ABRA Software s.r.o.
Jeremiášova 1422/7b
155 00 Praha 13

IČ 25097563
DIČ: CZ2597563
Zaps. v OR u Městského soudu
v Praze, odd. C, vložka 451504

DEFINOVATELNÉ IMPORTY

METODICKÁ PŘÍRUČKA

Informace pro uživatele o nastavení a použití definovatelných importů v systému ABRA FLORES®

Aktualizace dokumentace
14. října 2026

OBSAH

1. ZÁKLADNÍ INFORMACE 3
   1.1 NA CO SE DEFINOVATELNÝ IMPORT VYUŽÍVÁ 3
   1.2 TYPICKÉ SCÉNÁŘE VYUŽITÍ 3
   1.3 ZÁKLADNÍ DIAGRAM PROCESU IMPORTU 4
2. DEFINOVATELNÉ IMPORTY V IS FLORES 5
   2.1 LICENCE A POTŘEBNÁ NASTAVENÍ 5
   2.2 LOGOVÁNÍ DEFINOVATELNÝCH IMPORTŮ 5
   2.3 KÓDOVÁNÍ 5
   2.4 MAXIMÁLNÍ DOPORUČENÝ POČET ZÁZNAMŮ V JEDNOM IMPORTU 6
3. AGENDY DEFINOVATELNÉHO IMPORTU 7
   3.1 NOVÝ MODUL „IMPORTY A EXPORTY“ 7
   3.2 AGENDA ZDROJE DAT DEFINOVATELNÝCH IMPORTŮ 7
   3.3 AGENDA DEFINICE DEFINOVATELNÝCH IMPORTŮ 9
      3.3.1 HLAVIČKA DEFINICE DEFINOVATELNÝCH IMPORTŮ 9
      3.3.2 OBSAH DEFINICE DEFINOVATELNÉHO IMPORTU 13
      3.3.3 PRŮVODCE VYTVOŘENÍM DEFINIC DEFINOVATELNÝCH IMPORTŮ 24
      3.3.4 NAČTENÍ IMPORTNÍHO SOUBORU 25
      3.3.5 MOŽNOSTI A PROCES ZPRACOVÁNÍ SOUBORU 25
4. AGENDA DOKUMENTY 29
   4.1 NOVÁ KATEGORIE DOKUMENTŮ: „DEFINOVATELNÝ IMPORT“ 29
   4.2 AGENDA DOKUMENTY PRO DEFINOVATELNÉ IMPORTY 29
      4.2.1 DOPLNĚNÍ HODNOT PRO IMPORT 29
      4.2.2 OPAKOVANÉ ZPRACOVÁNÍ A PŘEGENEROVÁNÍ 30
5. PODPORA SKRIPTINGU 30
6. PODROBNÝ POPIS JEDNOTLIVÝCH TYPŮ DEFINOVATELNÝCH IMPORTŮ S PŘÍKLADY 33
   6.1 OBECNÉ PRINCIPY 33
   6.2 TYP IMPORTU CSV A TXT 33
      6.2.1 PŘÍKLADY: 33
   6.3 TYP IMPORTU XML 49
      6.3.1 PŘÍKLADY 49
   6.4 TYP IMPORTU XLS 56
      6.4.1 PŘÍKLADY: 56

1. ZÁKLADNÍ INFORMACE

1.1 NA CO SE DEFINOVATELNÝ IMPORT VYUŽÍVÁ

Nový modul „Definovatelné importy" je určen pro zakládání libovolných nových business objektů (BO) IS FLORES nebo aktualizace stávajících.

Vstupem pro import je externí soubor nebo tok dat (stream). Definovatelné importy podporují následující formáty importních dat:

- text poziční (TXT)
- text s oddělovači (CSV)
- formát XML
- Excel (XLSX)

Jednotlivé formáty importních dat jsou podrobněji popsány v dalších kapitolách.

Výsledkem zpracování souboru je založení nového nebo aktualizace (oprava) existujícího business objektu IS FLORES.

Novou funkcionalitu definovatelných importů můžou využívat všichni uživatelé IS FLORES. Modul definovatelných importů je samostatně licencován. Operace s business objekty zohledňují nastavení práv systému IS FLORES.

Nastavení a nadefinování definovatelných importů od uživatele vyžaduje znalosti struktury business objektů systému IS FLORES. Proto se předpokládá, že nastavování budou provádět nejčastěji konzultanti nebo technicky pokročilí uživatelé.
Pro snadnější orientaci ve strukturách Business Objektů lze využít nápovědu „GenDoc.chm". (umístěnou v instalačním adresáři s IS FLORES nebo dostupnou přes agendu Nástroje přizpůsobení, záložka Dokumentace).

Modul definovatelných importů nevyužívá import manager.

1.2 TYPICKÉ SCÉNÁŘE VYUŽITÍ

Příklad 1:

Typickým scénářem použití definovatelného importu je migrace dat zákazníka z jiného informačního systému do IS FLORES (většinou se jedná o jednorázový proces).

Pro tento případ se nejdříve provede export dat z cizího systému do souboru ve formátu, který má svojí strukturou předpoklady pro strojové zpracování.
Pro tyto data se v IS FLORES připraví definice pro definovatelné importy.
Před samotným zápisem do business objektů systému IS FLORES má uživatel možnost data zkontrolovat a vyřešit případné chyby.
Na závěr se ověřená data perzistentně uloží do systému IS FLORES.

Příklad 2:

Dalším typickým scénářem použití definovatelného importu je opakovaný import dat ze souborů s daty ve stále stejném formátu (jedná se o opakovatelné řešení výměny dat, která může probíhat i mnohokrát za den).
Typicky se jedná o strojovou výměnu dat mezi cizím informačním systémem a systémem IS FLORES.

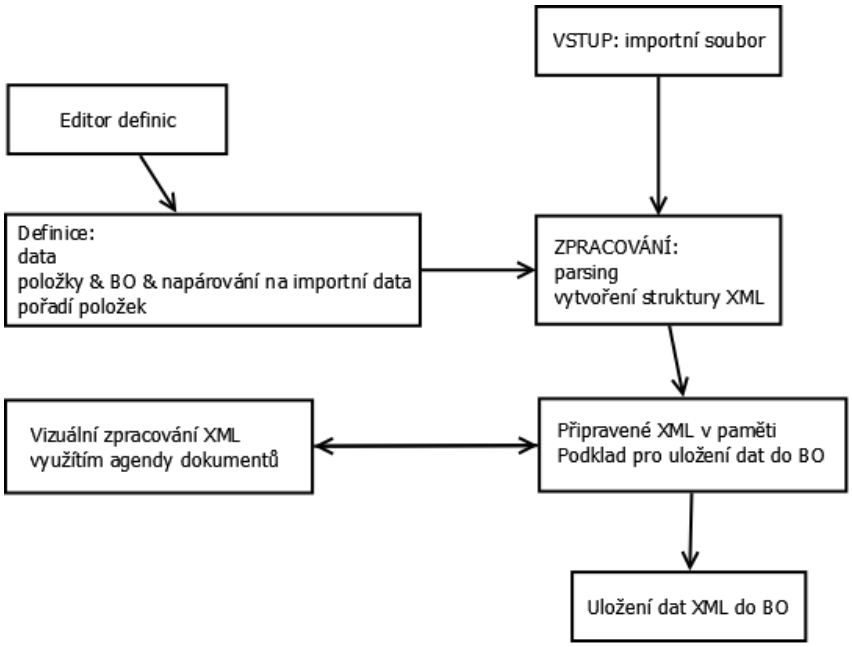
Nejdříve se nastaví definice pro definovatelný import podle struktury importního souboru.
V testovacím provozu je ideální využít možnost uživatelské kontroly importních dat před uložením do business objektů systému IS FLORES.
Po dostatečném ověření a otestování správné funkčnosti je rozumné vynechat mezikrok ověřování dat a data do IS FLORES ukládat přímo.

Příklad 3:

Definovatelným importem je možné nahradit i některá zakázková řešení. Typicky se jedná o tzv. „importní můstky"

atd. (opakovaná aktualizace ceníků, artiklů, firem, import dokladů atd.)

1.3 ZÁKLADNÍ DIAGRAM PROCESU IMPORTU



2. DEFINOVATELNÉ IMPORTY V IS FLORES

2.1 LICENCE A POTŘEBNÁ NASTAVENÍ

Modul definovatelných importů je uživateli dostupný, pokud má pro modul aktivní licenci.
Předpokládá se rozumné nastavení práv k business objektům pro jednotlivé uživatele sytému IS FLORES.

Pro automatické zakládání nových dokumentů s kategorií „definovatelný import" je nezbytné jednorázově nastavit řadu dokladů kategorie „definovatelný import". Přiřazení řady dokladů se provádí v agendě Kategorie dokumentů/záložka Detail/podzáložka Řady dokladů.

2.2 LOGOVÁNÍ DEFINOVATELNÝCH IMPORTŮ

Při provádění parsingu importních dat definicí definovatelného importu a při operacích s business objekty vždy vzniká nový záznam s logem. Záznamy logování jsou ukládány do agenda Logy. Záznam logu vzniká vždy, tedy pokud v průběhu procesu dojde k chybám, ale i pokud vše proběhne v pořádku. Úroveň logování je možné ovlivnit na hlavičce definovatelného importu volnou Úroveň logování. Logovat lze všechny podrobnosti, jen chyby nebo podrobnosti včetně XML parsovací struktury.

Rozšířené logování

Na hlavičce definovatelného importu lze zapnout i rozšířené logování, které následně umožní prohlížet log po jednotlivých řádcích. Volba na hlavičce má název **Logovat záznamy vzniklé zpracováním definovatelného importu** a funguje jen při samotném importu BO, nikoliv, pokud je provedeno pouze parsování.
Zatržítkem **Ukládat obsah chybných záznamů** je možné si zobrazit část XML rozparsovaného souboru s daty, která se nepovedlo uložit. Volba **Ukládat skutečná importní data jako přílohu logu** zajistí, že v rámci záznamu s logem bude vložena příloha obsahují reálná data před importem. Data před samotným importem mohou být díky skriptům modifikovaná a proto je důležité vidět jejich skutečnou podobu tak, jak jsou do IS FLORES importována.

U logů s velkým počtem řádků se log rozdělí do více samostatných záznamů logů po blocích. Limit pro jeden blok je 50 tisíc řádků.
Znamená to, že při zpracování jednoho definovatelného importu k němu může vzniknout více záznamů logů. V závěrečných zprávách o průběhu zpracování definovatelného importu se zobrazí informace o všech vytvořených záznamech logů.

U logů s velkým počtem řádků se log rozdělí do více samostatných záznamů logů po blocích. Limit pro jeden blok je 50 tisíc řádků.
Znamená to, že při zpracování jednoho definovatelného importu k němu může vzniknout více záznamů logů. V závěrečných zprávách o průběhu zpracování definovatelného importu se zobrazí informace o všech vytvořených záznamech logů.

2.3 KÓDOVÁNÍ

Podporované kódování:

- Pro typ importu XML: UTF-8
- Pro typ importu TXT a CSV: kódováni bez omezení - například Win-1250 nebo UTF-8 s BOM

POZNÁMKA
Pro správné zobrazování českých znaků (háčky, čárky apod. je potřeba u formátu UTF-8 vždy nastavit volbu UTF-8 s BOM.)

2.4 MAXIMÁLNÍ DOPORUČENÝ POČET ZÁZNAMŮ V JEDNOM IMPORTU

Maximální počet záznamů by optimálně neměl překročit 100.000. Tato hodnota je ověřená z praxe. Neznamená to že nelze importovat i větší soubory, nicméně je potřeba počítat s tím, že import větších souborů potrvá déle. Větší soubory nad 100.000 záznamů proto doporučujeme rozdělit do více menších souborů.

3. AGENDY DEFINOVATELNÉHO IMPORTU

3.1 NOVÝ MODUL „IMPORTY A EXPORTY"

V IS FLORES je k dispozici nový modul „Importy a Exporty" podléhající samostatné licenci.

Modul obsahuje agendy určené pro práci s definovatelnými importy:

- **Zdroje dat definovatelných importů**
- **Definice definovatelných importů**

3.2 AGENDA ZDROJE DAT DEFINOVATELNÝCH IMPORTŮ

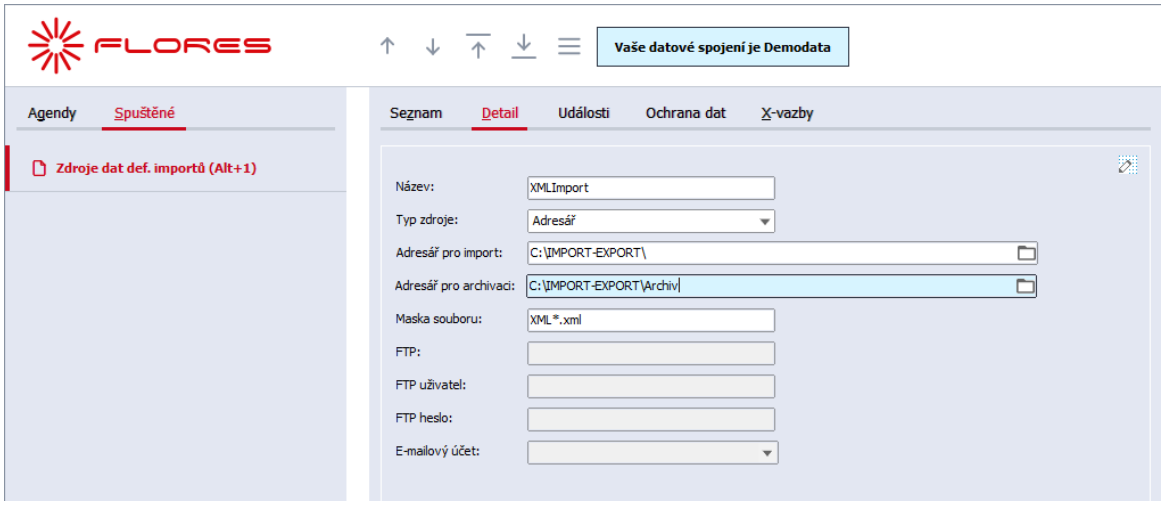
Číselníková agenda určená pro ukládání obecných nastavení definovatelných importů.

Popis položek:

Název	název pro daný zdroj dat
Typ zdroje	typ zdroje dat importu aktuálně podporované typy: Adresář plánované: FTP, Dokument, Email (z agendy „Doručené e-maily" systému IS FLORES)
Adresář pro import	cesta k adresáři, ze kterého se budou čerpat soubory pro automatický import (pokud není vyplněno, nelze importovat automaticky a je na uživateli, aby soubor před spuštěním definovatelného importu vybral ručně) **POZOR** Název importního souboru nesmí obsahovat zvláštní nebo speciální znaky, např. &, €, @, # apod. Tyto znaky nejsou v názvu souboru podporovány.
Adresář pro archivaci	cesta k adresáři pro archivaci zpracovaných importních souborů V případě změny cesty k archívu se změna adresáře projeví až po restartu agendy Definice definovatelných importů.
Adresář pro chyby	Cesta k adresáři pro chyby při zpracovaných importních souborů. Pokud je adresář pro chyby zadaný, dojde v případě chyby při zpracování konkrétního importního souboru definovatelným importem k přesunutí souboru do adresáře pro chyby. Pokud adresář pro chyby zadaný není, přesunují se soubory do archivního adresáře, pokud ani archivní adresář není zadaný, soubory se nepřesunují. V případě změny cesty k adresáři s chybami se změna adresáře projeví až po restartu (vypnutí/zapnutí) agendy Definice definovatelných importů.
Maska souboru	maska importního souboru pro výběr (filtraci) souborů při automatickém importu (např.: *.xml)
FTP	identifikace FTP pro připojení

(dostupné po přidání typu zdroje „FTP")	povinné pro typ zdroje dat „FTP"
FTP uživatel *(dostupné po přidání typu zdroje „FTP")*	FTP uživatel povinné pro typ zdroje dat „FTP"
FTP heslo *(dostupné po přidání typu zdroje „FTP")*	FTP heslo uživatele povinné pro typ zdroje dat „FTP"
E-mailový účet *(dostupné po přidání typu zdroje „Email")*	pro zdroj dat „Email" bude využito nové pravidlo zpracování doručené e-mailové pošty

Zdroj dat typu „webová služba" není v této verzi podporován. Je možné pro konkrétní případ řešit zakázkově - naskriptováním stahování souborů do adresáře.



3.3 AGENDA DEFINICE DEFINOVATELNÝCH IMPORTŮ

Agenda určená pro zadávání jednotlivých definic definovatelných importů.

V definici definovatelného importu se nastavuje, jakým způsobem se budou zpracovávat importní data.

Jednotlivé definice definovatelných importů je možné kopírovat.

Záznamy definic definovatelného importu je možné exportovat do souboru (tlačítko „Uložit definici"), případně si takto exportovanou definici do systému IS FLORES importovat (tlačítko „Načíst definici"). Součástí exportu definice importu je i příslušný zdroj dat definovatelných importů.

Po importu definice do systému je doporučeno si zkontrolovat správnost typu zdroje dat a jeho nastavení pro novou definici.

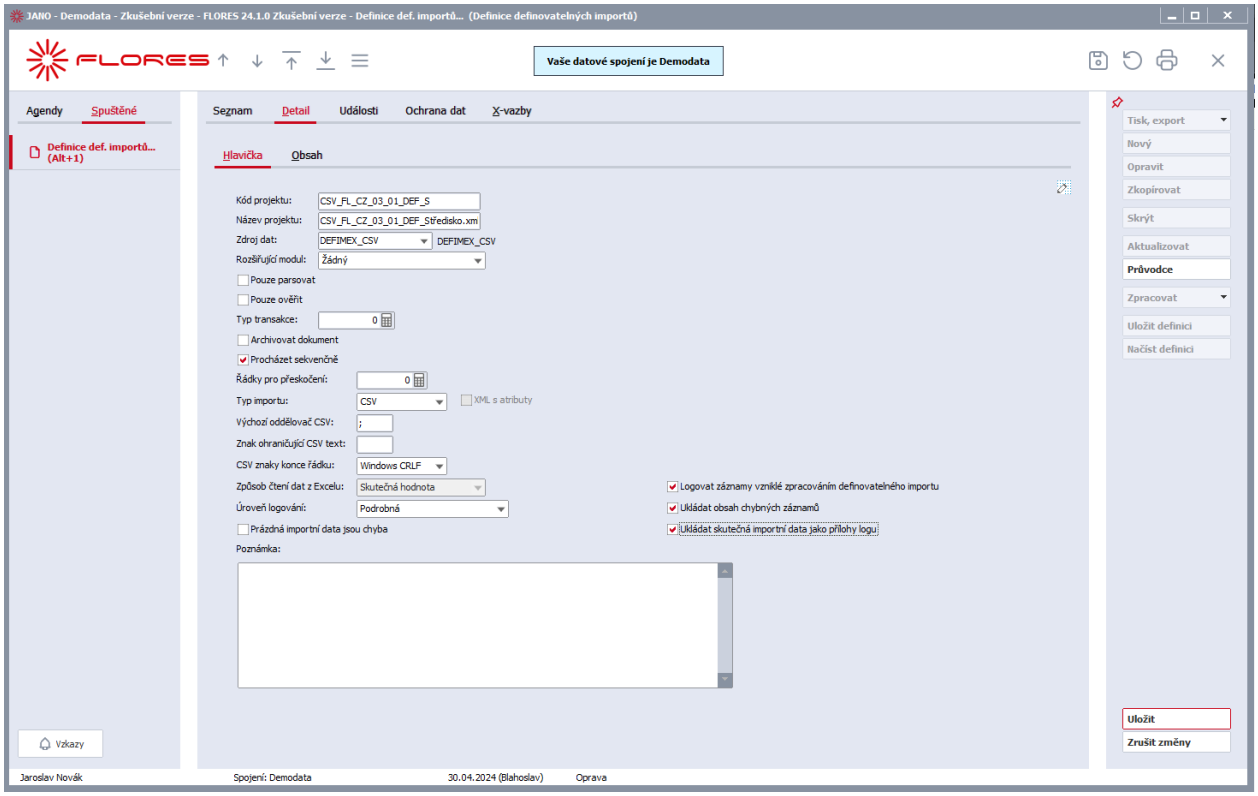
3.3.1 HLAVIČKA DEFINICE DEFINOVATELNÝCH IMPORTŮ

Popis položek hlavičky:

Kód projektu	kód projektu
Název projektu	název projektu
Zdroj dat	zdroj dat importu odkazující do číselníku „Zdroje dat definovatelných importů" povinný údaj
Rozšiřující modul	informace, jestli se má pro zpracování použít rozšiřující nástavba definovatelného importu výchozí hodnota: Žádný (možnost „EDI" bude aktivní až v budoucí verzi podporující elektronickou výměnu dat (EDI))
Pouze parsovat	informace, jestli se bude přímo zapisovat do BO (NE) nebo se jen provede parsování obsahu importního souboru a data se místo do BO uloží do dokladu dokumentů pro uživatelskou kontrolu (ANO) Pro přegenerování obsahu existujícího dokumentu funkcí „Přegenerovat DI" lze použít pouze definice se zatrženou touto volbou. výchozí hodnota: ANO
Pouze ověřit	informace, jestli se bude pouze testovat správnost definice pro založení nebo aktualizaci business objektu (ANO) při tomto nastavení se nevytváří žádný dokument ani se nezakládá/neaktualizuje business objekt výchozí hodnota: NE
Typ transakce	Toto pole je pouze připraveno do budoucna. Momentálně nastavení jakékoliv hodnoty nezmění typ importu. informace, jestli se bude import celého souboru zpracovávat v rámci jedné DB transakce (hodnota: 0), po blocích dle nastaveného počtu záznamů (hodnota > 1) nebo bude transakce samostatná pro každý jednotlivý BO (hodnota: 1) výchozí hodnota: 1 (podpora transakcí je plánovaná do jedné z dalších verzí)
Archivovat dokument	informace, jestli se bude obsah původního (neparsovaného) importního souboru archivovat jako příloha v novém dokladu dokumentu Archivace zdrojových dat je také podmínkou pro pozdější použití funkce „Přegenerovat DI" v agendě Dokumenty. Pokud zdrojová data nejsou v

	dokumentu uložena, nelze je znovu naparsovat. výchozí hodnota: NE
Procházet sekvenčně	informace, jestli se bude obsah importního souboru zpracovávat postupně tj. sekvenčně (ANO) pro XML Výchozí hodnota: ANO
Řádky pro přeskočení	pro textový soubor určuje počet řádků, které se na začátku zpracování souboru přeskočí výchozí hodnota: 0 (žádné řádky se nepřeskakují)
Typ importu	informace jakého typu je importní soubor povinná položka podporované hodnoty: XML, CSV (text s oddělovači definovanými v definici importu), TXT (text s položkami určenými pozicí a délkou) , Excel (obsah určený záložkou a sloupcem tabulky)
XML s atributy	Položka je přístupná jen pro typ importu XML. Pokud je položka zatržena, je možné importovat i XML, které obsahuje atributy. Importní definice pro XML elementy i atributy je prakticky shodná. Výchozí hodnota: Neaktivní (nezatrženo).
Výchozí oddělovač CSV	Znak oddělení položek CSV, použije se pro předvyplnění obdobné položky na položkách definice importu. Položka je pro typ importu CSV povinná.
Znak ohraničující CSV text	Ohraničení textu CSV v importních datech.
CSV znaky konce řádku	Znaky odřádkování. Výběr z přednastavených voleb odřádkování typického pro platformy Windows, Mac a Unix.
Způsob čtení dat z Excelu	Položka je k dispozici jen v případě, že je zvolen Typ importu Excel. Volba určuje způsob přebírání dat z Excelu. Skutečná hodnota - Data se budou přebírat bez ohledu na jejich formátování v MS Excel. Zformátovaná hodnota - Data budou přebírat včetně použitého formátování v MS Excel.
Úroveň logování	Volba ovlivňuje, které logované informace z procesu zpracování definovatelných importů budou do logu zapisovány. Podrobná - Zobrazují se všechny logy kromě zobrazení naparsované struktury importního XML při XML importu. Výchozí hodnota Jen chyby - Zobrazují se jen chyby, pokud nějaké při importu a parsování vzniknou. Podrobná s rozšířením pro XML - Zobrazují se všechny logy včetně zobrazení naparsované struktury importního XML při XML importu.
Prázdná importní data jsou chyba	Pokud je položka nastavena na hodnotu ANO, přistupuje se při zpracování definovatelného importu k prázdným importním datům jako k chybě. Prázdný soubor je označen jako chybný a přesune se do adresáře pro chyby, pokud je adresář pro chyby zadaný. Nevznikne interní XML, a při nastavení volby Pouze parsovat nevznikne ani dokument s interním XML. Pokud je položka nastavena na hodnotu NE, pracuje systém se souborem jako by nebyl prázdný. Prázdný soubor se import pokusí zpracovat a vznikne tedy v tomto případě prázdné interní XML, tedy vznikne i dokument pro

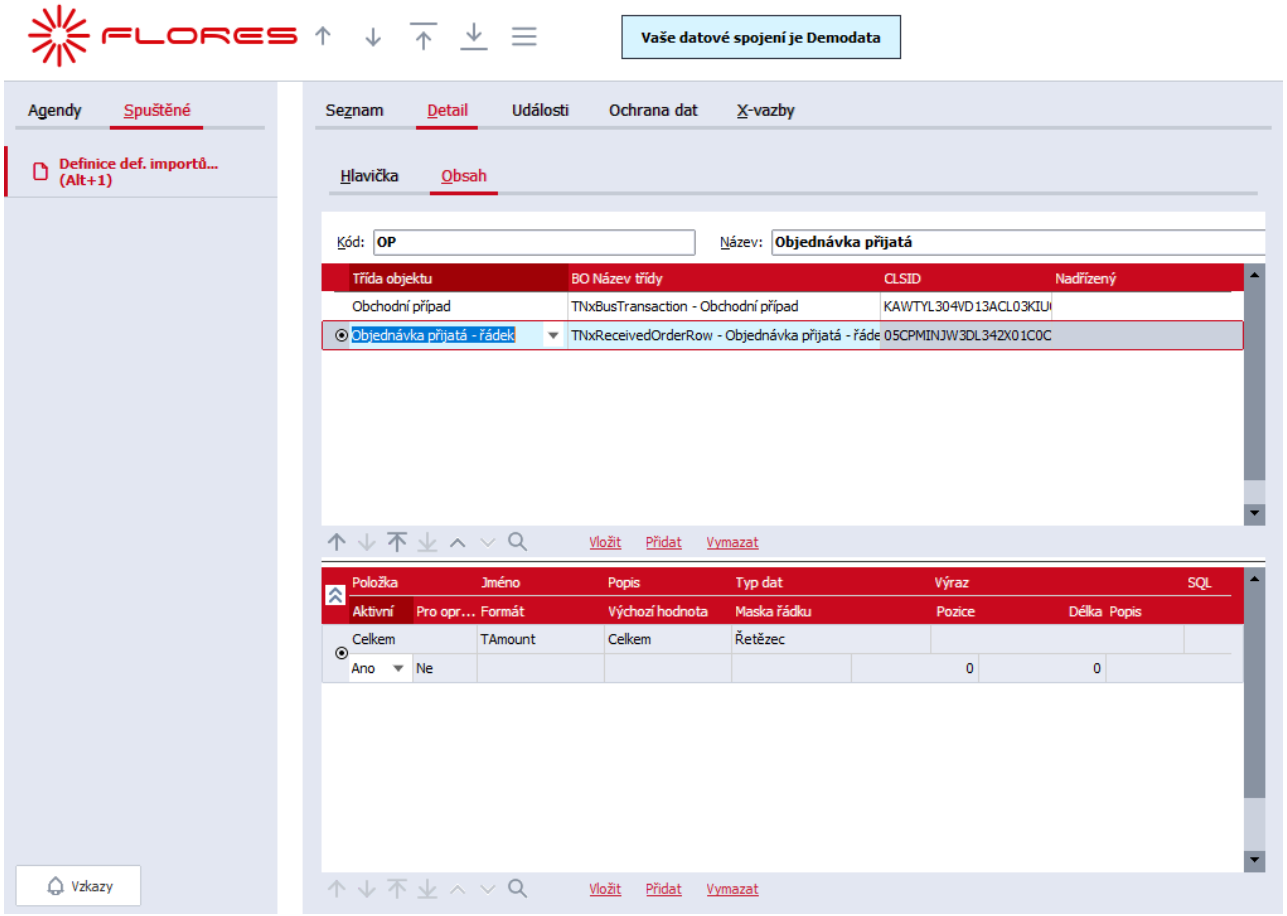
	volbu Pouze parsovat. Soubor se přesune do archivního adresáře, pokud je adresář zadaný. Výchozí hodnota Ne.
Logovat záznamy vzniklé zpracováním definovatelného importu	Pole má význam při samotném zpracování parsovaného souboru a ukládání Business objektu. Je-li položka zatržena, začnou se plnit řádky v agendě Logy, Subzáložka Řádky def.importu .V řádcích se lze snadněji orientovat, než ve standardním logování v subzáložce Poznámka. Položku má smysl mít aktivní (zatrženou) v případě, kdy logy obsahují často chyby a je třeba dohledávat jejich příčinu. **POZNÁMKA** V případě, kdy je na hlavičce def. importu zatržena volba Pouze parsovat, tak se do subbzáložky nic nezapíše. Zápis proběhne až při zpracování rozparsovaného souboru v agendě Dokumenty.
Ukládat obsah chybných záznamů	Zatržením toho pole se v případě chyby začne obsah Business objektu, dále jen BO, který se nepodařilo naimportovat (uložit) vypisovat do subzáložky Řádky def. importu v agendě Logy, položka **Obsah neuloženého BO**. Z tohoto obsahu je možné často snadněji identifikovat možnou příčinu neuložení BO.
Ukládat skutečná importní data jako přílohy logu	V případě, kdy je položka zatržena se do agendy , záložka vytvoří dokument, jehož obsahem jsou skutečná importní data. Skutečnými importními daty se rozumí data po jejich případně transformaci, například pomocí skriptingu (háček "IEImportExport_AfterSetImportDocument_Hook") nebo pomocí rozšiřujícího modulu definovatelných importů (EDI). Importní data se ukládají do obsahu samostatného dokumentu. Kategorie a další nastavení je stejné jako u dokumentů definovatelných importů vznikajících při volbě Pouze parsovat. Tedy kategorie IMP. Dokumenty nelze zpracovat a mají jiný prefix popisu. Pro kategorii je nutné mít přiřazenou řadu (stejně jako u EDI apod.). Dokument se skutečnými importními daty se automaticky připojí k záznamu logování jako příloha. V případě více záznamů logů se každý dokument připojuje pouze jednou.
Poznámka	Zde je možno vepsat poznámku vztahující se k definicím definovatelných importů.


JANO - Demodata - Zkušební verze - FLORES 24.1.0 Zkušební verze - Definice def. importů... (Definice definovatelných importů)
FLORES
Vaše datové spojení je Demodata
Agendy
Spuštěné
Definice def. importů... (Alt+1)
Seznam
Detail
Události
Ochrana dat
X-vazby
Hlavička
Obsah
Kód projektu:
CSV_FL_CZ_03_01_DEF_S
Název projektu:
CSV_FL_CZ_03_01_DEF_Středisko.xm
Zdroj dat:
DEFIMEX_CSV
DEFIMEX_CSV
Rozšiřující modul:
Žádný
Pouze parsovat
Pouze ověřit
Typ transakce:
Archivovat dokument
Procházet sekvenčně
Řádky pro přeskočení:
Typ importu:
CSV
XML s atributy
Výchozí oddělovač CSV:
Znak ohraničující CSV text:
CSV znaky konce řádku:
Windows CRLF
Způsob čtení dat z Excelu:
Skutečná hodnota
Úroveň logování:
Podrobná
Prázdná importní data jsou chyba
Logovat záznamy vzniklé zpracováním definovatelného importu
Ukládat obsah chybných záznamů
Ukládat skutečná importní data jako přílohy logu
Poznámka:
Tisk, export
Nový
Opravit
Zkopírovat
Skrýt
Aktualizovat
Průvodce
Zpracovat
Uložit definici
Načíst definici
Vzkazy
Uložit
Zrušit změny
Jaroslav Novák
Spojení: Demodata
30.04.2024 (Blahoslav)
Oprava

3.3.2 OBSAH DEFINICE DEFINOVATELNÉHO IMPORTU

Podzáložka Obsah na Detailu definice je rozčleněná na dvě části:

- horní seznam je určen pro zadání tříd business objektů
- spodní podseznam je určen pro zadání položek subkolekce třídy business objektu zadaného v nadřízeném seznamu



Popis položek řádku seznamu:
Kolekce je řazená a zpracovává se v takto nastaveném pořadí.

Třída objektu	interní jednoznačná identifikace business objektu (BO) lze zadat výběrem z nabízených možností, nebo se automaticky vyplní po zadání položky CLSID
BO Název třídy	interní název BO (např. „TNxReceivedOrder - Objednávka přijatá") jen ke čtení, předvyplní se po zadání CLSID nebo Třídy objektu
CLSID	interní jednoznačná identifikace business objektu (BO) lze zadat ručně, nebo se automaticky vyplní po zadání položky Třída objektu povinné
Nadřízený	odkaz na nadřízený business objekt (BO) povinný jen u kolekcí BO
Typ importu	určuje způsob práce s business objekty (platí pro hlavičky i položky kolekcí) tj. jestli se bude vytvářet nový BO nebo půjde o opravu již existujícího BO podporované volby: Nový: vždy se zakládá nový BO Oprava: vždy se provádí oprava BO (pokud se ID pro opravu nedohledá, záznam se nezpracuje) Oprava nebo nový: pokud se dohledá ID pro opravu BO, provede se oprava, jinak se založí nový BO výchozí hodnota: Nový
Aktivní	příznak, jestli se bude daný BO zpracovávat pokud je nastaveno na „NE": nastavení v podřízené subkolekci se ignoruje, tj. daný BO se importem nezpracuje výchozí hodnota: ANO (viz také kap. *3.3.3 Průvodce vytvořením definice importu*)
Provádět předvyplnění	Položka je ve výchozím stavu nastavena na hodnotu Ano. V závislosti na nastavení položky se na nových business objektech vznikajících zpracováním def. importu provádí předvyplnění (metoda prefill) nebo se tato metoda neprovádí. (pokud je nastaveno na Ne). **POZOR** Možnost vypnutí předvyplňování je zde především z důvodu možnosti zakládat skladové karty v jednom jediném kroku. Vypnutí prefillu u jiných situací nedoporučujeme. Importy budou vyžadovat doplnit všechny potřebné položky, které jsou povinné pro založení BO - někdy až na takové úrovni, kterou zvládne pouze uživatel, který vidí do kódu systému.
Název záložky Excelu	Položka je přístupná jen pro typ importu Excel. Položka určuje, z které záložky Excelu budou hodnoty z Excelu pro daný objekt načítány. Je podporováno načítání hodnot pro více BO ze stejných nebo i různých záložek Excelu. Vyplnění položky je kontrolováno soft validací. Pro spuštění importu je ale vyplnění položky povinné.

Popis položek řádku subkolekce na spodním podseznamu:
Kolekce je řazená a zpracovává se v nastaveném pořadí.

Seznam položek se aktualizuje při změně aktivního řádku v nadřízeném seznamu (viz vrchní seznam pro zadání tříd business objektů).

Položka	název položky business objektu (BO) v IS FLORES (např. „Firma“)
Jméno	interní název položky BO v IS FLORES (např. „Firm_ID“) povinné, pokud je položka „Pro opravu“ nastavena na hodnotu „NE“ vyplní se výběrem Položky nebo ji lze zadat položku lze využít pro definování proměnné nastavením prefixu *VAR* (vysvětlení dále v textu). **TIP** K položkám vlastněné kolekce (Vlastněná kolekce je v GenDoc.chm definována textem (Owned). Například: ResidenceAddress_ID(Owned)) se v definovatelných importech přistupuje jednoduše rozvinutím položky hlavního objektu pomocí názvu vlastněné kolekce a za tečkou přímo název položky vlastněné kolekce. Například takto: ResidenceAddress_ID.Street.
Typ dat	typ položky (Řetězec, Celé číslo, Desetinné číslo, Kolekce, Datum, Ano/Ne, XML kolekce, Neurčeno, Jedinečná přípona proměnné, Excel propojení, Podmínka)
Popis	libovolný popis nebo poznámka uživatele předvyplňuje se z popisu vybraného BO
Výraz	výraz, který se vyhodnotí nad daným BO a hodnotou získanou z importního souboru ve výrazu lze využívat proměnné (viz níže) Validace: Nelze zadat zároveň Výraz i SQL tj. nelze vyhodnocovat oba výrazy společně, vždy jen jeden z nich pomocí ▼ lze vyvolat Editor výrazů
SQL	SQL výraz, který se vyhodnotí pro hodnotu získanou z importního souboru ve výrazu lze využívat proměnné (viz níže) Validace: Nelze zadat zároveň Výraz i SQL tj. nelze vyhodnocovat oba výrazy společně, vždy jen jeden z nich pomocí ▼ lze vyvolat Editor výrazu pro SQL
Podmínka položky	podmínka, která se vyhodnotí pro hodnotu získanou z importního souboru v podmínce lze využívat proměnné (viz níže) pomocí ▼ lze vyvolat Editor výrazu pro podmínku
Vyhodnotit podmínku před zpracováním	Při nastavené hodnotě NE se podmínka z pole Podmínka položky vyhodnocuje na již zpracované hodnotě z importních dat. Zpracováním se rozumí, že je provedena na hodnotě příslušná datová konverze a vyhodnoceny výrazy nebo SQL dotazy. Při hodnotě ANO se podmínka z pole Podmínka položky vyhodnocuje na tzv “syrové” hodnotě z importních dat. To znamená dříve než je na hodnotě provedena příslušná datová konverze a před vyhodnocením výrazu nebo SQL dotazu. Výchozí hodnota: Ne **PŘÍKLAD** Mějme nadefinovaný základní import skladových karet v CSV formátu.

	Chceme v tomto importu kromě Kódu a Názvu skladové karty importovat také položku Procento obvyklé marže. Ta by měla být ve formátu desetinného čísla. Avšak v importním souboru zjistíme, že na dané pozici nejsou vždy jen čísla, ale třeba i text. Bez vyhodnocení podmínky před zpracováním by nám takový import skončil chybou v místě, kde by se místo čísla nacházel text. Díky tomu, že použijeme podmínku, například CfxStrToFloatDef('%s', -1, ',') <> -1, a nastavení Vyhodnotit podmínku před zpracováním na hodnotu Ano, jsou zpracovány jen ty záznamy, které budou podmínkou vyhodnoceny jinak, než -1. U záznamů, které podmínce nevyhoví, nedojde k importu této položky. Tzn. záznam se naimportuje, ale položka Procento obvyklé marže bude prázdná (nulová).
Aktivní	příznak, jestli se bude daná položka subkolekce zpracovávat (ANO) nebo ignorovat (NE) platí pro ověření, parsování, tvorbu BO Po nastavení hodnoty Ano či Ne se tato hodnota přebere pro další přidanou položku BO. výchozí hodnota: ANO (viz také kap. *3.3.3 Průvodce vytvořením definice importu*)
Pro opravu	využití u typu importu „Oprava" nebo „Oprava nebo nový" pro hodnotu „ANO" se vyhodnocením výrazu na řádku získá ID pro opravu výchozí hodnota: NE
Formát	speciální formátování pro typ položky, kde to má smysl (např. DateTime, Float ad.) např. formát datumu (YYYYMMDD), oddělovač desetinných čísel (tečka nebo čárka) atd.
Výchozí hodnota	výchozí hodnota, která se použije v případě, že se hodnotu nepodařilo z importního souboru získat položku lze využít pro získání hodnoty z proměnné definované prefixem **VAR**

Dále podseznam obsahuje položky nastavení pro napárování na importní soubor:
Vyplnění položek je povinné v závislosti na nastaveném typu importního souboru na hlavičce definice.

Maska řádku	Maska, jejímž vyhodnocením bude rozhodnutí, zda daný importní řádek zpracovávat či nikoliv (využití u TXT, CSV). Jinými slovy, pokud je na položce v Definici pro import maska zadaná, musí v importních datech řádek začínat znaky uvedenými v masce, jinak se nezpracuje. Zadávat lze na jedné položce pouze jednu masku. Pokud bychom zadali dvě masky oddělené středníkem, zpracovala by se jen ta data, která odpovídají celé masce včetně středníku. Maska rozlišuje velká a malá písmena (je key sensitive). Použití masky si vysvětíme na příkladu níže. Zadaná maska na položce BO objektu se automaticky kopíruje i na další přidávanou položku BO.
Sloupec Excelu	Položka je přístupná jen pro typ importu Excel. Určuje, ze kterého sloupce se daná položka bude z MS Excelu načítat. Hodnota sloupce musí být vždy velkými písmeny, při zadání malého písmena se provede automatická konverze na velké písmeno. Písmena se zadávají bez apostrofů.
Způsob aktualizace textové položky	Položka je k dispozici pro typy importů "Oprava" nebo "Oprava nebo nový" a pro řetězcové položky. V ostatních případech je položka nedostupná. Podle typu nastavení položky se při opravě řetězcové položky business objektu zapsání hodnoty chová takto:

	Přepsat: původní hodnota se vždy přepíše řetězcem z importu Přidat na začátek: hodnota z importu se přidá na začátek k již existujícímu řetězci Přidat na konec: hodnota z importu se přidá na konec k již existujícímu řetězci Zvolená varianta u jedné položky BO se automaticky přebere do dalších přidávaných položek BO.
Ořezávat mezery	Pokud je položka nastavena na Ano ořezávají se při parsingu automaticky koncové mezery zleva i zprava. Pokud je položka nastavena na Ne mezery se neořezávají. Na hodnotu položky se bere ohled pro typy importu TXT, CSV, Excel. Pro typ XML se vždy importuje bez ořezání mezer. Výchozí hodnota je Ano.
Pozice	celé číslo, pozice začátku položky na řádku importního souboru dostupná pro Typ importu: TXT, CSV
Délka	celé číslo, délka položky dostupná pro Typ importu: TXT
Oddělovač	oddělovač položek v textovém formátu CSV dostupná pro Typ importu: CSV
XML položka	jednoznačný název položky v importním XML souboru dostupná pro Typ importu: XML Pokud není pole vyplněno nebo není vyplněno pole Výchozí hodnota nedojde ke zpracování pole při importu. Na toto je uživatel upozorněn zprávou při uložení definice.
XML větev	název větve položek kolekce v importním XML souboru dostupná pro Typ importu: XML Povinná položka bez které není možné uložit definici.
Operátor kombinace podmínek	Nabízí hodnoty AND a OR. Výchozí AND. Používá se pro logické spojení více než jedné podmínky aktuálního business objektu. Položka je aktivní jen v případě, kdy v Typu dat je vybrána možnost: Podmínka. Dále je nutné mít vyplněnu položku Výraz.
Zastavit zpracování po nesplněné podmínce	Nabízí hodnoty Ano a Ne. Výchozí Ne. Pokud je aktuální stav vyhodnocení všech podmínek daného Business objektu, dále jen BO "NE" a položka je nastavena, ukončí se proces parsování všech následujících položek daného i podřízených BO. Při správném použití lze takto urychlit zpracování, typicky u položek, které jsou dále při nesplněné podmínce již zbytečné. Položka je aktivní jen v případě, kdy v Typu dat je vybrána možnost: Podmínka.
Kontrolovat duplicitu při ukládání	Nabízí hodnoty Ano a Ne. Výchozí hodnota: **Ne.** Hodnotu "Ano" lze nastavit pouze u položek datového typu: • Celé číslo • Ano/Ne • Desetinné číslo • Řetězec • Datum Pokud je položka nastavena na hodnotu „Ano" dojde v okamžiku ukládání nového business objektu (BO), připraveného zpracováním definovatelným importem (DI), k sestavení SQL dotazu podle DB tabulky daného BO a položky

	(fieldu). SQL obsahuje dotaz na hodnotu, která je do položky ukládaná. U dotazů do tabulek, které podporují skrývání, se do dotazu doplní nevyhledávání ve skrytých. **PŘÍKLAD** select ID from StoreCards where Hidden = 'N' and Code = 'cod3x' select ID from StoreUnits where ID <> '9M59000101' and Code = 'gr6' select ID from StoreEANs where EAN = 'ean76' Tímto způsobem se kontroluje duplicita unikátních položek. Pokud sestavený SQL dotaz vrátí nějaký záznam, znamená to, že se již stejná hodnota v databázi (DB) vyskytuje a daný BO se nebude ukládat. Více informací k této položce viz níže.
Režim editace položky	Tato výčtová položka má význam pouze v případě, že je na hlavičce definice importu zatržena volba Pouze parsovat. Určuje, zda a jak se daná datová položka nabídne při ručním zpracování dokumentu v dialogovém okně „Doplnění hodnot pro import". Režim editace nelze nastavit pro proměnné a nedatové položky. Výčtová položka má následující tři hodnoty: Nenabízet – položka se v dialogovém okně nenabízí a použije se její aktuální hodnota z interního XML. Vyplňovat jen prázdné – položka se nabídne pouze tehdy, pokud je její aktuální hodnota v interním XML prázdná. Vyplňovat vždy – položka se nabídne vždy a zadaná hodnota nahradí její aktuální hodnotu v interním XML. Výchozí hodnota: Nenabízet Uživatel může způsob vyplnění v dialogovém okně pro konkrétní zpracování změnit. Tím se upraví interní XML dokumentu, nastavení v definici importu se nezmění.

Při vyhodnocování výrazů a SQL dotazů lze využít dosazení hodnoty získané parsingem aktuální položky. Tato hodnota se ve výrazu vždy nahradí za konstantu %s.

POZOR
Použijete-li v jednom výrazu konstantu %s více než jednou je potřeba použít indexaci, tzn např %0:s.

Ve výrazech a SQL výrazech lze využívat i proměnné dříve nadefinované v definici, viz popis dále. Pomocí proměnných lze tedy ve výrazech využít i hodnoty z jiných položek než je aktuální.

Popis proměnných definovatelných importů:

V rámci provádění parsování importního souboru je možné si zapamatovat položku jako proměnnou a její hodnotu později využít v rámci importu daného business objektu (BO) při dalším zpracovávání na jiném řádku. Hodnota proměnné je platná pouze v průběhu zpracovávání jednoho hlavičkového BO. Proměnná je v definici importu identifikována prefixem *VAR*. Hodnotu proměnné je možné využít pro vyplnění položky „Výchozí hodnota", v podmínce vyhodnocované pro řádek subkolekce, ve výrazu a v SQL výrazu. Při využití proměnné ve výrazu nebo podmínce je nutné ji vždy použít s prefixem **VAR** a ukončit sufixem **VAREND**.

Proměnné *VAR* mohou být různých datových typů, např. Desetinné číslo, string, datum, atd. Datový typ proměnné určuje právě sloupec Typ dat

Proměnná *VAR* se neuvádí v uvozovkách ale pouze s hvězdičkami na začátku a na konci.

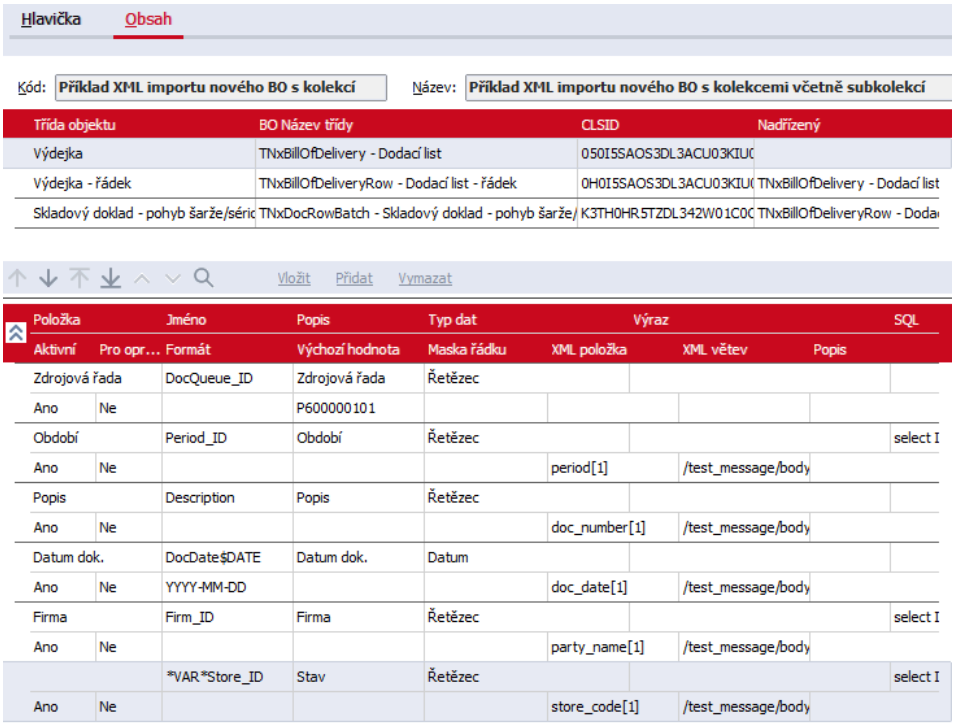
POZNÁMKA
V sekvenčních definovatelných importech typů TXT a CSV je povoleno v importních datech vynechat řádky libovolného nehlavičkového business objektu uvedeného v definici.
Je také povoleno v importních datech používat řádky, které nejsou určeny pro import, neexistuje k nim v importní definici odpovídající business objekt. Takové řádky jsou importem ignorovány.
Pro sekvenční zpracování importních dat platit, že řádky musí být uvozeny unikátním prefixem (pouze pokud je v importní definici nadefinováno více business objektů) a musí se dodržovat pořadí řádků odpovídající pořadí business objektů definovaných v importní definici.

Na příkladu níže je demonstrováno využití proměnné *VAR*Store_ID definované na hlavičkovém business objektu a její následné využití na řádku BO (dosazení do položky „Výchozí hodnota").

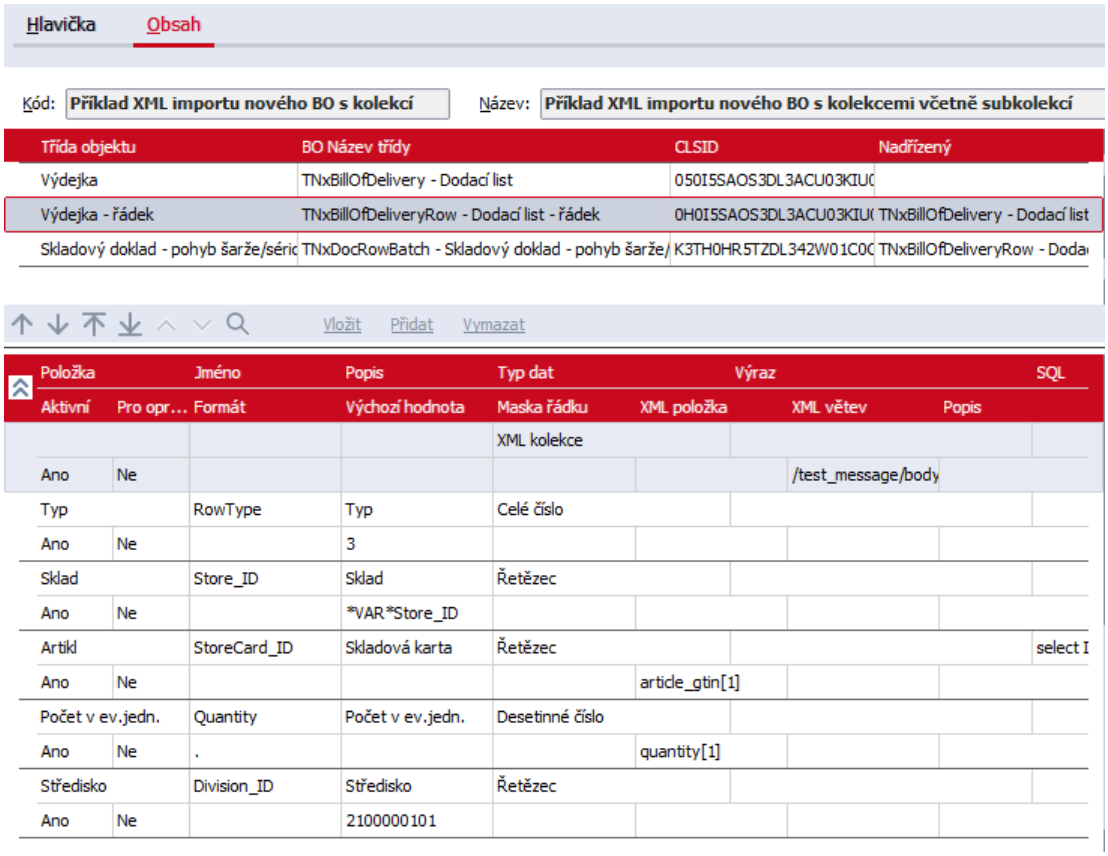
Konstanta „%s" je využita např. pro získání ID artiklu. Konstanta nabyde hodnotu XML položky pro daný řádek (*article_gtin[1]*) z importního souboru a využije se pro vyhodnocení SQL dotazu.

Datový typ konstanty „%s" a tedy i to, jestli bude v apostrofech, se řídí datovým typem položky.

PŘÍKLAD
Select ID from STORECARDCATEGORIES where COSTPRICESOURCETYPE = %s pro hodnotu 4
ale Select ID from STORECARDCATEGORIES where CODE = '%s' pro hodnotu 'Z', přičemž COSTPRICESOURCETYPE je integer a CODE je varchar(10). Datový typ řádku má odpovídat datovému typu položky, která je vyhledána (v tomto příkladu tedy ID, což je char(10) a tedy Řetězec.



Podrobněji o příkladu viz kap. 6.3.1 Příklady – Příklad č. 3.



POPIS A VYUŽITÍ DATOVÉHO TYPU PODMÍNKA

Na každém Business objektu (hlavičkové, kolekce i subkolekce.), dále jen BO definovatelných importů lze definovat podmínku, a to i kombinaci podmínek spojených logickými operátory. Logický operátor se použije pro spojení s předchozí hodnotou podmínky. Podmínka musí mít vždy hodnotu Ano/Ne vyhodnocovanou při zpracování podmínky parsingem (pokud je použito cokoliv jiného, je potřeba ověřit, jak se systém zachová - např. výsledek "True" bude vyhodnocovat jako "A", ale také integer bude vždy "A"). V podmínce lze použít proměnné definovatelných importů a výchozí hodnoty, tedy známé běžné možnosti.

Podmínka je platná vždy pouze pro příslušný BO definice definovatelného importu. Hodnota podmínka je vždy uložena v Interním XML na první pozici každého BO. Při zpracovávání Interního XML při přípravě BO pro uložení je hodnota podmínky u každého BO načtena. Pokud je podmínka neplatná, BO se neuloží

Plnění podmínky pomocí Výrazu a vyhodnocení podmínky

Podmínka je vždy definována jako Výraz a platí pro celý BO. Položka Výraz tedy nesmí být v případě použití datového typu Podmínka prázdná.

SQL je nepřístupné, protože nevrací logickou hodnotu. Pokud bude v poli SQL obsah zapsán, změnou typu dat na Podmínku dojde k jeho vymazání.

Při zpracovávání Interního XML při přípravě BO pro uložení je hodnota podmínky u každého BO načtena. Pokud je podmínka neplatná, BO se neuloží.

Použití funkce: Zastavit zpracování po nesplněné podmínce

Použitím této funkce dojde ke zrychlení zpracování importu, protože se nebudou parsovat následující "zbytečné" položky. Typickým příkladem použití podmínek je pro zamezení duplicitního ukládání již jednou uložených BO při opakovaném importu.

Uživatel, v případě výsledku vyhodnocení podmínky "Ne", má možnost proces parsingu při zpracování definovatelného importu zastavit. Uživatel si nastaví položku Zastavit zpracování po nesplněné podmínce na hodnotu: Ano. Takové nastavení dává smysl například pro jednu podmínku na objektu nebo pokud jsou 2 podmínky spojené pomocí operátoru AND (Pokud hned první vyhodnocení skončí jako "N", tak rovnou zastavíme import po první podmínce, protože výsledek kombinace bude vždy "N"). Naopak, pokud máme 2 podmínky s operátorem OR, tak na první podmínce ještě import zastavit nechceme, teprve pokud bude kombinace podmínek vyhodnocena jako "N", tj. zastavení importu nastavíme až na 2. podmínce.

> **POZOR**
> Již existující položka s názvem Podmínka položky se týká pouze konkrétní položky - vyhodnocuje se buď před zpracováním nebo po zpracování položky.

Při sestavování BO pro uložení může neplatná podmínka různě ovlivnit i podřízené BO kolekcí, které mají jako svůj nadřízený objekt BO, který se nezpracovává kvůli neplatné podmínce. Neexistující hlavičkový BO kvůli neplatné podmínce logicky způsobí i neuložení všech svých podřízených objektů. Při zpracování Interního XML je s tímto počítáno.

POPIS A VYUŽITÍ FUNKCE KONTROLAT DUPLICITU PŘI UKLÁDÁNÍ

Zde si vysvětlíme, jak přesně kontrola duplicit funguje.

Zamezení ukládání funguje pro hlavičkové BO i jednotlivé BO kolekce a subkolekce. Dohledávání duplicit se provádí napříč celou databází (DB). Tzn., že pokud kontrolujeme např. duplicitu položky EAN s hodnotou 123456 u jednotky artiklu „ks" pro artikl s názvem Televize a tento EAN se vyskytuje u jiné jednotky u zcela odlišného artiklu, systém ji i tak vyhodnotí jako duplicitu a neuloží. V případě podřízených kolekcí se neuloží ani všechny případné podřízené záznamy kolekcí závislé na neukládaném nadřízeném BO.

V následující tabulce si vysvětlíme chování kontroly duplicit na importu. Sloupec Výchozí stav prezentuje data, která jsou již uložena v databázi. Sloupec Importovaná data prezentuje data, která budeme chtít importovat. Tato tabulka je pro všechny varianty, které si představíme stejná, rozdíl je pouze v tom, na které položce budeme duplicitu kontrolovat, viz níže.

Výchozí stav	Kód	Název	Jednotka	EAN	Poznámka
	Kar01	Karta 01	kg	123456	
	Kar02	Karta 02	km	789123	
	Kar03	Karta 03	m	111111	
Importovaná data	**Kód**	**Název**	**Jednotka**	**EAN**	**Poznámka**
	Kar01	Karta 01	kg	123456	
	Kar04	Karta 04	km	789123	
	Kar05	Karta 05	m	111111	

1. varianta. Budeme mít kontrolu duplicity nastavenou na položce Kód.

Výsledek v databázi	Kód	Název	Jednotka	EAN	Poznámka
	Kar01	Karta 01	kg	123456	Další artikl s názvem Karta 01 se nenaimportuje (a ani subkolekce, které k ní patří), protože už v DB existuje artikl se stejným kódem

	Kar02	Karta 02	km	789123	
	Kar03	Karta 03	m	111111	
	Kar04	Karta 04	km	789123	
	Kar05	Karta 05	m	111111	

2. varianta. Budeme mít kontrolu duplicity nastavenou na položce Jednotka

Výsledek v databázi	Kód	Název	Jednotka	EAN	Poznámka
	Kar01	Karta 01	kg	123456	
	Kar01	Karta 01			Jednotka k artiklu se nenaimportuje, protože jednotka „kg" už v DB existuje. Nenaimportuje se ani EAN, který patří k této jednotce
	Kar02	Karta 02	km	789123	
	Kar03	Karta 03	m	111111	
	Kar04	Karta 04			Jednotka k artiklu se nenaimportuje, protože jednotka „km" už v DB existuje. Nenaimportuje se ani EAN, který patří k této jednotce
	Kar05	Karta 05			Jednotka k artiklu se nenaimportuje, protože jednotka „m" už v DB existuje. Nenaimportuje se ani EAN, který patří k této jednotce

..

3. varianta. Budeme mít kontrolu duplicity nastavenou na položce EAN

Výsledek v databázi	Kód	Název	Jednotka	EAN	Poznámka
	Kar01	Karta 01	kg	123456	
	Kar01	Karta 01	kg		EAN se k jednotce artiklu nenaimportuje, protože už v DB existuje.
	Kar02	Karta 02	km	789123	

	Kar03	Karta 03	m	111111	
	Kar04	Karta 04	km		EAN se k jednotce artiklu nenaimportuje, protože už v DB existuje.
	Kar05	Karta 05	m		EAN se k jednotce artiklu nenaimportuje, protože už v DB existuje.

POZNÁMKA
Všechny tyto akce jsou logovány v agendě Logy.
Pokud je položka určena k opravě, v SQL dotazu se vynechává záznam sama sebe.

Upravené logování

V mechanismu definovatelných importů bylo na různých místech upraveno i logování s ohledem na nové funkcionality. Z logu lze nyní vyčíst, jak byla podmínka BO vyhodnocena. V příkladu níže je situace, kdy se při parsování zjistilo, že podmínka BO pro řádek objednávky přijaté je vyhodnocena na "Ne". Nedojde tedy ani k uložení BO s informací o předchozí nesplněné podmínce.

PŘÍKLAD LOGOVÁNÍ
Parsing: Objednávka přijatá - řádek:, BOFieldName "_BOCOND_05CPMINJW3DL342X01C0CX3FCC", Maska "", RawValue "2", ProcessedValue(BOCondition after combining with previous BOcondition value) "N", pořadí importu "2", XMLCollection X-Path "/orion_message/body[1]/line_items[1]/item[2]"

Save-BO: IE TNxIESaveBusinessObjects: Fill BO values finished OK. BO name: TNxReceivedOrderRow - Objednávka přijatá - řádek.

05.03.2024 14:21:42 614, Jaroslav Novák, reRuntimeExe: ---IE TNxIESaveBusinessObjects: BO will not be processed because the condition for no processing is set. BO name: TNxReceivedOrderRow - Objednávka přijatá - řádek. (log may belong to next BO)

Oprava záznamu, kdy jsou v datech stejné údaje

Může se stát, že někdy máme v datech stejné údaje, např. máme dvě duplicitní střediska, nebo dva, či více duplicitních záznamů ve skladových kartách. Abychom chování při importu správně pochopili, ukážeme si jej na příkladu. Příklad je ryze informačního charakteru, duplicity by být neměly, ale duplicity povoluje.

PŘÍKLAD
Mějme vytvořených 5 středisek v agendě s Kódem 100 a Názvem střediska např. "Středisko Praha". Potřebujeme u všech pěti záznamů s Kódem 100 importem změnit Název středisek na "Středisko Brno" Použijeme klasickou definici s Typem importu Oprava nebo nový. V obsahu zadáme ID pro dohledání podle Kódu střediska a SQL dotaz na dohledání ID pro opravu. V importním souboru budeme mít jeden záznam s kódem 100 a střediskem Středisko Brno. Import proběhne a změní všech pět nalezených záznamů z původní hodnoty na novou.

Situaci, kdy bychom importem chtěli změnit všech 5 středisek a každé na jiný Název není možné tímto způsobem provést, jelikož by se všechna střediska změnila na Název střediska posledního řádku importu. Taková situace by musela být ošetřena individuálně za pomocí Podmínky položky.

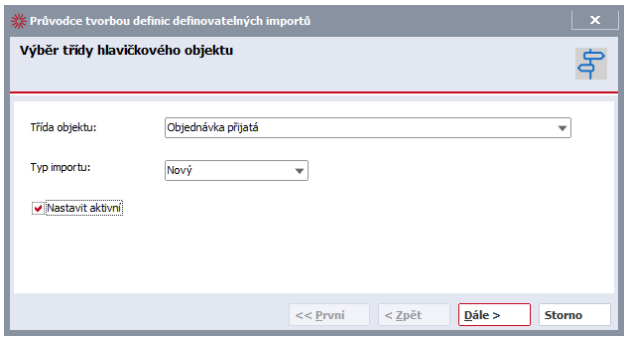
3.3.3 PRŮVODCE VYTVOŘENÍM DEFINIC DEFINOVATELNÝCH IMPORTŮ

Kromě ručního zadání jednotlivých tříd a subkolekcí business objektů je pro konfiguraci importní definice dostupná možnost „Průvodce tvorbou Definic definovatelných importů".

Tlačítko pro spuštění průvodce („Průvodce") je dostupné na záložce Detail v Panelu funkčních tlačítek a je aktivní v momentu editace definice (v případě tvorby nové definice po stisku tlačítka „Nový").

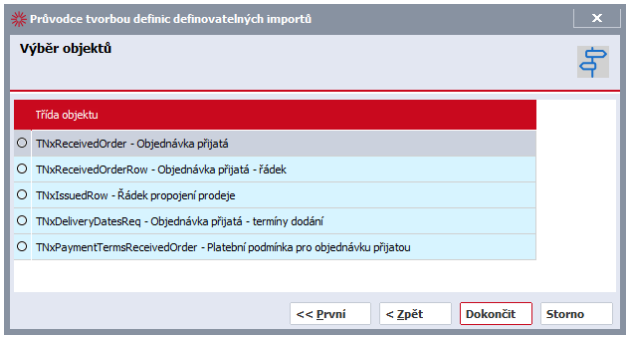
Stiskem tlačítka se zobrazí dialog pro zadání identifikace business objektu.

V prvním kroku je potřeba vybrat třídu hlavičkového objektu. Dále lze nastavit „Typ importu" (Nový, Oprava, Oprava nebo nový) a příznak „Nastavit aktivní" (viz „Aktivní" v popisu řádků definice). Zatržení příznaku „Nastavit aktivní" nastaví hodnotu „Ano" jednak na řádku třídy business objektu (horní seznam), jednak na řádcích všech položek subkolekce daného BO (položky spodního podseznamu).



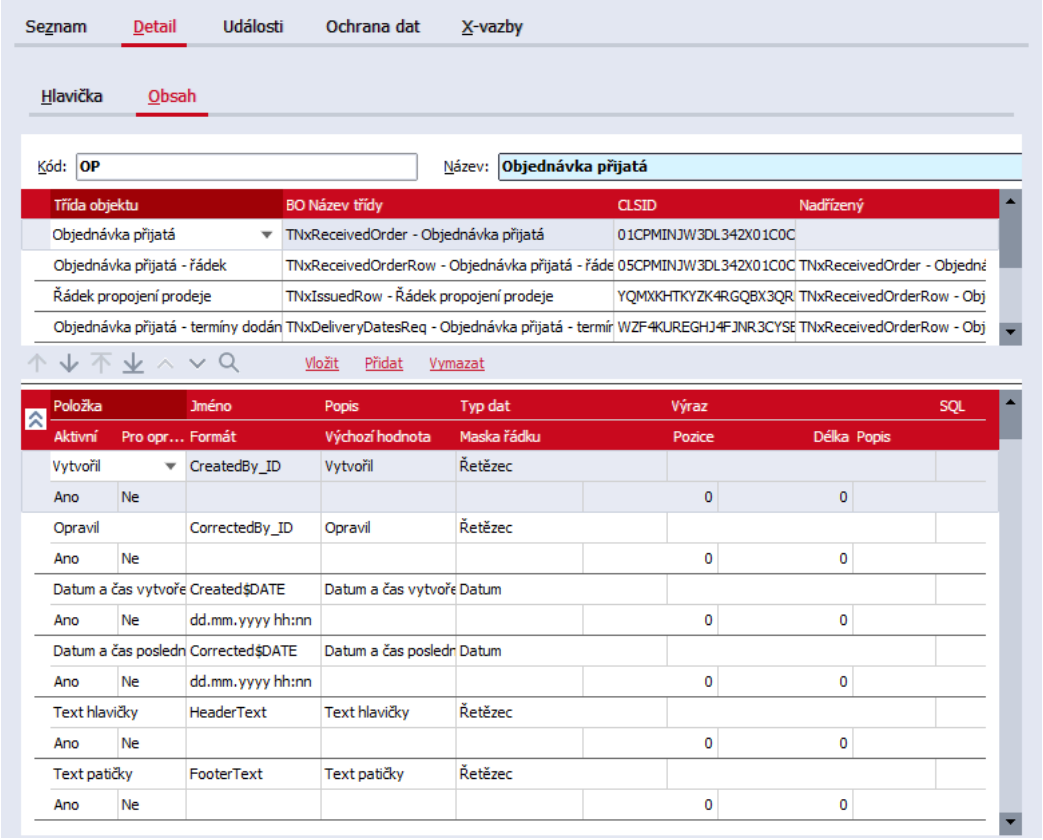
V druhém kroku průvodce systém nabídne business objekty zafiltrované dle třídy hlavičkového BO vybraného v prvním kroku. Business objekty, které již na definici zadané jsou, se nenabízí.

Předvyplnění položek BO se provede pro označené objekty.



Po dokončení práce s průvodcem se automaticky provede vyplnění položek seznamu zvolených tříd business objektů (vrchní seznam). Spodní podseznam načte všechny položky subkolekcí a předvyplní hodnoty položek, u kterých je to možné.

Doplnění dalších hodnot a správné nastavení definice je dále na uživateli.

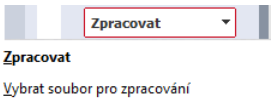


3.3.4 NAČTENÍ IMPORTNÍHO SOUBORU

V agendě „Definice definovatelných importů" je v panelu funkčních tlačítek k dispozici multifunkční tlačítko „Zpracovat" na spuštění definovatelného importu pro aktuální definici.

Volba „Importovat" provede automatický import souborů z adresáře definovaného na zdroji dat (viz „Adresář pro import"). Importní soubory v daném adresáři se zafiltrují dle zadané masky (viz „Maska souboru").

Druhá volba tlačítka „Vybrat soubor pro import" je určena k ručnímu výběru jednoho konkrétního souboru pro import.



Definovatelné importy podporují následující formáty importních dat:

- text poziční (TXT)
- text s oddělovači (CSV)
- formát XML

3.3.5 MOŽNOSTI A PROCES ZPRACOVÁNÍ SOUBORU

Výsledkem zpracování importních dat je založení nového nebo aktualizace (oprava) existujícího business objektu IS FLORES.

Možnosti zpracování importních dat:

1. **Pouze otestovat správnost definice pro založení nebo aktualizaci business objektu (BO):**

   Pro možnost pouze otestovat správnost definice je potřeba zatrhnout příznak „Pouze ověřit" na hlavičce definice definovatelných importů. Při tomto nastavení se nevytváří žádný dokument ani se nezakládá/neaktualizuje business objekt. Pro ověření správnosti se využije funkce „validate" BO.

2. **Provést nejprve parsing importního souboru do dokumentu a BO následně zpracovat samostatně:**

   Při zatržení příznaku „Pouze parsovat" se provede parsing importních dat do interní sktruktury XML určené k přímému zápisu do business objektů IS FLORES. Interní XML se uloží jako příloha do samostatného záznamu v agendě Dokumenty (viz dále kap. *3.3.7 Agenda Dokumenty*).

   Vzniklý dokument je v kategorií „Definovatelný import" (viz dále kap. *3.3.6 Nová kategorie dokumentů: „Definovatelný import"*) a řadou dokladů dle nastavení v agendě Kategorie dokumentů (viz kap. *2.1 Licence a potřebná nastavení*). Uživatel má možnost si interní XML strukturu zkontrolovat a případně upravit. Vytvoření nebo aktualizaci business objektu dle uložené interní XML struktury lze zahájit ručně tlačítkem „Zpracovat" v agendě Dokumenty nebo automaticky jako naplánovanou úlohu.

3. **Provést okamžité uložení nebo aktualizaci BO bez mezikroků:**

   Při nezatržených příznacích „Pouze ověřit" a „Pouze parsovat" se při spuštění importu vynechá „mezikrok" ukládání dokumentů. Vygeneruje se interní XML struktura a ihned se provede vytvoření nebo aktualizace business objektu.

Po importu je uživatel o výsledku akce informován a záznam je dostupný v příslušném logu.

Diagram průběhu zpracování nepovinné položky s nastaveným formátem:

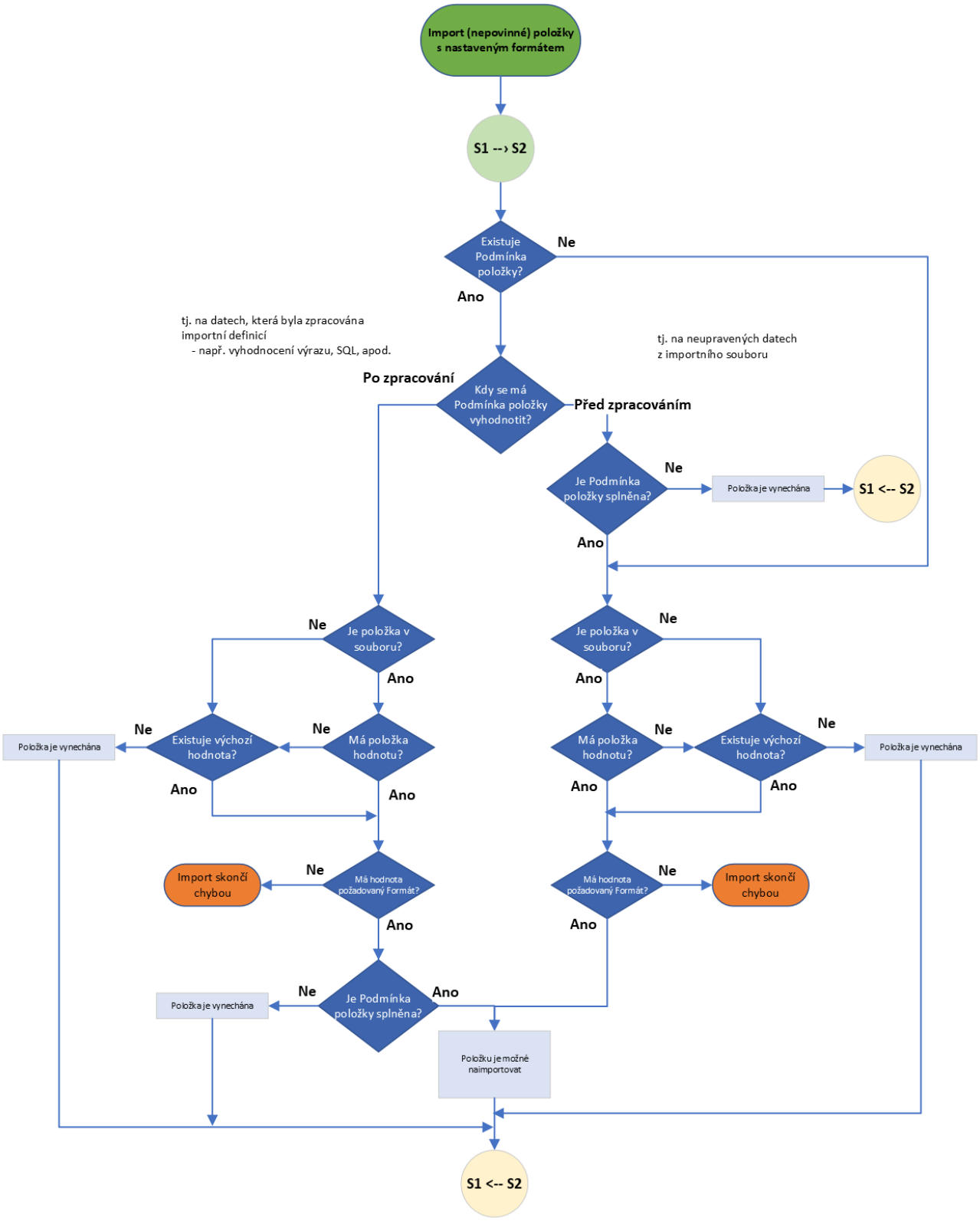
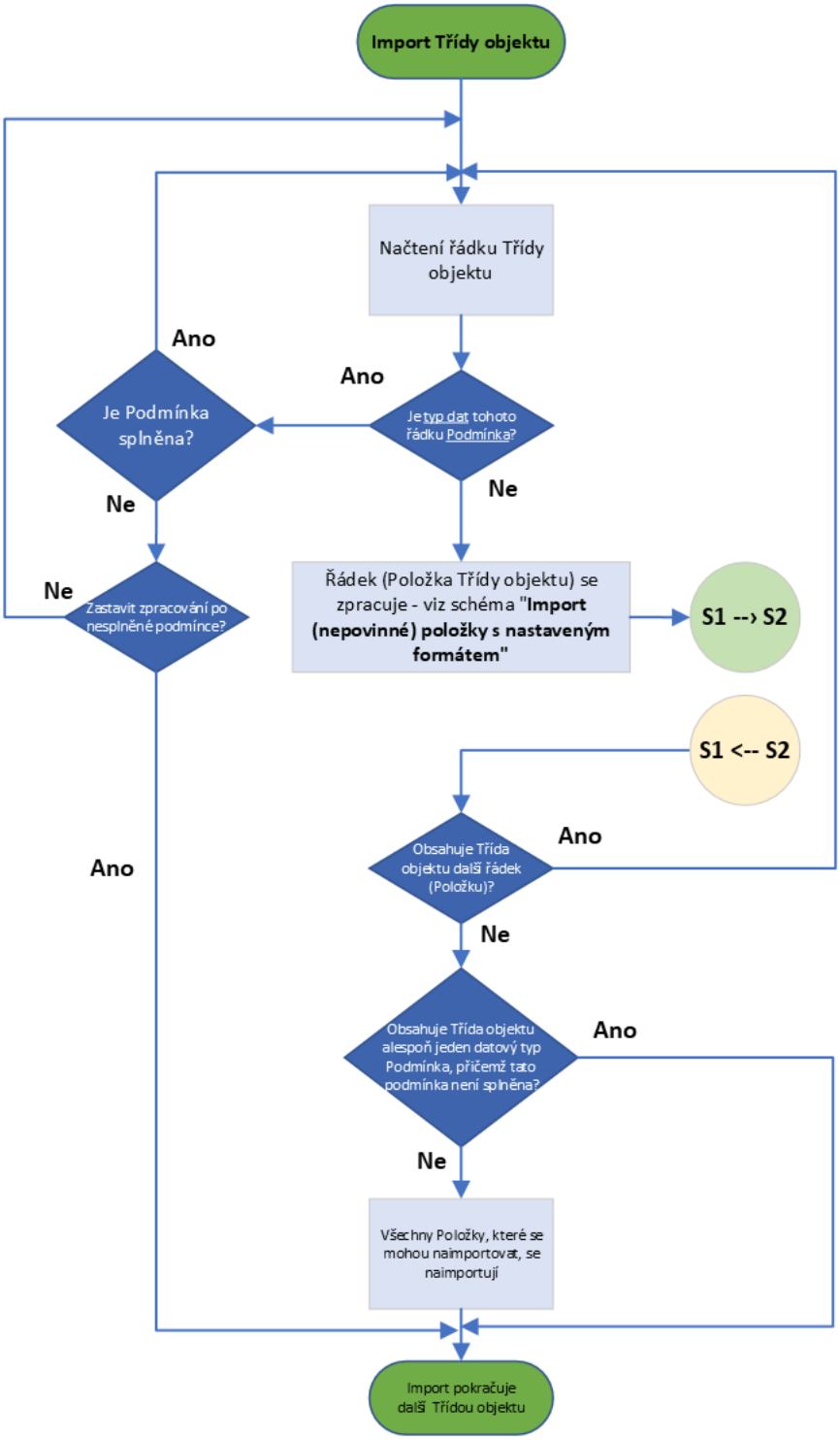


Diagram průběhu zpracování třídy objektu:


Import Třídy objektu
Načtení řádku Třídy objektu
Je typ dat tohoto řádku Podmínka?
Ano
Je Podmínka splněna?
Ano
Ne
Ne
Zastavit zpracování po nesplněné podmínce?
Ne
Řádek (Položka Třídy objektu) se zpracuje - viz schéma "Import (nepovinné) položky s nastaveným formátem"
S1 --› S2
S1 <-- S2
Obsahuje Třída objektu další řádek (Položku)?
Ano
Ne
Ano
Obsahuje Třída objektu alespoň jeden datový typ Podmínka, přičemž tato podmínka není splněna?
Ne
Všechny Položky, které se mohou naimportovat, se naimportují
Import pokračuje další Třídou objektu

4. AGENDA DOKUMENTY

4.1 NOVÁ KATEGORIE DOKUMENTŮ: „DEFINOVATELNÝ IMPORT"

Je vytvořena nová kategorie dokumentů „Definovatelný import".

Dokumenty s touto kategorií typicky vznikají při parsování importních dat pomocí definic definovatelných importů (je podrobněji popsáno v následujících kapitolách). Při zpracování takto vygenerovaných dokumentů dojde k uložení naparsovaných dat do business objektů IS FLORES.

4.2 AGENDA DOKUMENTY PRO DEFINOVATELNÉ IMPORTY

Hlavička dokumentu kategorie „Definovatelný import" obsahuje nové položky „Kód chyby" a „Text chyby", které slouží k zapsání informací o chybě v případě neúspěšného zpracování dokumentu (text chyby lze také vždy dohledat v záznamu logování definovatelných importů). Tyto dvě položky jsou na detailu dokumentu viditelné jen v případě, že dojde k chybě. V seznamu dokumentů jsou sloupce k dispozici vždy.

Do položky „Popis/název dokument:" se zapisuje informace o definici definovatelného importu, která byla pro import daného souboru použita.

Dokument vytvořený importem vstupního souboru (viz možnost „Pouze parsovat" v Definici definovatelných importů) obsahuje strukturu XML vzniklou parsováním vstupních dat připravenou k importu do systému IS FLORES.

V případě volby archivace importního souboru (viz možnost „Archivovat dokument" v Definici definovatelných importů) vzniklý dokument obsahuje také původní, nerozparsovaná, importní data.

Obsah interní struktury XML pro ukládání dat do IS FLORES je k nahlédnutí na záložce Obsah/podzáložka Obsah, případně v strukturované formě na podzáložce XML.

Zpracování dokumentu spustí vytvoření či aktualizaci business objektů. Zpracování dokumentu lze ručně spustit tlačítkem „Zpracovat".

Každý dokument lze zpracovat pouze jednou.

Pokud proces zpracování dokumentu proběhl bez chyb, na dokumentu se nastaví příznak „Vyřízeno".

Pokud se při zpracovaní dokumentu vyskytne chyba (např. XML soubor neobsahuje povinnou položku pro daný BO – BO tedy nelze vytvořit), text chyby se zapíše do položky „Text chyby" a příznak dokumentu se nastaví na „Přerušeno". O průběhu zpracování se vždy generuje nový záznam logu.

XML vzniklé parsováním lze v dokumentu samozřejmě i ručně upravit (XML si lze uložit, upravit a upravené XML nahrát zpět).

4.2.1 DOPLNĚNÍ HODNOT PRO IMPORT

Pokud má některá datová položka v použité definici nastaven Režim editace položky na hodnotu Vyplňovat jen prázdné nebo Vyplňovat vždy, zobrazí se při ručním použití funkce „Zpracovat" dialogové okno „Doplnění hodnot pro import". Uživatel v něm může před vytvořením nebo aktualizací business objektů upravit hodnoty uložené v interním XML dokumentu.

U každé nabízené položky je uveden business objekt, položka, způsob vyplnění a hodnota. Způsob vyplnění lze pro konkrétní zpracování změnit na Nevyplnit, Vyplnit jen prázdné nebo Vyplnit vždy. Změna se promítne pouze do interního XML zpracovávaného dokumentu, nikoli do definice importu.

Při zpracování více označených dokumentů se zadané hodnoty použijí na všechny zpracovávané dokumenty s příslušnou definicí. Dokumenty, které vyžadují uživatelské zadání podle více různých definic importu, nelze zpracovat společně.

Pomocí tlačítka „Načíst výchozí" lze obnovit hodnoty a způsoby vyplnění podle definice importu. Tlačítkem „Vybrat nastavení" lze načíst dříve uložené pojmenované nastavení a tlačítkem „Uložit nastavení" aktuální nastavení uložit, přepsat nebo smazat.

Pojmenovaná nastavení jsou společná pro firmu a konkrétní definici importu. Mohou je používat všichni uživatelé s přístupem k definici. Ukládat, přepisovat a mazat je mohou uživatelé s oprávněním definici měnit. Tato nastavení se nepřenášejí instalačními sadami; v instalační sadě se přenese pouze Režim editace položky uložený v definici importu.

Dialogové okno se zobrazuje pouze při ručním použití funkce „Zpracovat". Při zpracování naplánovanou úlohou se použijí hodnoty, které jsou již uložené v interním XML; pojmenovaná nastavení dialogu se neuplatní. Pokud dokument nelze bez doplnění hodnot zpracovat, zpracování se přeruší a dokument je nutné dokončit ručně.

4.2.2 OPAKOVANÉ ZPRACOVÁNÍ A PŘEGENEROVÁNÍ

Opakovaným použitím funkce „Zpracovat" se zpracuje interní XML struktura uložená v dokumentu. Zdrojová data z přílohy se znovu neparsují, proto se při opakovaném zpracování samy neprojeví změny provedené v definici importu.

Pokud byla opravena definice importu nebo je potřeba původní zdrojová data naparsovat jinou definicí, použijte na záložce Detail funkci „Přegenerovat DI". Funkce znovu předá zdrojová data definovatelnému importu a interní XML uložené v dokumentu nahradí novým výsledkem. Nový dokument nevzniká.

Funkce je dostupná mimo editační režim pro dokument kategorie s ovladačem Definovatelný import, který je ve stavu Chyba. Uživatel musí mít oprávnění dokument upravovat a dokument musí obsahovat právě jednu přílohu se zdrojovými importními daty. Zdrojový soubor je proto nutné při prvním importu uložit do dokumentu pomocí volby „Archivovat dokument".

Pokud dokument již obsahuje interní XML, použije se definice importu uvedená na dokumentu. Pokud interní XML neobsahuje, uživatel vybere definici ze seznamu. Nabízejí se pouze definice se zatrženou volbou „Pouze parsovat".

Po úspěšném přegenerování přejde dokument do stavu Nezpracováno, zruší se příznaky Přerušeno a Vyřízeno a dokument lze zpracovat běžnou funkcí „Zpracovat". Pokud definice zdrojová data nezpracuje, dokument zůstane beze změny. Pokud zpracování skončí chybou, dokument zůstane ve stavu Chyba a důvod se zapíše do položky „Text chyby".

XML vzniklé parsováním lze v dokumentu také ručně upravit: XML lze uložit do souboru, upravit a poté nahrát zpět.

5. PODPORA SKRIPTINGU

Do balíčků skriptů IS FLORES je zveřejněna třída BO "TNxIEImportDefinition" se speciální metodou určenou ke spuštění parsingu definovatelného importu "ProcessImport".

Do balíčků skriptů IS FLORES je přidán nový aplikační modul "Definovatelný import". V modulu jsou přístupné nové skriptingové háčky umožňující ovlivnit způsob zpracování importů:

IEImportExport_AfterFinishedCompleteInternalXML_Hook:
Vyvolává se po sestavení interního XML - umožňuje změnit obsah interního XML.

IEImportExport_AfterEvaluateItemImportCondition_Hook:
Vyvolává se po vyhodnocení podmínky pro import položky - umožňuje změnit výsledek vyhodnocení podmínky.

IEImportExport_AfterParseItem_Hook:
Vyvolává se po parsingu položky z importního dokumentu - lze využít pro změnu hodnoty položky.

IEImportExport_AfterSetImportDocument_Hook:
Vyvolává se po nastavení importního dokumentu před spuštěním parsingu - umožňuje změnit obsah importního dokumentu.

BeforeSaveImportedBO_Hook:

Vyvolá se před uložením každého business objektu připraveného zpracováním definice definovatelných importů samotným definovatelným importem. V háčku je možné předaný business objekt v parametru háčku "AImported_BO" libovolně upravovat. Parametr háčku "AIsSavedByScriptingHook" určuje, jestli jde pouze o úpravu business objektu a uložení objektu provede definovatelný import (hodnota parametru False) nebo jestli je vše řešeno přímo skriptem a objekt již definovatelný import ukládat nebude (hodnota parametru True).

Zajímavým příkladem využití háčku BeforeSaveImportedBO_Hook je použití importního manažeru pro uložení výsledného business objektu.

V přiloženém příkladu je demonstrováno vytvoření dokladu dobropisu vydaného importním manažerem na základě dat získaných z business objektu připraveného definovatelným importem.

V příkladu jsou importní data "Importni data.csv". Importní data jsou společná pro obě definice definovatelných importů.

Dále je v příkladu balíček skriptů "TEST_BeforeSaveImportedBO_Hook" a dvě importní definice "_Review-01_TEST BeforeSave skript FV" a "_Review-02_TEST BeforeSave skript DV".

Pomocí definovatelného importu "_Review-01_TEST BeforeSave skript FV" dojde k založení dvou dokladů faktur vydaných.

Zajímavé je zpracování definovatelného importu "_Review-02_TEST BeforeSave skript DV". V definici je nadefinován business objekt dobropisu vydaného, který ale není možné samostatně uložit. Dobropis vydaný se musí ukládat vždy pomocí importního manažeru. V definici tedy jde jen o přípravu dat objektu, vyplnění zejména vazebních položek. Při zpracování importu je takto připravený a dosud neuložený objekt předaný do zpracování skriptingovým háčkem BeforeSaveImportedBO_Hook . V háčku se převezmou data z předaného objektu do importního manažeru pro tvorbu dobropisů vydaných podle faktur vydaných. Importní manažer se získanými daty vyplní a dojde k uložení dokladu dobropisu vydaného importním manažerem. Objekt připravený definovatelným importem se následně zahodí (AIsSavedByScriptingHook je nastaveno na True).

Výsledkem příkladu je hromadný doklad dobropisu vydaného, který čerpá obě faktury vydané založené prvním importem a čerpá jen některé řádky faktur.

funkce dostupná na třídě "TNxIEImportDefinition" s názvem "ConvertOnlyRowsTXTData"

Práce s definovatelnými importy: Definovatelné importy (txt, xls a csv) vyžadují, aby v importním souboru byla rozlišena hlavička a řádky pomocí masky. Pokud tomu tak není, jsou jednotlivé řádky ze souboru importovány jako samostatné doklady.

Pro případ, kdy jsou v importních datech POUZE řádkové položky VÍCE než JEDNÉ hlavičky a zároveň jde o řádky jen JEDNOHO druhu business objektu, byly vytvořeny nové skritptingové funkce dostupné na třídě "TNxIEImportDefinition".

- Soubor typu txt:

  Funkce ConvertOnlyRowsTXTData, metoda ConvertOnlyRowsTXTData, procedure ConvertOnlyRowsTXTData(var ATXTInputData: TStringList; const AKeyPositions: TStringList; const AHeadersPrefix, ARowsPrefix: string; ATrimSpaces: Boolean).

  Metoda provede konverzi vstupních, pouze řádkových txt, dat do formátu, který lze zpracovat definovatelným importem.

  Data seřadí do skupin podle vstupního klíče definovaného txt pozicemi a délkami a ke skupinám přidá hlavičkový řádek.

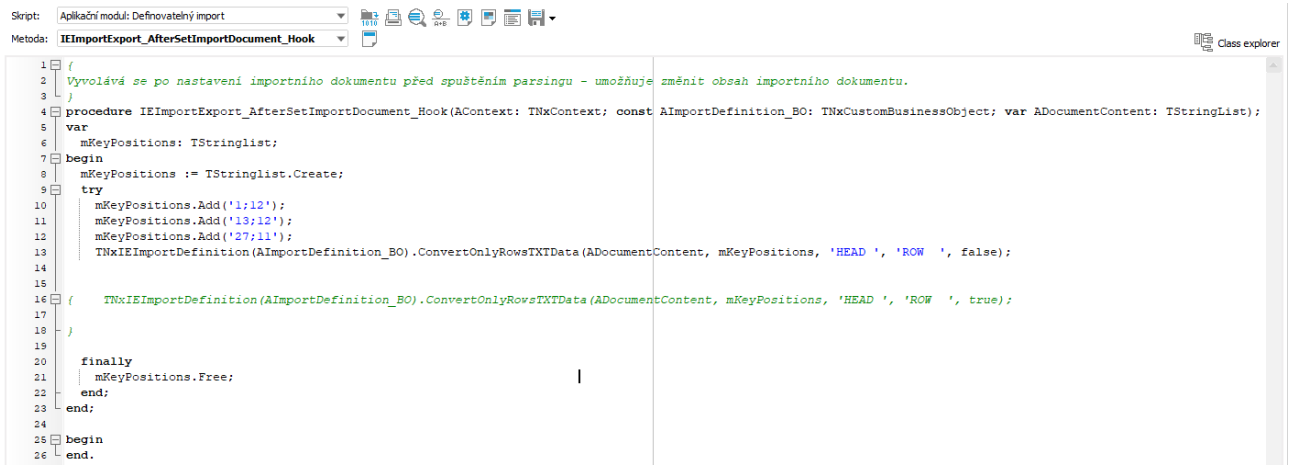
  Hlavičkám i řádkům nastaví předané prefixy.

  Metoda je dostupná pouze pro importní definice s přesně dvěma business objekty, přičemž jeden objekt je hlavičkou druhého objektu.

  Je možné zvolit, jestli se u položek mají ořezávat mezery.

Použití funkce se očekává ve skriptingovém háčku definovatelných importů "IEImportExport_AfterSetImportDocument_Hook".

Příklad:



- Soubor typu csv a xls (soubor xls se při importu převede na csv formát):

  Funkce ConvertOnlyRowsCSVData, metoda ConvertOnlyRowsCSVData, procedure ConvertOnlyRowsCSVData(var ACSVInputData: TStringList; const AKeyPositions: TStringList; const AHeadersPrefix, ARowsPrefix: string).

  Metoda provede konverzi vstupních, pouze řádkových csv dat do formátu, který lze zpracovat definovatelným importem.

  Data seřadí do skupin podle vstupního klíče definovaného csv pozicemi a ke skupinám přidá hlavičkový řádek.

  Hlavičkám i řádkům nastaví předané prefixy.

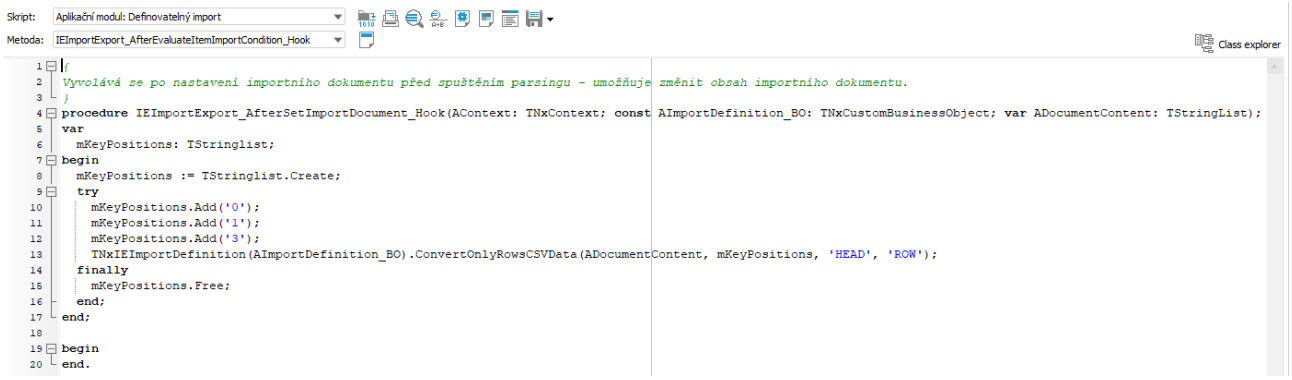
  Metoda je dostupná pouze pro importní definice s přesně dvěma business objekty, přičemž jeden objekt je hlavičkou druhého objektu.

  Metoda nepodporuje csv text přes více řádků.

-

  Použití funkce se očekává ve skriptingovém háčku definovatelných importů "IEImportExport_AfterSetImportDocument_Hook".

  Příklad:



6. PODROBNÝ POPIS JEDNOTLIVÝCH TYPŮ DEFINOVATELNÝCH IMPORTŮ S PŘÍKLADY

6.1 OBECNÉ PRINCIPY

Importní data jsou definovatelným importem zpracovávána vždy v pořadí, v jakém jsou zadána v importní definici. Je to nezbytné, protože v systému IS FLORES záleží na pořadí zadávání položek business objektů. Pokud při práci s jednotlivými business objekty dojde k chybě, proces zpracování se nepřeruší a pokračuje se zpracováním dalšího business objektu. O chybě je uložena informace do logu. Log se vytváří a ukládá vždy, tj. jak v případě zpracování bez chyb, tak v případě zpracování s chybami. Veškeré údaje vztažené ke stavu databáze se vyhodnocují v okamžiku parsování importních dat.

6.2 TYP IMPORTU CSV A TXT

Typ importu TXT:
Jedná se o import dat v textovém formátu. Položky pro import se v importních datech definují pozicí a délkou.

Typ importu CSV:
Jedná se o import dat v textovém formátu. Položky pro import se v importních datech definují oddělovačem a pořadím položky v rámci jednoho řádku.

POZOR
Pro správné importování diakritiky při použitém formátování UTF 8 je potřeba využít volbu **UTF 8 s BOM**.

Importy TXT a CSV se definovatelným importem zpracovávají obdobně. Na příkladech je dále popisován jen typ CSV. Použití pro TXT je stejné.

6.2.1 PŘÍKLADY:

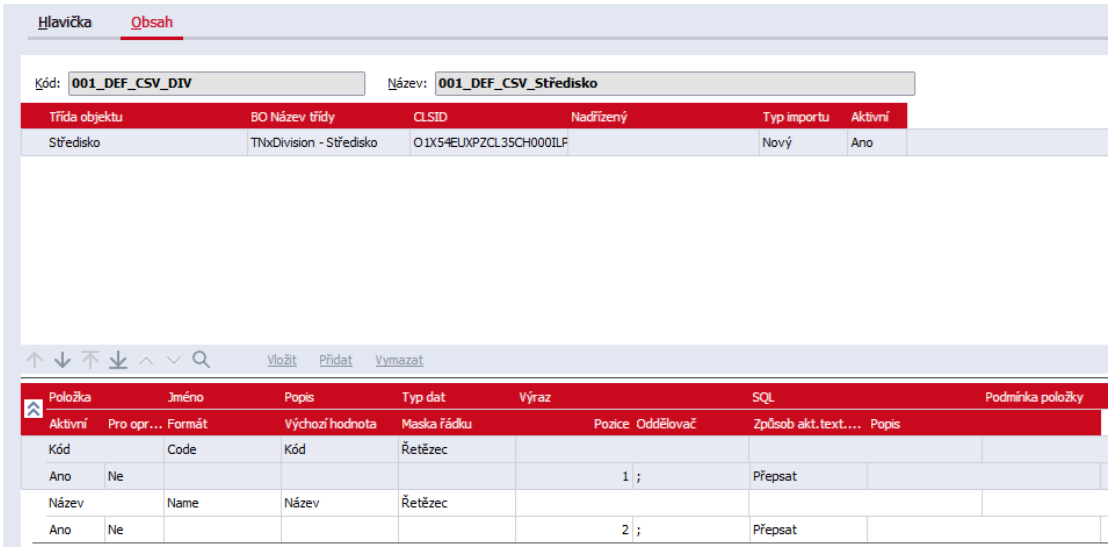
POZOR
Všechny příklady jsou připraveny pro testovaní nad demodaty (demodata je možné vytvořit jako další databázové spojení). V případě, že budete chtít příklady otestovat na vašich datech, je potřeba změnit data v příkladových importních souborech na data z vaší databáze.

Součástí příkladů jsou i vzorové definice importů v XML a importní soubory v CSV. Definice i soubory ke všem příkladům jsou ke stažení v jednom ZIP souboru zde.

1) Příklad importu nového střediska (nových BO bez kolekcí):
Import nových středisek, vyplňuje se kód a název. Data jsou v importním souboru vždy uvedena za sebou.

Příklad importních dat:
DIV01;Provozovna 01
DIV02;Provozovna 02

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 0, typ importu = CSV



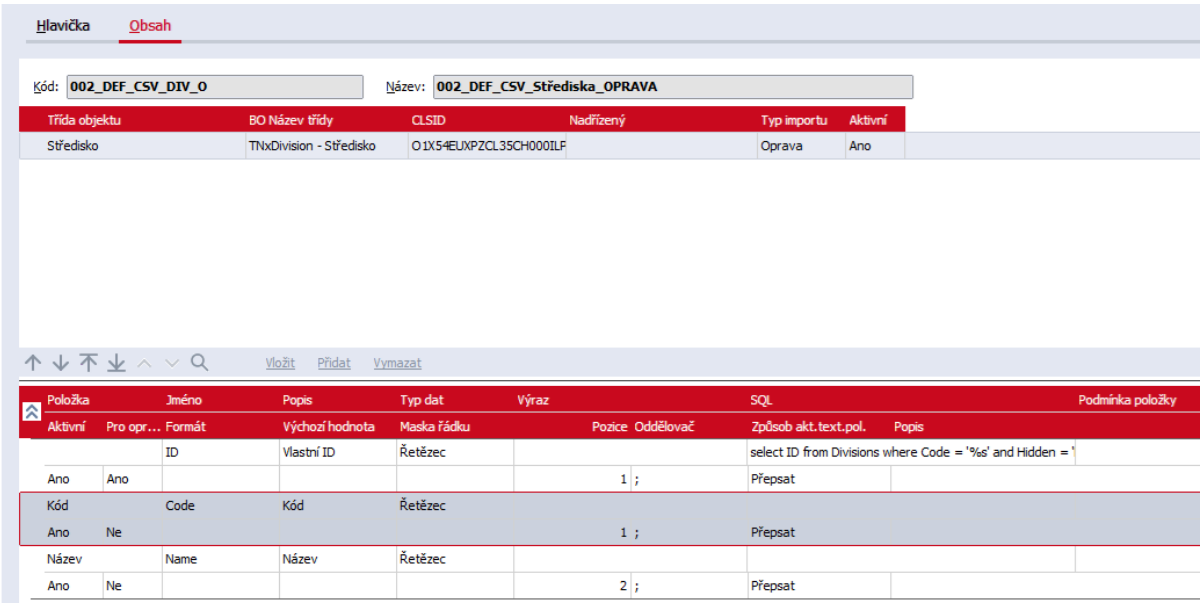
Výsledkem zpracování importních dat touto definicí definovatelného importu jsou dva nové záznamy středisek.

Přílohy:
Importní data: 001_Soubor_Středisko.csv
Definice definovatelných importů: 001_DEF_CSV_Středisko.xml

2) Příklad opravy existujících středisek (opravy BO bez kolekcí):
Opravy již existujících středisek. Záznam střediska pro opravu se dohledává podle kódu. Aktualizuje se kód a název. Data jsou v importním souboru vždy uvedena za sebou.

Příklad importních dat:
DIV01;Provozovna V Údolí
DIV02;Provozovna Na potoce

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 0, typ importu = CSV



Dohledání střediska je řešeno pomocí řádku s nastaveným příznakem „pro opravu" a SQL výrazem.

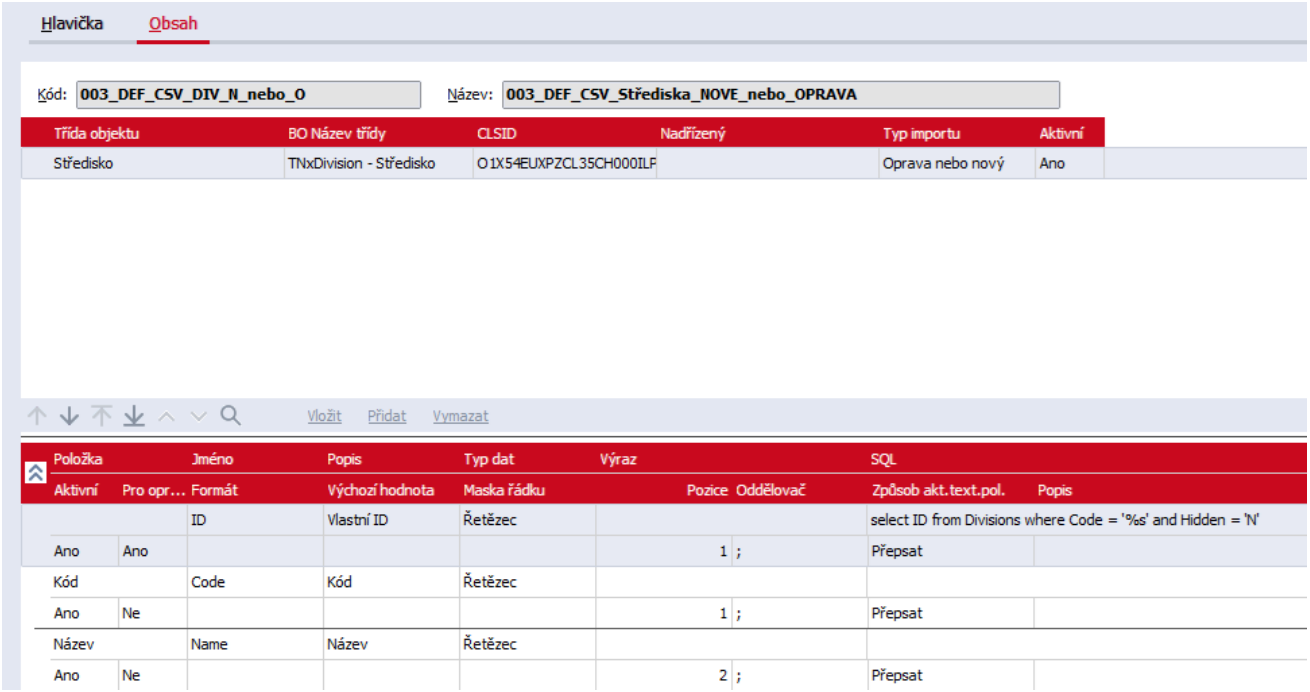
Výsledkem zpracování importních dat touto definicí definovatelného importu jsou aktualizované záznamy středisek. Pokud se záznam střediska s daným kódem nedohledá, záznam se pro aktualizaci ignoruje.

Přílohy:
Importní data: 002_Soubor_Střediska_OPRAVA.csv
Definice definovatelného importu: 002_DEF_CSV_Střediska_OPRAVA.xml

3) Příklad opravy střediska nebo založení nového v případě, kdy neexistuje (oprava nebo založení nového BO bez kolekcí:
Opravy již existujících středisek nebo založení nových záznamů středisek. Záznam střediska pro opravu se dohledává podle kódu. Pokud se záznam pro opravu nedohledá, založí se záznam nový. V novém nebo opravovaném záznamu středisek se zapisuje kód a název. Data jsou v importním souboru vždy uvedena za sebou.

Příklad importních dat:
DIV01;Provozovna 01
DIV02;Provozovna 02

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 0, typ importu = CSV



Dohledání střediska je řešeno pomocí řádku s nastaveným příznakem „pro opravu" a SQL výrazem.

Výsledkem zpracování importních dat touto definicí definovatelného importu jsou buď aktualizované záznamy středisek nebo záznamy nové. Pokud se záznam střediska s daným kódem nedohledá, založí se záznam nový.

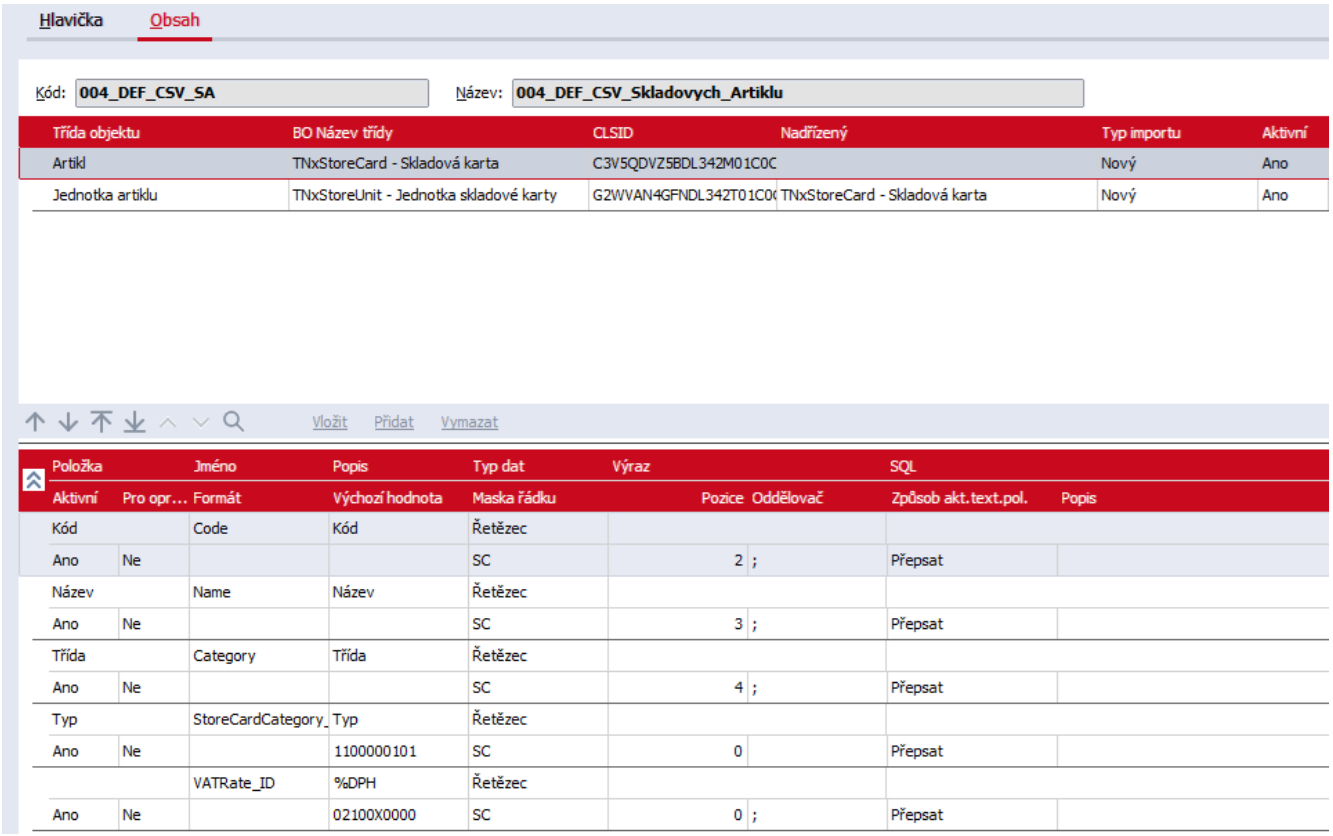
Přílohy:
Importní data: 003_Soubor_Střediska_NOVE_nebo_OPRAVA.csv
Definice definovatelného importu: 003_DEF_CSV_Střediska_NOVE_nebo_OPRAVA.xml

4) Příklad importu nových artiklů včetně jednotek a EANů (import BO s kolekcemi včetně subkolekce):
Import nových artiklů s novými jednotkami a EANy jednotek. Data jsou v importním souboru vždy uvedena za sebou.

Příklad importních dat:
Maska řádku;Kod artiklu;Název artiklu;Třída artiklu
Maska řádku;Kod existující jednotky;EAN
SC;OIE/4001/CSV;Karafiat4;0
UNIT;bal;11220401
SC;OIE/4002/CSV;Petrklic4;1
UNIT;m;11220402
UNIT;bal;11220403
SC;OIE/4003/CSV;Sedmikraska4;2
SC;OIE/4301/CSV;Balici papir;4
UNIT;m;112301
SC;OIE/4302/CSV;Plastovy box;4
UNIT;pal;112302
UNIT;bal;112303
SC;OIE/4303/CSV;Krabice;4

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 2, typ importu = CSV

Definice BO StoreCards:

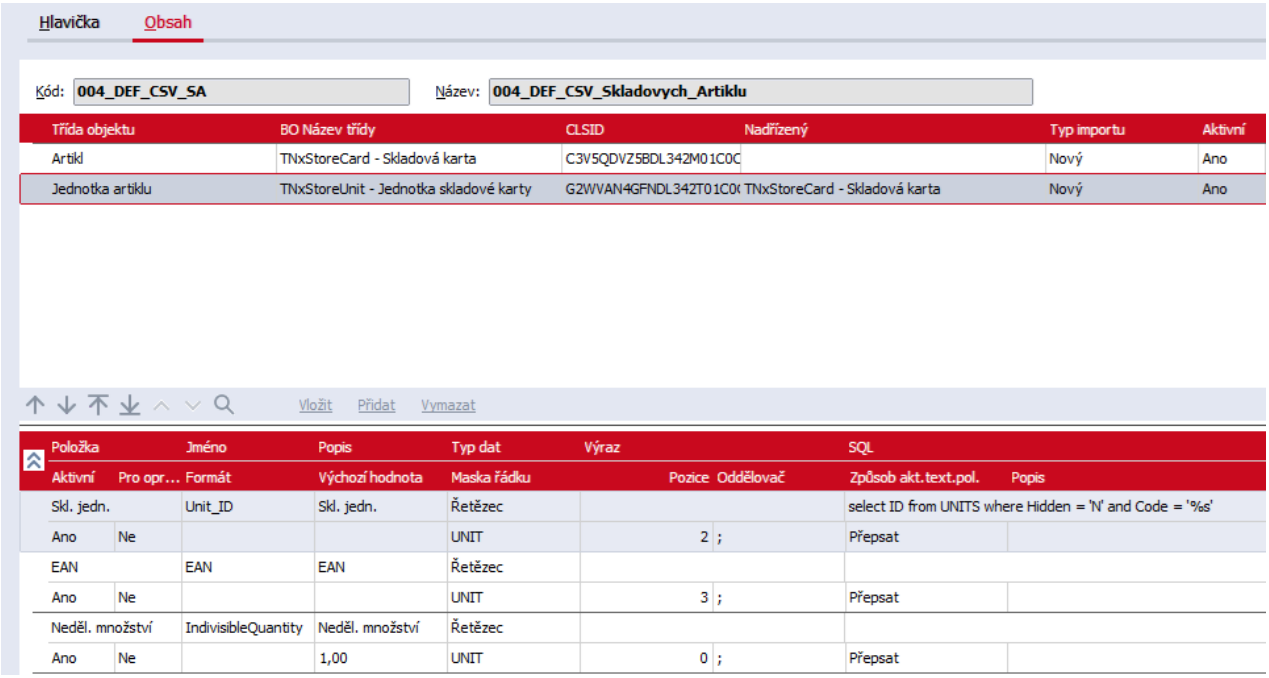


Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano

Položka		Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Kód		Code	Kód	Řetězec				
Ano	Ne			SC	2	;	Přepsat	
Název		Name	Název	Řetězec				
Ano	Ne			SC	3	;	Přepsat	
Třída		Category	Třída	Řetězec				
Ano	Ne			SC	4	;	Přepsat	
Typ		StoreCardCategory_	Typ	Řetězec				
Ano	Ne		1100000101	SC	0		Přepsat	
		VATRate_ID	%DPH	Řetězec				
Ano	Ne		02100X0000	SC	0	;	Přepsat	

Pro povinné položky, které nejsou k dispozici v importních datech jsou použity výchozí hodnoty. Pro identifikaci řádku v importních datech, který odpovídá danému BO definice je využita maska řádku.

Definice BO StoreUnit:



Není nutné nic jiného. Jednotky artiklu jsou kolekcí hlavičky, tedy artiklů. Nadřízená artiklu může být jen jedna, není tedy nutné specifikovat nic dalšího. Pro identifikaci řádku v importních datech, který odpovídá danému BO definice je využita maska řádku.

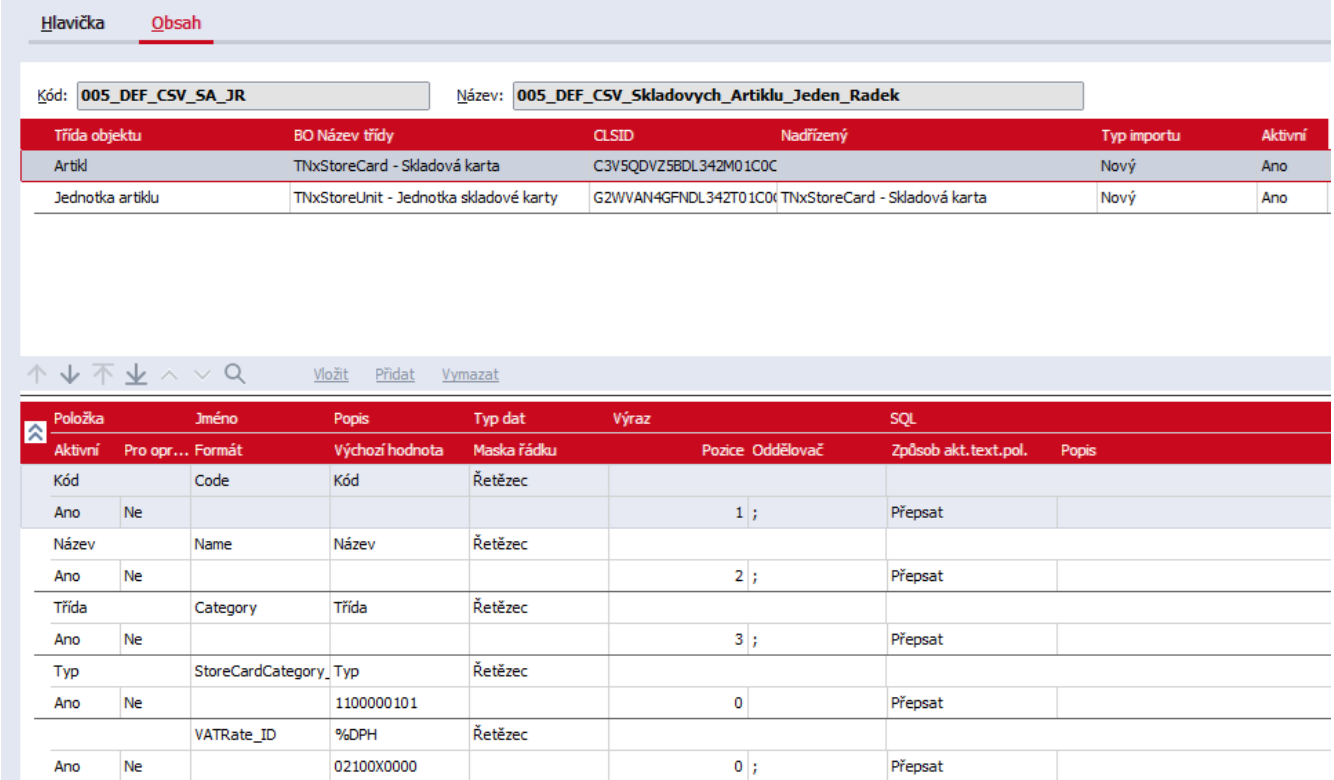
Přílohy:
Importní data: 004_Soubor_Skladove_Artikly.csv
Definice definovatelného importu: 004_DEF_CSV_Skladove_Artikly.xml

5) Příklad importu nových artiklů včetně jednotek a EANů v jednom řádku (import BO s kolekcemi včetně subkolekce v jednom řádku):
Import nových artiklů s novými jednotkami a EANy jednotek. Data jsou v importním souboru v jednom řádku, čímž se tento příklad liší od předchozího příkladu.

Příklad importních dat:
Kod artiklu;Název artiklu;Třída artiklu;Kod existující jednotky;EAN
OIE/5004/CSV;Tulipan5;0;bal;11220504
OIE/5005/CSV;Fialka5;1;m;11220505
OIE/5006/CSV;Narcis5;2;kg;11220506

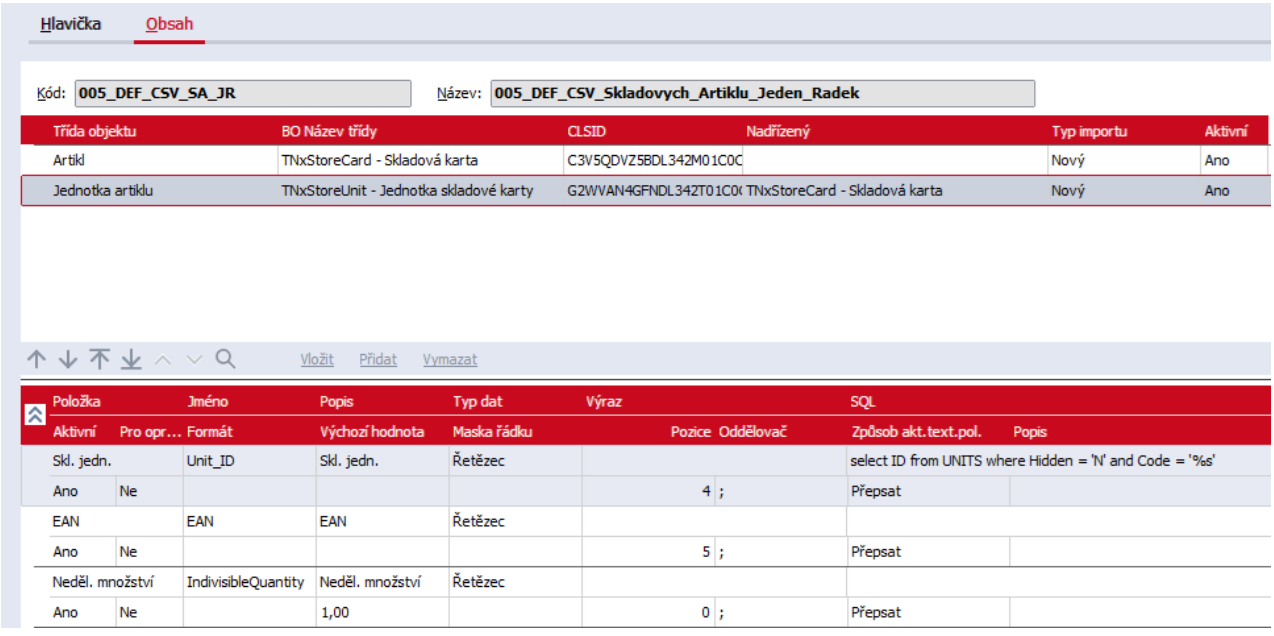
Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 1, typ importu = CSV

Definice BO StoreCards:



Pro povinné položky, které nejsou k dispozici v importních datech jsou použity výchozí hodnoty. Tam kde jsou použity výchozí hodnoty musí být vždy pozice nastavena na neexistující, v našem případě 0.

Definice BO StoreUnit:



Jednotka se dohledá pomocí SQL dotazu. Pro EAN, který je na jednotce není třeba nic dalšího nastavovat. V našem příkladu je konstantou nastaveno nedělitelné množství.

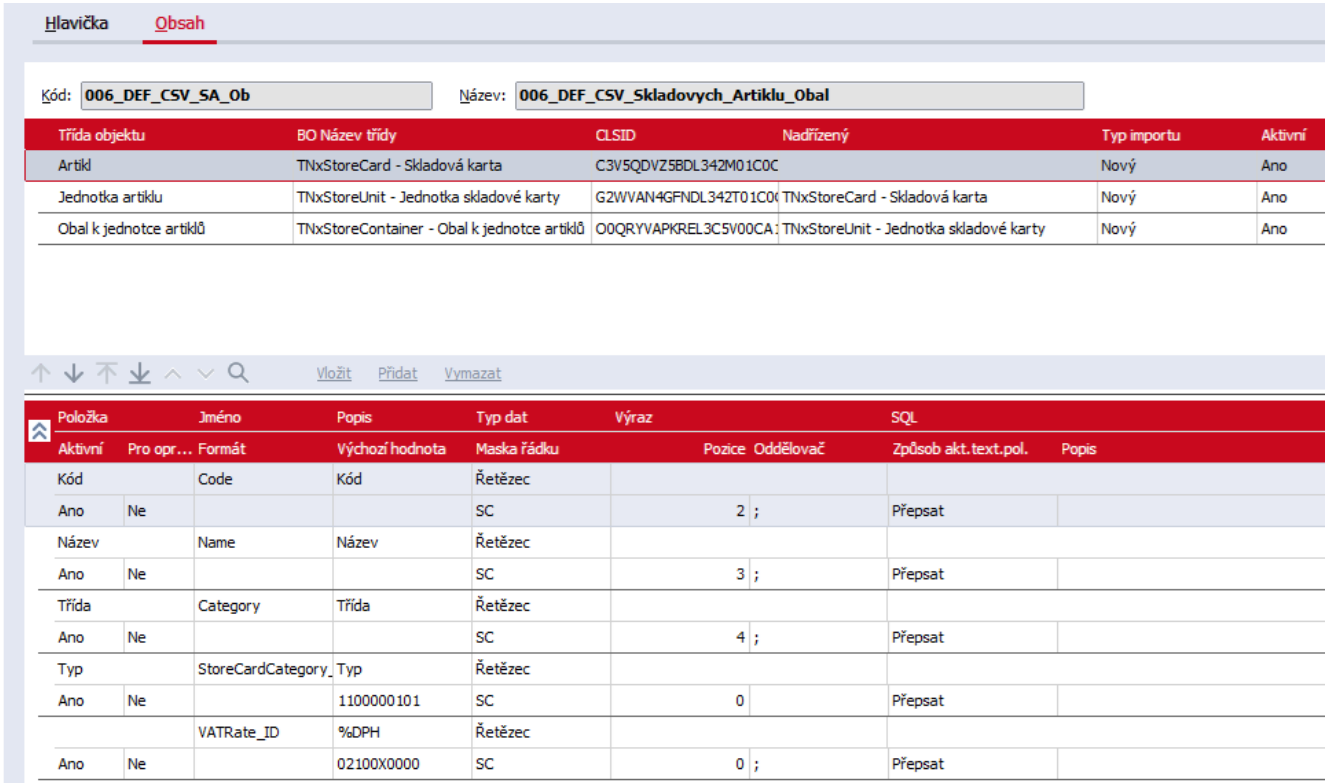
Přílohy:
Importní data: 004_Soubor_Skladove_Artikly.csv
Definice definovatelného importu: 005_DEF_CSV_Skladove_Artikly_Jeden_Rade.xml

6) Příklad importu nových artiklů včetně jednotek, EANů a obalů (import BO s kolekcemi včetně subkolekce):
Import nových artiklů s novými jednotkami, EANy jednotek a obaly. Data vždy začínají maskou řádku, která je nezbytná pro správné rozpoznání jednotlivých objektů.

Příklad importních dat:
ARTIKL: Maska řádku;Kod artiklu;Název artiklu;Třída artiklu
JEDNOTKA ARTIKLU: Maska řádku;Kod existující jednotky;EAN
OBAL ARTIKLU: Maska řádku;Zkratka jednotky;Kod obalu;Množství
SC;OIE/6001/CSV;Serik6;0
UNIT;bal;11220601
CONT;bal;OIE/4301/CSV;2,00
SC;OIE/6002/CSV;Kopretina6;1
UNIT;m;11220602
UNIT;bal;11220603
CONT;m;OIE/4302/CSV;5,00
CONT;bal;OIE/4303/CSV;1000,00
SC;OIE/6003/CSV;Sedmikraska6;2

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 3, typ importu = CSV

Definice BO StoreCards:


Hlavička Obsah

Kód: 006_DEF_CSV_SA_Ob Název: 006_DEF_CSV_Skladovych_Artiklu_Obal

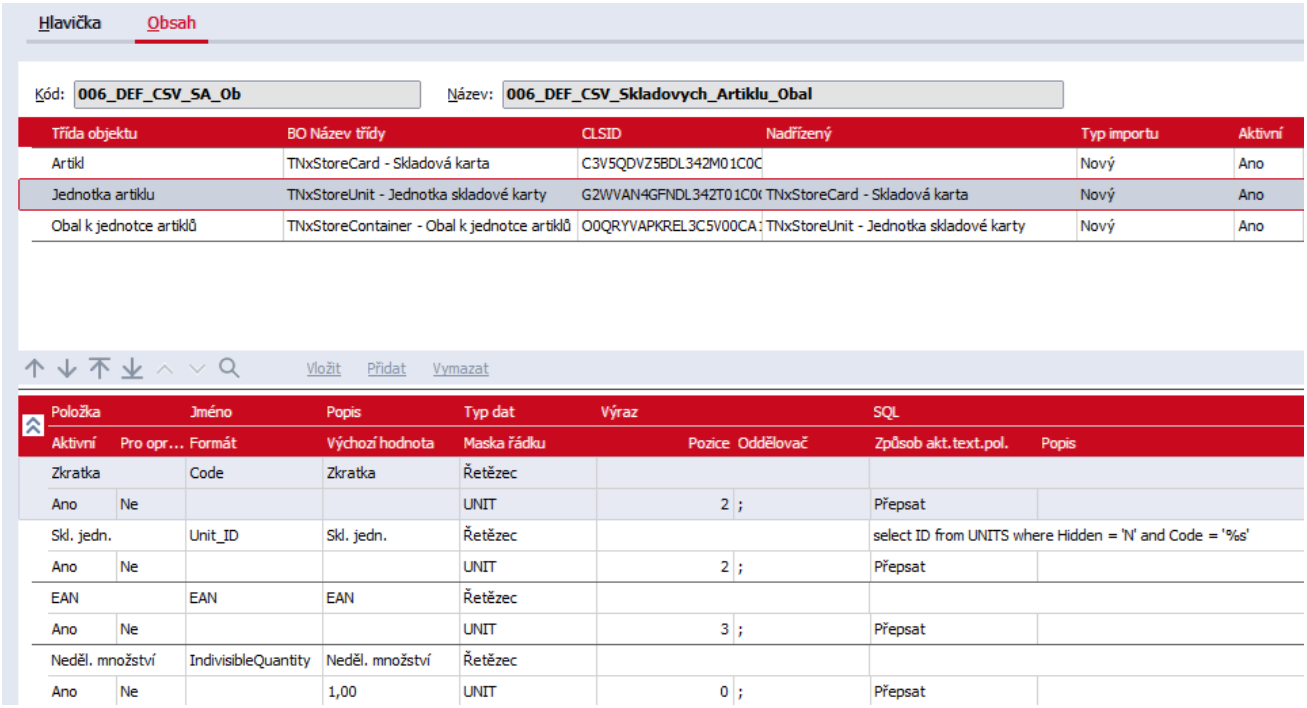
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA:	TNxStoreUnit - Jednotka skladové karty	Nový	Ano

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr... Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Kód	Code	Kód	Řetězec				
Ano	Ne		SC	2	;	Přepsat	
Název	Name	Název	Řetězec				
Ano	Ne		SC	3	;	Přepsat	
Třída	Category	Třída	Řetězec				
Ano	Ne		SC	4	;	Přepsat	
Typ	StoreCardCategory_	Typ	Řetězec				
Ano	Ne	1100000101	SC	0		Přepsat	
	VATRate_ID	%DPH	Řetězec				
Ano	Ne	02100X0000	SC	0	;	Přepsat	

Pro povinné položky, které nejsou k dispozici v importních datech jsou použity výchozí hodnoty. Tam kde jsou použity výchozí hodnoty musí být vždy pozice nastavena na neexistující, v našem případě 0.

Definice BO StoreUnit:


Hlavička Obsah

Kód: 006_DEF_CSV_SA_Ob Název: 006_DEF_CSV_Skladovych_Artiklu_Obal

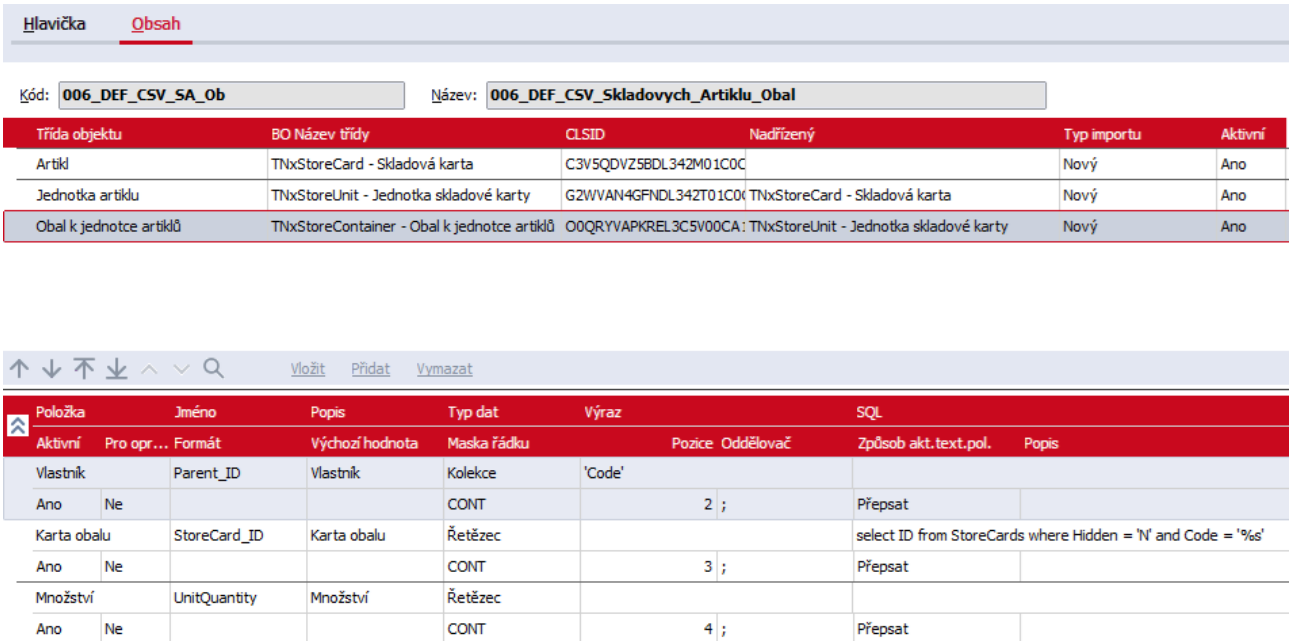
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA:	TNxStoreUnit - Jednotka skladové karty	Nový	Ano

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr... Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Zkratka	Code	Zkratka	Řetězec				
Ano	Ne		UNIT	2	;	Přepsat	
Skl. jedn.	Unit_ID	Skl. jedn.	Řetězec			select ID from UNITS where Hidden = 'N' and Code = '%s'	
Ano	Ne		UNIT	2	;	Přepsat	
EAN	EAN	EAN	Řetězec				
Ano	Ne		UNIT	3	;	Přepsat	
Neděl. množství	IndivisibleQuantity	Neděl. množství	Řetězec				
Ano	Ne	1,00	UNIT	0	;	Přepsat	

Jednotka se dohledá pomocí SQL dotazu. Pro EAN, který je na jednotce není třeba nic dalšího nastavovat. V našem příkladu je konstantou nastaveno nedělitelné množství.

Definice BO StoreContainer:


Hlavička Obsah

Kód: 006_DEF_CSV_SA_Ob Název: 006_DEF_CSV_Skladovych_Artiklu_Obal

Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA:	TNxStoreUnit - Jednotka skladové karty	Nový	Ano

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr... Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Vlastník	Parent_ID	Vlastník	Kolekce	'Code'			
Ano	Ne		CONT	2	;	Přepsat	
Karta obalu	StoreCard_ID	Karta obalu	Řetězec			select ID from StoreCards where Hidden = 'N' and Code = '%s'	
Ano	Ne		CONT	3	;	Přepsat	
Množství	UnitQuantity	Množství	Řetězec				
Ano	Ne		CONT	4	;	Přepsat	

Parent_ID, tedy nadřazená jednotka obalu se dohledává pomocí kódu a výrazu ‚Code'. Karta obalu se pak dohledává za pomocí SQL dotazu do tabulky artiklů.

Přílohy:
Importní data: 006_Soubor_Skladove_Artikly_Obal.csv
Definice definovatelného importu: 006_DEF_CSV_Skladove_Artikly_Obal.xml

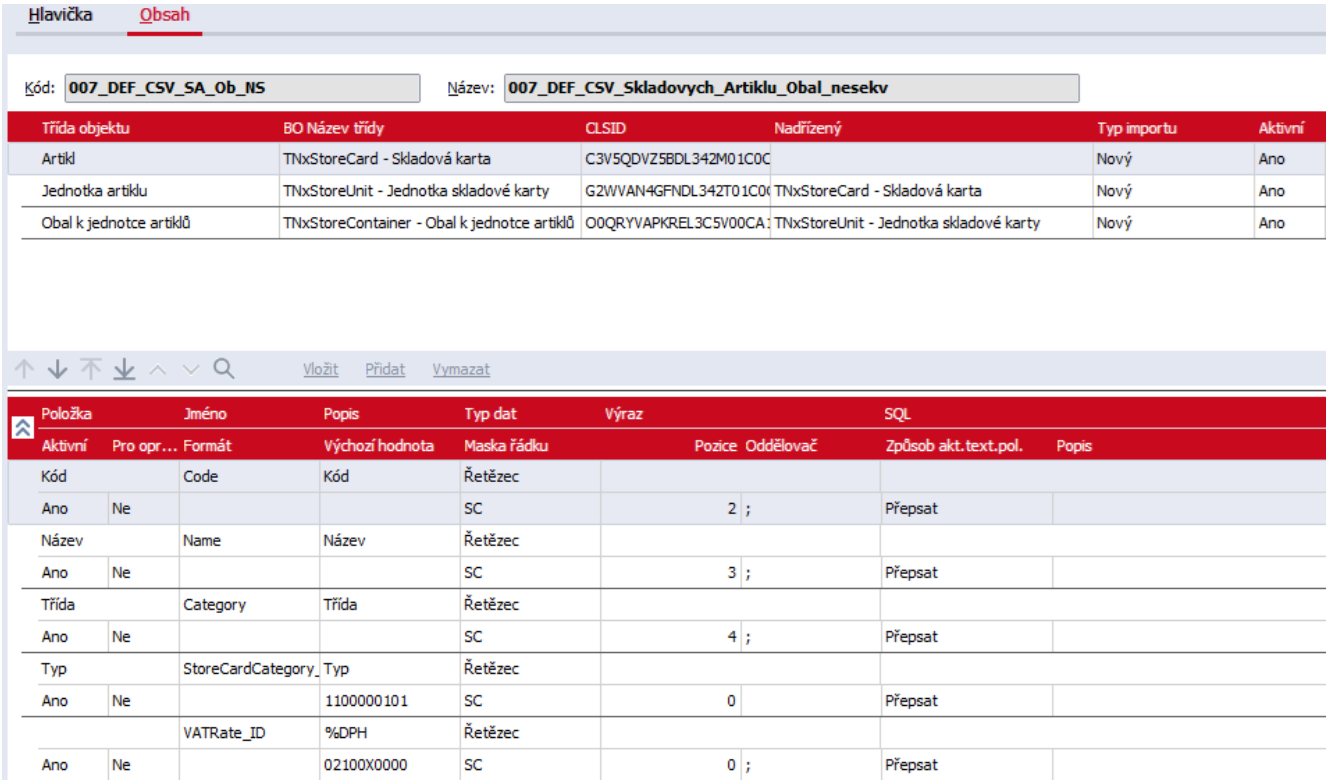
7) Příklad importu nových artiklů včetně jednotek, EANů a obalů nesekvenčně (import BO s kolekcemi včetně subkolekce):

Import nových artiklů s novými jednotkami, EANy jednotek a obaly nesekvenčně. V případě nesekvenčního importu lze mít v datech více hlavičkových BO a k nim kolekce a subkolekce, ale vždy jen o jednom záznamu. Např. jeden artikl a k němu jednotky, EANy a obaly, jedno středisko atd. Data vždy začínají maskou řádku, která je nezbytná pro správné rozpoznání jednotlivých objektů.
Definice tohoto příkladu je nastavená zcela identicky jako v předchozím příkladu.

Příklad importních dat:
ARTIKL: Maska řádku;Kod artiklu;Název artiklu;Třída artiklu
JEDNOTKA ARTIKLU: Maska řádku;Kod existující jednotky;EAN
OBAL ARTIKLU: Maska řádku;Kod obalu;Množství
SC;OIE/7008/CSV;Kopretina7;1;
UNIT;m;11220702;
CONT;m;OIE/4302/CSV;5,00;
UNIT;bal;11220703;
CONT;bal;OIE/4303/CSV;1000,00;

Definice definovatelného importu:
Hlavička: sekvenčně = NE (volba nebude zatržena), řádky pro přeskočení = 3, typ importu = CSV

Definice BO StoreCards:


Hlavička Obsah

Kód: 007_DEF_CSV_SA_Ob_NS Název: 007_DEF_CSV_Skladovych_Artiklu_Obal_nesekv

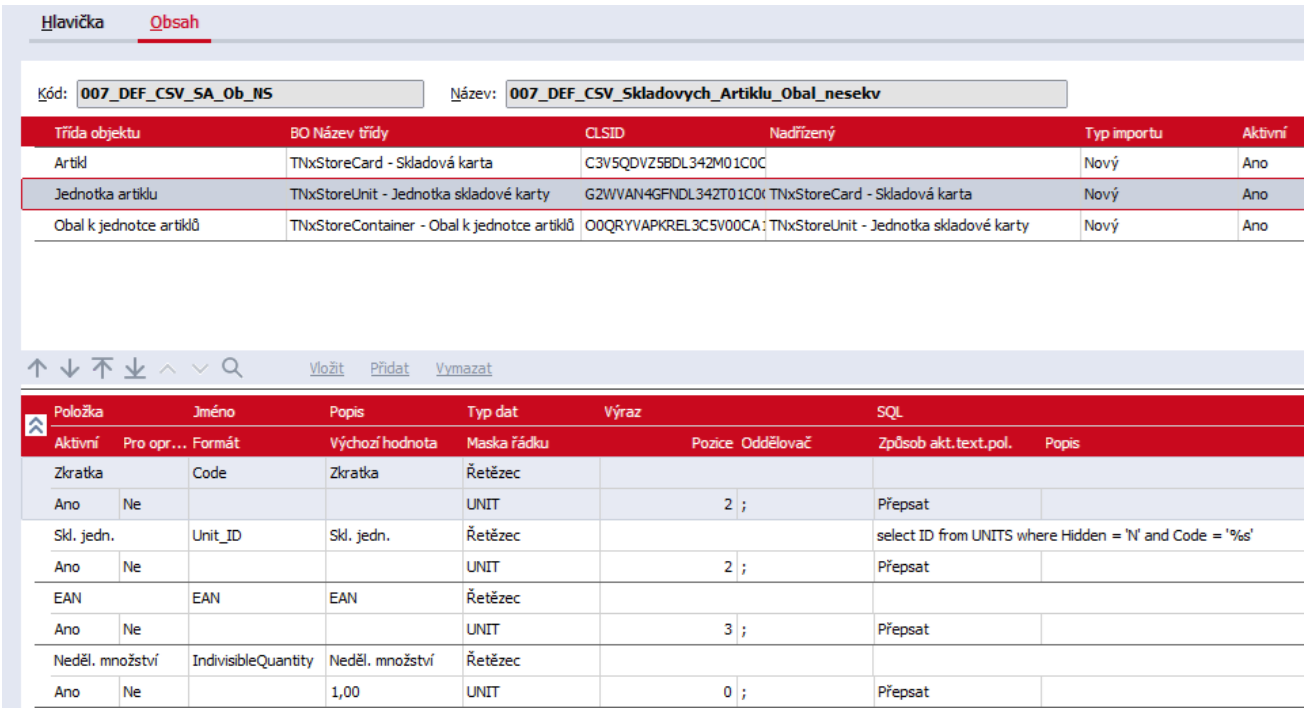
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA1	TNxStoreUnit - Jednotka skladové karty	Nový	Ano

Vložit Přidat Vymazat

Položka		Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Kód		Code	Kód	Řetězec				
Ano	Ne			SC	2	;	Přepsat	
Název		Name	Název	Řetězec				
Ano	Ne			SC	3	;	Přepsat	
Třída		Category	Třída	Řetězec				
Ano	Ne			SC	4	;	Přepsat	
Typ		StoreCardCategory_	Typ	Řetězec				
Ano	Ne		1100000101	SC	0		Přepsat	
		VATRate_ID	%DPH	Řetězec				
Ano	Ne		02100X0000	SC	0	;	Přepsat	

Pro povinné položky, které nejsou k dispozici v importních datech jsou použity výchozí hodnoty. Tam kde jsou použity výchozí hodnoty musí být vždy pozice nastavena na neexistující, v našem případě 0.

Definice BO StoreUnit:



Hlavička | Obsah

Kód: 007_DEF_CSV_SA_Ob_NS Název: 007_DEF_CSV_Skladovych_Artiklu_Obal_nesekv

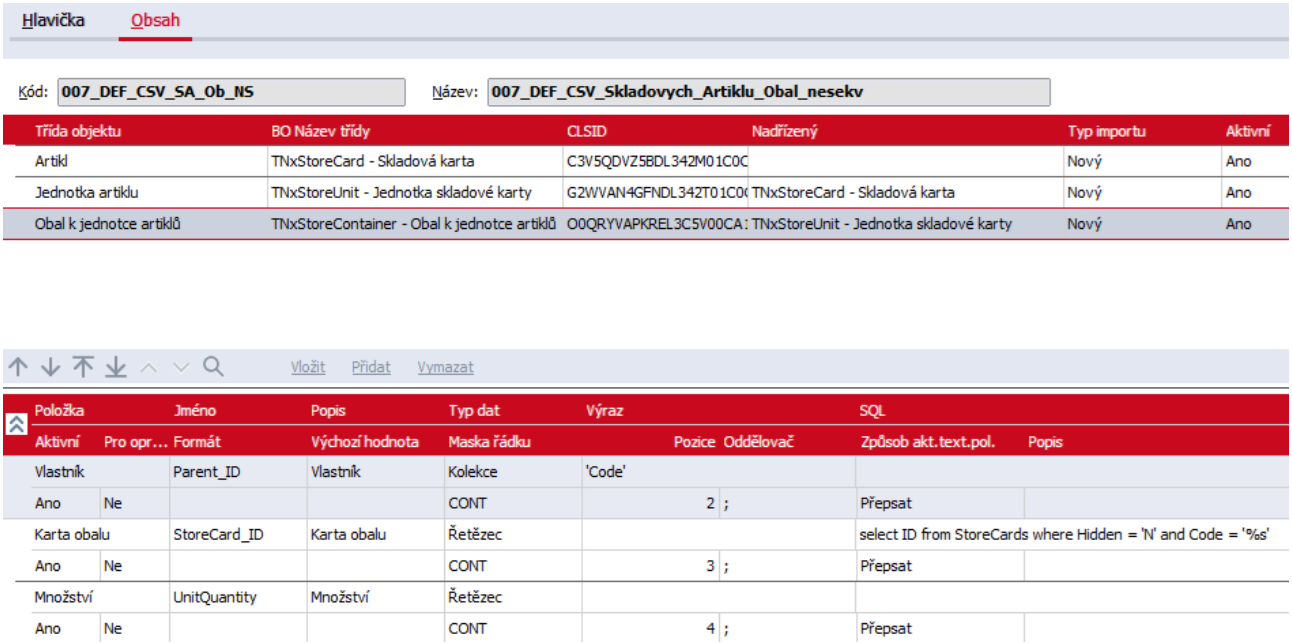
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA	TNxStoreUnit - Jednotka skladové karty	Nový	Ano

Vložit | Přidat | Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL			
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Zkratka		Code	Zkratka	Řetězec				
Ano	Ne			UNIT	2	;	Přepsat	
Skl. jedn.		Unit_ID	Skl. jedn.	Řetězec			select ID from UNITS where Hidden = 'N' and Code = '%s'	
Ano	Ne			UNIT	2	;	Přepsat	
EAN		EAN	EAN	Řetězec				
Ano	Ne			UNIT	3	;	Přepsat	
Neděl. množství		IndivisibleQuantity	Neděl. množství	Řetězec				
Ano	Ne		1,00	UNIT	0	;	Přepsat	

Jednotka se dohledá pomocí SQL dotazu. Pro EAN, který je na jednotce není třeba nic dalšího nastavovat. V našem příkladu je konstantou nastaveno nedělitelné množství.

Definice BO StoreContainer:



Hlavička | Obsah

Kód: 007_DEF_CSV_SA_Ob_NS Název: 007_DEF_CSV_Skladovych_Artiklu_Obal_nesekv

Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA	TNxStoreUnit - Jednotka skladové karty	Nový	Ano

Vložit | Přidat | Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL			
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Vlastník		Parent_ID	Vlastník	Kolekce	'Code'			
Ano	Ne			CONT	2	;	Přepsat	
Karta obalu		StoreCard_ID	Karta obalu	Řetězec			select ID from StoreCards where Hidden = 'N' and Code = '%s'	
Ano	Ne			CONT	3	;	Přepsat	
Množství		UnitQuantity	Množství	Řetězec				
Ano	Ne			CONT	4	;	Přepsat	

Parent_ID, tedy nadřazená jednotka obalu se dohledává pomocí kódu a výrazu ‚Code'. Karta obalu se pak dohledává za pomocí SQL dotazu do tabulky artiklů.

Přílohy:
Importní data: 007_Soubor_Skladove_Artikly_Obal_nesekv.csv
Definice definovatelného importu: 007_DEF_CSV_Skladove_Artikly_Obal_nesek.xml

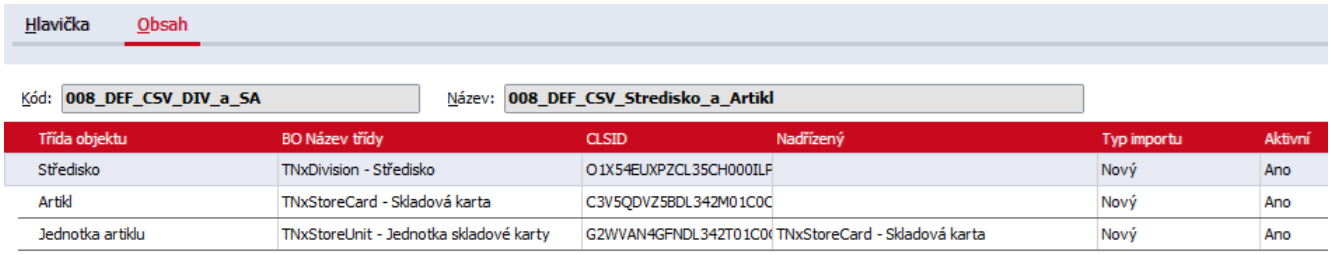
8) Příklad importu nových artiklů včetně jednotek, EANů a obalů a nových středisek (import BO s kolekcemi včetně subkolekce):

Import nových artiklů s novými jednotkami, EANy jednotek a obaly a nových středisek. V jednom importu mohou být různá data. Tento příklad ukazuje kombinaci artiklu a střediska. Data vždy začínají maskou řádku, která je nezbytná pro správné rozpoznání jednotlivých objektů.

Příklad importních dat:
DIV: Maska řádku;Kod střediska;Název střediska
SC: Maska řádku;Kod artiklu;Název artiklu;Třída artiklu
UNIT: Maska řádku;Kod existující jednotky;EAN
DIV;DIV03;Trziste
DIV;DIV04;Supermarket
DIV;DIV05;Pojizdna prodejna
SC;OIE/8001/CSV;Brambory8;0
UNIT;bal;11220801
SC;OIE/8002/CSV;Cokolada8;1
UNIT;m;11220802
UNIT;bal;11220803
SC;OIE/8003/CSV;Polevka8;2

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 3, typ importu = CSV

Definice BO Division:



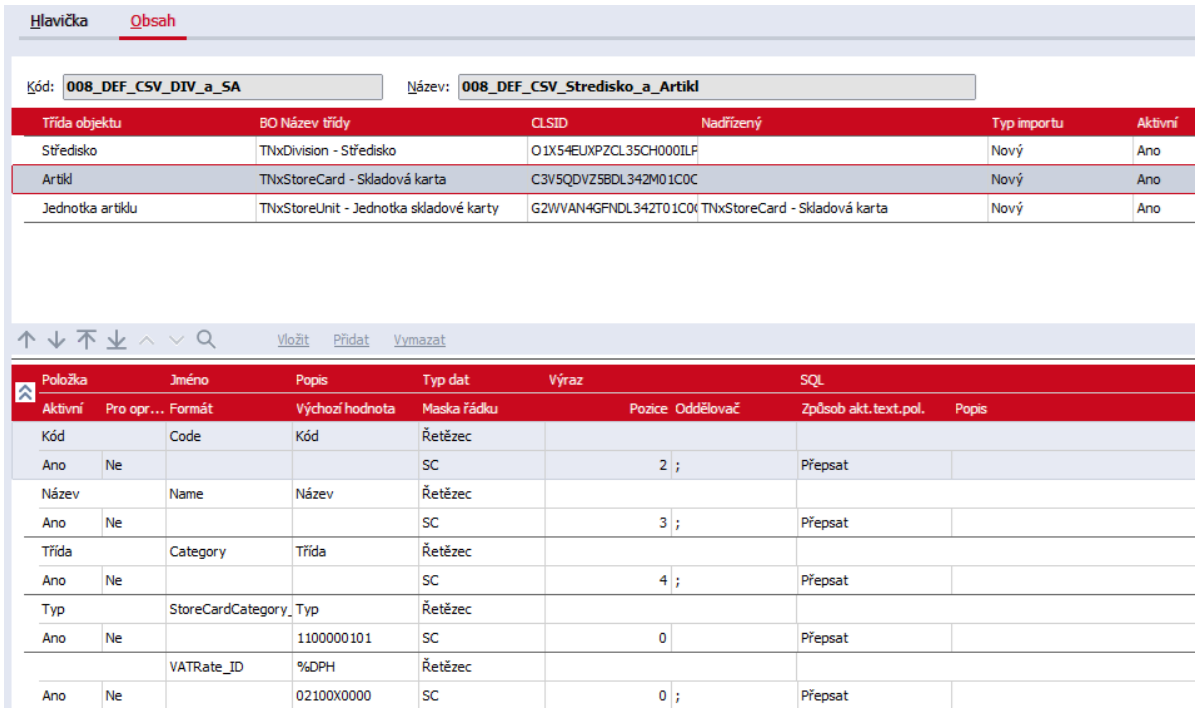
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Středisko	TNxDivision - Středisko	O1X54EUXPZCL35CH000ILP		Nový	Ano
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano



Položka		Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Kód		Code	Kód	Řetězec				
Ano	Ne			DIV	2	;	Přepsat	
Název		Name	Název	Řetězec				
Ano	Ne			DIV	3	;	Přepsat	

Kód i název střediska se importují z datového souboru. Důležitá je Maska řádku, aby bylo jasně definované, co se kam má importovat.

Definice BO StoreCards:


Hlavička Obsah

Kód: 008_DEF_CSV_DIV_a_SA Název: 008_DEF_CSV_Stredisko_a_Artikl

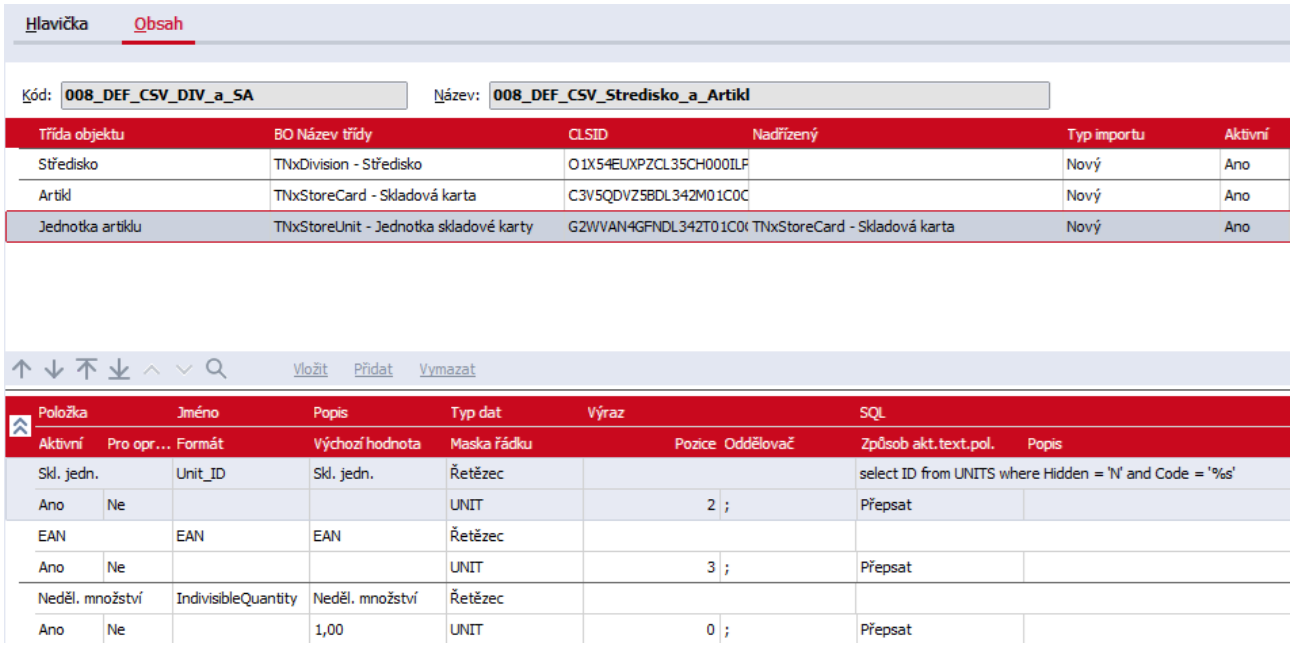
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Středisko	TNxDivision - Středisko	O1X54EUXPZCL35CH000ILF		Nový	Ano
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr... Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Kód	Code	Kód	Řetězec				
Ano	Ne		SC	2	;	Přepsat	
Název	Name	Název	Řetězec				
Ano	Ne		SC	3	;	Přepsat	
Třída	Category	Třída	Řetězec				
Ano	Ne		SC	4	;	Přepsat	
Typ	StoreCardCategory_	Typ	Řetězec				
Ano	Ne	1100000101	SC	0		Přepsat	
	VATRate_ID	%DPH	Řetězec				
Ano	Ne	02100X0000	SC	0	;	Přepsat	

Pro povinné položky, které nejsou k dispozici v importních datech jsou použity výchozí hodnoty. Tam kde jsou použity výchozí hodnoty musí být vždy pozice nastavena na neexistující, v našem případě 0.

Definice BO BO StoreUnit:


Hlavička Obsah

Kód: 008_DEF_CSV_DIV_a_SA Název: 008_DEF_CSV_Stredisko_a_Artikl

Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Středisko	TNxDivision - Středisko	O1X54EUXPZCL35CH000ILF		Nový	Ano
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr... Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Skl. jedn.	Unit_ID	Skl. jedn.	Řetězec			select ID from UNITS where Hidden = 'N' and Code = '%s'	
Ano	Ne		UNIT	2	;	Přepsat	
EAN	EAN	EAN	Řetězec				
Ano	Ne		UNIT	3	;	Přepsat	
Neděl. množství	IndivisibleQuantity	Neděl. množství	Řetězec				
Ano	Ne	1,00	UNIT	0	;	Přepsat	

Jednotka se dohledá pomocí SQL dotazu. Pro EAN, který je na jednotce není třeba nic dalšího nastavovat. V našem příkladu je konstantou nastaveno nedělitelné množství.

Přílohy:
Importní data: 008_Soubor_Stredisko_a_Artikl.csv
Definice definovatelného importu: 008_DEF_CSV_Stredisko_a_Artikl.xml

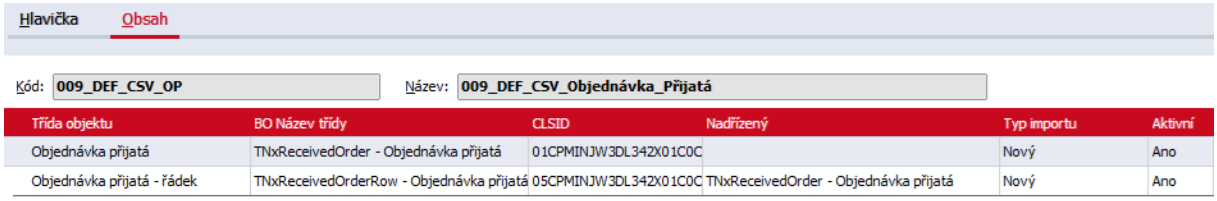
9) Příklad importu nové Objednávky přijaté (import BO s kolekcemi včetně subkolekce)

Import dvou nových objednávek přijatých s řádky typu 0 a 3.. Data jsou v importním souboru vždy uvedena za sebou.
Příklad importních dat:

```
Header;Alza.cz a.s.;Objednávka jedna
Rows;0;Řádek jedna
Rows;3;;01;003514;4
Header;CK Mare;Objednávka dva
Rows;3;;01;008015;2
Rows;0;Řádek dva
Rows;3;;01;10000023;6
```

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 0, typ importu = CSV

Definice BO ReceivedOrder:



Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Objednávka přijatá	TNxReceivedOrder - Objednávka přijatá	01CPMINJW3DL342X01C0C		Nový	Ano
Objednávka přijatá - řádek	TNxReceivedOrderRow - Objednávka přijatá	05CPMINJW3DL342X01C0C	TNxReceivedOrder - Objednávka přijatá	Nový	Ano



Položka	Jméno	Popis	Typ dat	Výraz	SQL
Aktivní / Pro opr... / Formát		Výchozí hodnota	Maska řádku	Pozice / Oddělovač	Způsob akt.text.pol. / Popis
Zdrojová řada	DocQueue_ID	Zdrojová řada	Řetězec		
Ano / Ne		I700000101	Header	0 / ;	Přepsat
Firma	Firm_ID	Firma	Řetězec		select ID from Firms where Hidden = 'N' and Name = '%s'
Ano / Ne			Header	2 / ;	Přepsat
Popis	Description	Popis	Řetězec		
Ano / Ne			Header	3 / ;	Přepsat

Pro povinné položky, které nejsou k dispozici v importních datech jsou použity výchozí hodnoty.
Pro identifikaci řádku v importních datech, který odpovídá danému BO definice je využita maska řádku.
Pro dohledání firmy je použit SQL dotaz s konstantou odkazující do importního souboru na pozici 2.

Definice BO ReceivedOrderRow:

Hlavička Obsah

Kód: 009_DEF_CSV_OP Název: 009_DEF_CSV_Objednávka_Přijatá

Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Objednávka přijatá	TNxReceivedOrder - Objednávka přijatá	01CPMINJW3DL342X01C0C		Nový	Ano
Objednávka přijatá - řádek	TNxReceivedOrderRow - Objednávka přijatá	05CPMINJW3DL342X01C0C	TNxReceivedOrder - Objednávka přijatá	Nový	Ano

Vložit Přidat Vymazat

Položka		Jméno	Popis	Typ dat	Výraz		SQL	
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Pozice	Oddělovač	Způsob akt.text.pol.	Popis
Typ řádku		RowType	Typ	Celé číslo				
Ano	Ne			Rows	2	;	Přepsat	
Text		Text	Text	Řetězec				
Ano	Ne			Rows	3	;	Přepsat	
Sklad		Store_ID	Sklad	Řetězec			select ID from Stores where Hidden = 'N' and Code = '%s'	
Ano	Ne			Rows	4	;	Přepsat	
Artikl		StoreCard_ID	Skladová karta	Řetězec			select ID from StoreCards where Hidden = 'N' and Code = '%s'	
Ano	Ne			Rows	5	;	Přepsat	
Počet v ev.jedn.		Quantity	Počet v ev.jedn.	Desetinné číslo				
Ano	Ne	.	0	Rows	6	;	Přepsat	
Středisko		Division_ID	Středisko	Řetězec				
Ano	Ne		2100000101	Rows	0	;	Přepsat	
%DPH		VATRate_ID	%DPH	Řetězec				
Ano	Ne		02100X0000	Rows	0	;	Přepsat	

Pro povinné položky (v tomto případě Středisko a DPH sazba), které nejsou k dispozici v importních datech jsou použity výchozí hodnoty. Proto je pozice nastavená jako nulová.
Pro identifikaci řádku v importních datech, který odpovídá danému BO definice je využita maska řádku.
Pro dohledání skladu a artiklu jsou použity SQL dotazy s konstantou odkazující do importního souboru na pozice, kde jsou tyto informace vyplněny.

Výsledkem zpracování importních dat touto definicí definovatelného importu jsou dvě nové Objednávky přijaté v řádky typu 0 a 3.

Přílohy:
Importní data: 009_Soubor_Objednávka_Přijatá.csv
Definice definovatelného importu: 009_DEF_CSV_Objednávka_Přijatá.xml

10) Příklad importu kalendáře do agendy Pracovních kalendářů

Importem jsou přidávány řádky s datem směny a následně mohou být k směnám naimportovány přestávky, pokud je potřeba je evidovat. Z pohledu systému ABRA Gen je přestávku potřeba evidovat jen u noční směny.
Směny je možné importovat i pro budoucí období.

V importním souboru jsou očekávány 3 typy řádků:

- **AHLPK** – pro identifikace pracovního kalendáře v systému ABRA Gen
- **BSPK** – hodnoty pro import na záložku „Kalendář" pro odpracované/plánované směny, svátek místo směny nebo směnové volno
- **CPSPK** – hodnoty pro evidenci čerpané přestávky (nutné pouze pro noční směny přes půlnoc)

POZOR
BSPK – směny – Definují-li se přestávky pozor na zadávání začátku směny vzhledem počtu odpracovaných hodin, po kterých začíná přestávka. Pokud se nedefinují přestávky musí se délka směny rovnat počtu odpracovaných hodin.

Import umožňuje načítat vždy jen jednu záložku Excelovského souboru, přeskakuje první tři řádky, kde mohou být o v hlavičkové popisy sloupců. Název listu v souboru lze ručně upravit v Definici definovatelných importů. Ve výchozím stavu předpokládáme název listu „K**alendarSmenaPrestavka**".

Před inportem musí existovat hlavička Pracovního kalendáře se stejným kodem a týdenním pracovním úvazkem jako v importovaném souboru. Vzor po generování nesmí být vyplněn.

Pro import je nutné mít založenou řadu dokladů typu LO (logy) se zkratkou a názven LOGIE. (Před 2024 neexistovala.)

Současně s importem se zapisuje do uvedené řady log s popisem průběhu importu.

Sloupce pro masku AHLPK

Pozice CSV	Název sloupce	Datový typ	Hodnota	Význam
1	Maska	String	AHLPK	Určuje kalendář, do kterého se importuje
2	Kód	String	10 znaků	Kód pracovního kalendáře (povinné)
3	Název	String	Max. 50 znaků	Název kalendáře (nepovinné)
4	Týdenní úvazek	Numeric (10/2)	Př. 37,50; 38,75; 40,00; 20,00	Úvazek (povinné)

Sloupce pro masku BSPK

Pozice CSV	Název sloupce	Datový typ	Hodnota	Význam
1	Maska	String	BSPK	Určuje směny, které se importují
2	Datum směny	Date	dd.MM.YYYY	Datum směny (povinné)
3	Typ plánované směny	String	Hodnoty: S; SV; V	S – směna, SV – svátek místo směny, V – volno (celý den) (povinné) pozn. V jeden den může být pouze plánovaná směna nebo volno místo směny. Mix zatím nedovolujeme.
4	Hodina od	Time	HH:MM Př. 06:00; 08:00; 18:00	V kolik hodin směna nebo svátek místo směny začíná. (povinné pro typ S) Prázdné pro V a SV.
5	Délka	Numeric	#00,00 Př. 8,75; 8,30	Délka směny včetně přestávek. (povinné pro typ S) Prázdné pro V a SV.
6	Hodiny	Numeric	#00,00 Př. 7,75; 8,00	Počet odpracovaných hodin. (povinné pro typ S) Prázdné pro V a SV.
7	V noci	Numeric	#00,00 Př. 8,00	Počet hodin pro příplatek za práci v noci. (nepovinné)

				Prázdné pro V a SV.
8	V prostředí	Numeric	#00,00 Př. 4,00	Počet hodin pro příplatek za práci ve škodlivém prostředí. (nepovinné) Prázdné pro V a SV.
9	Ve svátek	Numeric	#00,00 Př. 7,75	Počet hodin volna ve svátek místo plánované směny (povinné pro typ SV) Prázdné pro S a V.

Sloupce pro masku CPSPK

Nemusí být vyplněno, pokud se u BSPK hodnoty Délka a Hodiny rovnají. Pokud se délka směny a počet odpracovaných hodin nerovná, potom přestávka musí být definovaná!

Pozice CSV	Název sloupce	Datový typ	Hodnota	Význam
1	Maska	String	CPSPK	Určuje přestávky, které se importují, k již importovaným směnám
2	Datum směny	Date	dd.MM.YYYY	Datum směny (povinné)
3	Začátek přestávky od	Time	HH:MM Př. 22:00; 02:00	V kolik hodin přestávka začíná (nepovinné)
4	Začátek po	Numeric	#00,00 Př. 4,00; 8,00	Počet hodin od začátku směny. (povinné) Musí platit, že začátek směny „BSPK.Hodina od“ plus CPSPK.Začátek po“ se musí rovnat CPSPK.Začátek přestávky od (hodině Začátku přestávky)
5	Délka	Numeric	#00,00 Př. 0,50; 1,00	Délka přestávky (povinná hodnota)

Přílohy:
Importní data: 011_Soubor_pro_CSV_PracovniKalendar_VzorStruktury.csv
Definice 011_DEF_CSV_O_PracovniKalendar_I-O_Smena_Prestavky.xml

6.3 TYP IMPORTU XML

Jedná se o import dat ve formátu XML.

Je podporováno kódování UTF-8.

Nastavení „Procházet sekvenčně" je pro typ importu XML irelevantní.

Položky pro import se v importních datech definují pomocí X-path. Absolutní cesta k položce je uložena v položce „XML větev" a konkrétní cílová položka je v importní definici uložena v „XML položka".

V případě kolekcí je nezbytné vždy definovat v importní položce s typem dat „XML kolekce" dynamickou cestu k odpovídající XML kolekci v importních XML datech. U dalších položek kolekce v importní definici není potřeba zadávat „XML větev", stačí „XML položka", konkrétní XML větev se při importu doplňuje dynamicky podle konkrétních importních XML dat. Vazby mezi hlavičkami a řádky, i pro subkolekce, se pro typ importu nastavují automaticky, není tedy nutné v importní definici řešit vazby typu vlastník (typicky pro Parent_ID).

Nastavení dynamické XML kolekce v importní definici musí být vždy ve formátu například … /documents[N]/document[], kde documents[N] (N je povinné) je dynamická cesta k hlavičce kolekce v xml datech a document[] ([] je povinné) je dynamická cesta ke konkrétní skupině položek kolekce v XML datech. Při zpracování se za [N] a [] dynamicky dosadí kompletní X-path cesta k položkám podle toho, jak jsou konkrétně zadány v XML importních datech. Při zpracování se tedy bude cesta v závislosti na konkrétní struktuře XML importních dat vyhodnocovat například takto:

… /documents[1]/document[1]/položka[1], /documents[1]/document[2]/položka[1], /documents[1]/document[3]/položka[1], /documents[2]/document[1]/položka[1] atd. Je podporován i stav, kdy v XML importních datech nějaká položka kolekce chybí.

6.3.1 PŘÍKLADY

POZOR
Všechny příklady jsou připraveny pro testovaní nad demodaty (demodata je možné vytvořit jako další databázové spojení). V případě, že budete chtít příklady otestovat na vašich datech, je potřeba změnit data v příkladových importních souborech na data z vaší databáze.

Součástí příkladů jsou i vzorové definice importů v XML a importní soubory v XML. Definice i soubory ke všem příkladům jsou ke stažení v jednom ZIP souboru zde.

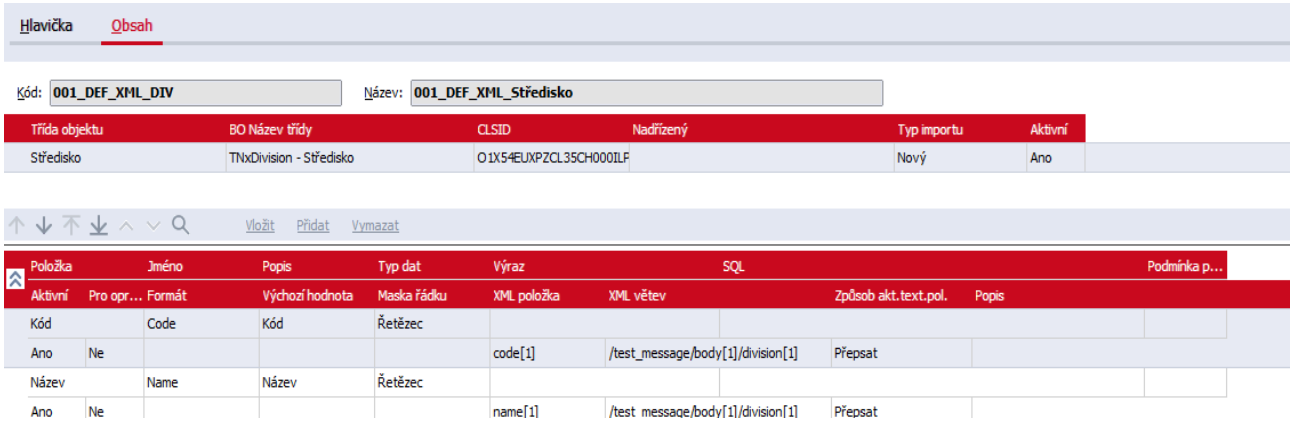
1) Příklad XML importu jednoho nového střediska (jednoho nového BO
Základní import jednoho nového záznamu středisek.

Příklad importních XML dat:

```
<?xml version="1.0" encoding="UTF-8"?>
<test_message>
    <body>
        <division>
            <code>DIV01</code>
            <name>Provozovna_01</name>
        </division>
    </body>
</test_message>
```

Definice definovatelného importu:
Hlavička: řádky pro přeskočení = 0, typ importu = XML

Definice BO Division:



Jde o seznam položek definovaných statickými x-path.

Výsledkem zpracování importních dat touto definicí definovatelného importu je jeden nový záznam střediska s vyplněným kódem a názvem.

Přílohy:
Importní data: 001_Soubor_Středisko.xml
Definice definovatelného importu: 001_DEF_XML_Středisko.xml

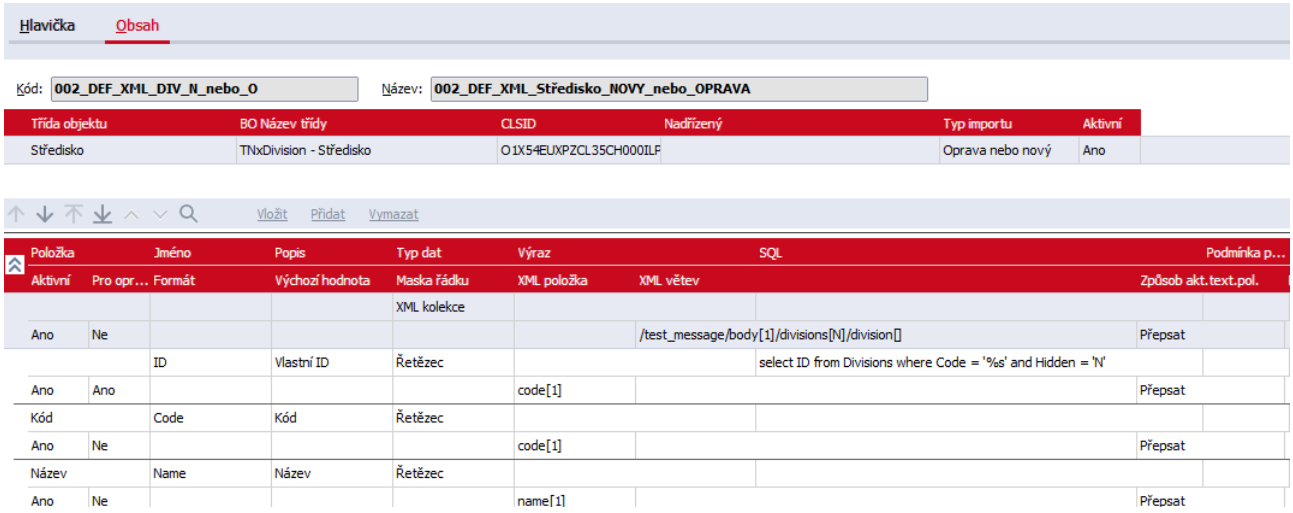
2) Příklad XML importu více BO – oprava nebo nový
Definice pro opravu nebo zadání více středisek.

Příklad importních XML dat:

```
<?xml version="1.0" encoding="UTF-8"?>
<test_message>
    <body>
        <divisions>
            <division>
                <code>DIV01</code>
                <name>Provozovna U potoka</name>
            </division>
            <division>
                <code>DIV02</code>
                <name>Provozovna Na kopci</name>
            </division>
            <division>
                <code>DIV03</code>
                <name>Provozovna V údolí</name>
            </division>
        </divisions>
    </body>
</test_message>
```

Definice definovatelného importu:
Hlavička: řádky pro přeskočení = 0, typ importu = XML

Definice BO Division:



Jde o dynamickou kolekci, která je definována v položce s typem dat „XML kolekce". Pro demonstraci je využita položka „Pro opravu" pro dohledání ID střediska pro opravu.

Výsledkem zpracování importních dat touto definicí definovatelného importu jsou opravené záznamy středisek, pokud byly dohledány podle kódu. Pokud se středisko nedohledá, zadá se automaticky nový záznam středisek. Vyplňují nebo aktualizují se položky kód a název.

Přílohy:
Importní data: 002_Soubor_Středisko_NOVY_nebo_OPRAVA.xml
Definice definovatelného importu: 002_DEF_XML_Středisko_NOVY_nebo_OPRAVA.xml

3) Příklad XML aktualizace položek a cen skladového ceníku

Provede aktualizaci položek a cen zvoleného skladového ceníku. Pokud položka nebo cena v ceníku neexistuje, založí se, v opačném případě se provede aktualizace ceny.

Příklad importních XML dat:

```
<?xml version="1.0" encoding="UTF-8"?>
<cenik_message>
	<body>
		<storeprices>
			<storeprice>
				<PriceListCode>Hlavní</PriceListCode>
				<SC_Code>VAN_KO</SC_Code>
				<prices>
					<price>
				<SC_Code>VAN_KO</SC_Code>
						<definice_ceny>1</definice_ceny>
						<jednotka>ks</jednotka>
						<amount>10.5</amount>
					</price>
					<price>
				<SC_Code>VAN_KO</SC_Code>
						<definice_ceny>2</definice_ceny>
						<jednotka>ks</jednotka>
						<amount>22.15</amount>
					</price>
				</prices>
			</storeprice>
			<storeprice>
				<PriceListCode>Hlavní</PriceListCode>
				<SC_Code>ZG01501</SC_Code>
```

```
                                        <prices>
                                                <price>
                                        <SC_Code>ZG01501</SC_Code>
                                                        <definice_ceny>1</definice_ceny>
                                                        <jednotka>kg</jednotka>
                                                        <amount>33.5</amount>
                                                </price>
                                                <price>
                                        <SC_Code>ZG01501</SC_Code>
                                                        <definice_ceny>2</definice_ceny>
                                                        <jednotka>kg</jednotka>
                                                        <amount>44.35</amount>
                                                </price>
                                        </prices>
                                </storeprice>
                        </storeprices>
                </body>
</cenik_message>
```

Definice definovatelného importu:
Hlavička: řádky pro přeskočení = 0, typ importu = XML

Definice BO StorePrice:

Hlavička | Obsah

Kód: 003_DEF_XML_Akt_Cen Název: 003_DEF_XML_Aktualizace_skladových_ceníků

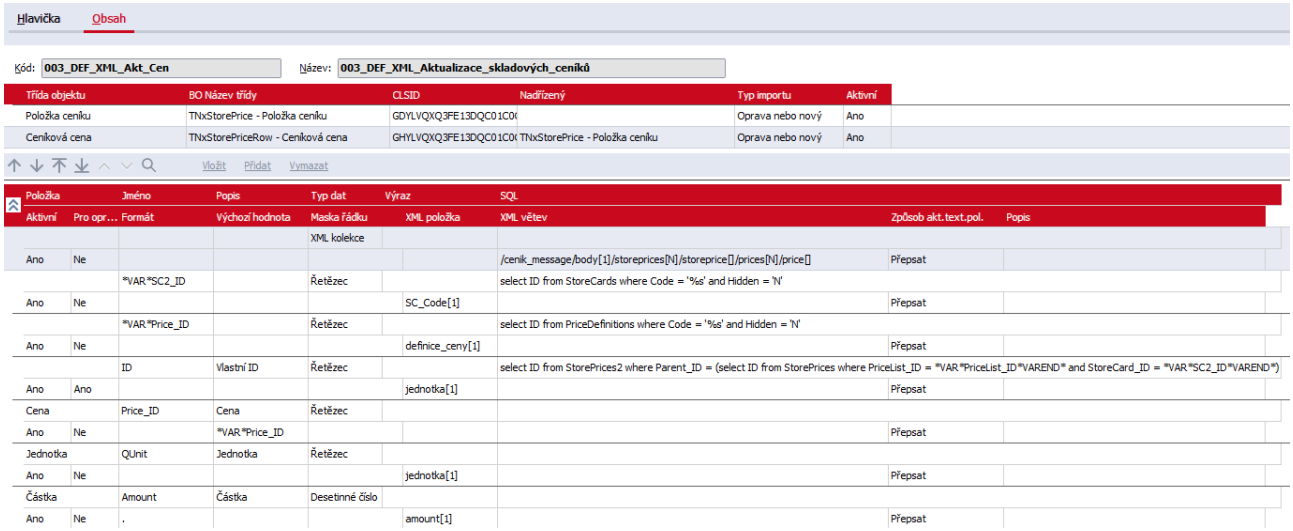
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Položka ceníku	TNxStorePrice - Položka ceníku	GDYLVQXQ3FE13DQC01C0(		Oprava nebo nový	Ano
Ceníková cena	TNxStorePriceRow - Ceníková cena	GHYLVQXQ3FE13DQC01C0(	TNxStorePrice - Položka ceníku	Oprava nebo nový	Ano

Vložit | Přidat | Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL	Podmínka p...
Aktivní / Pro opr...	Formát	Výchozí hodnota	Maska řádku	XML položka	XML větev	Způsob akt.text.pol.
			XML kolekce			
Ano / Ne					/cenik_message/body[1]/storeprices[N]/storeprice[]	Přepsat
	*VAR*PriceList_ID		Řetězec		select ID from PriceLists where Code = '%s' and Hidden = 'N'	
Ano / Ne				PriceListCode[1]		Přepsat
	*VAR*SC_ID		Řetězec		select ID from StoreCards where Code = '%s' and Hidden = 'N'	
Ano / Ne				SC_Code[1]		Přepsat
	*VAR*StorePrice_OI		Řetězec		select ID from StorePrices where PriceList_ID = *VAR*PriceList_ID*VAREND*	
Ano / Ne				SC_Code[1]		Přepsat
	ID	Vlastní ID	Řetězec			
Ano / Ano		*VAR*StorePrice_OI				Přepsat
Ceník	PriceList_ID	Ceník	Řetězec		select ID from PriceLists where Code = '%s' and Hidden = 'N'	
Ano / Ne		*VAR*PriceList_ID		PriceListCode[1]		Přepsat
Artikl	StoreCard_ID	Skl. karta	Řetězec			
Ano / Ne		*VAR*SC_ID				Přepsat
Datum platnosti	PriceListValidity_ID	Datum platnosti	Řetězec			
Ano / Ne		0000000000				Přepsat

Kolekce položek ceníků. Definice dynamické kolekce je v položce s typem dat „XML kolekce". Demonstrace získání ID pro opravu.

Definice BO StorePriceRow:



Kolekce cen ceníků. Definice dynamické kolekce je v položce s typem dat „XML kolekce". Demonstrace získání ID pro opravu.
Výsledkem zpracování importních dat touto definicí importu je aktualizaci položek a cen zvoleného skladového ceníku. Pokud položka nebo cena v ceníku neexistuje, založí se, v opačném případě se provede aktualizace ceny.

Přílohy:
Importní data: 003_Soubor_Aktualizace_skladových_ceníků.xml
Definice definovatelného importu: 003_DEF_XML_Aktualizace_skladových_ceníků.xml

4) Příklad XML Výdejka

Import hlavičky a řádků výdejky včetně výběru šarží/sériových čísel.

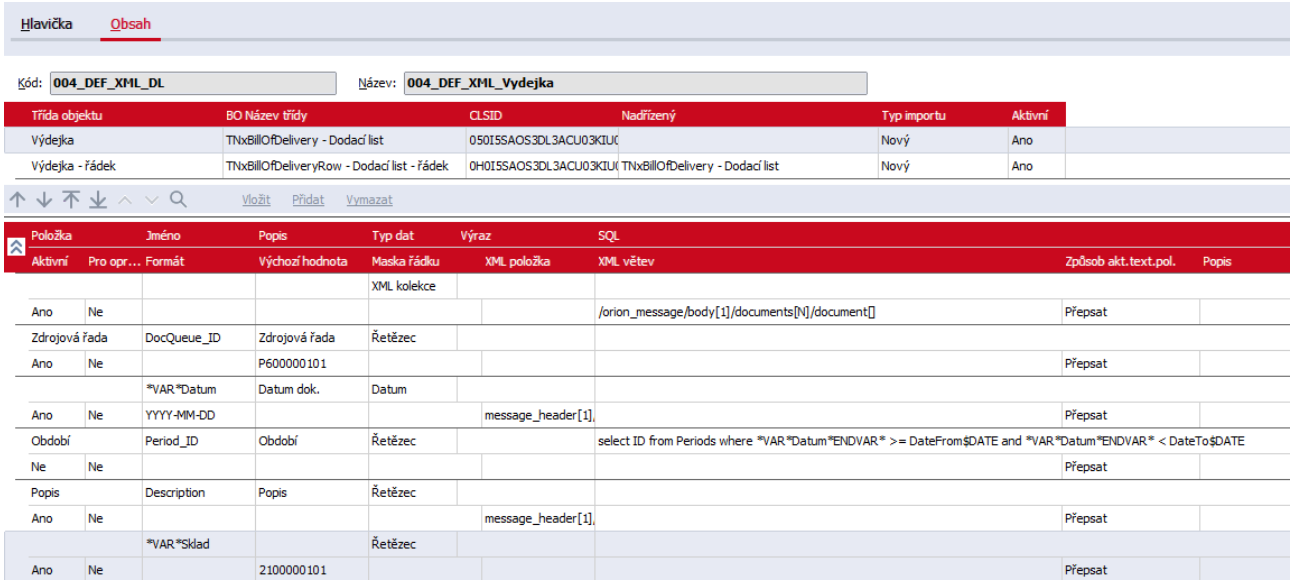
Příklad importních XML dat:

```
<?xml version="1.0" encoding="UTF-8"?>
<orion_message>
	<body>
		<documents>
			<document>
				<message_header>
					<doc_number>Dodak1</doc_number>
					<doc_date_of_issue>2024-05-05</doc_date_of_issue>
				</message_header>
				<articles>
					<article>
						<item_number>1</item_number>
						<article_code>10</article_code>
						<quantity>3.000</quantity>
						<unit>ks</unit>
						<article_serialnumbers>
							<article_serialnumberitem>
								<article_serialnumber>SCX/000169-
AA</article_serialnumber>
							</article_serialnumberitem>
							<article_serialnumberitem>
								<article_serialnumber>SCX/000170-
AA</article_serialnumber>
							</article_serialnumberitem>
							<article_serialnumberitem>
								<article_serialnumber>SCX/000171-
AA</article_serialnumber>
							</article_serialnumberitem>
```

```
                                        </article_serialnumbers>
                                    </article>
                                    <article>
                                        <item_number>2</item_number>
                                        <article_code>08</article_code>
                                        <quantity>20.000</quantity>
                                        <unit>ks</unit>
                                    </article>
                                </articles>
                            </document>
                        </documents>
                    </body>
</orion_message>
```

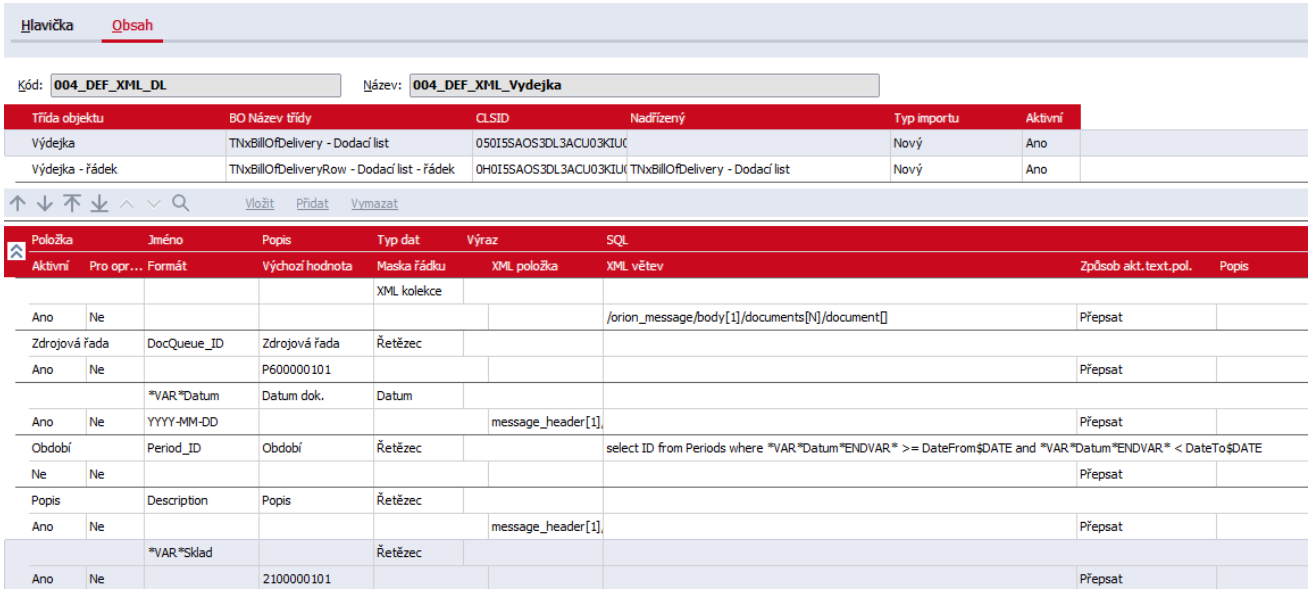
Definice definovatelného importu:
Hlavička: řádky pro přeskočení = 0, typ importu = XML

Definice BO BillOfDelivery:



Kolekce položek hlavičky. Definice dynamické kolekce je v položce s typem dat „XML kolekce“. Demonstrace načtení ID skladu z konstanty do proměnné.

Definice BO BillOfDeliveryRow:



Hlavička Obsah

Kód: 004_DEF_XML_DL Název: 004_DEF_XML_Vydejka

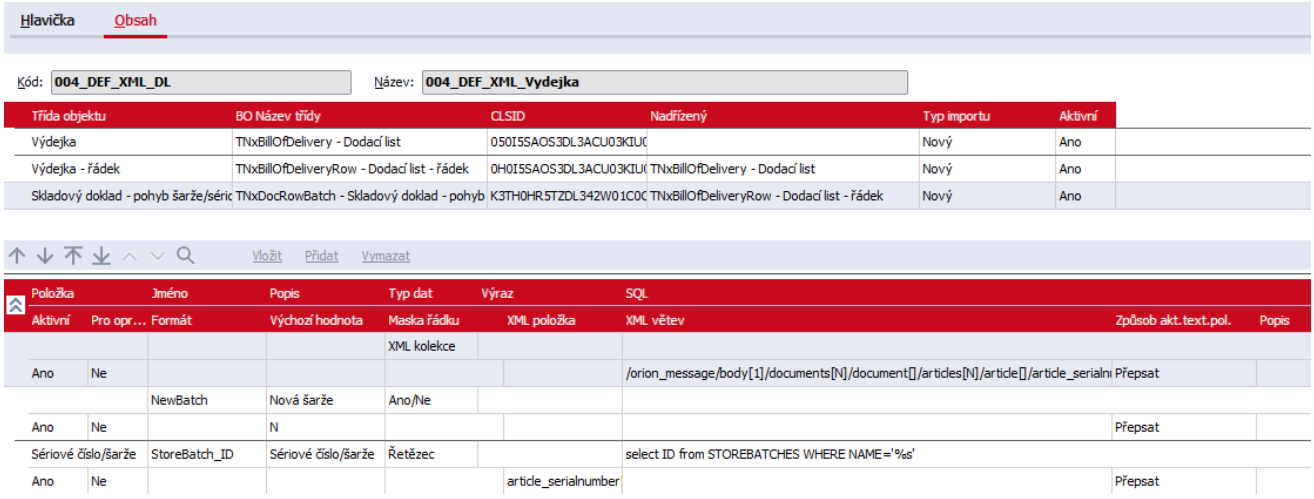
Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Výdejka	TNxBillOfDelivery - Dodací list	050I5SAOS3DL3ACU03KIU(		Nový	Ano
Výdejka - řádek	TNxBillOfDeliveryRow - Dodací list - řádek	0H0I5SAOS3DL3ACU03KIU(	TNxBillOfDelivery - Dodací list	Nový	Ano

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL		
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	XML položka	XML větev	Způsob akt.text.pol.	Popis
				XML kolekce				
Ano	Ne					/orion_message/body[1]/documents[N]/document[]	Přepsat	
Zdrojová řada		DocQueue_ID	Zdrojová řada	Řetězec				
Ano	Ne		P600000101				Přepsat	
		*VAR*Datum	Datum dok.	Datum				
Ano	Ne	YYYY-MM-DD			message_header[1]		Přepsat	
Období		Period_ID	Období	Řetězec		select ID from Periods where *VAR*Datum*ENDVAR* >= DateFrom$DATE and *VAR*Datum*ENDVAR* < DateTo$DATE		
Ne	Ne						Přepsat	
Popis		Description	Popis	Řetězec				
Ano	Ne				message_header[1]		Přepsat	
		*VAR*Sklad		Řetězec				
Ano	Ne		2100000101				Přepsat	

Kolekce řádků. ID skladu je načteno z proměnné na hlavičce, Typ řádku a Středisko je zadáno konstantou. Artikl se dohledává za pomoci SQL dotazu.

Definice BO DocRowBatch:



Hlavička Obsah

Kód: 004_DEF_XML_DL Název: 004_DEF_XML_Vydejka

Třída objektu	BO Název třídy	CLSID	Nadřízený	Typ importu	Aktivní
Výdejka	TNxBillOfDelivery - Dodací list	050I5SAOS3DL3ACU03KIU(		Nový	Ano
Výdejka - řádek	TNxBillOfDeliveryRow - Dodací list - řádek	0H0I5SAOS3DL3ACU03KIU(	TNxBillOfDelivery - Dodací list	Nový	Ano
Skladový doklad - pohyb šarže/sério	TNxDocRowBatch - Skladový doklad - pohyb	K3TH0HR5TZDL342W01C0(	TNxBillOfDeliveryRow - Dodací list - řádek	Nový	Ano

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL		
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	XML položka	XML větev	Způsob akt.text.pol.	Popis
				XML kolekce				
Ano	Ne					/orion_message/body[1]/documents[N]/document[]/articles[N]/article[]/article_serialn	Přepsat	
		NewBatch	Nová šarže	Ano/Ne				
Ano	Ne		N				Přepsat	
Sériové číslo/šarže		StoreBatch_ID	Sériové číslo/šarže	Řetězec		select ID from STOREBATCHES WHERE NAME='%s'		
Ano	Ne				article_serialnumber		Přepsat	

Kolekce šarží. ID šarže je dohledáváno pomocí SQL dotazu.

Přílohy:
Importní data: 004_Soubor_Vydejka.xml
Definice definovatelného importu: 003_DEF_XML_Aktualizace_skladových_ceníků.xml

6.4 TYP IMPORTU XLS

Jedná se o import dat ve formátu tabulkového editoru Excel. Položky pro import se v importních datech definují záložkou sešitu (třídy BO) a sloupcem listu (jednotlivé položky - subkolekce).
Pokud je použito více BO (tedy záložek) v jedné importní definici, definují se vazby mezi těmito objekty prostřednictvím sloupce v excelu a položkou v definici s typem dat „Excel propojení".

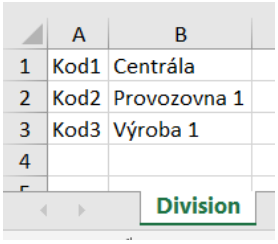
6.4.1 PŘÍKLADY:

> **POZOR**
> Všechny příklady jsou připraveny pro testovaní nad demodaty (demodata je možné vytvořit jako další databázové spojení.). V případě, že budete chtít příklady otestovat na vašich datech, je potřeba změnit data v příkladových importních souborech na data z vaší databáze.

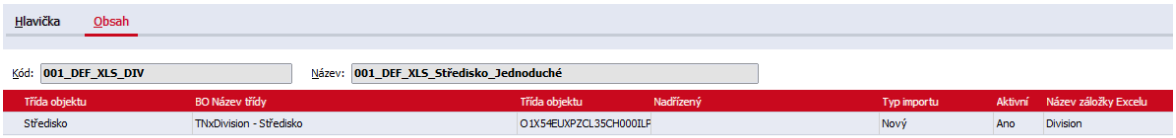
Součástí příkladů jsou i vzorové definice importů v XML a importní soubory v XLS. Definice i soubory ke všem příkladům jsou ke stažení v jednom ZIP souboru zde

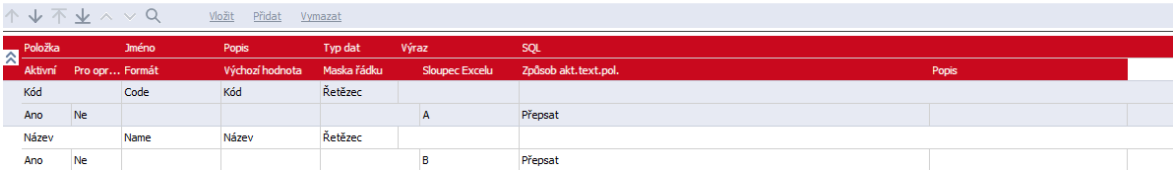
1) Příklad importu nových středisek (nových BO bez kolekcí):
Import nových středisek, vyplňuje se kód a název. Data jsou v importním souboru uváděna ve sloupcích a řádcích.

Příklad importních dat:



Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 0, typ importu = Excel





Výsledkem zpracování importních dat touto definicí definovatelného importu jsou tři nové záznamy středisek.

Přílohy:
Importní data: 001_Soubor_Středisko_Jednoduché.xlsx
Definice definovatelných importů: 001_DEF_XLS_Středisko_Jednoduché.xml

Následovat bude sada příkladů, které na sebe budou navazovat. Nejprve v příkladu provedeme import nových artiklů, poté uděláme v dalším příkladu jejich update. Dále naimportujeme tyto artikly do příjemky a nakonec z těchto artiklů vytvoříme fakturu vydanou.

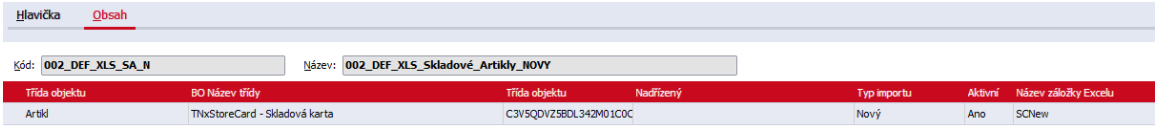
2) Příklad importu nových artiklů

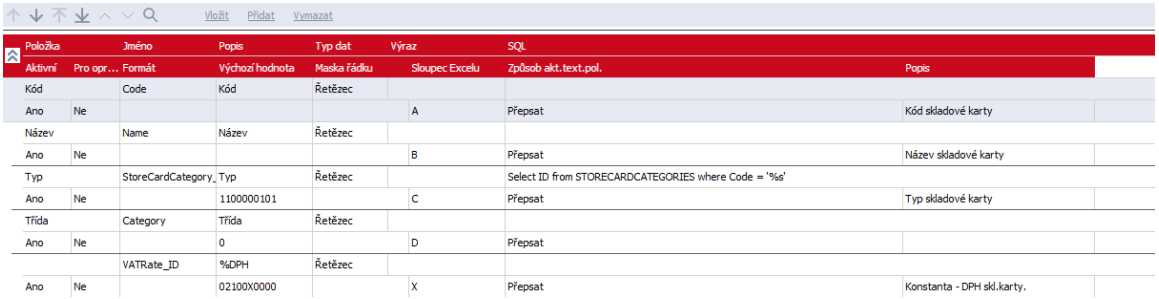
Import nových skladových karet, vyplňuje se kód a název, typ a třída. Data jsou v importním souboru uváděna ve sloupcích a řádcích.

Příklad importních dat:

	A	B	C	D
1	kód	název	Typ	Třída
2	OIE/1001/XLS	Karafiat1	Z	0
3	OIE/1002/XLS	Petrklic1	Z	0
4	OIE/1003/XLS	Sedmikraska1	Z	0
5	OIE/1004/XLS	Konvalinka1	Z	0
6	OIE/1005/XLS	Krabice1	M	4
7	OIE/1006/XLS	Košík1	M	4
8	OIE/1007/XLS	Paleta1	M	4

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 0, typ importu = Excel
Obsah:
V importovaném souboru je záložek více, je potřeba na hlavičku BO nastavit správnou záložku SCNew
Typ artiklu - ve vzoru použita výchozí hodnota, prezentována jako ID. Na Typ artiklu se doptáváme přes SQL dotaz.
Třída artiklu – ve vzoru použita výchozí hodnota 0.
%DPH – zapomenout nesmíme doplnit položku % DPH sazby. Ve vzoru je použita výchozí hodnota prezentována jako ID. Jako sloupec je zde použitý sloupec X, který však není úmyslně vyplněn, hodnoty se budou čerpat z výchozí hodnoty.
Výsledkem zpracování importních dat touto definicí definovatelného importu jsou nové vytvořené záznamy v agendě artikly.





Přílohy:
Importní data: 002_DEF_XLS_Skladové_Artikly_NOVY.xlsx
Definice definovatelných importů: 002_Soubor_Skladové_Artikly_NOVY.xml

3) Příklad opravy artiklu se založením subkolekcí

Komplexní příklad na opravu existujících hlaviček artiklů (oprava nebo nový) s kolekcí jednotek a přidání dalších položek do subkolekcí jednotek. Příklad použití proměnné na podřízené kolekci, pokud je proměnná definovaná na nadřízené kolekci.

Příklad importních dat:

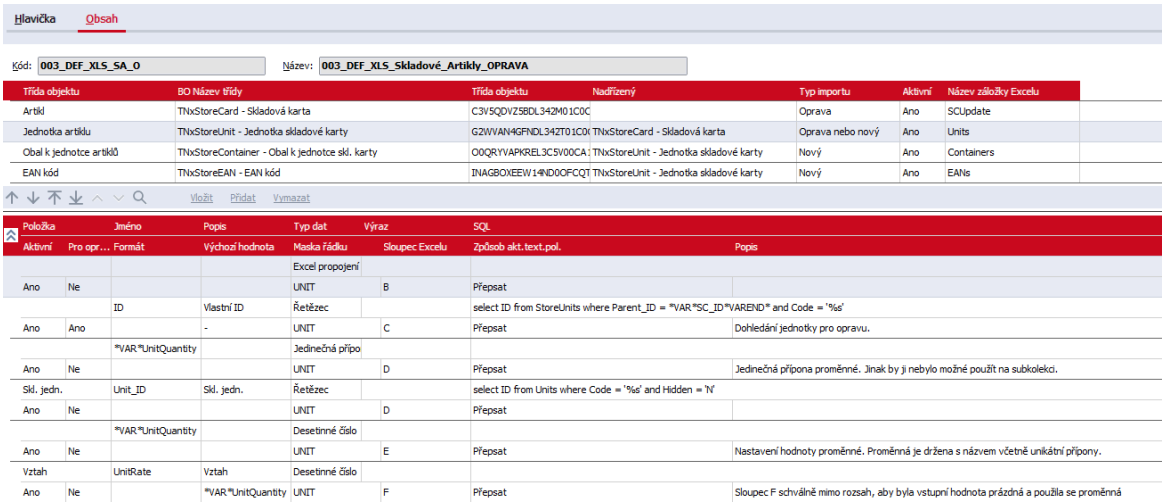
	A	B	C	D	E	F
1	prefix	identifikace	kód	název	popis	cizí název
2	SC	Karta_1	OIE/1001/	Karafiat1	řádek jedna řádek dva	cizí název 1
3	SC	Karta_2	OIE/1002/	Petrklic1	řádek";" dvě-jedna řádek "dva-dva, ";" text2 řádek ; dva-tři	cizí název2
4	SC	Karta_3	OIE/1004/XLS			
5						
6						

Definice definovatelného importu:
Hlavička: sekvenčně = ANO, řádky pro přeskočení = 0, typ importu = Excel
Třída Skladová karta:
Typ importu: Oprava
První řádek má datový typ Excel propojení, což je typ, dostupný pouze pro importy z Excelu. Jde o identifikátor, který bude použit na všech dalších záložkách. Zde je umístěn ve sloupci B.



Na dalším řádku je použita proměnná z předchozího řádku a řádek je nastaven pro opravu. Ve výrazu je pak také nastaveno ukončení proměnné *VAR*SC_ID*VAREND*
Ostatní řádky pak souvisejí s daty v tabulce a jsou navázány na jednotlivé sloupce.

Třída Jednotka artiklu:
Typ importu Oprava nebo nový
Na prvním řádku je opět datový typ Excel propojení, který ve sloupci B obsahuje stejná data jako na záložce SCUpdate pro třídu Artikl.
V dalším řádku, který slouží pro opravu je využita proměnná z nadřízené kolekce v SQL dotazu, a navíc doplněna o sloupec C. Řádek slouží pro dohledání jednotky pro opravu.


Hlavička Obsah

Kód: 003_DEF_XLS_SA_O Název: 003_DEF_XLS_Skladové_Artikly_OPRAVA

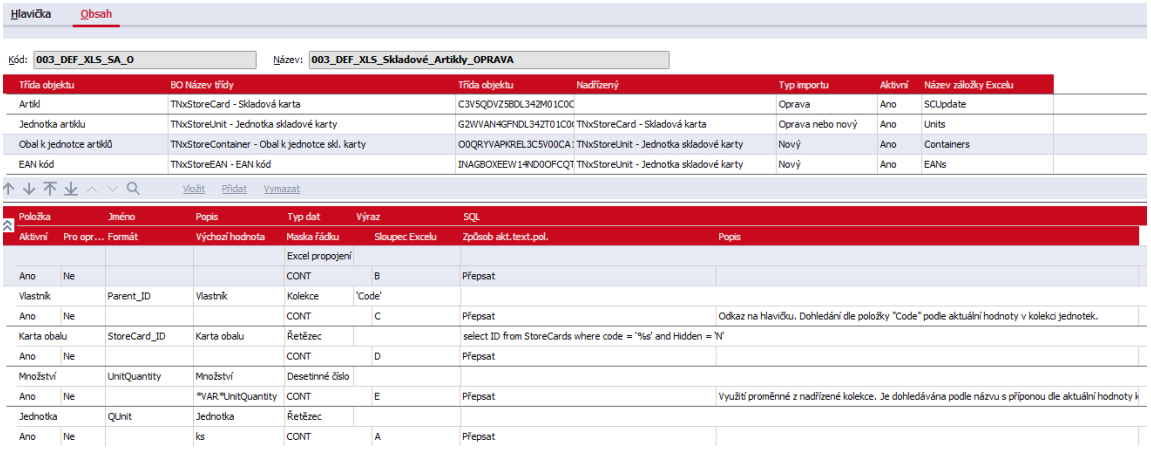
Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Oprava	Ano	SCUpdate
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Oprava nebo nový	Ano	Units
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce skl. karty	O0QRYVAPKREL3C5V00CA	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	Containers
EAN kód	TNxStoreEAN - EAN kód	INAGBOXEEW14ND0OFCQT	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	EANs

Vložit Přidat Vymazat

Položka		Jméno	Popis	Typ dat	Výraz	SQL	
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text.pol.	Popis
				Excel propojení			
Ano	Ne			UNIT	B	Přepsat	
		ID	Vlastní ID	Řetězec		select ID from StoreUnits where Parent_ID = *VAR*SC_ID*VAREND* and Code = '%s'	
Ano	Ano		-	UNIT	C	Přepsat	Dohledání jednotky pro opravu.
		*VAR*UnitQuantity		Jedinečná přípo			
Ano	Ne			UNIT	D	Přepsat	Jedinečná přípona proměnné. Jinak by ji nebylo možné použít na subkolekci.
Skl. jedn.		Unit_ID	Skl. jedn.	Řetězec		select ID from Units where Code = '%s' and Hidden = 'N'	
Ano	Ne			UNIT	D	Přepsat	
		*VAR*UnitQuantity		Desetinné číslo			
Ano	Ne			UNIT	E	Přepsat	Nastavení hodnoty proměnné. Proměnná je držena s názvem včetně unikátní přípony.
Vztah		UnitRate	Vztah	Desetinné číslo			
Ano	Ne		*VAR*UnitQuantity	UNIT	F	Přepsat	Sloupec F schválně mimo rozsah, aby byla vstupní hodnota prázdná a použila se proměnná

V dalším řádku je použita proměnná *VAR*UnitQuantity a datový typ je nastaven jako Jedinečná přípona proměnné, aby ji bylo možné použít pro subkolekce.
Pro vztah musíme založit ještě jednu proměnnou, také s názvem *VAR*UnitQuantity. Tu následně použijeme v posledním řádku Unitrate, kde sloupec Excelu nastavíme úmyslně na jiné písmeno (v našem příkladu F), než kde máme data a to proto, aby byla hodnota prázdná a použila se proměnná.

Třída Obal k jednotce artiků:
Typ importu Nový
Zde potřebujeme kód jednotky skladové karty pro párování. V řádku použijeme Parent_ID a typ dat nastavíme na kolekci.


Hlavička Obsah

Kód: 003_DEF_XLS_SA_O Název: 003_DEF_XLS_Skladové_Artikly_OPRAVA

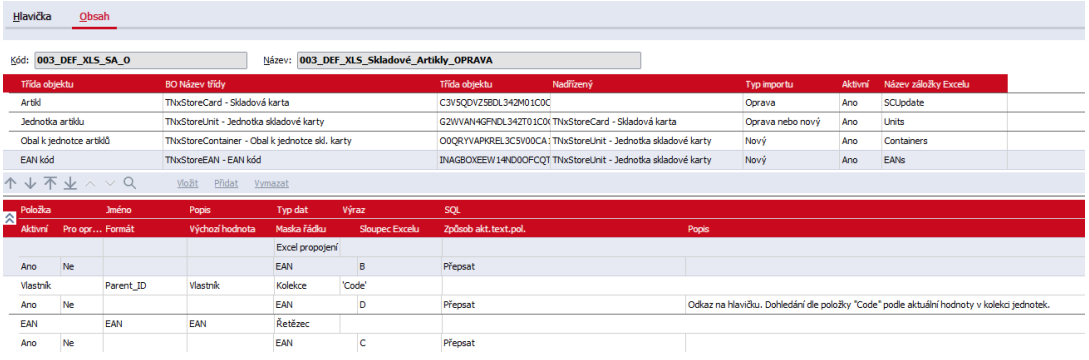
Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Oprava	Ano	SCUpdate
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Oprava nebo nový	Ano	Units
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce skl. karty	O0QRYVAPKREL3C5V00CA	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	Containers
EAN kód	TNxStoreEAN - EAN kód	INAGBOXEEW14ND0OFCQT	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	EANs

Vložit Přidat Vymazat

Položka		Jméno	Popis	Typ dat	Výraz	SQL	
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text.pol.	Popis
				Excel propojení			
Ano	Ne			CONT	B	Přepsat	
Vlastník		Parent_ID	Vlastník	Kolekce	'Code'		
Ano	Ne			CONT	C	Přepsat	Odkaz na hlavičku. Dohledání dle položky "Code" podle aktuální hodnoty v kolekci jednotek.
Karta obalu		StoreCard_ID	Karta obalu	Řetězec		select ID from StoreCards where code = '%s' and Hidden = 'N'	
Ano	Ne			CONT	D	Přepsat	
Množství		UnitQuantity	Množství	Desetinné číslo			
Ano	Ne		*VAR*UnitQuantity	CONT	E	Přepsat	Využití proměnné z nadřízené kolekce. Je dohledávána podle názvu s příponou dle aktuální hodnoty k
Jednotka		QUnit	Jednotka	Řetězec			
Ano	Ne		ks	CONT	A	Přepsat	

Pro získání množství použijeme založenou proměnnou *VAR*UnitQuantity z nadřízené kolekce.

Třída EAN kód k jednotce artiklu:
Typ importu Nový
První řádek je opět jen pro spárování se skladovou kartou na záložce SCUpdate.
V druhém řádku Parent_ID s odkazem na kód z jednotky. Do výrazu píšeme Code viz obrázek.


Hlavička Obsah

Kód: 003_DEF_XLS_SA_O Název: 003_DEF_XLS_Skladové_Artikly_OPRAVA

Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Oprava	Ano	SCUpdate
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Oprava nebo nový	Ano	Units
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce skl. karty	O0QRYVAPKREL3C5V00CA	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	Containers
EAN kód	TNxStoreEAN - EAN kód	INAGBOXEEW14ND0OFCQT	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	EANs

Vložit Přidat Vymazat

Položka		Jméno	Popis	Typ dat	Výraz	SQL	
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text.pol.	Popis
				Excel propojení			
Ano	Ne			EAN	B	Přepsat	
Vlastník		Parent_ID	Vlastník	Kolekce	'Code'		
Ano	Ne			EAN	D	Přepsat	Odkaz na hlavičku. Dohledání dle položky "Code" podle aktuální hodnoty v kolekci jednotek.
EAN		EAN	EAN	Řetězec			
Ano	Ne			EAN	C	Přepsat	

Přílohy:
Importní data: 003_Soubor_Skladové_Artikly_OPRAVA.xlsx
Definice definovatelných importů: 003_DEF_XLS_Skladové_Artikly_OPRAVA.xml

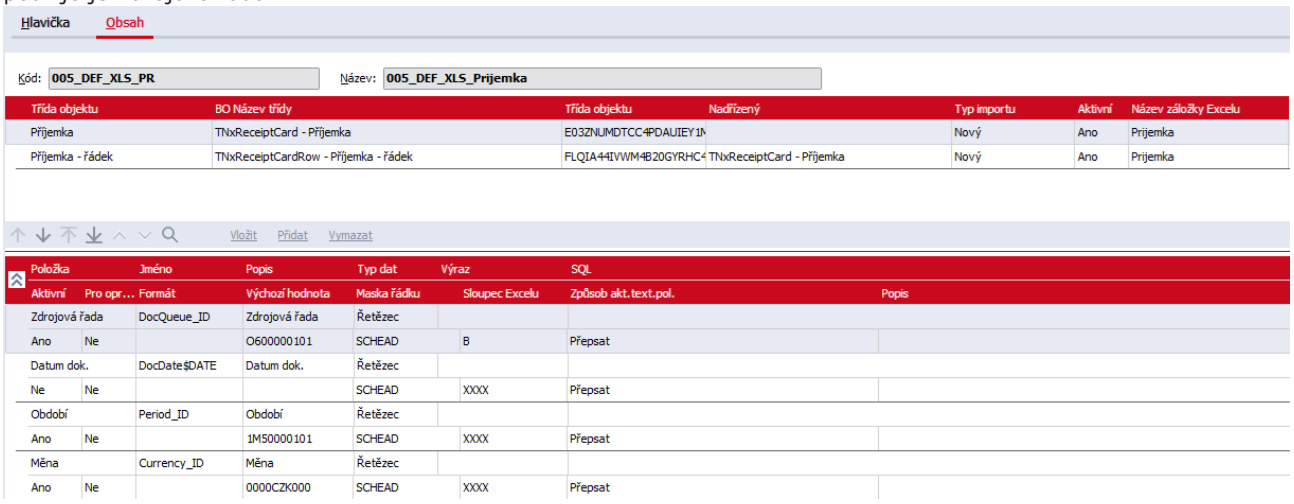
4) Příklad definice příjemky artiklů

Tento příklad provede import příjemky artiklů, které máme již vytvořeny v předchozích příkladech.

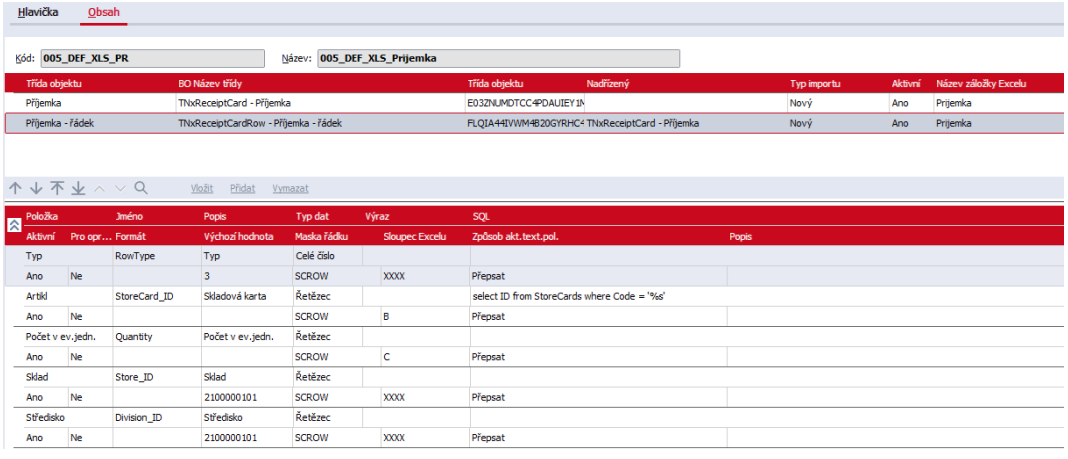
Příklad importních dat:

	A	B	C
1	Maska SCHEA	Zdrojová řada	
2	SCHEAD	O600000101	
3	Maska SCROW	Kod Skladové karty	Množství
4	SCROW	OIE/1001/XLS	100
5	SCROW	OIE/1002/XLS	100
6	SCROW	OIE/1003/XLS	100
7	SCROW	OIE/1004/XLS	100
8	SCROW	OIE/1005/XLS	100
9	SCROW	OIE/1006/XLS	100
10	SCROW	OIE/1007/XLS	100
11	SCROW	OIE/1101/XLS	100
12	SCROW	OIE/1102/XLS	100
13	SCROW	OIE/1103/XLS	100
14	SCROW	OIE/1104/XLS	100

Pro import hlavičky je většina údajů předvyplněných přímo v definici a jediné, co se z excelové tabulky použije je Zdrojová řada.



Obsah:
V řádcích se z Excelové tabulky dotahuje pouze artikl přes SQL dotaz a množství. Ostatní údaje jsou také předvyplněny přímo v definici.



Přílohy:
Importní data: 005_Soubor_Prijemka.xlsx
Definice definovatelných importů: 005_DEF_XLS_Prijemka.xml

Poznámka: V příkladech může zdánlivě chybět popsaná definice 004. Ta je však popsána v posledním příkladu 6, a to proto, že nenavazuje na příklady 2,3,4 a 5, ale stojí samostatně. Z tohoto důvodu je v příkladu 4 použita definice 005 a v příkladu 5 použita definice 006.

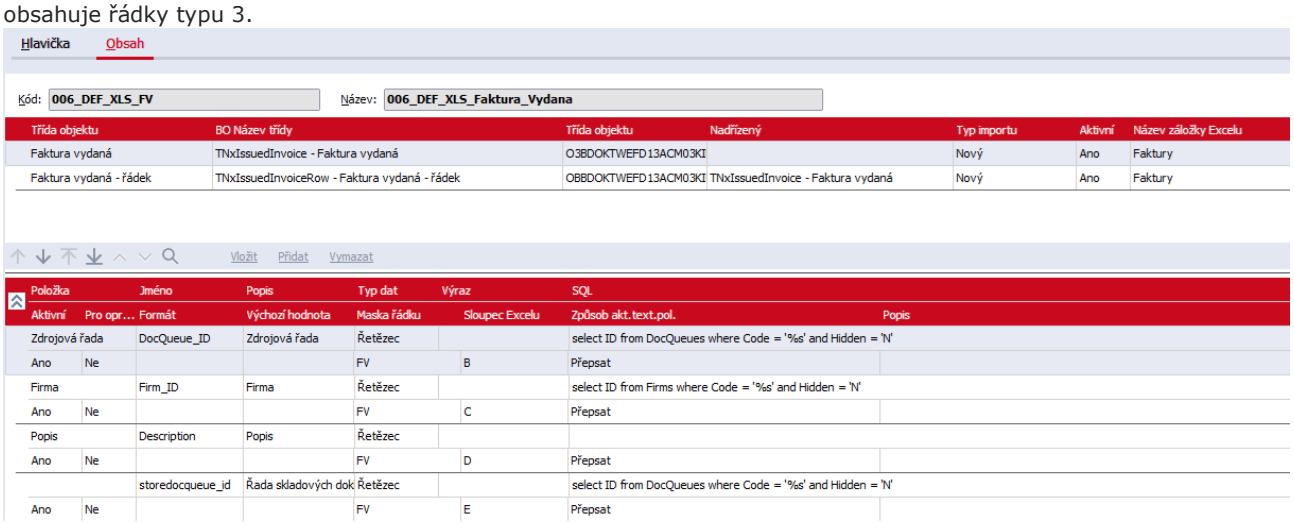
5) Příklad definice faktury vydané

Příklad pro import nové Faktury vydané. Pro její správné fungování u řádků typu 3 je nutné, aby existovaly použité artikly a byly naskladněny. Proto jsme v příkladu číslo 4 provedli nejprve import příjemky, naskladnili artikly a nyní na ně vytvoříme fakturu vydanou.

Příklad importních dat:

	A	B	C	D	E	F	G	H	I
1	Hlavicka faktury	kod rady	Firma	popis	Dodací listy				
2	FV	FV	0	TEST Faktura v	DL				
3	radky faktury	Typ radku	Sklad	Skladova karta	Množství	Jednotka	J. cena	DPH sazba	kod strediska
4	ROWFV	1	01	OIE/1101/XLS	10	ks	100	21	000
5	ROWFV	2	01	OIE/1102/XLS	10	ks	200	21	000
6	ROWFV	3	01	OIE/1103/XLS	10	ks	300	21	000
7	Hlavicka faktury	kod rady	Firma	popis	Dodací listy				
8	FV	FV	00016		DL				
9	radky faktury	typ radku					J. cena	DPH sazba	kod strediska
10	ROWFV	3	01	OIE/1101/XLS	5	ks	1.95	21	000
11	ROWFV	3	01	OIE/1102/XLS	15	ks	5.63	21	000
12	ROWFV	3	01	OIE/1103/XLS	20	ks	4.96	21	000
13	ROWFV	3	01	OIE/1104/XLS	25	ks	5.69	21	000

Hlavička Faktury vydané se kompletně dotahuje z Excelové tabulky. Zapomenout se tam nesmí vyplnit také kód pro výdejky, protože vyplnění výdejky jsou v IS FLORES povinné při vytváření faktury, pokud doklad obsahuje řádky typu 3.



V řádcích se pak ve sloupci SQL vyskytují SQL dotazy, kterými se doptáváme na konkrétní položky. SQL dotaz na DPH sazbu je uveden nikoliv ve sloupci SQL, ale ve sloupci výraz. Důvodem, proč je SQL ve výrazu je ten, že je zde použita funkce CfxStrToFloatDef pro převod řetězce na desetinné číslo.

Hlavička Obsah

Kód: 006_DEF_XLS_FV Název: 006_DEF_XLS_Faktura_Vydana

Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu
Faktura vydaná	TNxIssuedInvoice - Faktura vydaná	O3BDOKTWEFD13ACM03KI		Nový	Ano	Faktury
Faktura vydaná - řádek	TNxIssuedInvoiceRow - Faktura vydaná - řádek	OBBDOKTWEFD13ACM03KI	TNxIssuedInvoice - Faktura vydaná	Nový	Ano	Faktury

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL			
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text.pol.	Popis	
Typ řádku		RowType	Typ řádku	Celé číslo				
Ano	Ne			ROWFV	B	Přepsat		
Sklad		Store_ID	Sklad	Řetězec		select ID from Stores where Code = '%s'		
Ano	Ne		2100000101	ROWFV	C	Přepsat		
Artikl		StoreCard_ID	Skladová karta	Řetězec		select ID from StoreCards where Code = '%s'		
Ano	Ne			ROWFV	D	Přepsat		
Počet v ev.jedn.		Quantity	Počet v ev.jedn.	Řetězec				
Ano	Ne			ROWFV	E	Přepsat		
Jednotka		QUnit	Jednotka	Řetězec				
Ano	Ne			ROWFV	F	Přepsat		
J.cena		UnitPrice	J.cena	Desetinné číslo				
Ano	Ne	.		ROWFV	G	Přepsat		
%DPH		VATRate_ID	%DPH	Řetězec	NxSQLSelect('select ID fro			
Ano	Ne			ROWFV	H	Přepsat		
Středisko		Division_ID	Středisko	Řetězec		select ID from Divisions where Code = '%s' and Hidden = 'N'		
Ano	Ne			ROWFV	I	Přepsat		

Přílohy:
Importní data: 006_Soubor_Faktura_Vydana.xlsx
Definice definovatelných importů: 006_DEF_XLS_Faktura_Vydana.xml

6) Příklad vytvoření obalu k artiklu

V tomto příkladu importu vytváříme zcela nové artikly typu obal včetně jednotek.

Příklad importních dat:

	A	B	C	D	E	F
1	OBAL ARTIKLU: Maska řádku	Zkratka jednotky	Kod obalu	Množství	Excel propojení	
2	CONT	bal	OIE/2005/XLS	2	Karta-1	
3	CONT	m	OIE/2004/XLS	5	Karta-2	
4	CONT	bal	OIE/2007/XLS	3	Karta-2	
5						
6						

BO Artikl
Téměř celý se dotahuje z importního souboru až na Typ a sazbu DPH.
První řádek tvoří Excel propojení, které musíme použít, pokud importujeme z více záložek.

Hlavička Obsah

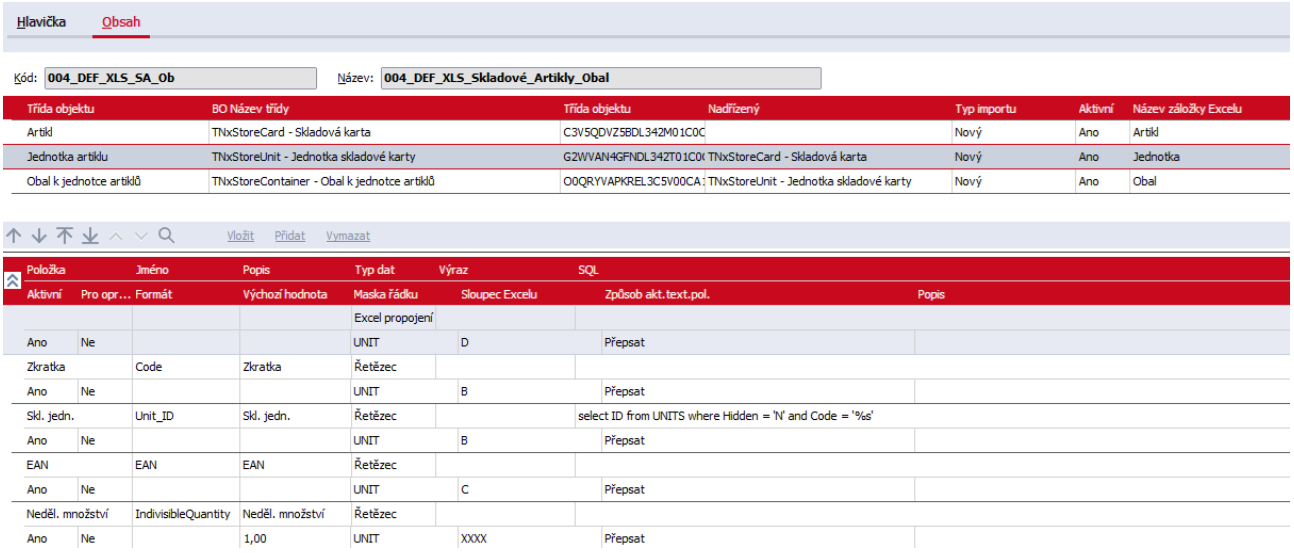
Kód: 004_DEF_XLS_SA_Ob Název: 004_DEF_XLS_Skladové_Artikly_Obal

Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano	Artikl
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano	Jednotka
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	Obal

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL		
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text.pol.	Popis
				Excel propojení			
Ano	Ne			SC	E	Přepsat	
Kód		Code	Kód	Řetězec			
Ano	Ne			SC	B	Přepsat	
Název		Name	Název	Řetězec			
Ano	Ne			SC	C	Přepsat	
Třída		Category	Třída	Řetězec			
Ano	Ne			SC	D	Přepsat	
Typ		StoreCardCategory_	Typ	Řetězec			
Ano	Ne		1100000101	SC	XXXX	Přepsat	
		VATRate_ID	%DPH	Řetězec			
Ano	Ne		02100X0000	SC	XXXX	Přepsat	

BO Jednotka artiklu
Na prvním řádku je použito excel propojení. SQL dotazem ve druhém řádku se doptáváme na kód jednotky artiklu.


Hlavička Obsah

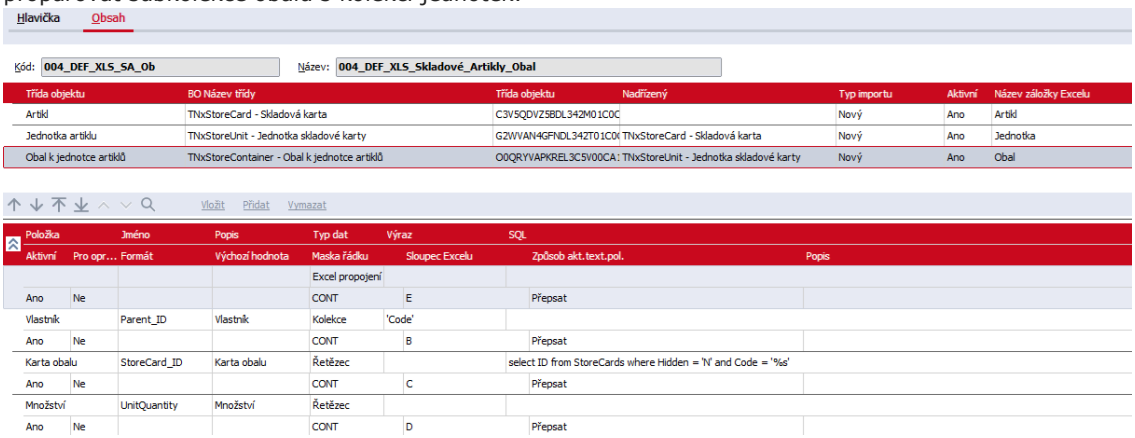
Kód: 004_DEF_XLS_SA_Ob Název: 004_DEF_XLS_Skladové_Artikly_Obal

Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano	Artikl
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano	Jednotka
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA:	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	Obal

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL		
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text.pol.	Popis
				Excel propojení			
Ano	Ne			UNIT	D	Přepsat	
Zkratka		Code	Zkratka	Řetězec			
Ano	Ne			UNIT	B	Přepsat	
Skl. jedn.		Unit_ID	Skl. jedn.	Řetězec		select ID from UNITS where Hidden = 'N' and Code = '%s'	
Ano	Ne			UNIT	B	Přepsat	
EAN		EAN	EAN	Řetězec			
Ano	Ne			UNIT	C	Přepsat	
Neděl. množství		IndivisibleQuantity	Neděl. množství	Řetězec			
Ano	Ne		1,00	UNIT	XXXX	Přepsat	

BO obal k jednotce artiklu
První řádek Excel propojení. V druhém řádku musíme nastavit kolekci a Parent_ID, protože potřebujeme propárovat subkolekce obalů s kolekcí jednotek.


Hlavička Obsah

Kód: 004_DEF_XLS_SA_Ob Název: 004_DEF_XLS_Skladové_Artikly_Obal

Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu
Artikl	TNxStoreCard - Skladová karta	C3V5QDVZ5BDL342M01C0C		Nový	Ano	Artikl
Jednotka artiklu	TNxStoreUnit - Jednotka skladové karty	G2WVAN4GFNDL342T01C0(	TNxStoreCard - Skladová karta	Nový	Ano	Jednotka
Obal k jednotce artiklů	TNxStoreContainer - Obal k jednotce artiklů	O0QRYVAPKREL3C5V00CA:	TNxStoreUnit - Jednotka skladové karty	Nový	Ano	Obal

Vložit Přidat Vymazat

Položka	Jméno	Popis	Typ dat	Výraz	SQL		
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text.pol.	Popis
				Excel propojení			
Ano	Ne			CONT	E	Přepsat	
Vlastník		Parent_ID	Vlastník	Kolekce	'Code'		
Ano	Ne			CONT	B	Přepsat	
Karta obalu		StoreCard_ID	Karta obalu	Řetězec		select ID from StoreCards where Hidden = 'N' and Code = '%s'	
Ano	Ne			CONT	C	Přepsat	
Množství		UnitQuantity	Množství	Řetězec			
Ano	Ne			CONT	D	Přepsat	

Třetím řádkem dohledáváme artikl, ke kterému bude obal patřit. Dohledání probíhá pomocí SQL dotazu.

Přílohy:
Importní data: 004_Soubor_Skladové_Artikly_Obal.xlsx
Definice definovatelných importů: 004_DEF_XLS_Skladové_Artikly_Obal.xml

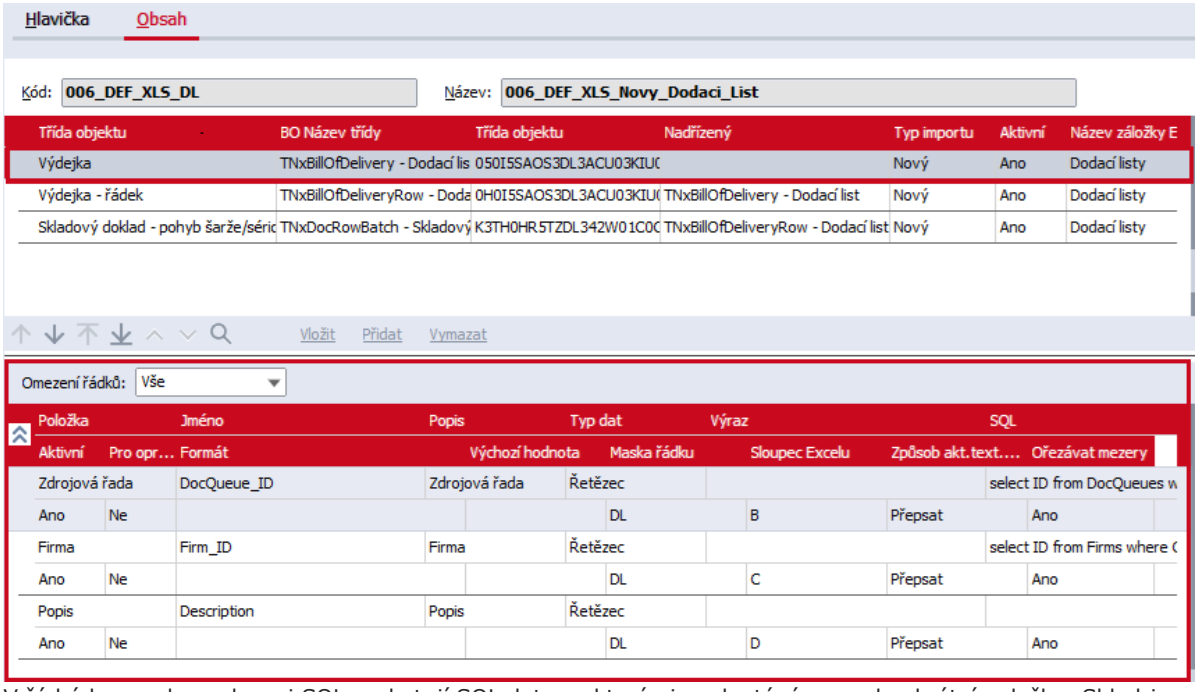
7) Příklad definice výdejky z jednoho listu excelu

V rámci funkce Export mohou být data exportována z ABRA FLORES tak, že jeden doklad bude mít hlavičku, řádky a sériová čísla pod sebou na řádcích tak, jak to vidíte na obrázku níže. Z tohoto důvodu zde uvádíme příklad na import právě takových dat, vyexportovaných stejným způsobem jako z ABRA FLORES.

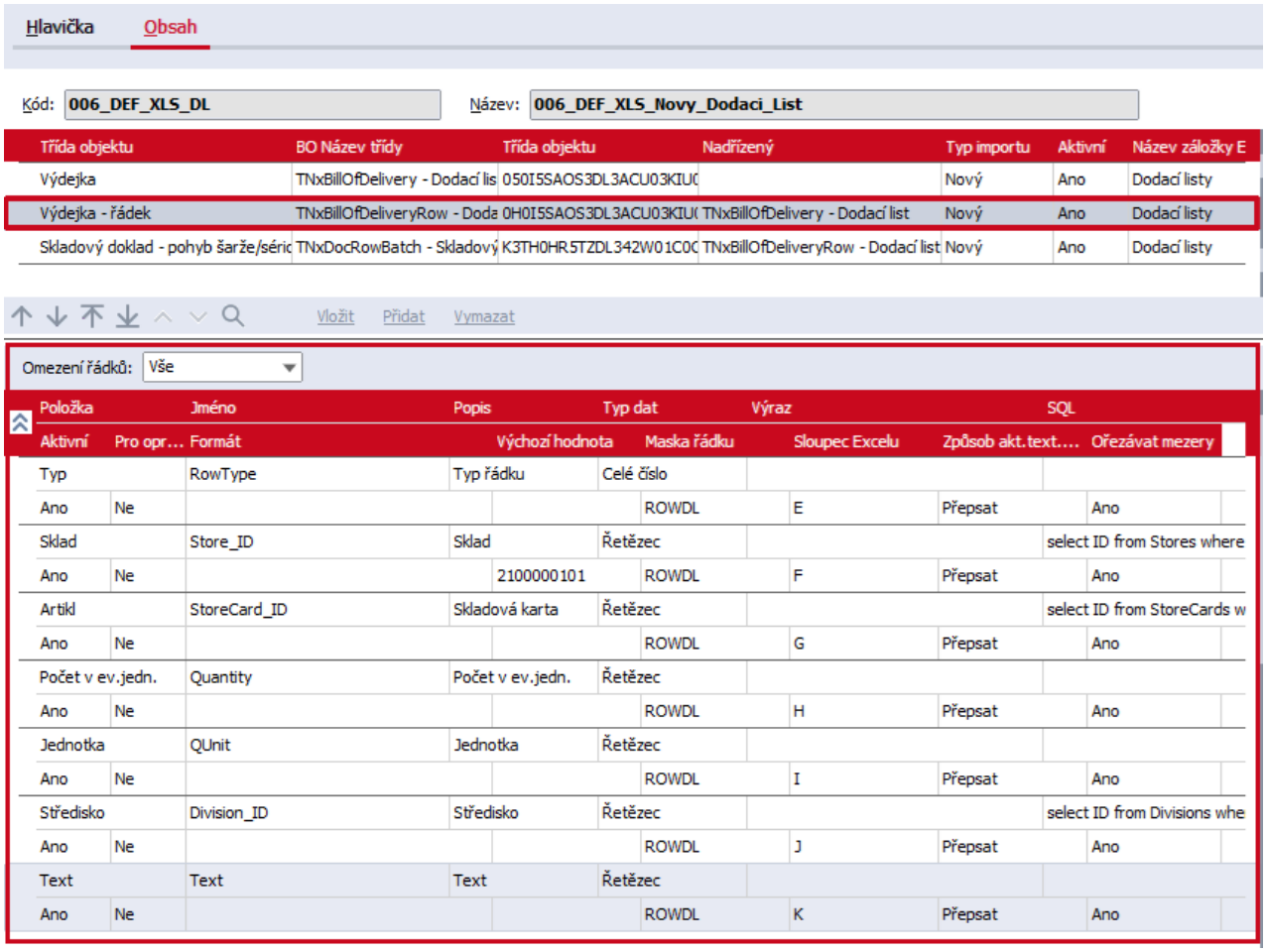
Příklad importních dat:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Typ	Řada dokladů	Kód firmy	Popis	Typ řádku	Kód skladu	Kód karty	Počet	Jednotka	Kód střediska	Text / Odkaz na řádek z šarže		Sériové číslo/šarže (ID)
2	DL	DL	0	TEST Dodací list									
3	ROWDL	DL	0		1					000	TEST Text řádku typu 1		
4	ROWDL	DL	0		2			10	ks	000	TEST Text řádku typu 2		
5	ROWDL	DL	0		3	01	OIE/2103/XLS	10	ks	000			
6													
7	DL	DL	00016										
8	ROWDL	DL	00016		3	01	OIE/2101/XLS	5	ks	000			
9	ROWDL	DL	00016		3	01	OIE/2102/XLS	15	ks	000			
10	ROWDL	DL	00016		3	01	OIE/2103/XLS	20	ks	000			
11	ROWDL	DL	00016		3	01	OIE/2104/XLS	25	ks	000			
12													
13													
14	DL	DL	00011										
15	ROWDL	DL	00011		3	01	10	3	ks	000	DL_ROW_1		
16	ROWDL	DL	00011		3	01	10	3	ks	000	DL_ROW_2		
17	BATCHDL	DL	00011		3	01	10	1	ks	000	DL_ROW_1		3300000101
18	BATCHDL	DL	00011		3	01	10	1	ks	000	DL_ROW_1		2300000101
19	BATCHDL	DL	00011		3	01	10	1	ks	000	DL_ROW_1		1300000101
20	BATCHDL	DL	00011		3	01	10	1	ks	000	DL_ROW_2		E300000101
21	BATCHDL	DL	00011		3	01	10	1	ks	000	DL_ROW_2		F300000101
22	BATCHDL	DL	00011		3	01	10	1	ks	000	DL_ROW_2		D300000101

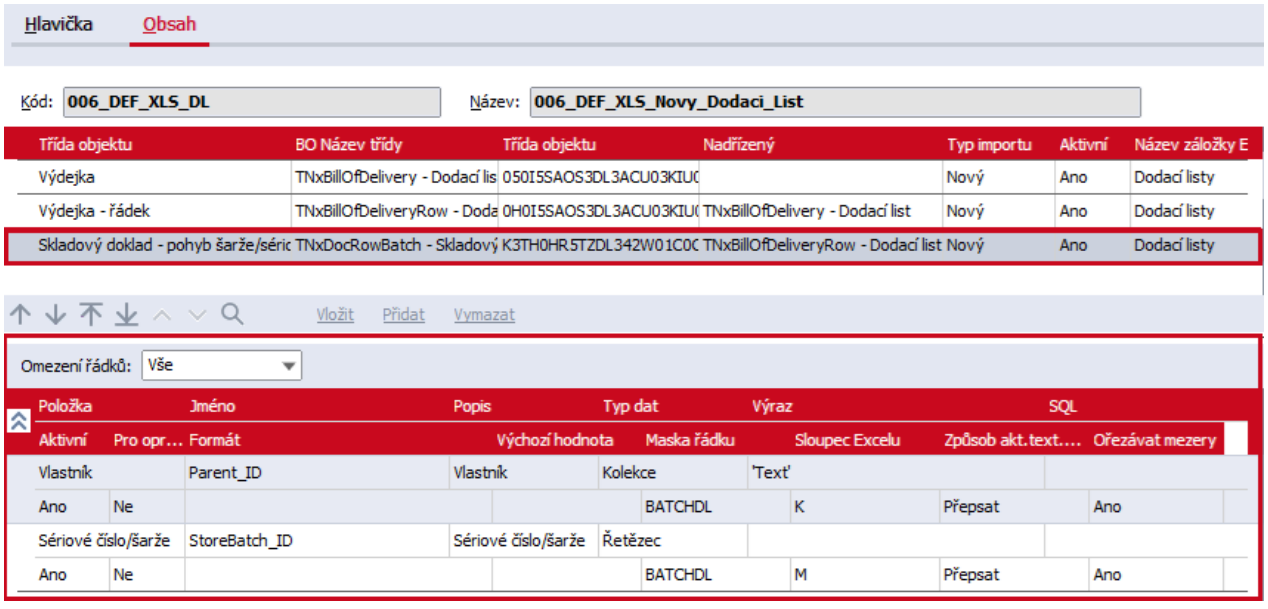
Hlavička výdejky se kompletně dotahuje z Excelové tabulky. Zdrojová řada a firma se dohledávají v excelu pomocí SQL dotazů. Název záložky excelu je dodací listy. Pro rozlišení hlavičky od ostatních hodnot se používá maska řádku DL.



V řádcích se pak ve sloupci SQL vyskytují SQL dotazy, kterými se doptáváme na konkrétní položky. Sklad je zadán výchozí hodnotou pro případ, pokud by v excelu nebyla hodnota vyplněna. Pro rozlišení řádků od ostatních hodnot se používá maska řádku ROWDL.



Sériová čísla se také importují z XLS sešitu. Pro rozlišení sériových čísel od ostatních hodnot se používá maska řádku BATCHDL V Excelu jsou sériová čísla uvedena jako ID existujících sériových čísel v ABRA FLORES.



Přílohy:
Importní data: 006_Soubor_Dodaci_Listy.xlsx
Definice definovatelných importů: 006_DEF_XLS_DL_FL.XM

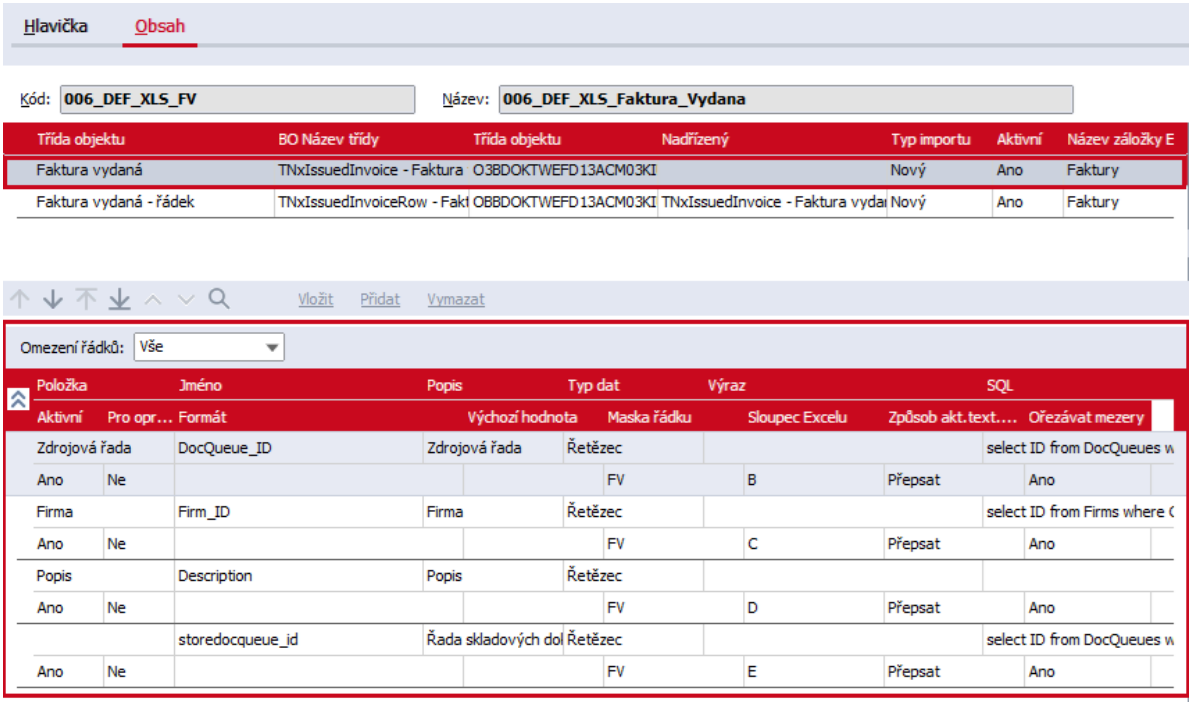
8) Příklad definice faktury vydané z jednoho listu excelu

V rámci funkce Export mohou být data exportována z ABRA FLORES tak, že jeden doklad bude mít hlavičku a řádky pod sebou na řádcích tak, jak to vidíte na obrázku níže. Z tohoto důvodu zde uvádíme příklad na import právě takových dat, vyexportovaných stejným způsobem jako z ABRA FLORES.

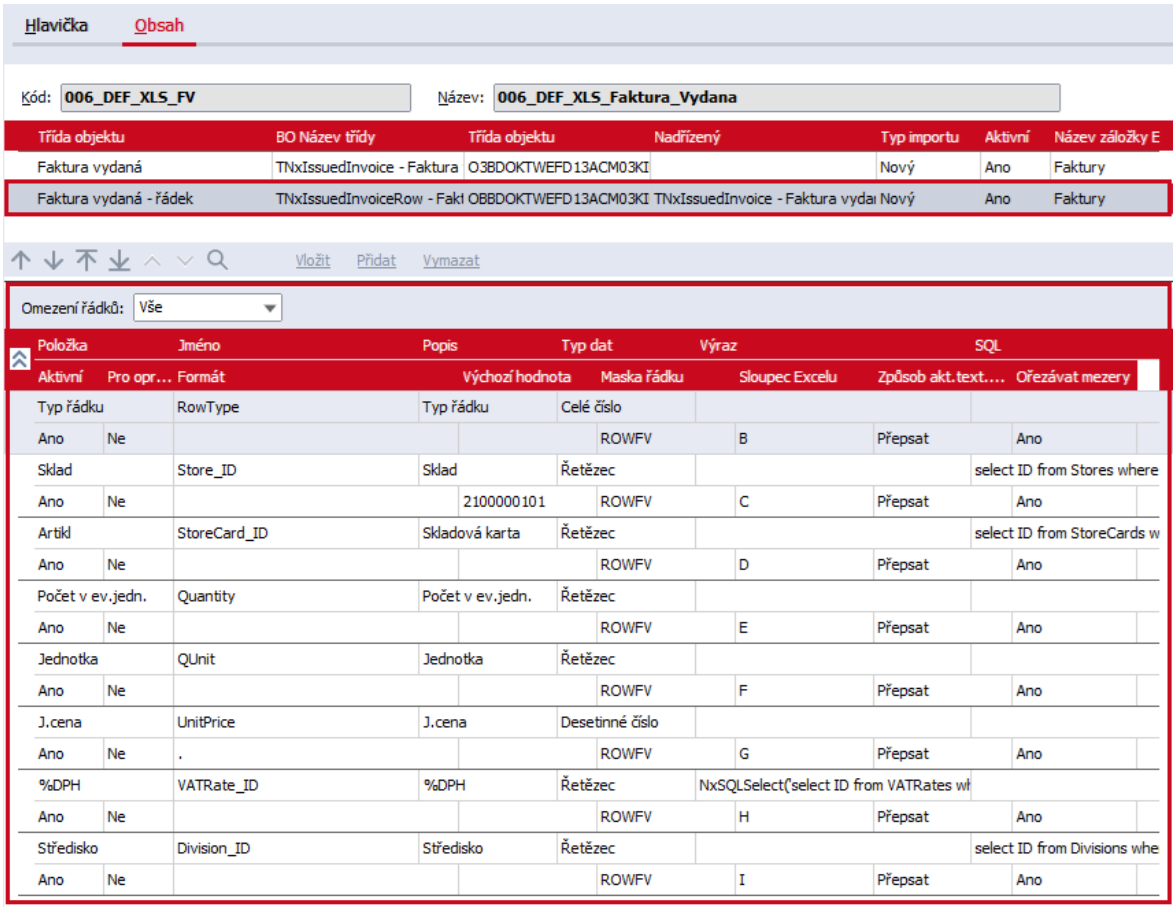
Příklad importních dat:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Typ	Řada dokladů	Kód firmy	Popis	Řada dodacích listů	Typ řádku	Kód skladu	Kód karty	Počet	Jednotka	Cena za jednotku	DPH	Kód střediska
2	FV	FV	0	TEST Faktura vydaná	DL								
3	ROWFV	FV	0	TEST Faktura vydaná	DL	1			1	ks	1000	21	000
4	ROWFV	FV	0	TEST Faktura vydaná	DL	2			10	ks	200	21	000
5	ROWFV	FV	0	TEST Faktura vydaná	DL	3	01	OIE/2103/XLS	10	ks	300	21	000
6													
7	FV	FV	00016		DL								
8	ROWFV	FV	00016		DL	3	01	OIE/2101/XLS	5	ks	1,95	21	000
9	ROWFV	FV	00016		DL	3	01	OIE/2102/XLS	15	ks	5,63	21	000
10	ROWFV	FV	00016		DL	3	01	OIE/2103/XLS	20	ks	4,96	21	000
11	ROWFV	FV	00016		DL	3	01	OIE/2104/XLS	25	ks	5,69	21	000

Hlavička faktury vydané se kompletně dotahuje z Excelové tabulky. Zdrojová řada, firma a řada skladových dokladů se dohledávají v excelu pomocí SQL dotazů. Název záložky excelu je Faktury. Pro rozlišení hlavičky od ostatních hodnot se používá maska řádku FV



V řádcích se ve sloupci SQL vyskytují SQL dotazy, kterými se doptáváme na konkrétní položky. Sklad je zadán výchozí hodnotou pro případ, že by nebyl ve zdroji dat vyplněn. Pro rozlišení řádků od ostatních hodnot se používá maska řádku ROWFW.



Přílohy:
Importní data: 007_Soubor_Faktury.xlsx
Definice definovatelných importů: 006_DEF_XLS_DL_FL.XML

9) Příklad importu kalendáře do agendy Pracovních kalendářů

Importem jsou přidávány řádky s datem směny a následně mohou být k směnám naimportovány přestávky, pokud je potřeba je evidovat. Z pohledu systému ABRA Gen je přestávku potřeba evidovat jen u noční směny. Směny je možné importovat i pro budoucí období.

V importním souboru jsou očekávány 3 typy řádků:

- **AHLPK** – pro identifikace pracovního kalendáře v systému ABRA Gen
- **BSPK** – hodnoty pro import na záložku „Kalendář" pro odpracované/plánované směny, svátek místo směny nebo směnové volno
- **CPSPK** – hodnoty pro evidenci čerpané přestávky (nutné pouze pro noční směny přes půlnoc)

POZOR
BSPK – směny – Definují-li se přestávky pozor na zadávání začátku směny vzhledem počtu odpracovaných hodin, po kterých začíná přestávka. Pokud se nedefinují přestávky musí se délka směny rovnat počtu odpracovaných hodin.

Import umožňuje načítat vždy jen jednu záložku Excelovského souboru, přeskakuje první tři řádky, kde mohou být o v hlavičkové popisy sloupců. Název listu v souboru lze ručně upravit v Definici definovatelných importů. Ve výchozím stavu předpokládáme název listu „K**alendarSmenaPrestavka**".

Před inportem musí existovat hlavička Pracovního kalendáře se stejným kodem a týdenním pracovním úvazkem jako v importovaném souboru. Vzor po generování nesmí být vyplněn.

Pro import je nutné mít založenou řadu dokladů typu LO (logy) se zkratkou a názven LOGIE. (Před 2024 neexistovala.)

Současně s importem se zapisuje do uvedené řady log s popisem průběhu importu.

Sloupce pro masku AHLPK

Sloupec XLS	Název sloupce	Datový typ	Hodnota	Význam
A	Maska	String	AHLPK	Určuje kalendář, do kterého se importuje
B	Kód	String	10 znaků	Kód pracovního kalendáře (povinné)
C	Název	String	Max. 50 znaků	Název kalendáře (nepovinné)
D	Týdenní úvazek	Numeric (10/2)	Př. 37,50; 38,75; 40,00; 20,00	Úvazek (povinné)

Sloupce pro masku BSPK

Sloupec XLS	Název sloupce	Datový typ	Hodnota	Význam
A	Maska	String	BSPK	Určuje směny, které se importují
B	Datum směny	Date	dd.MM.YYYY	Datum směny (povinné)
C	Typ plánované směny	String	Hodnoty: S; SV; V	S – směna, SV – svátek místo směny, V – volno (celý den) (povinné) pozn. V jeden den může být pouze plánovaná směna nebo volno místo směny. Mix zatím nedovolujeme.
D	Hodina od	Time	HH:MM Př. 06:00; 08:00; 18:00	V kolik hodin směna nebo svátek místo směny začíná. (povinné pro typ S) Prázdné pro V a SV.
E	Délka	Numeric	#00,00 Př. 8,75; 8,30	Délka směny včetně přestávek. (povinné pro typ S) Prázdné pro V a SV.
F	Hodiny	Numeric	#00,00 Př. 7,75; 8,00	Počet odpracovaných hodin. (povinné pro typ S) Prázdné pro V a SV.
G	V noci	Numeric	#00,00 Př. 8,00	Počet hodin pro příplatek za práci v noci. (nepovinné) Prázdné pro V a SV.
H	V prostředí	Numeric	#00,00 Př. 4,00	Počet hodin pro příplatek za práci ve škodlivém prostředí. (nepovinné) Prázdné pro V a SV.
I	Ve svátek	Numeric	#00,00 Př. 7,75	Počet hodin volna ve svátek místo plánované směny (povinné pro typ SV) Prázdné pro S a V.

Sloupce pro masku CPSPK

Nemusí být vyplněno, pokud se u BSPK hodnoty Délka a Hodiny rovnají. Pokud se délka směny a počet odpracovaných hodin nerovná, potom přestávka musí být definovaná!

Sloupec XLS	Název sloupce	Datový typ	Hodnota	Význam
A	Maska	String	CPSPK	Určuje přestávky, které se importují, k již importovaným směnám
B	Datum směny	Date	dd.MM.YYYY	Datum směny (povinné)
C	Začátek přestávky od	Time	HH:MM Př. 22:00; 02:00	V kolik hodin přestávka začíná (nepovinné)
D	Začátek po	Numeric	#00,00 Př. 4,00; 8,00	Počet hodin od začátku směny. (povinné) Musí platit, že začátek směny „BSPK.Hodina od" plus CPSPK.Začátek po" se musí rovnat CPSPK.Začátek přestávky od (hodině Začátku přestávky)
E	Délka	Numeric	#00,00 Př. 0,50; 1,00	Délka přestávky (povinná hodnota)

Přílohy:
Importní data: 008_Soubor_pro_XLS_PracovniKalendar_VzorStruktury.xlsx
Definice 008_DEF_XLS_O_PracovniKalendar_I-O_Smena_Prestavky.xml
Další soubory: 008_100.xlsx, 008_PRÁZDNY VZOR XLS.xlsx

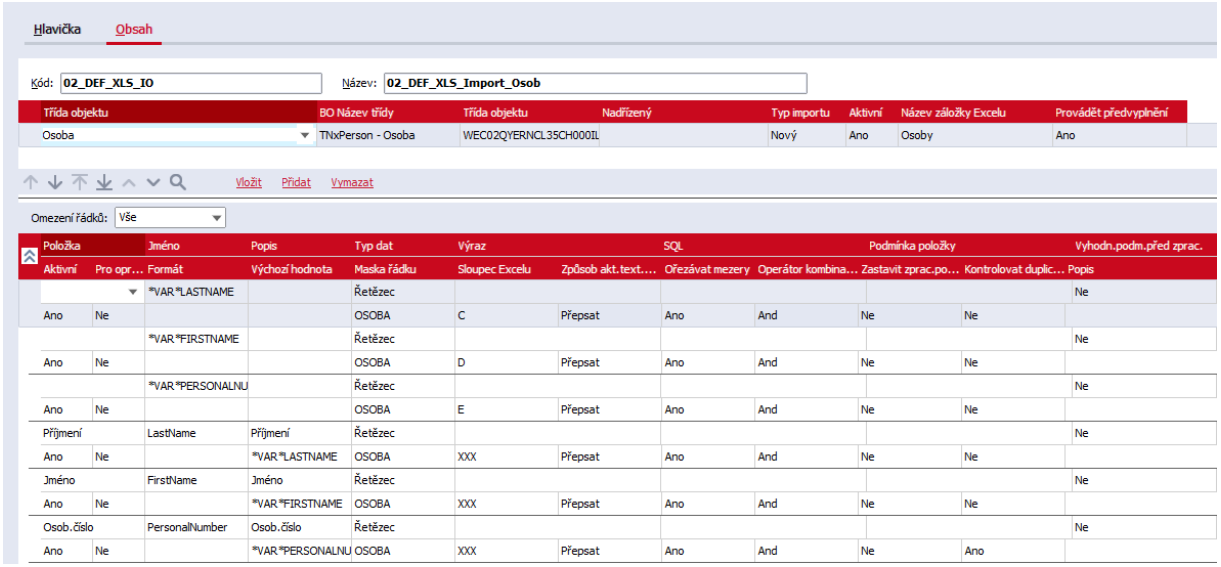
10) Příklad vytvoření Osob a jejich následný import k firmám.

Tento příklad obsahuje dva importy, které pokud se provedou ručně za sebou nebo mohou být součástí agendy Sady importních a exportních definic, tak naimportují osoby, které následně přiřadí k firmám.
Příklad importních dat (záložka Osoby a záložka Firmy):

	Maska OSOBA	Firma	Příjmení	Jméno	Osob.číslo
2	OSOBA	00011	Metelka	Jarka	101
3	OSOBA	00011	Metelka	Jarka	102
4	OSOBA	00011	Čert	Petr	999
5	OSOBA	00011	Kuchta	Václav	004
6	OSOBA	00011	Rasák	Daniel	005
7	OSOBA	00015	Metelka	Jarka	101
8	OSOBA	00015	Metelka	Jarka	102
9	OSOBA	00015	Hojer	Jindra	109
10	OSOBA	00015	Kuchta	Václav	004
11	OSOBA	00015	Rasák	Daniel	005
12	OSOBA	00016	Kusá	Eva	006
13	OSOBA	00016	Přesný	Jaroslav	015
14	OSOBA	00016	Kladivoun	Zdeněk	017
15	OSOBA	00016	Bohém	Tadeáš	018
16	OSOBA	00016	Svoboda	Erik	019
17	OSOBA	00016	Pečlivá	Jaroslava	020

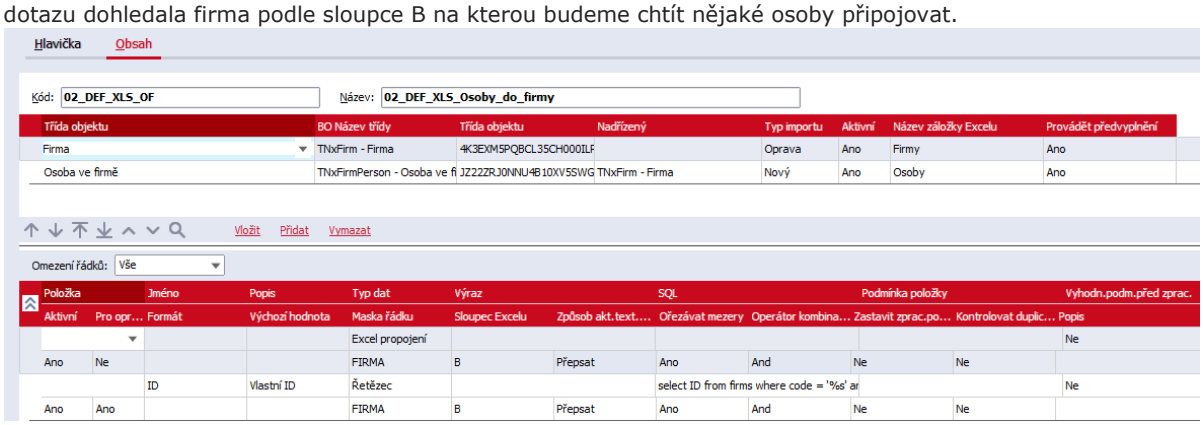
	A	B	C
1	Maska FIRMA	Firma	
2	FIRMA	00011	
3	FIRMA	00015	
4	FIRMA	00016	
5	FIRMA	00017	
6	FIRMA	00019	
7			
8			

Import s názvem 02_DEF_XLS_Import_Osob nejprve importuje osoby. Jednotlivé položky jsou načítány do Proměnných (z důvodu jednoznačnosti) z přiřazených buněk a následně jsou jako výchozí hodnoty použity pro naplnění jednotlivých položek.

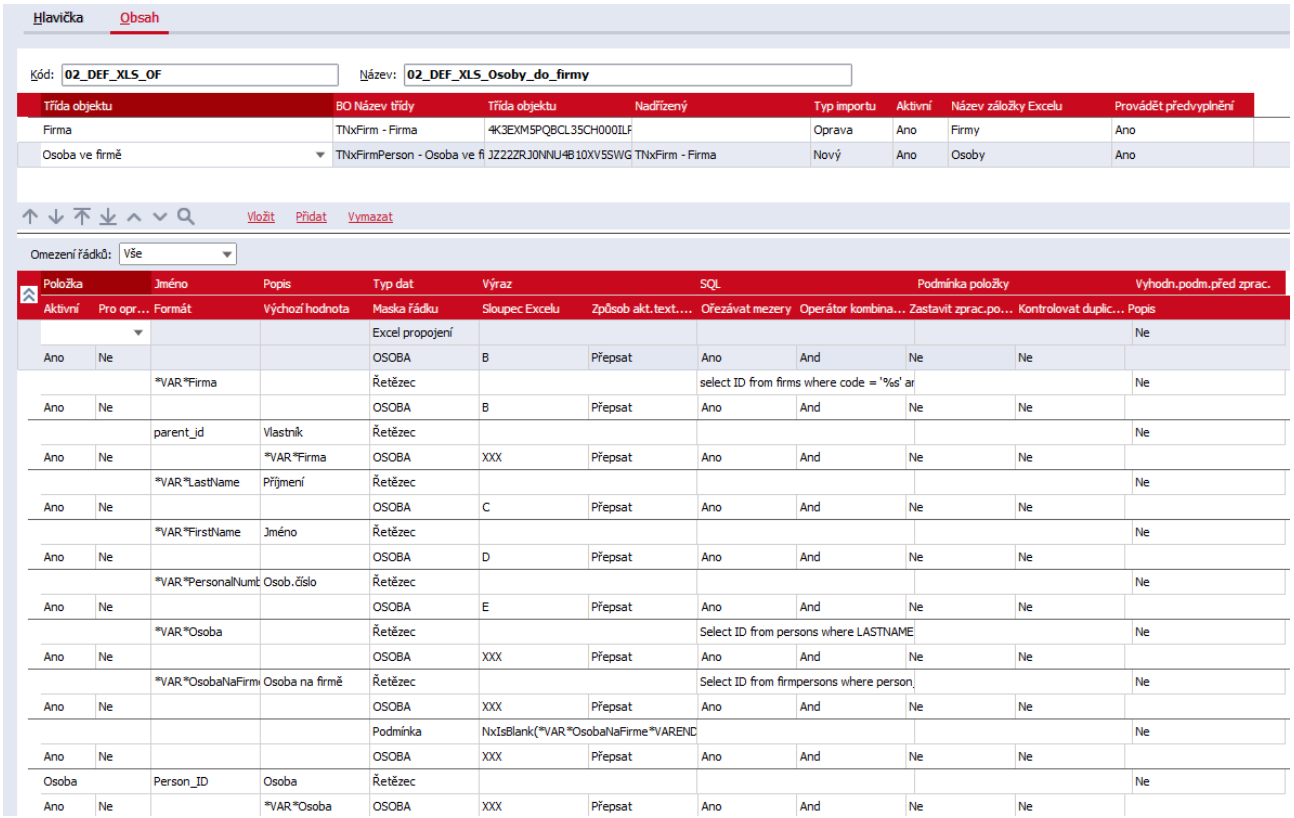


Po následném importu lze manuálně nebo automaticky přes agendu Sady importních a exportních definic vyvolat import, který vytvořené osoby přiřadí do jednotlivých firem.

Import z názvem 02_DEF_XLS_Osoby_do_firmy má dva Business objekty (BO). První Firma – je propojena v excelem přes identifikaci firmy, sloupec B – firma. Druhý řádek zde slouží k tomu, aby se pomocí SQL dotazu dohledala firma podle sloupce B na kterou budeme chtít nějaké osoby připojovat.



Druhý BO Osoba na firmě se objet nejprve propojí přes identifikační sloupec v Excelu B, tedy kód firmy. Následuje načtení firmy do proměnné a následné načtení proměnní do parent_ID. Dále se naplní do proměnných jméno, příjmení, osobní číslo a tyto údaje se následně použijí při naplnění proměnné osoby,kde se použije SQL dotaz Select ID from persons where LASTNAME=*VAR*LASTNAME*VAREND* AND FIRSTNAME=*VAR*FIRSTNAME*VAREND* AND PERSONALNUMBER=*VAR*PERSONALNUMBER*VAREND* and Hidden = 'N'
Následně je použita podmínka exportu. Ta zajistí že pokud osoba na firmě není, tak ta m bude doplněna. Pokud tam již připojena je, nebude se připojovat znova.



Hlavička Obsah

Kód: 02_DEF_XLS_OF Název: 02_DEF_XLS_Osoby_do_firmy

Třída objektu	BO Název třídy	Třída objektu	Nadřízený	Typ importu	Aktivní	Název záložky Excelu	Provádět předvyplnění
Firma	TNxFirm - Firma	4K3EXM5PQBCL35CH000ILF		Oprava	Ano	Firmy	Ano
Osoba ve firmě	TNxFirmPerson - Osoba ve fi	JZ22ZRJ0NNU4B10XV5SWG	TNxFirm - Firma	Nový	Ano	Osoby	Ano

Vložit Přidat Vymazat

Omezení řádků: Vše

Položka	Jméno	Popis	Typ dat	Výraz		SQL		Podmínka položky		Vyhodn.podm.před zprac.
Aktivní	Pro opr...	Formát	Výchozí hodnota	Maska řádku	Sloupec Excelu	Způsob akt.text....	Ořezávat mezery	Operátor kombina...	Zastavit zprac.po...	Kontrolovat duplic...	Popis
				Excel propojení							Ne
Ano	Ne			OSOBA	B	Přepsat	Ano	And	Ne	Ne	
		*VAR*Firma		Řetězec			select ID from firms where code = '%s' ar				Ne
Ano	Ne			OSOBA	B	Přepsat	Ano	And	Ne	Ne	
		parent_id	Vlastník	Řetězec							Ne
Ano	Ne		*VAR*Firma	OSOBA	XXX	Přepsat	Ano	And	Ne	Ne	
		*VAR*LastName	Příjmení	Řetězec							Ne
Ano	Ne			OSOBA	C	Přepsat	Ano	And	Ne	Ne	
		*VAR*FirstName	Jméno	Řetězec							Ne
Ano	Ne			OSOBA	D	Přepsat	Ano	And	Ne	Ne	
		*VAR*PersonalNumb	Osob.číslo	Řetězec							Ne
Ano	Ne			OSOBA	E	Přepsat	Ano	And	Ne	Ne	
		*VAR*Osoba		Řetězec			Select ID from persons where LASTNAME				Ne
Ano	Ne			OSOBA	XXX	Přepsat	Ano	And	Ne	Ne	
		*VAR*OsobaNaFirm	Osoba na firmě	Řetězec			Select ID from firmpersons where person				Ne
Ano	Ne			OSOBA	XXX	Přepsat	Ano	And	Ne	Ne	
				Podmínka	NxIsBlank(*VAR*OsobaNaFirme*VAREND						Ne
Ano	Ne			OSOBA	XXX	Přepsat	Ano	And	Ne	Ne	
Osoba		Person_ID	Osoba	Řetězec							Ne
Ano	Ne		*VAR*Osoba	OSOBA	XXX	Přepsat	Ano	And	Ne	Ne	

Přílohy:
Importní data: Soubor_Osoby_k_firmě.xlsx
Definice importu osoby: 02_DEF_XLS_Import_Osob.xml
Definice importu osoby na firmu: 02_DEF_XLS_Osoby_do_firmy.xml
Soubor instalační sady, který přidá obě definice do agendy Sady importních a exportních definic:
Osoby_k_firme_V2.ais