitate: DocumentTransformareStoc.

UIName: G.DTS (Meniul Productie -> Document de transformare stoc).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de DTS
- <Detaliu> reprezinta detaliul de DTS; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → DTS Direct 2 → DTS Productie 3 → DTS Stornare Productie
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de DTS-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit)
Tert_ID	Int	Tert_ID (trebuie sa fie Client)
Agent_ID	Int	Agent_ID (daca este NULL se lasa ce propune obiectul)
Responsabil_ID	Int	Responsabil_ID (daca este NULL se lasa ce propune obiectul)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
ModDescarcare (0 = Denumire)	Int	TipSplit
Lot	Decimal(19, 6)	Lot

Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
PretUnitar	Decimal(19, 6)	Pret (daca Iesiri = 1 trebuie sa vina NULL pentru produsele stocabile si completat pentru cele speciale; daca Iesiri = 0 si campul este NULL se lasa ce propune obiectul; daca Iesiri = 0 si TipDoc = 1 campul trebuie sa vina NULL)
TVAP	Decimal(19, 6)	TVAP (daca Iesiri = 1 trebuie sa vina NULL; daca Iesiri = 0 si campul este NULL se lasa ce propune obiectul)
Garantie	Int	Garantie (daca Iesiri = 1 trebuie sa vina NULL; daca Iesiri = 0 si campul este NULL se lasa ce propune obiectul)
SerieLot	VarChar(25)	SerieLot (daca este NULL se lasa ce propune obiectul)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca SerieLot este NULL atunci DataExpirare trebuie sa fie NULL)
DeclaratieConformitate	VarChar(25)	DeclaratieConformitate (Daca Iesiri = 1 trebuie sa vina NULL)
Iesiri (0 = NU, 1 = DA)	Bit	Daca are valoarea 1 se considera o linie de iesiri, altfel o linie de intrari

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (DTS).

Observatii:

Detaliul trebuie sa aiba liniile cu Iesiri = 1 primele, apoi restul liniilor. Daca TipDoc = 1 sau optiunea "Sume egale la transformare stoc" (din Optiuni Generale/Validari) este setata trebuie sa vina doar o singura linie pentru intrari si PretUnitar = NULL. In detaliu pot sa vina produse normale si speciale pentru iesiri si doar produse normale pentru intrari.

## 20. Transfer

Importa un document de transfer primit ca si parametru ca si AIMT sau BTG.

Nume entitate: Transfer.

UIName: G.AIMT (Meniul Stocuri -> Aviz de tranfer).

### A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de transfer
- <Detaliu> reprezinta detaliul de transfer; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → BTG + BTGB 2 → AIMT (fara NIRT) 3 → AIMT + NIRT
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de AIMT/BTG-uri din baza de date)
CTGB_ID	Int	CTGB_ID (daca este NULL se face Transfer liber; daca este completat se face Transfer pe baza de CTGB)
InchideCTG (0 = NU, 1 = DA)	Bit	InchideCTG (daca CTGB_ID este NULL atunci campul trebuie sa fie NULL; daca CTGB_ID este completat si campul este NULL atunci se lasa ce propune obiectul)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit sau Magazin; daca se face Transfer pe baza de CTGB atunci validam ca Gestiune_ID = CTGB.Gestiune_ID)
GestiuneDestinatie_ID	Int	GestiuneLegatura_ID (trebuie sa fie o Gestiune de tip Depozit sau Magazin; daca se face Transfer pe baza de CTGB atunci validam ca GestiuneDestinatie_ID = CTGB.GestiuneSursa_ID)
Prefix.Numar.Sufix Serie	As VarChar(22)	Serie (Doar pentru TipDoc IN (2, 3). Daca este valoarea AUTO atunci se propune o serie din plaja, altfel se salveaza cu valoarea trimisa; se permite ca Seria sa fie si dintr-o plaja definita la AIM)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Delegat	VarChar(50)	Delegat (doar pentru TipDoc IN (2, 3); daca este NULL se lasa ce propune obiectul, adica Delegatul Tertului)
Transport	VarChar(30)	Transport (doar pentru TipDoc IN (2, 3); se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NrMasina	VarChar(10)	NrMasina (doar pentru TipDoc IN (2, 3); se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
SerieCI	VarChar(10)	SerieCI (doar pentru TipDoc IN (2, 3); se completeaza tot timpul daca Delegat



		este completat; altfel se ia in considerare doar daca este diferit de NULL)
NumarCI	VarChar(20)	NumarCI (doar pentru TipDoc IN (2, 3); se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
PolitieCI	VarChar(50)	PolitieCI (doar pentru TipDoc IN (2, 3); se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
DataExpeditie	DateTime	DataLivrare (doar pentru TipDoc IN (2, 3); se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica Data documentului) OraLivrare (doar pentru TipDoc IN (2, 3); se elimina componenta de Data, daca exista; daca este NULL se lasa ce propune obiectul, adica Ora curenta)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
Observatii	VarChar(500)	Observatii
Serie	VarChar(100)	Serie (poate sa fie NULL; daca este completata atunci trebuie sa avem Cantitate = 1 si SerieLot = NULL)
SerieLot	VarChar(25)	SerieLot (poate sa fie NULL)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca SerieLot este NULL atunci DataExpirare trebuie sa fie NULL)

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (AIMT/BTG).

Observatii:

In detaliu pot sa vina doar produse normale si descarcarea de face dupa Denumire. Daca se transmite informatie de lot pe cel putin una dintre linii atunci se face descarcarea fiselor corespunatoare lotului.

Daca TipDoc = 3 atunci NIRT.NumarAuto = True si NIRT.Data = AIME.Data.

## 21. Consum

Importa un document de consum primit ca si parametru ca si BCD sau DSS.

Nume entitate: Consum.

UIName: G.BCD (Meniul Stocuri -> Bon de Consum Direct).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de consum
- <Detaliu> reprezinta detaliul de consum; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → BCD Normal 2 → BCD Protocol 3 → BCD Confiscare 4 → BCD Sponsorizare 5 → BCD Bonus 6 → BCD Verificari 7 → BCD Perisabil 8 → BCD Mostre Publicitate 20 → DSS
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de BCD/DSS-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit, Magazin sau Van-sales)
Tert_ID	Int	Tert_ID
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
NumarComanda	VarChar(30)	Referinta (Doar pentru TipDoc IN (1, 2, 3, 4, 5,6, 7, 8))
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului; pe gestiunile de tip Van-sales acest camp trebuie sa fie completat si sa fie un agent de tip van-sales)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	... Stoc ID
Cantitate	Decimal(19, 6)	Cantitate
Observatii	VarChar(500)	Observatii
Serie	VarChar(100)	Serie (poate sa fie NULL; daca este completata atunci trebuie sa avem Cantitate = 1 si SerieLot = NULL)
SerieLot	VarChar(25)	SerieLot (poate sa fie NULL)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca SerieLot este NULL atunci DataExpirare trebuie sa fie NULL)
CentruCost_ID	Int	CentruCost_ID (poate sa fie NULL; daca TipDoc = 20 atunci campul trebuie sa fie NULL)

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (BCD/DSS).

Observatii:

In detaliu pot sa vina doar produse normale si descarcarea de face dupa Denumire. Daca se transmite informatie de lot pe cel putin una dintre linii atunci se face descarcarea fisele corespunzatoare lotului.

## 22. FacturaStorno

Importa o factura stornatoare (multipla) primita ca si parametru ca si FCS, cu Valuta = LEI.

Nume entitate: FacturaStorno.

UIName: G.FCSMultipla (Meniul Vanzari -> Stornari facturi -> Stornare Multipla).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de factura storno
- <Detaliu> reprezinta detaliul de factura storno; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False
		Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de FC-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit)
Prefix.Numar.Sufix Serie	As VarChar(22)	Serie (Daca este valoarea AUTO atunci se propune o serie din plaja, altfel se salveaza cu valoarea trimisa; se permite ca Seria sa fie si dintr-o plaja definita la FC)
Tert_ID	Int	Tert_ID (Trebuie sa fie Client)
ContTert_ID	Int	ContTert_ID (Contul bancar al tertului; daca este NULL se lasa ce propune obiectul, adica Contul bancar implicit al tertului)
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului)
ProcentAgent	Decimal(19, 6)	ProcentAgent (Procent Comision agent; daca este NULL se lasa ce propune obiectul, adica comisionul agentului)
EvidentiereDiscount (0 = fara, 1 = cu)	Bit	EvidentiereDiscount
BancaFacturare_ID	Int	BancaFacturare_ID (daca este NULL se lasa ce propune obiectul, adica Banca implicita la iesiri a gestiunii facturii)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)

DataScadenta	DateTime	DataScadenta (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica se tine cont de zilele de scadenta ale tertului)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
PersoanaLivrare_ID	Int	PersoanaLivrare_ID (daca este NULL se lasa ce propune obiectul, adica o persoana de livrare identica cu utilizatorul, sau cea implicita de pe gestiune)
Delegat	VarChar(50)	Delegat (daca este NULL se lasa ce propune obiectul, adica Delegatul Tertului)
Transport	VarChar(30)	Transport (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NrMasina	VarChar(10)	NrMasina (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
SerieCI	VarChar(10)	SerieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NumarCI	VarChar(20)	NumarCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
PolitieCI	VarChar(50)	PolitieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
DataExpeditie	DateTime	DataLivrare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica Data documentului)
		OraLivrare (se elimina componenta de Data, daca exista; daca este NULL se lasa ce propune obiectul, adica Ora curenta)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
GrupID	Int	GrupID
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
Pret	Decimal(19, 6)	Pret (poate sa fie NULL)
DiscountP	Decimal(19, 6)	DiscountP (poate sa fie NULL)
TVAP	Decimal(19, 6)	TVAP (poate sa fie NULL)
Observatii	VarChar(500)	Observatii

Serie	VarChar(100)	Serie (poate sa fie NULL; daca este completata atunci trebuie sa avem Cantitate = 1 si SerieLot = NULL)
SerieLot	VarChar(25)	SerieLot (poate sa fie NULL)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca SerieLot este NULL atunci DataExpirare trebuie sa fie NULL)

Criteriu de grupare: Liniile din detaliu se grupeaza dupa Stoc\_ID si Pret (da eroare daca liniile din grupare difera prin unul dintre campurile DiscountP sau TVAP).

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (FCS).

Observatii:

(\*) Transmiterea discountului la nivel de detaliu depinde de modul de evidentiare al discountului EvidentiazaDiscount

= 0 → se transmite discountul inclus în pret (pret e pretul discountat)

= 1 → se transmite discountul evidentiat (pret e pretul fara discountul evidentiat aplicat)

Ambele variante preiau pretul si discountul si il inregistreaza nemodificat (deci fara sa-l mai discounteze).

GrupID trebuie sa aiba valoarea NULL pentru ambele variante de evidentiare.

Nu se verifica numarul de linii.

## 23. Aviz

Importa un aviz/bon fiscal depozit primit ca si parametru ca si AIMC/AIMC din FP/AIML (cu Valuta=LEI) sau AIM din FP/AIM din FCA (cu orice Valuta) sau BFD/BFD din FPM (cu Valuta = LEI).

Nume entitate: Aviz.

UIName: G.AIM (Meniul Stocuri -> Aviz de insotire a marfii).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de aviz
- <Detaliu> reprezinta detaliul de aviz; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → AIMC 2 → AIM (din FP) 3 → AIMC (din FP) 4 → AIM (din FCA), AIMM (din FCAM) 6 → AIML Lucrari 7 → AIML Configurari 21 → BFD 22 → BFD (din FPM)
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de AIM/BFD-uri din baza de date)
FP_ID	Int	FP_ID (daca TipDoc IN (1, 4, 6, 7, 21))

		atunci campul trebuie sa fie NULL)
InchideFP (0 = NU, 1 = DA)	Bit	InchideFP (daca TipDoc IN (1, 4, 6, 7, 21) atunci campul trebuie sa fie NULL; daca TipDoc IN (2, 3, 22) si campul este NULL atunci se lasa ce propune obiectul, adica optiunea pe firma "Inchidere FP dupa avizare")
FC_ID	Int	FC_ID (daca TipDoc IN (1, 2, 3, 6, 7, 21, 22) atunci campul trebuie sa fie NULL)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit; pentru TipDoc IN (2, 3, 22) validam ca Gestiune_ID = FP.Gestiune_ID; pentru TipDoc = 4 validam ca Gestiune_ID = FCA.Gestiune_ID; pentru TipDoc IN (21, 22) validam ca Gestiune_ID = CasaMarcat.Gestiune_ID)
Prefix.Numar.Suffix Serie	As VarChar(22)	Serie (Daca este valoarea AUTO atunci se propune o serie din plaja, altfel se salveaza cu valoarea trimisa; se permite ca Seria sa fie si dintr-o plaja definita la AIM; daca TipDoc IN (21, 22) atunci campul trebuie sa fie NULL)
CasaMarcat_ID	Int	CasaMarcat_ID (daca TipDoc IN (1, 2, 3, 4, 6, 7) atunci campul trebuie sa fie NULL)
Casier_ID	Int	Casier_ID (Agent de tip Casier; daca TipDoc IN (1, 2, 4, 3, 6, 7) atunci campul trebuie sa fie NULL)
Tert_ID	Int	Tert_ID (Trebuie sa fie Client; pentru TipDoc IN (2, 3, 22) validam ca Tert.TertMaster_ID = FP.Tert.TertMaster_ID daca campul este completat; pentru TipDoc = 4 validam ca Tert.TertMaster_ID = FCA.Tert.TertMaster_ID daca campul este completat)
ContTert_ID	Int	ContTert_ID (Contul bancar al tertului; daca TipDoc IN (1, 6, 7) si campul este NULL se lasa ce propune obiectul, adica Contul bancar implicit al tertului; daca TipDoc IN (2, 3, 4, 21, 22) atunci campul trebuie sa fie NULL)
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca TipDoc IN (1, 6, 7, 21) si campul este NULL se lasa ce propune obiectul, adica Agentul Tertului; daca TipDoc IN (2, 3, 4, 22) atunci campul trebuie sa fie NULL)
ProcentAgent	Decimal(19, 6)	ProcentAgent (Procent Comision agent; daca TipDoc IN (1, 3, 6, 7, 21, 22) atunci campul trebuie sa fie NULL; daca TipDoc = 2 si campul este NULL se lasa ce propune obiectul, adica FP.ProcentAgent; daca TipDoc = 4 si campul este NULL se lasa ce propune obiectul, adica FCA.ProcentAgent)

CodFiscal	VarChar(30)	CodFiscal (daca TipDoc IN (1, 2, 3, 4, 6, 7) atunci campul trebuie sa fie NULL)
EvidentiereDiscount (0 = fara, 1 = cu)	Bit	EvidentiereDiscount (daca TipDoc IN (2, 3, 4, 6, 7, 22) atunci campul trebuie sa fie NULL)
BancaFacturare_ID	Int	BancaFacturare_ID (daca TipDoc IN (1, 6, 7, 21) si campul este NULL se lasa ce propune obiectul, adica Banca implicita la iesiri a gestiunii avizului; daca TipDoc IN (2, 3, 4, 22) atunci campul trebuie sa fie NULL)
Data	DateTime	Data (se elimina componenta de Timp, daca exista) Ora (se elimina componenta de Data; ora la care s-a listat bonul fiscal pe casa de marcat pentru TipDoc IN (21, 22))
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca TipDoc IN (1, 3) si campul este NULL se lasa ce propune obiectul; daca TipDoc IN (2, 4, 6, 7, 21, 22) atunci campul trebuie sa fie NULL)
RecalculeazaPretUnitar (0 = NU, 1 = DA)	Bit	RecalculeazaPretUnitar (daca este NULL se considera valoarea 0 = NU; daca TipDoc IN (2, 3, 4, 6, 7, 21, 22) atunci campul trebuie sa fie NULL)
TotalCHS	Decimal(19, 6)	TotalCHS – Numerar (daca TipDoc IN (1, 2, 3, 4, 6, 7) atunci campul trebuie sa fie NULL; daca TipDoc IN (21, 22) atunci campurile TotalCHS, TotalTV si TotalCI pot fi toate completate sau toate NULL si se considera incasare integrala cu numerar)
TotalTV	Decimal(19, 6)	TotalTV - Tichet Valoric (daca TipDoc IN (1, 2, 3, 4, 6, 7) atunci campul trebuie sa fie NULL; daca TipDoc IN (21, 22) atunci campurile TotalCHS, TotalTV si TotalCI pot fi toate completate sau toate NULL si se considera incasare integrala cu numerar)
TotalCI	Decimal(19, 6)	TotalCI - Card (daca TipDoc IN (1, 2, 3, 4, 6, 7) atunci campul trebuie sa fie NULL; daca TipDoc IN (21, 22) atunci campurile TotalCHS, TotalTV si TotalCI pot fi toate completate sau toate NULL si se considera incasare integrala cu numerar)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Delegat	VarChar(50)	Delegat (daca este NULL se lasa ce propune obiectul, adica Delegatul Tertului)
Transport	VarChar(30)	Transport (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NrMasina	VarChar(10)	NrMasina (se completeaza tot timpul

		daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
SerieCI	VarChar(10)	SerieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NumarCI	VarChar(20)	NumarCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
PolitieCI	VarChar(50)	PolitieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
DataExpeditie	DateTime	DataLivrare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica Data documentului)
		OraLivrare (se elimina componenta de Data, daca exista; daca este NULL se lasa ce propune obiectul, adica Ora curenta)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
GrupID	Int	GrupID (daca TipDoc IN (2, 3, 4, 6, 7, 22) atunci campul trebuie sa fie NULL)
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
Pret	Decimal(19, 6)	Pret (daca TipDoc IN (2, 3, 4, 22) atunci campul trebuie sa fie NULL; daca TipDoc = 21 atunci campul are sensul de PretAmanunt)
PretValuta	Decimal(19, 6)	PretValuta (daca TipDoc IN (2, 3, 4, 21, 22) atunci campul trebuie sa fie NULL)
Valuta_ID	Int	Valuta_ID (daca TipDoc IN (2, 3, 4, 21, 22) atunci campul trebuie sa fie NULL)
DiscountP	Decimal(19, 6)	DiscountP (daca TipDoc IN (1, 6, 7, 21) si campul este NULL se lasa ce propune obiectul; daca TipDoc IN (2, 3, 4, 22) atunci campul trebuie sa fie NULL)
TVAP	Decimal(19, 6)	TVAP (daca TipDoc IN (1, 6, 7, 21) si campul este NULL se lasa ce propune obiectul; daca TipDoc IN (2, 3, 4, 22) atunci campul trebuie sa fie NULL)
Garantie	Int	Garantie (Se ia pentru toate liniile mai putin pentru linia de discount de grup; daca TipDoc IN (1, 6, 7, 21) si campul este NULL se lasa ce propune obiectul; daca TipDoc IN (2, 3, 4, 22) atunci campul trebuie sa fie NULL)
Observatii	VarChar(500)	Observatii (daca TipDoc IN (2, 3, 4, 22) si campul este NULL se lasa ce propune obiectul, adica FPItem.Observatii)



Serie	VarChar(100)	Serie (poate sa fie NULL; daca este completata atunci trebuie sa avem Cantitate = 1 si SerieLot = NULL)
SerieLot	VarChar(25)	SerieLot (poate sa fie NULL)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca SerieLot este NULL atunci DataExpirare trebuie sa fie NULL)

Criteriu de grupare: Liniile din detaliu care au completat campul SerieLot se grupeaza dupa Stoc\_ID si Pret (da eroare daca liniile din grupare difera prin unul dintre campurile GrupID, DiscountP, TVAP, Garantie sau Observatii).

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (AIMC/AIM/AIML/BFD).

Observatii:

(\*) Transmiterea discountului la nivel de detaliu depinde de modul de evidentiare al discountului  
EvidentiazaDiscount

= 0 → se transmite discountul inclus în pret (pret e pretul discountat)

= 1 → se transmite discountul evidentiat (pret e pretul fara discountul evidentiat aplicat)

Ambele variante preiau pretul si discountul si il inregistreaza nemodificat (deci fara sa-l mai discounteze).

Pentru varianta cu evidentiare putem avea mai multe grupuri si mai multe linii libere. Un grup se identifica prin acelasi GrupID  $\neq 0$  si contine una sau mai multe linii de produse normale si o singura linie de discount de grup, care trebuie sa fie ultima linie in grup, cu Cantitate = 1 si Pret = suma discounturilor de pe liniile cu produse normale. O linie libera are GrupID = 0 si poate contine un produs de tip discount sau un produs special. Atentie: In cadrul unui Grup toate produsele (inclusiv cel de discount) trebuie sa aiba acelasi TVAP. In caz contrar se da eroare.

Pentru varianta fara evidentiare putem avea numai linii libere. O linie libera are GrupID = 0 si poate contine orice tip de produs.

In functie de valorile lui Pret si PretValuta avem urmatoarele variante de utilizare:

- Pivotal este Pret: Pret  $\neq$  NULL si PretValuta = orice; daca PretValuta  $\neq$  NULL si Valuta\_ID  $\neq$  NULL se pastreaza acest PretValuta, altfel PretValuta = Pret / Cursul bancii de facturare
- Pivotal este PretValuta: PretValuta  $\neq$  NULL si Pret = NULL; Pret se calculeaza din PretValuta la cursul bancii de facturare
- Pivotal este Nomenclatorul: Pret = NULL, PretValuta = NULL, Valuta\_ID = NULL si DiscountP = orice; se aduc din nomenclator valorile pentru PretValuta, Valuta\_ID si DiscountP daca trebuie; Pret se calculeaza din PretValuta la cursul bancii de facturare

Pentru linia de Discount de Grup trebuie sa vina toate campurile dar se copiaza doar Stoc\_ID-ul si se verifica GrupID-ul sa fie corect.

Nu se verifica numarul de linii si creditul tertului.

Daca TipDoc IN (21, 22) BFD-ul va avea Sursa Import. Daca campurile TotalCHS, TotalTV si TotalCI sunt completate atunci Tipariri = 1 (si nu se va putea sterge), altfel Tipariri = 0.

## 24. AvizIncarcare

Importa un aviz de incarcare primit ca si parametru ca si AIMI Normal (cu Valuta=LEI).

Nume entitate: AvizIncarcare.

UIName: G.AIMI (Meniul Stocuri -> Aviz de Incarcare / Livrare).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de aviz
- <Detaliu> reprezinta detaliul de aviz; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False
		Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de AIM-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit sau Van-sales)
Prefix.Numar.Suffix Serie	As VarChar(22)	Serie (Daca este valoarea AUTO atunci se propune o serie din plaja, altfel se salveaza cu valoarea trimisa; se permite ca Seria sa fie si dintr-o plaja definita la AIM)
Ruta_ID	Int	Ruta_ID (poate sa fie NULL)
Agent_ID	Int	Agent_ID (trebuie sa fie de Tip Livrator; daca este NULL se lasa ce propune obiectul, adica Agentul Rutei)
ActualizeazaExpeditie (0 = NU, 1 = DA)	Bit	ActualizeazaLivrare (daca este NULL atunci se lasa ce propune obiectul, adica optiunea pe utilizator "Actualizare livrare dupa aviz incarcare")
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Delegat	VarChar(50)	Delegat (daca este NULL se lasa ce propune obiectul, adica Delegatul Tertului)
Transport	VarChar(30)	Transport (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NrMasina	VarChar(10)	NrMasina (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
SerieCI	VarChar(10)	SerieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
NumarCI	VarChar(20)	NumarCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
PolitieCI	VarChar(50)	PolitieCI (se completeaza tot timpul daca Delegat este completat; altfel se ia in considerare doar daca este diferit de NULL)
DataExpeditie	DateTime	DataLivrare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica Data documentului)

		OraLivrare (se elimina componenta de Data, daca exista; daca este NULL se lasa ce propune obiectul, adica Ora curenta)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
AIM_ID	Int	AIM_ID

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

#### A. OUT

Se returneaza ID-ul de Document (AIMI).

Observatii:

Nu se verifica numarul de linii.

## 25. ReturCustodie

Importa un document de retur din custodie (multiplu) primita ca si parametru ca si DRC.

Nume entitate: ReturCustodie.

UIName: G.DRCMultiplu (Meniul Aprovizionare -> Document de Retur din Custodie Multiplu).

#### A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de retur custodie
- <Detaliu> reprezinta detaliul de retur custodie; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de DRC-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit)
DocumentInsotitor	VarChar(30)	Referinta
Tert_ID	Int	Tert_ID (Trebuie sa fie Client)
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului)
EvidentiereDiscount (0 = fara, 1 = cu)	Bit	EvidentiereDiscount
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
VineMarfa (0 = NU, 1 = DA)	Bit	VineMarfa
RecalculeazaPretUnitar (0 = NU, 1 = DA)	Bit	RecalculeazaPretUnitar (daca este NULL se considera valoarea 0 = NU)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa

		ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
GrupID	Int	GrupID
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
Pret	Decimal(19, 6)	Pret (poate sa fie NULL)
DiscountP	Decimal(19, 6)	DiscountP (poate sa fie NULL)
TVAP	Decimal(19, 6)	TVAP (poate sa fie NULL)
Observatii	VarChar(500)	Observatii
Serie	VarChar(100)	Serie (poate sa fie NULL; daca este completata atunci trebuie sa avem Cantitate = 1 si SerieLot = NULL)
SerieLot	VarChar(25)	SerieLot (poate sa fie NULL)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca SerieLot este NULL atunci DataExpirare trebuie sa fie NULL)

Criteriu de grupare: Liniile din detaliu se grupeaza dupa Stoc\_ID si Pret (da eroare daca liniile din grupare difera prin unul dintre campurile DiscountP sau TVAP).

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

#### A. OUT

Se returneaza ID-ul de Document (DRC).

Observatii:

(\*) Transmiterea discountului la nivel de detaliu depinde de modul de evidentiare al discountului  
EvidentiazaDiscount

= 0 → se transmite discountul inclus în pret (pret e pretul discountat)

= 1 → se transmite discountul evidentiat (pret e pretul fara discountul evidentiat aplicat)

Ambele variante preiau pretul si discountul si il inregistreaza nemodificat (deci fara sa-l mai discounteze).

GrupID trebuie sa aiba valoarea NULL pentru ambele variante de evidentiare.

## 26. ComandaFurnizor

Importa o comanda primita ca si parametru ca si CF sau CTG + CTGB.

Nume entitate: ComandaFurnizor.

UIName: G.CF (Meniul Comenzi -> Comanda Furnizor).

#### A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de comanda
- <Detaliu> reprezinta detaliul de comanda; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → CF 2 → CTG + CTGB

Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de CF/CTG-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit sau Magazin)
GestiuneDestinatie_ID	Int	GestiuneLegatura_ID (doar pentru TipDoc = 2; trebuie sa fie o Gestiune de tip Depozit sau Magazin)
Serie	VarChar(30)	Referinta
Tert_ID	Int	Tert_ID (doar pentru TipDoc = 1)
BancaFacturare_ID	Int	BancaFacturare_ID (doar pentru TipDoc = 1; daca este NULL se lasa ce propune obiectul, adica Banca implicita la iesiri a gestiunii facturii)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
Agent_ID	Int	Agent_ID (doar pentru TipDoc = 2; poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
CantitateConfirmata	Decimal(19, 6)	CantitateConfirmata (trebuie sa fie <= Cantitate; daca este NULL se lasa ce propune obiectul)
DataLivrare	DateTime	DataLivrare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul)
Pret	Decimal(19, 6)	Pret (doar pentru TipDoc = 1)
PretValuta	Decimal(19, 6)	PretValuta (doar pentru TipDoc = 1)
Valuta_ID	Int	Valuta_ID (doar pentru TipDoc = 1)
DiscountP	Decimal(19, 6)	DiscountP (doar pentru TipDoc = 1; daca este NULL se lasa ce propune obiectul)
TVAP	Decimal(19, 6)	TVAP (doar pentru TipDoc = 1; daca este NULL se lasa ce propune obiectul)
Garantie	Int	Garantie (daca este NULL se lasa ce propune obiectul)
TipOperatieStoc	Int	1 = Rezerva tot 2 = Comanda tot 3 = Confirma cat poti, restul rezerva 4 = Confirma cat poti, restul comanda 5 = Confirma tot (daca nu se poate bloca toata catitatea se da eroare) (doar pentru TipDoc = 2)
Observatii	VarChar(500)	Observatii

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (CF/CTG), Suma si TVA.

Observatii:

(\*) Transmiterea discountului la nivel de detaliu: se transmite discountul inclus în pret (pret e pretul discountat); se preia pretul si discountul si il inregistreaza nemodificat (deci fara sa-l mai discounteze).

O linie din detaliu poate contine doar produse stocabile.

In functie de valorile lui Pret si PretValuta avem urmatoarele variante de utilizare:

- Pivotal este Pret: Pret  $\neq$  NULL si PretValuta = orice; daca PretValuta  $\neq$  NULL si Valuta\_ID  $\neq$  NULL se pastreaza acest PretValuta, altfel PretValuta = Pret / Cursul bancii de facturare
- Pivotal este PretValuta: PretValuta  $\neq$  NULL si Pret = NULL; Pret se calculeaza din PretValuta la cursul bancii de facturare
- Pivotal este Nomenclatorul: Pret = NULL, PretValuta = NULL, Valuta\_ID = NULL si DiscountP = orice; se aduc din Info aprovizionare valorile pentru PretValuta, Valuta\_ID si DiscountP daca trebuie; Pret se calculeaza din PretValuta la cursul bancii de facturare

## 27. FacturaFurnizor

Importa o factura/aviz furnizor primita ca si parametru ca si FF/AF in orice Valuta.

Nume entitate: FacturaFurnizor.

UIName: G.FF (Meniul Aprovizionare -> Factura Furnizor).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de factura furnizor
- <Detaliu> reprezinta detaliul de factura furnizor; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → FF (normal) 10 → AF (normal)
Numar	Int	Daca = NULL → NumarAuto = True Daca $\neq$ NULL → NumarAuto = False Numar (Daca $\neq$ NULL) (Se verifica unicitatea intre numerele de FF/AF-uri din baza de date)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit sau Magazin)
Serie	VarChar(22)	Referinta
Tert_ID	Int	Tert_ID (Trebuie sa fie Furnizor)
Valuta_ID	Int	Valuta_ID (daca este NULL se considera valoarea 3 = LEI)

TVA_ID	Int	TVA_ID (permite alegerea Regimului de TVA; daca este NULL se lasa ce propune obiectul; daca TipDoc = 10 atunci campul trebuie sa aiba valorile NULL, 0 sau 11) 0 → Taxabila (pentru TipDoc = 10 semnifica Taxabila - conform facturii) 35 → Taxabila – TVA la incasare 7 → Scutita (forma de 0) 11 → Taxabila DVI - import (forma de 0) 2 → Taxabila - cota TVA externa etc.
EvidentiereDiscount (0 = fara, 1 = cu)	Bit	EvidentiereDiscount
BancaFacturare_ID	Int	BancaFacturare_ID (daca este NULL se lasa ce propune obiectul, adica Banca implicita la intrari a gestiunii facturii/avizului furnizor)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
DataScadenta	DateTime	DataScadenta (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul, adica se tine cont de zilele de scadenta ale tertului; daca TipDoc = 10 atunci campul trebuie sa fie NULL)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)
Generat	Int	Generat (daca este NULL se lasa ce propune obiectul adica valoarea 0; daca nu este NULL trebuie sa aiba valoarea 0 sau valoarea 3) 0 → face note contabile 3 → nu face note contabile

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
GrupID	Int	GrupID
NomGeneral_ID	Int	... Stoc_ID
InCost (0 = NU, 1 = DA)	Bit	InCost (daca este NULL se lasa ce propune obiectul, in functie de tipul de produs din detaliu)
Cantitate	Decimal(19, 6)	Cantitate
Pret	Decimal(19, 6)	Pret (daca FF/AF.Valuta <> LEI atunci campul trebuie sa fie NULL)
PretValuta	Decimal(19, 6)	PretValuta
Valuta_ID	Int	Valuta_ID (daca FF/AF.Valuta <> LEI atunci campul trebuie sa aiba valorile NULL sau FF/AF.Valuta)
DiscountP	Decimal(19, 6)	DiscountP

TVAP	Decimal(19, 6)	TVAP (daca este NULL se lasa ce propune obiectul; daca TVA_ID din antet corespunde unei forme de 0 atunci campul trebuie sa fie cu valoarea NULL sau 0)
Garantie	Int	Garantie (Se ia pentru toate liniile mai putin pentru linia de discount de grup; daca este NULL se lasa ce propune obiectul)
Observatii	VarChar(500)	Observatii
SerieLot	VarChar(25)	SerieLot
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista)
DeclaratieConformitate	VarChar(25)	DeclaratieConformitate
CentruCost_ID	Int	CentruCost_ID (poate sa fie NULL)
CF_ID	Int	CF_ID (poate sa fie NULL; daca Cantitate < 0 atunci trebuie sa fie NULL)

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Document (FF/AF).

Observatii:

(\*) Transmiterea discountului la nivel de detaliu depinde de modul de evidentiare al discountului

EvidentiazaDiscount

= 0 → se transmite discountul inclus în pret (pret e pretul discountat)

= 1 → se transmite discountul evidentiat (pret e pretul fara discountul evidentiat aplicat)

Ambele variante preiau pretul si discountul si il inregistreaza nemodificat (deci fara sa-l mai discounteze).

Pentru varianta cu evidentiare putem avea mai multe grupuri si mai multe linii libere. Un grup se identifica prin acelasi GrupID  $\neq$  0 si contine una sau mai multe linii de produse normale si o singura linie de discount de grup, care trebuie sa fie ultima linie in grup, cu Cantitate = 1 si Pret = suma discounturilor de pe liniile cu produse normale. O linie libera are GrupID = 0 si poate contine un produs de tip discount sau un produs special. Atentie: In cadrul unui Grup toate produsele (inclusiv cel de discount) trebuie sa aiba acelasi TVAP. In caz contrar se da eroare.

Pentru varianta fara evidentiare putem avea numai linii libere. O linie libera are GrupID = 0 si poate contine orice tip de produs.

Daca FF/AF.Valuta = LEI, in functie de valorile lui Pret si PretValuta avem urmatoarele variante de utilizare:

- Pivotal este Pret: Pret  $\neq$  NULL si PretValuta = orice; daca PretValuta  $\neq$  NULL si Valuta\_ID  $\neq$  NULL se pastreaza acest PretValuta, altfel PretValuta = Pret / Cursul bancii de facturare

Daca FF/AF.Valuta  $\neq$  LEI, in functie de valorile lui PretValuta avem urmatoarele variante de utilizare:

- Pivotal este PretValuta: PretValuta  $\neq$  NULL; Pret se calculeaza din PretValuta la cursul BNR

Pentru linia de Discount de Grup trebuie sa vina toate campurile dar se copiaza doar Stoc\_ID-ul si se verifica GrupID-ul sa fie corect.

## 28. Intrare

Importa un document de intrare primit ca si parametru ca si DIS/NIR.

Nume entitate: Intrare.

UIName: G.DISD (Meniul Stocuri -> Document Introducere in Stoc).

## A. IN

In *pstrDateAsXML* apar sectiunile:



- <Antet> reprezinta un antet de intrare
- <Detaliu> reprezinta detaliul de intrare; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
TipDoc	Int	1 → DIS 2 → NIR (Uz intern)
Numar	Int	Daca = NULL → NumarAuto = True Daca <> NULL → NumarAuto = False Numar (Daca <> NULL) (Se verifica unicitatea intre numerele de DIS/NIR-uri din baza de date)
FF_ID	Int	FF_ID (daca TipDoc = 1 atunci campul trebuie sa fie NULL)
Gestiune_ID	Int	Gestiune_ID (trebuie sa fie o Gestiune de tip Depozit, Magazin si daca TipDoc = 1 poate sa fie si Van-sales; daca TipDoc = 2 atunci validam ca Gestiune_ID = FF.Gestiune_ID)
Tert_ID	Int	Tert_ID (daca TipDoc = 2 atunci validam ca Tert.TertMaster_ID = FF.Tert.TertMaster_ID daca campul este completat)
Data	DateTime	Data (se elimina componenta de Timp, daca exista)
Agent_ID	Int	Agent_ID (poate sa fie de orice Tip; daca este NULL se lasa ce propune obiectul, adica Agentul Tertului; pe gestiunile de tip Van-sales acest camp trebuie sa fie completat si sa fie un agent de tip van-sales; daca TipDoc = 2 campul trebuie sa fie NULL)
Observatii	VarChar(1024)	Observatii (daca este DEFAULT se lasa ce propune obiectul)
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra documentul (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
NomGeneral_ID	Int	... Stoc_ID
Cantitate	Decimal(19, 6)	Cantitate
Pret	Decimal(19, 6)	Pret
PretValuta	Decimal(19, 6)	PretValuta (pe Gestiuni de tip Magazin trebuie sa fie NULL sau egal cu Pret)
Valuta_ID	Int	Valuta_ID (pe Gestiuni de tip Magazin trebuie sa fie NULL sau 3, adica LEI)
DiscountP	Decimal(19, 6)	DiscountP (daca este NULL se lasa ce propune obiectul)
TVAP	Decimal(19, 6)	TVAP (daca este NULL se lasa ce propune obiectul)
Acciza	Decimal(19, 6)	Acciza (se considera doar pentru produsele accizabile)

Garantie	Int	Garantie (daca este NULL se lasa ce propune obiectul)
Observatii	VarChar(500)	Observatii
SerieLot	VarChar(25)	SerieLot (daca este NULL se lasa ce propune obiectul)
DataExpirare	DateTime	DataExpirare (se elimina componenta de Timp, daca exista; daca este NULL se lasa ce propune obiectul)
DeclaratieConformitate	VarChar(25)	DeclaratieConformitate

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

Data TipDoc = 2 atunci nu trebuie sa se transmita informatiile de detaliu (NIR are detaliul identic cu FF).

## B. OUT

Se returneaza ID-ul de Document (DIS/NIR).

Observatii:

Pretul si discountul se preiau si se inregistreaza nemodificate (deci fara sa mai discounteze pretul).

In functie de valorile lui Pret si PretValuta avem urmatoarele variante de utilizare:

- Pivotal este Pret: Pret  $\diamond$  NULL si PretValuta = orice; daca PretValuta  $\diamond$  NULL si Valuta\_ID  $\diamond$  NULL se pastreaza acest PretValuta, altfel PretValuta = Pret / Cursul bancii de facturare

## 29. NotaContabila

Importa o nota contabila primita ca si parametru ca si NC.

Nume entitate: NotaContabila.

UIName: C.NC (Meniul Operatii -> Note Contabile).

## A. IN

In *pstrDateAsXML* apar sectiunile:

- <Antet> reprezinta un antet de nota contabila
- <Detaliu> reprezinta detaliul de nota contabila; in aceasta sectiune apar oricate sectiuni de tip <Linie>

Poate sa exista o singura sectiune <Antet> si aceasta trebuie sa fie prima. Dupa aceasta poate sa existe o singura sectiune <Detaliu>.

Campuri pentru Antet:

Camp	Tip	Camp ERP
Generare_ID	Int	Generare_ID (se permit doar surse de import definite de utilizator)
Valuta_ID	Int	Valuta_ID
TipNC_ID	Int	TipNC_ID (de ex. 4 = Diverse)
Explicatii	VarChar(255)	Explicatii
Utilizator	VarChar(50)	Utilizatorul cu care se va inregistra nota contabila (Trebuie sa fie un utilizator definit in aplicatie)

Campuri pentru Detaliu:

Camp	Tip	Camp ERP
NrIC	Int	NrIC
TipCompunere	Int	TipCompunere (note contabile) NULL → % in Credit sau % in Debit sau simple 21 → % in Debit sau simple
TipDoc	VarChar(5)	TipDoc

NumarDoc	VarChar(30)	NumarDoc
DataIC	DateTime	DataIC
DataDoc	DateTime	DataDoc
Debit_ID	Int	Debit_ID
Credit_ID	Int	Credit_ID
TertDebit_ID	Int	TertDebit_ID (Trebuie sa fie NULL daca Debit_ID nu este un cont de Tert)
TertCredit_ID	Int	TertCredit_ID (Trebuie sa fie NULL daca Credit_ID nu este un cont de Tert)
AgentDebit_ID	Int	AgentDebit_ID (Trebuie sa fie NULL daca Debit_ID nu este un cont de Agent)
AgentCredit_ID	Int	AgentCredit_ID (Trebuie sa fie NULL daca Credit_ID nu este un cont de Agent)
Suma	Decimal(19, 6)	Suma
SumaValuta	Decimal(19, 6)	Suma_V
Curs	Decimal(19, 6)	Curs
Explicatii	VarChar(255)	Explicatii
TVA_ID	Int	TVA_ID
TipJurnal	Int	TipJC 0 → JCD 6 → JCD50 2 → JFD 3 → JPR 4 → JCS 5 → JVS
CentruCost_ID	Int	CentruCost_ID (poate sa fie NULL)

Criteriu de ordonare: Datele se importa in ordinea in care sunt in fisier, nu se schimba ordinea.

## B. OUT

Se returneaza ID-ul de Nota contabila (NC).

Observatii:

In cadrul unei Note Contabile (NC) se pot importa mai multe inregistrari contabile simple sau compuse (IC-uri identificate prin acelasi NrIC). In cadrul unei IC compus campurile TipDoc, NumarDoc, DataIC si DataDoc trebuie sa aiba aceleasi valori. Campul NrIC este un numar relativ in cadrul unei NC si trebuie sa aiba valori crescatoare pentru IC-uri diferite.

Campul DataIC trebuie sa fie in aceeasi luna pentru toate liniile din detaliu.