



informační systémy



VarioSoap

Technická dokumentace webových služeb

Platné k verzi SOAP: 0.0.6.241 (15.03.2022)

Obsah

Obecný popis řešení.....	3
Instalace	3
Instalace triggerů a funkcí do databází Varia.....	4
Instalace a nastavení služby.....	4
Nastavení INI souboru	4
Obsah INI souboru	5
Instalovaná služba	7
Aktualizace služby.....	7
Objekty	7
Čtení kompletních dat	7
Čtení změněných dat	7
Metody.....	8
Čtení dat	9
Čtení jednotlivých objektů	9
Čtení kompletních dat	11
Čtení změněných dat.....	13
Zápis dat.....	14
Výjimky.....	16
Objekty.....	17
Přehled aktualizací	36

Obecný popis řešení

V principu jde o službu, která je spuštěna na serveru. Tato služba zajišťuje komunikaci s databází Altus VARIO. Libovolná aplikace (jiný IS, e-shop, web) pak tuto službu může volat přes internet. Služba vrací změny v datech, ve struktuře, která je nezávislá na VARIU.

VarioSOAP je konektor umožňující přístup k datům AltusVario především pro e-shopy. Komunikace je pomocí SOAP rozhraní a jeho metod. Rozhraní SOAP serveru je přístupné přes adresu serveru na definovaném portě (obvykle 1024), VarioSOAP je tak dostupné např. na <http://server:1024/soap/IVarioSOAPService>. WSDL popis je pak např. na <http://server:1024/wsdl/IVarioSOAPService>

Rozhraní poskytuje změny dat, které proběhnou v databázi AltusVario a umožňuje zakládat nové adresy a doklady. Pokud dojde k chybě ve čtení nebo zápisu dat, je vyvolána výjimka.

Hlavní charakteristika:

- založeno na vzdáleném volání procedur pomocí protokolu SOAP (TCP/IP); na serveru běží Windows služba, která poskytuje rozhraní
- služba vrací sadu objektů sloužících k přenosu dat
- jde o rozhraní nad daty, nejde o klasický přímý přístup k databázi => není nutné znát datovou strukturu
- vyšší bezpečnost
- nedochází k zamykání databáze, jak je tomu při replikacích na úrovni SQL serveru
- vrací pouze změny v databázi mezi dvěma synchronizacemi, nebo také kompletní data
- změny se dají stahovat kdykoliv, takže není problém s výpadkem spojení nebo serveru
- jednoduchost použití: třetí strana nemusí znát strukturu VARIA - služba publikuje svůj vlastní popis, který se dá programově zpracovat a tak automaticky vygenerovat základ komunikační vrstvy na straně uživatele služby
- lze volat z libovolného programovacího jazyku

Instalace

Samotná instalace SOAPu je velice jednoduchá a je tvořena 3 kroky.

- Instalace triggerů a funkcí do databází Varia
- Instalace a nastavení služby
- Nastavení INI souboru

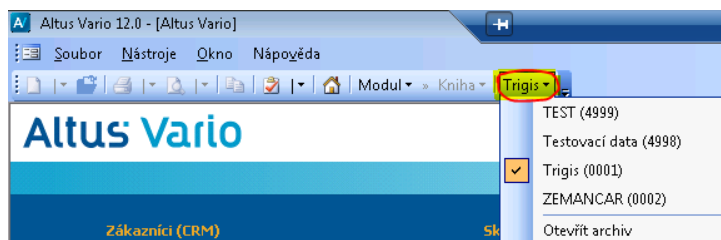
Instalace triggerů a funkcí do databází Varia

Pomocí SQL management studia je třeba spustit tyto skripty, které se nacházejí v instalační složce SOAP můstku:

```
script VarioSOAP_Data.sql
script VarioSOAP_DWPR.sql
```

V obou scriptech je před spuštěním **potřeba nahradit** všechny výskyty řetězce #### za číslo dat firmy Varia, do které se instaluje.

Číslo dat je čtyřmístný kód, který naleznete ve Variu v seznamu firem v závorkách.



Instalace a nastavení služby

Instalace musí být prováděna na stejný server, kde již běží aplikačně Vario.

Nejdříve si vytvořte adresář na serveru, do kterého se následně nakopírují programové soubory.

Adresář si můžete zvolit libovolně, vhodné je založit adresář TRIGIS_SOAP v C:\Program Files\

Do tohoto adresáře následně nakopírujte tyto soubory:

VarioSOAPService.exe	(tento soubor je v případě potřeby možné přejmenovat a umožnit tak registraci více SOAP můstků pro více firem).
trigis.dll	
VarioSOAPService.ini	(Pokud se přejmenuje VarioSOAPService.exe, je třeba ekvivalentně tomuto názvu přejmenovat i tento soubor)

Nyní spusťte v tomto adresáři pomocí příkazové řádky příkaz

```
VarioSOAPService.exe /install
```

Pokud máte zapnuté UAC, bude nutné spustit příkazový řádek přes pravé tlačítko myši „jako správce“. Tím se povolí registrace sdílených knihoven a vše proběhne korektně.

Nastavení INI souboru

Soubor VarioSOAPService.ini slouží k celkovému nastavení služby, nastavují se zde a upravují parametry důležité pro běh služby (IP adresa a port a další nezbytné parametry). Vše je detailně popsáno v INI souboru. Pokud se provede změna nastavení INI souboru, doporučujeme službu restartovat.

Je-li správně služba nainstalována, je možné ji spravovat ze sekce SLUŽBY na serveru. Zde je možné službu zastavit, případně ji restartovat nebo zapnout.

Obsah INI souboru

```
[service]
;výchozí XML Parser (DOM) - "MSXML" (výchozí), nebo "Omni XML"
;DefaultDOMVendor=Omni XML
;port, na kterém služba přijímá požadavky
port=1024
;MS SQL server
server=server\instance
;jméno a heslo pro přihlášení na MS SQL, pokud není uvedeno, použije se Windows Authentication
;user=login
;password=heslo
;číslo dat firmy ve Variu
datanumber=4999
;pokud je počet klientů větší, než 1, jde o multidoménovost
;pocetklientu=2
;maximální velikost souboru v MB pro přenos, větší soubor nebude poslán
;limitvelikostisouboru=10
;logování veškeré komunikace
logovatvse=0
;logování odchozí komunikace
logovatodchozi=0
;logování příchozí komunikace
logovatprichozi=0

;nastavení zpracování rezervací externím VariolibNastavitStav.exe
[rezervace]
;pocetopakovani=1
;v milisekundách
;intervalopakovani=500
;pocetparalelnich=5

;nastavení profilu Varia
[vario]
firma=název firmy ve variu
;cesta k profilu
profil=d:\profily\sql\_sql12
;login do Varia
uzivatel=uživatel
;heslo do Varia
heslo=
;používat adekvátní pole Varia 12 místo polí Parametrů Plus
;0=pouze Parametry Plus, 1=ParametryPlus a pole produktu z Varia, 2=pouze Vario
pouzivatpolev12mistoparametru=1
;kniha pro nové adresy
kniha_adresar=Adresář

;nastavení (změny) dotazů
[queries]
;dsProducts=select top 25 rowguid from Katalog
;AvailableSupplyStates=SELECT 'RezervovatObjednано' as stav, convert(float, Sum(pDotaz.prezervovat)) +
convert(float, Sum(pDotaz.pOBJEDNANO)) AS mnozstvi, min(pdotaz.datum) AS datum from (SELECT
PDP.Produkt as Produkt, PDP.Varianta, PDP.Sklad as Sklad, Sum(case when [PDP].[Stav]=1 then
[PDP].[Mnozstvi] else case when [PDP].[Stav] Is Null then Null else 0 end end) AS prezervovat, Sum(case when
```

```
([PDP].[Stav]=0 AND [PDP].[PV]=-1) then [PDP].[Mnozstvi] else case when ([PDP].[Stav] Is Null AND [PDP].[PV] = -1) then Null else 0 end end) AS pOBJEDNANO, Sum(case when [PDP].[Stav]=2 then [PDP].[Mnozstvi] else case when [PDP].[Stav] Is Null then Null else 0 end end) AS pPOTVRZENO, ZO.Datum FROM Polozky_dokladu_produkty as PDP with(nolock) left join (select pd.rowguid, isnull(pd.datum_dodani, d.datum) AS Datum from polozky_dokladu as pd with (nolock) inner join doklady as d with (nolock) on pd.cislo_dokladu=d.cislo_dokladu where d.typ_dokladu='ZO') AS ZO on pdp.polozka_obchodniho_dokladu=zo.rowguid WHERE Produkt=:parProdukt and Sklad=:parSklad and Varianta=:parVarianta GROUP BY PDP.Produkt, PDP.Varianta, PDP.Sklad, PDP.Stav, PDP.PV, ZO.Datum) AS pDotaz GROUP BY pdotaz.Produkt, pdotaz.sklad, pdotaz.Varianta
```

;nastavení jednotlivých typů dokladů

;zakázky

[ZZ]

kniha=Zakázky

zaznam=Zakázka

pv=1

;šablona (výchozí hodnoty) ve Variu

;šablon ve Variu pro doklad je možné mít víc, vždy začínají na nastavení [sablon],

;dál je podtržítka _ následované DIČ_ a dvěma písmeny DIČa (pokud je), případně Státem nebo ISO země,

;případně "EU", nebo "TZ" (třetí země), taková šablona se použije pro doklad s adresou odběratele daného

;státu, případně EU, pokud je stát členem, nebo TZ

;jako poslední možnost je šablona s posledními dvěma písmeny odpovídajícími prvním dvěma písmenům pole DIČ

;příklady šablon ve Variu:

;Výchozí_DIČ_SK, Výchozí_SK, Výchozí_SVK, Výchozí_EUsablona=Výchozí

;hodnota pro pole Polozky_dokladu.Vlastnosti

;polozkydokladuvlastnosti=19

;výchozí sklad položky, pokud není nastaven u produktu

sklad=

;výchozí stav položky mimo služby

vychoziStavPolozky=

;výchozí stav služby

vychoziStavPolozkySL=

;výchozí stav položky bez produktu

vychoziStavPolozkyBezProduktu=

;způsoby úhrady na zakázce pro automatické vystavení zálohové faktury, oddělené čárkou

;pokud je vlastnost Zakazka.Payed>0, uhradí se vystavená zálohová faktura touto částkou

zpusobyUhradyProVystaveniZalohy=

;přenos uživatelských polí pro následující tabulky, názvy polí se oddělují čárkou

;[Katalog]

;UzivatelaskaPole=

;[Firmy]

;UzivatelaskaPole=

;[Doklady]









;UzivatelaskaPole=

;[Polozky_dokladu]

;UzivatelaskaPole=

Instalovaná služba

Služba se jmenuje VarioSOAPService

 TeamViewer 9	Team...	Spušt...	Automaticky	Local System
 Technologie Windows Connect Now – Regi...	Služb...	Spušt...	Ručně	Local Service
 Telefonní subsystém	Posk...		Ručně	Síťová služba
 Telnet	Umož...		Zakázáno	Local Service
 VarioSOAPService	Posk...	Spušt...	Automaticky	.\soapservice
 VarioSOAPServiceYouuu	Posk...	Spušt...	Automaticky	Local System
 Virtuální disk	Posk...	Spušt...	Ručně	Local System
 VMware USB Arbitration Service	Arbit...	Spušt...	Automaticky	Local System

Aktualizace služby

Budete-li aktualizovat Vario, není automaticky vyžadována aktualizace SOAP služby. Pokud bude třeba službu aktualizovat, postupujte takto:

1. Službu ve správci služeb zastavíte.
2. Přepište EXE soubor služby aktualizovaným a službu opět nainstalujte.
Pokud máte EXE soubor pojmenovaný vlastním názvem, stejně pojmenujte i nový soubor a přepište původní.
3. Službu ve správci služeb nainstalujte.

Objekty

Veškerá výměna dat probíhá pomocí objektů VarioSOAP, které jsou předávány jako návratové hodnoty metod, případně jsou rozhraní předávány jako parametry metod. Objekt obsahuje data většinou jedné datové entity SW AltusVario. Tato entita je obvykle identifikována pomocí GUID pole ID.

Čtení kompletních dat

VarioSOAP obsahuje metody na čtení kompletních dat určitého typu. Tyto metody jsou doplňkové a používají se zpravidla při úvodním naplnění databáze shopu. Dále není doporučeno jejich používání, pokud totiž databáze obsahuje větší množství dat, může při pokusu o jejich jednorázové kompletní přenesení dojít k překročení timeoutu, případně až vyčerpáním systémových prostředků serveru. Pro běžné použití jsou vhodné jen metody, u nichž lze očekávat řádově maximálně stovky vrácených objektů (seznam ceníků, seznam skladů apod.), bezpečné maximum jsou tisíce vrácených objektů (záleží na rychlosti internetového připojení a výkonnosti serveru).

Pokud je potřeba stáhnout kompletní data (např. při nasazení), je výhodnější nechat na straně serveru vygenerovat joby (zajistí dodavatel můstku), a ty na straně shopu načíst standardně tak, jak se budou načítat i v ostrém provozu.

Čtení změněných dat

Jde o doporučený a většinou i jediný možný způsob přenosu dat z AltusVario na shop. Principem je opakované volání vhodné metody, která vrací buď ID změněných objektů, nebo přímo kompletní změněné objekty,

zpracování těchto objektů a následné zpětné potvrzení zpracování objektu. Jednotlivé změněné objekty lze přenášet pomocí metod vracejících objekt zadáním ID objektu.

Metody

GetVersion: double

Vrací aktuální verzi VarioSOAP

Čtení dat

Čtení jednotlivých objektů

GetCategory(const ID: string): TCategory

Vrací Kategorii ze stromu kategorií podle ID

GetProductCategory(const ID: string): TProductCategory

Vrací informace o zařazení produktu do kategorie stromu

GetManufacturer(const ID: string): TManufacturer

Vrací výrobce podle ID

GetVariant(const ID: string): TVariant

Vrací data varianty podle jejího ID

GetParameterGroup(const ID: string): TParameterGroup

Vrací skupinu paramerů podle ID

GetParameter(const ID: string): TParameter

Vrací parametr podle ID

GetProduct(const ID: string): TProduct

Vrací produkt podle ID

GetProductSimple(const ID: string): TProductSimple

Vrací základní data o produktu podle ID

GetProductsArray(const AIDList: TStringArray): TProducts

Vrací pole produktů podle zadaných ID v poli

GetProductsByCriteria(const Code, ProductNumber, EAN, EANInternal, ProductName: string; const UseOr: boolean): TProducts

Vrací pole produktů podle zadaných kritérií. Pokud je některý parametr prázdný, při hledání se neuvažuje. Zadané podmínky platí současně (UseOr=false), nebo musí být splněna alespoň jedna (UseOr=true).

GetStore(const ID: string): TStore

Vrací informace o skladu podle ID

GetPricelist(const ID: string): TPricelist

Vrací informace o ceníku podle ID

GetPriceMatrixSetting(const ID: string): TPriceMatrixSetting;

Vrací nastavení Matice slev

GetDiscount(const ID: string): TDiscount

Vrací informace o slevě podle ID

GetPriceMatrix(const CustomerID, Currency: string; const ProductIDList: TStringArray): TPriceMatrixPrices;

Vrací ceny produktů pro danou firmu z Matice slev

GetSupply(const ID: string): TSupply

Vrací informaci o stavu skladu podle ID

GetSuppliesByIDs(const IDList: TStringArray): TSupplies;

Vrací pole informací o stavu skladu podle ID skladových zásob zadaných v poli

GetSuppliesArray(const ProductIDList: TStringArray): TSupplies

Vrací pole informací o stavu skladu podle ID produktů zadaných v poli

GetSuppliesByProduct(const ProductID: string): TSupplies

Vrací informace o stavech skladů podle ID produktu

GetAvailableSuppliesByIDs(const IDList: TStringArray): TAvailableSupplies

Vrací informace o disponibilních zásobách na skladech podle ID skladových zásob zadaných v poli

GetAvailableSuppliesByProduct(const ProductID: string): TAvailableSupplies

Vrací informace o disponibilních zásobách na skladech podle ID produktu

GetAvailableSupply(const ID: string): TAvailableSupply

Vrací disponibilní zásobu podle ID

GetAvailableSuppliesArray(const ProductIDList: TStringArray): TAvailableSupplies

Vrací pole informací o disponibilním stavu skladu podle ID produktů zadaných v poli

GetCustomer(const ID: string): TCustomer

Vrací zákazníka podle ID

GetCustomersByCriteria(const Identificator, CustomerName, Email, IC, DIC, Telephone, Street, City, ZIP, Country: string; const UseOr: boolean): TCustomers

Vrací pole zákazníků podle zadaných kritérií. Pokud je některý parametr prázdný, při hledání se neuvažuje. Pokud je UseOr = false, vyhledání jsou zákazníci, pro které zadané podmínky platí současně, pokud je UseOr = true, pro hledání platí alespoň jedna podmínka

GetContact(const ID: string): TContact deprecated ,nahrazeno GetCustomer'

Vrací kontakt (osobu zákazníka) podle ID

GetProductExt(const ID: string): TProductExt

Vrací rozšířené informace k produktu podle ID

GetProductExtsArray(const IDList: TStringArray): TProductExts

Vrací rozšířené informace k produktu podle ID zadaných produktů nebo variant produktů

GetProductExts(const ID: string): TStringArray

Vrací pole ID rozšíření produktů podle ID produktu nebo ID varianty produktu

GetElectroFee(const ID: string): TElectroFee

Vrací definici elektropoplatku podle ID

GetAuthorshipFee(const ID: string): TAuthorshipFee

Vrací definici autorského poplatku podle ID

GetDocument(const ID: string): TDocument

Vrací doklad podle ID

GetDocumentsByCriteria(const CustomerIdentificator, DocumentType: string; const DateFrom, DateTo, SettlementDateFrom, SettlementDateTo: TXSDateTime; const Payed: TXSBoolean): TDocuments;

GetDocumentsByCriteria1(const CustomerIdentificator, DocumentType: string; const DateFrom, DateTo, SettlementDateFrom, SettlementDateTo: TXSDateTime; const PageSize, PageNumber: TXSInteger; const OrderBy: string; const OrderDesc: boolean): TDocuments;

Vrací pole dokladů podle zadaných kritérií (vybrané vlastnosti TDocument). Pokud je některý parametr prázdný, při hledání se neuvažuje. Zadané podmínky platí současně. Pokud má funkce „stránkovat“, zadá se PageSize (počet záznamů na „stránku“) větší než 0. PageNumber pak znamená číslo strany, první strana má číslo 0. OrderBy je název pole, podle kterého se řadí (Customeridentificator, DocumentType, Date, SettlementDate, pokud je prázdné, řadí se podle čísla dokladu), pokud je OrderDesc true, řadí se sestupně.

GetReturn(const ID: string): TReturn

Vrací reklamaci podle ID

GetActivityLog(const ID: string): TActivityLog

Vrací záznam deníku podle ID

GetActivityLogsByRecordID(const RecordID: string): TActivityLogs

Vrací pole záznamů v deníku k zadanému ID dokladu nebo ID firmy

GetTranslation(const ID: string): TTranslation

Vrací překlad pro produkt podle ID

GetTranslationsArray(const AIDList: TStringArray): TTranslations

Vrací pole překladů podle zadaných ID v poli

GetPrice(const ID: string): TPrice

Vrací cenu podle ID

GetDocumentFiles(const DocumentID: string): TFiles

Vrací pole souborů k zadanému ID dokladu

Čtení kompletních dat

GetStores: TStringArray

Vrací pole ID skladů

GetCategories: TStringArray

Vrací pole ID kategorií stromu

GetProductCategories: TStringArray

Vrací pole ID informací o zařazení produktů do kategorií stromu

GetManufacturers: TStringArray

Vrací pole ID výrobců

GetVariants: TStringArray

Vrací pole ID variant

GetParameterGroups: TStringArray

Vrací pole ID skupin parametrů

GetParametres: TStringArray

Vrací pole ID parametrů

GetProducts: TStringArray

Vrací pole ID produktů

GetProductsPage(const PageSize: integer, PageNumber: integer): TStringArray

Vrací pole ID produktů; PageSize je počet záznamů na stránce (maximální velikost vráceného pole), PageNumber je pořadí požadované stránky (číslováno od 0)

GetProductsByStore(const ID: string): TStringArray

Vrací pole ID produktů podle ID skladu

GetPricelists: TStringArray

Vrací pole ID ceníků

GetPriceMatrixSettings: TStringArray;

Vrací pole ID nastavení Matice slev

GetDiscounts: TStringArray

Vrací pole ID slev

GetSupplies: TStringArray

Vrací pole ID stavů skladů

GetAvailableSupplies: TStringArray

Vrací pole ID disponibilních zásob na skladě

GetCustomers: TStringArray

Vrací pole ID zákazníků

GetContacts(const CustomerID: string): TStringArray deprecated ,nahrazeno GetCustomer'

Vrací pole ID osob (zákazníků)

GetElectroFees: TStringArray

Vrací pole ID elektropoplatků

GetAuthorshipFees: TStringArray

Vrací pole ID autorských poplatků

GetDocuments: TStringArray

Vrací pole ID dokladů

GetReturns: TStringArray

Vrací pole ID reklamací

GetActivityLogs: TStringArray

Vrací pole ID záznamů deníku

GetTranslations: TStringArray

Vrací pole ID překladů produktů

GetPricesByPriceListValid(const APriceListID: string): TPrices

Vrací pole platných cen podle ID ceníku

GetPricesByPriceList(const APriceListID: string): TPrices

Vrací pole cen podle ID ceníku

GetPricesByProduct(const AProductID: string): TPrices

Vrací pole cen podle ID produktu

GetVATRatesData(const Country: string; const ValidFrom, ValidTo: TXSDateTime): TVATRates

Vrací pole definic sazeb DPH, které jsou platné v daném období. Country je zkratka země (2 znaky), pokud je nevyplněná, vrací vše. Pokud je ValidFrom prázdné (0), uvažuje se platnost do minulosti neomezená, pokud je ValidTo prázdné (0), uvažuje se platnost do budoucnosti neomezená

GetSQLData(const ADBName: string; ASQL: string): string;

Vrací XML jako string se záznamy podle SQL dotazu nad konkrétní databází (ADBName="" použije se výchozí databáze; lze použít např. ADBName='Data####', kde se #### nahradí příslušným číslem dat)

GetFile(const AFileName: string): TByteDynArray;

Vrací zadaný soubor z disku nebo sítě

Čtení změněných dat

Pokud je použita multidoménovost, je třeba při volání následujících metod posílat SOAP hlavičku typu TAuthHeader

GetJobs: TJobs

Vrací pole „jobů“, tedy zatím nezpracovaných změn dat v databázi. Podle objektu a jeho ID v jobu lze pak volat funkce pro čtení dat jednotlivých objektů

GetJobsByType(const ACount: integer; const AObjectType: TObjectType): TJobs

Vrací pole „jobů“ určitého typu objektu a určitého počtu. Podle objektu a jeho ID v jobu lze pak volat funkce pro čtení dat jednotlivých objektů

GetJobsData(const ACount: integer; const AObjectType: TObjectType): TJobsData

Vrací zadaný počet „jobů“ určitého typu v poli objektů, které zároveň nese i data změněných objektů. Funkce umožňuje efektivnější, a tedy rychlejší zpracování, ale pouze pro „joby“ typu otProduct, otProductSimple, otPrice, otCategory, otSupply a otAvailableSupply

SetJob(const ID: Cardinal; const Done: boolean; const Error: string): boolean

Označí „job“ jako zpracovaný (dále nebude vrácen), pokud je parametr Done=true. V Error lze zadat případnou chybu při zpracování jobu

SetJobs(const AIDs: TCardinalArray; const Done: boolean; const Error: string): boolean

Označí zadané „joby“ podle pole ID (array of cardinal) jako dokončené (Done=true) hromadně. Jde o efektivnější, tedy rychlejší variantu okapovaného volání funkce SetJob

Zápis dat

Pokud mají následující funkce aktualizovat existující data, je potřeba v objektu vyplnit ID. Pokud ID nebude zadáno, funkce vždy vytváří nový záznam, což může vést k duplicitě.

CreateOrUpdateCustomer(ACustomer: TCustomer): string

Založí nebo aktualizuje zákazníka, vrací ID založeného/aktualizovaného objektu

DeleteCustomer(const ID: string; const DeleteSubordinate: boolean)

Smaže kontakt podle ID, pokud je DeleteSubordinate rovno True, smaže i podřízené kontakty, jinak zůstanou podřízené kontakty osamoceny

DeleteBank(const ID: string)

Smaže bankovní spojení podle ID

CreateOrUpdateContact(AContact: TContact): string deprecated ,nahrazeno CreateOrUpdateCustomer'

Založí nebo aktualizuje osobu zákazníka, vrací ID založeného/aktualizovaného objektu

CreateOrUpdateDocument(ADocument: TDocument): string

Založí nebo aktualizuje doklad, vrací ID založeného/aktualizovaného objektu. Pokud není vyplněno číslo dokladu bude doplněno podle číselné řady. Při aktualizaci musí doklad obsahovat také položky, neposlané položky budou z dokladu smazány

UpdateDocumentBody(ADocument: TDocument): string

Aktualizuje hlavičku dokladu (není nutné posílat položky), vrací ID Aktualizovaného objektu

UpdateDocumentBodyFields(ADocument: TDocument; const Actualize: TStringArray): string

Aktualizuje hlavičku dokladu (není nutné posílat položky), vrací ID Aktualizovaného objektu. Pole Actualize je možné vyplnit názvy vlastností objektu TDocument, které se mají v databázi aktualizovat. Ostatní pole se pak ignorují.

CreateOrUpdateDocumentItem(ADocumentItem: TDocumentItem): string

Založí nebo aktualizuje položku dokladu, vrací ID založeného/aktualizovaného objektu

ChangeDocumentItemsState(const IDs: TStringArray; const State: string)

Změní stav položek dokladů, položky se zadávají pomocí pole jejich IDček

CreateRequestAdvance(const ADocumentID: string; const ARequested, APaid: Currency)

Dodatečně vytvoří zálohu k zakázce s ID ADocumentID ve výši ARequested, zálohu případně uhradí částkou APaid

CreateOrUpdateReturn(AReturn: TReturn): string

Založí nebo aktualizuje reklamaci, vrací ID založeného/aktualizovaného objektu

CreateOrUpdateReturnItem(AReturnItem: TReturnItem): string

Založí nebo aktualizuje položku reklamace, vrací ID založeného/aktualizovaného objektu

CancelOrder(const ID: string)

Stornuje doklad (zakázku) podle ID, pokud je to možné

CreateRelationship(const ID1, ID2: string; RType: TRelationType): string

Založí vazbu mezi dvěma doklady a vrací ID vazby

CreateOrUpdateActivityLog(AActivityLog: TActivityLog): string

Založí nebo aktualizuje záznam deníku, vrací ID založeného/aktualizovaného objektu

CreateOrUpdatePrices(const Prices: TPrices)

Založí nebo aktualizuje ceny v existujícím ceníku

CreateOrUpdateProduct(AProduct: TProduct): string

Založí nebo aktualizuje produkt v Katalogu, vrací ID založeného/aktualizovaného objektu. Pokud není vyplněno pole Produkt (Code) bude doplněno podle číselné řady. Povinné je pole ProductType. Zapisuje pouze do tabulek Katalog, Katalog_varianty_produkty nebo Firmy.Ciselnik, Kategorie_produkty, Cenik, Souvisejici_produkty.

UpdateProduct(AProduct: TProduct; const Actualize: TStringArray): string

Aktualizuje produkt v Katalogu, vrací ID aktualizovaného objektu. Pole Actualize je možné vyplnit názvy vlastností objektu TProduct, které se mají v databázi aktualizovat. Ostatní pole se pak ignorují. Pokud je pole Actualize prázdné, aktualizují se všechna pole v databázi.

CreateOrUpdateVariant(AVariant: TVariant): string

Založí nebo aktualizuje variantu produktu, pokud se produkt na varianty sleduje. Vrací ID založené varianty. Povinná jsou pole AVariant.ProductID, AVariant.VariantName.

CreateOrUpdateProductExt(AProductExt: TProductExt): string

Založí nebo aktualizuje rozšířené informace k produktu, vrátí ID záznamu.

CreateOrUpdateProductExts(AProductExts: TProductExts)

Efektivnější volání CreateOrUpdateProductExt pro více (pole) TProductExt

CreateOrUpdateStockLevel(const AStockLevel: TStockLevel): string

Založí nebo aktualizuje skladovou kartu (množství zásoby na skladě), vrátí ID skladové karty. Množství skladové zásoby lze měnit jen skladové karty z externích (dodavatelských) skladů (Typ knihy = E). U ostatních skladů mění jen textové informace a množství pro objednávání. Souvisí s (Available)Supply, kdy se naopak skladová zásoba čte.

CreateOrUpdateStockLevels(const AStockLevels: TStockLevels)

Efektivnější volání CreateOrUpdateStockLevel pro více (pole) TStockLevel

CreateOrUpdateCategory(ACategory: TCategory): string

Založí nebo aktualizuje větev stromu kategorií, vrátí ID založeného/aktualizovaného objektu

DeleteCategory(const ID: string)

Smaže větev stromu kategorií podle ID

CreateCoupon(const ACoupon: TCoupon): string

Založí slevový kupón, vrátí ID založeného objektu (nemusí být dostupné)

CreatePayment(const ADocumentID: string; const AAmount: Currency): string

Uhradí předaný doklad (podle ID), úhradu vloží jako novou položku denního bankovního výpisu, pokud výpis neexistuje, založí ho

FinishPayment(const APaidDocumentID, APayingItemID: string; const AAmount: Currency): string

Provede úhradu dokladu APaidDocumentID položkou (hradícího dokladu, výpisu) APayingItemID částkou AAmount

ChangeID(const AObjectType: TObjectType; const AOldID, ANewID: string)

Změna interního ID objektu, používat jen v nezbytném případě po předchozí konzultaci

CreateOrUpdatePriceMatrixSetting(const APriceMatrixSetting: TPriceMatrixSetting): string

Založí nebo aktualizuje záznam v nastavení Matice slev, vrátí ID založeného/aktualizovaného záznamu

DeletePriceMatrixSetting(const ID: string)

Smaže nastavení matice slev podle ID

Výjimky

TExceptionCode

TExceptionCode = (ecInternal, ecDuplicated, ecMustHaveValue, ecUnknowValue, ecNotFound, ecNotAllowed)

Kód chyby SOAP

EVarioSOAPException

Třída výjimky zpracování SOAP požadavku, vyvolá se, pokud jsou přesně známy příčiny chyby.

ExceptionCode: TExceptionCode – kód chyby

- **ecInternal** – interní chyba nastavení serveru
- **ecDuplicated** – duplicita dat, nebo pokus o vytvoření duplicitního záznamu
- **ecMustHaveValue** – pole nebo proměnná, která je povinná, nemá hodnotu
- **ecUnknowValue** – neznámá nebo neplatná hodnota proměnné nebo pole
- **ecNotFound** – požadovaný záznam, nebo objekt nenalezen
- **ecNotAllowed** – operace není povolena

ExceptionSource: string – zdroj chyby, pokud ze znám

- pro **ecDuplicated** jde o pole, které způsobilo duplicitu
- pro **ecMustHaveValue** jde o pole nebo proměnnou metody, která musí mít vyplněnu hodnotu
- pro **ecUnknowValue** jde o pole nebo proměnnou metody, která má neznámou nebo neplatnou hodnotu
- pro **ecNotFound** jde o objekt, který nebyl nalezen
- pro **ecNotAllowed** jde o objekt, nad kterým nelze operaci provést

Objekty

TAuthHeader

SOAP hlavička sloužící k autorizaci klienta (shopu), pokud je použita multidoménovost, nutno posílat pro metody uvedené v sekci Čtení změněných dat

ClientNumber: integer – kladné číslo dat (databází) AltusVario, nad kterými má daná metoda probíhat

TPropHeader

Nepovinná SOAP hlavička s parametry připojení.

DataNumber: integer – kladné číslo, které je přiděleno klientovi

TObjectType

TObjectType = (otCategory, otManufacturer, otParameterGroup, otParameter, otProduct, otProductExt, otSupply, otDiscount, otElectroFee, otAuthorshipFee, otCustomer, otContact, otDocument, otReturn, otAvailableSupply, otActivityLog, otTranslation, otProductSimple, otPrice, otAll, otPriceList, otPriceMatrixSetting, otProductCategory)

Typ objektu, výčet možných hodnot odpovídá jednotlivým objektům potažmo metodám, které je vrací

TAction = (acInsert, acUpdate, acDelete)

Typ změny, která se s objektem stala (vlození, aktualizace, smazání)

Vazba mezi TObjectType a tabulkou VarioSynchrono

Hodnota pole `variosynchro.TableName`

Katalog
Kategorie
Kategorie_produkty
Parametr
Struktura
Vyrobcí
T_Katalog_Exts
Sklad
Slevy
Elektrozarizeni
AutorskePopl
Firmy
Osoby
Doklady
Reklamace
SkladDisponibilni
Denik
Preklad
Produkt
Cenik
Ceniky
T_Matice_Slev

TObjectType

otProduct
otCategory
otProductCategory
otParameter
otParameterGroup
otManufacturer
otProductExt
otSupply
otDiscount
otElectroFee
otAuthorshipFee
otCustomer
otContact
otDocument
otReturn
otAvailableSupply
otActivityLog
otTranslation
otProductSimple
otPrice
otPriceList
otPriceMatrixSetting

TJob

„job“ popisující změnu v databázi

ID: Cardinal – ID (číselné) jobu

ObjectID: string – ID (GUID) změněného objektu

ObjectType: TObjectType – typ změněného objektu

Created: **TDateTime** – datum a čas změny

Action: **TAction** – Typ změny

TJobs = array of TJob

TJobData

„job“ popisující změnu v databázi včetně změněných dat

Job: **TJob**

Data: **TRemotable** – změněný objekt, který popisuje vlastnost Job. Jde o základní třídu, kterou lze na daný typ objektu přetypovat

TJobsData = array of TJobData

TParameterValue

ParameterID: **string** – (Parametry_hodnoty.Parametr, nebo T_Parametry.rowguid) ID parametru

Value: **string(100)** – (Parametry_hodnoty.Hodnota, nebo T_Struktury_Rozpis.HodnotaPoleText) hodnota parametru

TParameterValues = array of TParameterValue

TProductSetPart

Materiál k produktu

ProductID: **string** – (Katalog.rowguid) ID produktu (materiálu)

Authorized: **boolean** – (Katalog_material.Schvaleno) příznak schválení materiálu

Count: **Double** – (Katalog_material.Mnozstvi) počet jednotek produktu (materiálu)

TProductSet = array of TProductSetPart

TPrice

Cena produktu z ceníku

ID: **string** – (Cenik.rowguid) ID ceny

ProductID: **string** – (Katalog.rowguid) ID produktu

PriceID: **string** – (Ceniky.rowguid) ID ceníku

VariantID: **string** – (Katalog_varianty_produkту.rowguid – Cenik.Varianta) ID varianty produktu

Value: **Currency** – (Cenik.Cena) cena

ValidFrom: **TDateTime** – (Cenik.Platnost_od) platnost ceny od

ValidTo: **TDateTime** – (Cenik.Platnost_do) platnost ceny do

QuantityFrom: **integer** – (Cenik.Mnozstvi_od) Množství, od kdy platí cena

Data1: **string(30)** – (Cenik.Udaj_ceniku_1) údaj 1 ceníku

TPrices = array of TPrice

TPriceMatrixPrice = class(TRemotable)

ProductID: **string** – (Katalog.rowguid) ID produktu

Value: **Currency** – cena (0 – cena není definována)

TPriceMatrixPrices = array of TPriceMatrixPrice;

TDiscountType = (dtPercentage, dtPerUnit, dtValue, dtValuePerUnit)

Typ slevy

TDiscountRate

Definice položky slevy

ID: string – (Slevy_sazby.rowguid) ID položky slevy

PriceFrom: Currency – (Slevy_sazby.Cena_od) cena od

Value: Currency – (Slevy_sazby.Sazba) hodnota slevy

ValueMax: Currency – (Slevy_sazby.Sazba_maximalni) maximální hodnota slevy

DiscountCategory: string(30) – (Slevy.Kategorie_slev) název kategorie slev

DiscountType: TDiscountType – typ slevy

TDiscountRates = array of TDiscountRate;

TDiscount

Definice slevy

ID: string – (Slevy.rowguid) ID slevy

Name: string(30) – (Slevy.Sleva) název slevy

Currency: string(3) – (Slevy.Mena) měna slevy

ValidFrom: TDateTime – (Slevy.Platnost_od) začátek platnosti slevy

ValidTo: TDateTime – (Slevy.Platnost_do) konec platnosti slevy

WithVAT: boolean – (Slevy.Ceny_vcetne_DPH) ceny jsou včetně DPH

PriceGroup: string(30) – (Slevy.Kategorie_cen) název kategorie cen pro slevu

TDiscounts = array of TDiscount

TEAN

Informace o čárovém kódu

ID: string – (Carove_kody.rowguid) ID čárového kódu

EAN: string – (Carove_kody.Carovy_kod): čárový kód

TEANs = array of TEAN

TVariant

Varianta produktu

ID: string – (Katalog_varianty_produkту.rowguid nebo Ciselnik.rowguid) ID varianty produktu

VariantName: string(30) – (Katalog_varianty_produkту.Varianta, nebo Ciselnik.Hodnota) název varianty

ProductID: string – (Katalog.rowguid) ID produktu

ProductName: string(255) – (Katalog_varianty_produkту.Popis) popis produktu v této variantě

ProductNumber: string(30) – (Katalog_varianty_produkту.Katalogove_cislo) katalogové číslo produktu v této variantě

Image: TByteDynArray – (Katalog_varianty_produkту.Obrazek) obrázek varianty

ImageRef: string(255) – (Katalog_varianty_produkту.Obrazek) odkaz na obrázek

SmallImage: TByteDynArray – (Katalog_varianty_produkту.Obrazek_maly) malý obrázek varianty

SmallImageRef: string(255) – (Katalog_varianty_produkту.Obrazek_makly) odkaz na malý obrázek

Note: string(255) – (Katalog_varianty_produkту.Poznamka, nebo Ciselnik.Poznamka) poznámka

Description: string(8000) – (Katalog_varianty_produkту.Text) dlouhý text

Visible: boolean – (Katalog_varianty_produkту.Publikovat) má se varianta zobrazovat
Manufacturer: string(30) – (Katalog_varianty_produkту.Vyrobce) výrobce varianty
Supplier: string(30) – (Katalog_varianty_produkту.Dodavatel) dodavatel varianty
Weight: double – (Katalog_varianty_produkту.Hmotnost) hmotnost
QuantityMinimum: double – (Katalog_varianty_produkту.Mnozstvi_minimalni) minimální množství na skladě pro objednání
QuantityMaximum: double – (Katalog_varianty_produkту.Mnozstvi_maximalni) maximální množství na skladě pro objednání
QuantityReorderPoint: double – (Katalog_varianty_produkту.Mnozstvi_bodu_objednani) množství bodu objednání na skladě
Warranty: integer – (Katalog_varianty_produkту.Zaruka) délka záruky
WWW: string(255) – (Katalog_varianty_produkту.Internet) odkaz na internetové stránky
Data1: string(100) – (Katalog_varianty_produkту.Udaj_1) údaj 1
Data2: string(100) – (Katalog_varianty_produkту.Udaj_1) údaj 2
Number1: double – (Katalog_varianty_produkту.Pocet_1) počet 1
Number2: double – (Katalog_varianty_produkту.Pocet_1) počet 2
Dimensions: string(100) – (Katalog_varianty_produkту.Rozmery) rozměry varianty
Width: integer – (Katalog_varianty_produkту.Sirka) šířka
Height: integer – (Katalog_varianty_produkту.Vyska) výška
Length: integer – (Katalog_varianty_produkту.Hloubka) hloubka
QualityStandard: string(30) – (Katalog_varianty_produkту.Jakostni_norma) jakostní norma
PurchasingStrategy: string(30) – (Katalog_varianty_produkту.Vyrobni_strategie) nákupní strategie
New: boolean – (Katalog_varianty_produkту.Novinka) novinka
Action: boolean – (Katalog_varianty_produkту.Akce) akce
TopProduct: boolean – (Katalog_varianty_produkту.Top_produkту) top produkt
Availbaility: integer – (Katalog_varianty_produkту.Doba_dodani) doba dodání
SellOff: boolean – (Katalog_varianty_produkту.Vyprodej) výprodej
Annotation: string(255) – (Katalog_varianty_produkту.Anotace) anotace
Sale: boolean – (Katalog_varianty_produkту.Nabizet_v_dokladech) nabízet
EANS: TEANs – pole čárových kódů varianty

TVariants = array of TVariant

TProduct

Produkt včetně dalších údajů k němu

Id: string – (Katalog.rowguid) ID produktu
Book: string(30) – (Katalog.Kniha) zařazení do knihy
CategoriesID: TStringArray – (Kategorie.rowguid) pole ID kategorií stromu, kam je produkt zařazen
ManufacturerID: string – (Firmy.rowguid) ID výrobce
Manufacturer: string(30) – (Katalog.Vyrobce) název výrobce
Supplier: string(30) – (Katalog.Dodavatel) název dodavatele
ParameterGroupID: string – (T_Struktury.rowguid) ID skupiny parametrů
Parametres: TParameterValue – hodnoty parametrů
ProductSet: TProductSet – materiál produktu
AlternativesID: TStringArray – (Katalog.rowguid) pole ID alternativních produktů
IncompatiblesID: TStringArray – (Katalog.rowguid) pole ID nekompatibilních produktů
RelatedProductsID: TStringArray – (Katalog.rowguid) pole ID souvisejících produktů
AccessoriesID: TStringArray – (Katalog.rowguid) pole ID produktů příslušenství
DesignProductsID: TStringArray – (Katalog.rowguid) pole ID produktů provedení
FilesID: TStringArray – prázdné
Code: string(30) – (Katalog.Produkt) kód produktu (produkt)
ProductName: string(255) – (Katalog.Popis) popis produktu
ProductNumber: string(30) – (Katalog.Katalogove_cislo) katalogové číslo

Units: string(30) – (Katalog.Jednotky) jednotky
Description: string(8000) – (Katalog.Text) dlouhý text
Note: string(255) – (Katalog.Poznamka) poznámka
Classification: string(255) – (Katalog.Zarazeni) zařazení produktu
EAN: string(30) – (Katalog.Carovy_kod) čárový kód
EANInternal: string(30) – (Katalog.Carovy_kod_vlastni) čárový kód vlastní
EANs: TEANs – pole čárových kódů produktu
GPL: Currency – prázdné
ProductType: string(2) – (Katalog.Typ_zaznamu) typ produktu (MA – materiál, ZB – zboží, SL – služba, BU – sada, SE – sestava, VY – výrobek), při zápisu nutné zadat
ProductTypeName: string(30) – (Katalog.Typ_produkту) název typu produktu
VAT: string(30) – (Katalog.Sazba_DPH) sazba DPH
Price: Currency – (Katalog.Cena) cena zadaná přímo u produktu (ne z ceníku)
PriceWithVAT: boolean – (nastavení Ceny katalogu včetně DPH) je cena Price s daní?
PricePurchase: Currency – (Katalog.Cena_nakupni) nákupní cena produktu
WWW: string(255) – (Katalog.Internet) URL adresa produktu
Dimensions: string(100) – (Katalog.Rozmery) rozměry
Variants: TVariants – pole variant produktu
Warranty: integer – (Katalog.Zaruka) záruka
QualityStandard: string(30) – (Katalog.Jakostni_norma) jakostní norma
Weight: Double – (Katalog.Hmotnost) hmotnost
Data1: string(100) – (Katalog.Udaj_1) údaj 1
Data2: string(100) – (Katalog.Uda_2) údaj 2
Number1: double – (Katalog.Pocet_1) číslo 1
Number2: double – (Katalog.Pocet_2) číslo 2
Action: Boolean – (T_Katalog.B2CAkce) akce
TIP: boolean – (T_Katalog.B2CHiTydne) tip
New: boolean – (T_Katalog.B2CNovinka, nebo Katalog.Novinka) novinka
Sale: boolean – (Katalog.Nabizet_v_dokladech) nabízet
SellOff: boolean – (Katalog.Vyprodej) výprodej
TitlePage: boolean – (T_Katalog.B2CTitulka) titulní strana
InvisibleComponent: boolean – prázdné
Visible: boolean – (T_Katalog.B2C, nebo Katalog.Publikovat) viditelný
HideOnWeb: boolean – (T_Katalog.SkrytyNaInt) skryt na webu
B2B: boolean – (T_Katalog.B2B) viditelný pro B2B
B2CPriority: boolean – (T_Katalog.B2CUprednostnit) viditelný pro B2C
DiscountCategory: string – (Katalog.Kategorie_cen) Kategorie cen
Prices: TPrices – pole cen produktu v cenících
Image: TByteDynArray – (Katalog.Obrazek) obrázek
SmallImage: TByteDynArray – (Katalog.Obrazek_maly) malý obrázek
ElectroFee: string – (Elektrozarizeni.rowguid) ID elektropoplatku
AuthorshipFee: string – (T_Autorske_poplatky.rowguid) ID autorského poplatku
AlwaysAvailable: boolean – (T_Katalog.StaleSkladem) vždy dostupný
Availbaility: string(50) – (T_Katalog.B2CDodaciLhuta, nebo Katalog.Doba_dodani) dodací lhůta
BasicUnit: string(30) – (Katalog.Z_akladni_jednotka) základní jednotka
BasicUnitQuantity: integer – (Katalog.Mnozstvi_zakladnich_jednotek) množství základních jednotek
Discounts: TDiscounts – pole slev
Height: double – (Katalog.Vyska) výška
Width: double – (Katalog.Sirka) šířka
Length: double – (Katalog.Hloubka) délka
AlternativeDescription: string(255) – (T_Katalog.Popis_2) alternativní popis
AllowOrderByCustomer: boolean – (T_Katalog.PovolitObjOdberatelem) má odběratel povoleny objednávky
AlternativeNote: string(8000) – (T_Katalog.PoznamkaVelka) alternativní poznámka
ImageRef: string(255) – (Katalog.Obrazek) odkaz na obrázek

SmallImageRef: string(255) – (Katalog.Obrazek_makly) odkaz na malý obrázek
B2BAction: boolean – (T_Katalog.B2BAkce) B2B akce
ActionDescription: string(100) – (T_Katalog.Akce) popis akce
MarketingAction: string(100) – (T_Katalog.MarketingovaAkce) popis marketingové akce
B2BTIP: boolean – (T_Katalog.B2BHitTydne) B2B tip
B2BNew: boolean – (T_Katalog.Novinka) B2B novinka
B2BTitlePage: boolean – B2B titulní stránka
B2BPriority: boolean – (T_Katalog.B2BUprednostnit) B2B prioritní
B2BAvailaility: string(50) – (T_Katalog.B2BDodaciLhuta) B2B dostupnost
TopProduct: boolean – (T_Katalog.TopProdukt, nebo Katalog.Top_produk) top produkt
B2BMinimumOrderQuantity: double – (Katalog.u_Mnozstvi_minimalni_shop) B2B minimální objednáací množství
InPricelist: boolean – (T_Katalog.VCeniku) označení produktu „v ceníku“
DefaultStoreID: string – (Knihy.rowguid) ID výchozího skladu
AccountCodeSell: string(30) – (Katalog.Predkontace_prodej) předkontace produktu pro prodej
Annotation: string(255) – (Katalog.Anotace) anotace produktu
RecyclingCosts: currency – (Katalog.Naklady_na_recyklaci) náklady na recyklaci elektrozařízení na ks nebo kg
CustomsTariff: string(30) – (Katalog.Celni_sazebnik) zařazení do celního sazebníku
CustomsNumberOfUnits: double – (Katalog.Mnozstvi_jednotek_clo) Množství jednotek pro clo
QuantityMinimum: double – (Katalog.Mnozstvi_minimalni) Množství minimální pro objednávání
QuantityMaximum: double – (Katalog.Mnozstvi_maximalni) Množství maximální pro objednávání
PurchasingStrategy: string(30) – (Katalog.Vyrobní_strategie) Nákupní strategie
UserFields: string – hodnoty uživatelských polí v XML formátu, např.
<pole1>hodnota</pole1><pole2>hodnota</pole2>

TProductSimple

Zjednodušený objekt produktu, používá se pro urychlení přenosů u katalogů s velkým počtem produktů, neobsahuje informace, které přenáší jiné objekty

Id: string – (Katalog.rowguid) ID produktu
Book: string(30) – (Katalog.Kniha) zařazení do knihy
ManufacturerID: string – (Firmy.rowguid) ID výrobce
Manufacturer: string(30) – (Katalog.Vyrobce) název výrobce
Supplier: string(30) – (Katalog.Dodavatel) název dodavatele
Code: string(30) – (Katalog.Produkt) kód produktu (produkt)
ProductName: string(255) – (Katalog.Popis) popis produktu
ProductNumber: string(30) – (Katalog.Katalogove_cislo) katalogové číslo
Units: string(30) – (Katalog.Jednotky) jednotky
Description: string(8000) – (Katalog.Text) dlouhý popis
Note: string(255) – (Katalog.Poznamky) poznámky
Classification: string(255) – (Katalog.Zarazeni) zařazení
EAN: string(30) – (Katalog.Carovy_kod) čárový kód
EANInternal: string(30) – (Katalog.Carovy_kod_vlastni) čárový kód vlastní
ProductType: string(2) – (Katalog.Typ_zaznamu) typ produktu (MA – materiál, ZB – zboží, SL – služba, BU – sada, SE – sestava, VY – výrobek)
ProductTypeName: string(30) – (Katalog.Typ_produk) název typu produktu
VAT: string(30) – (Katalog.Sazba_DPH) sazba DPH
Price: Currency – (Katalog.Cena) cena zadaná přímo u produktu (ne z ceníku)
PriceWithVAT: boolean – (nastavení Ceny v katalogu včetně DPH) je cena produktu s DPH?
PricePurchase: Currency – (Katalog.Cena_nakupni) nákupní cena produktu
WWW: string(255) – (Katalog.Internet) URL adresa produktu
Sale: boolean – (Katalog.Nabizet_v_dokladech) nabízet
DiscountCategory: string – (Katalog.Kategorie_cen) Kategorie cen
Warranty: integer – (Katalog.Zaruka) záruka

QualityStandard: string(30) – (Katalog.Jakostni_norma) jakostní norma
Weight: Double – (Katalog.Hmotnost) hmotnost
Data1: string(100) – (Katalog.Udaj_1) údaj 1
Data2: string(100) – (Katalog.Udaj_2) údaj 2
Number1: double – (Katalog.Pocet_1) číslo 1
Number2: double – (Katalog.Pocet_2) číslo 2
Image: TByteDynArray – (Katalog.Obrazek) obrázek
SmallImage: TByteDynArray – (Katalog.Obrazek_maly) malý obrázek
ElectroFee: string – (Elektrozarizeni.rowguid) ID elektropoplatku
BasicUnit: string(30) – (Katalog.Z_akladni_jednotka) základní jednotka
BasicUnitQuantity: integer – (Katalog.Mnozstvi_zakladnich_jednotek) množství základních jednotek
ImageRef: string(255) – (Katalog.Obrazek) odkaz na obrázek
SmallImageRef: string(255) – (Katalog.Obrazek_maly) odkaz na malý obrázek
B2BMinimumOrderQuantity: double – (Katalog.u_Mnozstvi_minimalni_shop) B2B minimální objednáací množství
TopProduct: boolean – (Katalog.Top_produkty) top produkt (nepoužívat, pokud se používají Parametry Plus)
Dimensions: string(100) – (Katalog.Rozmery) rozměry
Annotation(255) – (Katalog.Anotace) anotace produktu
Visible: boolean – (Katalog.Publikovat) viditelný (obsahuje hodnotu z Varia, nepoužívat, pokud se používají Parametry Plus)
New: boolean – (Katalog.Novinka) novinka (obsahuje hodnotu z Varia, nepoužívat, pokud se používají Parametry Plus)
Action: boolean – (Katalog.Akce) akce (obsahuje hodnotu z Varia, nepoužívat, pokud se používají Parametry Plus)
Height: double – (Katalog.Vyska) výška (obsahuje hodnotu z Varia, nepoužívat, pokud se používají Parametry Plus)
Width: double – (Katalog.Sirka) šířka (obsahuje hodnotu z Varia, nepoužívat, pokud se používají Parametry Plus)
Length: double – (Katalog.Hloubka) hloubka (obsahuje hodnotu z Varia, nepoužívat, pokud se používají Parametry Plus)
Availability: integer – (Katalog.Doba_dodani): předpokládaná doba dodání
RecyclingCosts: currency – (Katalog.Naklady_na_recyklaci) náklady na recyklaci elektrozařízení na ks nebo kg
CustomsTariff: string(30) – (Katalog.Celni_sazebnik) zařazení do celního sazebníku
CustomsNumberOfUnits: double – (Katalog.Mnozstvi_jednotek_clo) Množství jednotek pro clo
UserFields: string – hodnoty uživatelských polí v XML formátu, např.
<pole1>hodnota</pole1><pole2>hodnota</pole2>

TProducts = array of TProduct

TCategory

Kategorie stromu

ID: string – (Kategorie.rowguid) ID kategorie
ParentID: string – (Kategorie.rowguid) ID nadřazené kategorie
Order: Cardinal – (Kategorie.Poradi) pořadí
CategoryName: string(255) – (Kategorie.Kategorie) název kategorie
HyperLink: string(255) – (Kategorie.Odkaz) hypertextový odkaz
Description: string(8000) – (Kategorie.Popis) popis kategorie
Shop: string(255) – (Kategorie.Obchod) obchod
Image: TByteDynArray – (Kategorie.Obrazek) obrázek kategorie
ImageRef: string(255) – (Kategorie.Obrazek) jméno souboru s obrázkem
Visible: boolean – (Kategorie.Publikovat, pokud se používají Parametry Plus, tak T_Kategorie.B2C_TK) má se zobrazovat
B2BVisible: boolean – (T_Kategorie.B2B_TK) má se zobrazovat pro B2B (z Parametrů Plus)

TProductCategory

Zařazení produktu do kategorie stromu

ID: string – (Kategorie_produkty.rowguid) ID zařazení

ProductID: string – (rowguid produktu Kategorie_produkty.Produkt) ID produktu

CategoryID: string – (rowguid větve kategorie Kategorie_produkty.ID_Kategorie) ID kategorie

Section: byte – (Kategorie_produkty.Sekce) sekce

Order: integer – (Kategorie_produkty.Poradi) pořadí

TManufacturer

Výrobce

ID: string – (Firmy.rowguid) ID výrobce

ManufacturerName: string(255) – (Firmy.Nazev_firmy) název výrobce

Identificator: string(30) – (Firmy.Firma) identifikátor výrobce

WWW: string(100) – (Firmy.Internet) webové stránky výrobce

Description: string(255) – (nic) popis výrobce

TParameterStruct

Struktura parametrů

ID: string – (T_Parametry.rowguid) ID parametru

Order: word (T_Struktury_Definice.Poradi) – pořadí

PSUnit: string(15) – (T_Struktury_Definice.MernaJednotka) měrná jednotka

TParameterStructs = array of TParameterStruct;

TParameterGroup

Skupina parametrů

ID: string – (T_Struktury.rowguid) ID skupiny

GroupName: string(255) – (TStruktury.PopisS) název skupiny

ParameterStructs: TParameterStructs

TParameter

Parametr produktu

ID: string – (Parametry.rowguid, nebo T_Parametry.rowguid) ID parametru

ParameterName: string(150) – (Parametry.Nazev, nebo T_Parametry.IDPole) název parametru

Note: string(255) – (Parametry.Titulek, nebo T_Parametry.PopisParametru) poznámka

ParentID: string – (Parametry.Parent) ID nadřizovaného parametru (pro vytváření skupin parametrů)

Type: string – (Parametry.Typ) typ parametru

Definition: string – (Parametry.Definice) definice parametru v XML

TStore

Sklad

ID: string – (Knihy.rowguid) ID skladu

StoreName: string(30) – (Knihy.Kniha) název skladu

StoreType: string(1) – (Knihy.Typ_knihy) typ skladu, U – uživatelský (nastavení pro běžné sklady), E – externí (sklad dodavatele, kde je skladová zásoba informativní)

DoNotUse: boolean – (Knihy.Nepouzivate) ano, pokud se sklad nemá nabízet a používat

TPricelist

Ceník (hlavička)

ID: string – (Ceniky.rowguid) ID ceníku

PriceListName: string(30) – (Ceniky.Cenik) název ceníku

Currency: string(5) – (Ceniky.Mena) měna ceníku

WithVAT: boolean – (Ceniky.Ceny_vcetne_DPH) ceník s DPH

Sale: boolean – (Ceniky.Prodejni) je ceník prodejní

PriceGroup: string(30) – (Ceniky.Kategorie_cen) kategorie cen, do které ceník patří

Supplier: string(30) – (Ceniky.Dodavatel) název dodavatele

TQuantityGroupState

= (sqgPOTVRZENO, sqgOBJEDNANO, sqgPrijmout, sqgPrijimaSe, sqgPRIJATO, sqgDODAT, sqgRezervovat, sqgREZERVOVANO, sqgVydát, sqgVydavaSe, sqgVYDANO, sqgFakturovat, sqgFAKTUROVANO, sqgOK)

TQuantityGroupStates = array of TQuantityGroupState;

TQuantityGroup

Množstevní skupina

ID: string – (Polozky_dokladu_produkty.rowguid) ID množstevní skupiny

ItemID: string – (Polozky_dokladu_produkty.rowguid) ID položky

GroupName: string(30) – název množstevní skupiny

Description: string(255) – (Polozky_dokladu_produkty.Seriove_cislo, nebo Sarze) popis

LOT: string – (Polozky_dokladu_produkty.Sarze) šarže

SerialNumber: string – (Polozky_dokladu_produkty.Seriove_cislo) sériové číslo

Durability: TXSDateTime – (Polozky_dokladu_produkty.Trvanlivost) trvanlivost

Quantity: Double – (Polozky_dokladu_produkty.Mnozstvi) množství

States: TQuantityGroupStates – (Polozky_dokladu_produkty.Stav, Stav_pozadovany a kominace dalších polí) stav

TQuantityGroups = array of TQuantityGroup

TSupply

Skladová zásoba

ID: string – (Sklad.rowguid) ID skladové zásoby

ProductID: string – (Katalog.rowguid) ID produktu

VariantID: string – (Katalog_varianty_produkty.rowguid) ID varianty produktu

StoreID: string – (Knihy.rowguid) ID skladu

Quantity: double – (Sklad.Skladem) množství

QuantityMinimum: double – (Sklad.Mnozstvi_minimalni_skladu) množství minimální na skladové kartě pro objednávání

QuantityMaximum: double – (Sklad.Mnozstvi_maximalni_skladu) množství maximální na skladové kartě pro objednávání

QuantityReorderPoint: double – (Sklad.Mnozstvi_bodu_objednani) množství bodu objednání na skladové kartě pro objednávání

PurchasingStrategy: string(30) – (Sklad.Nakupni_strategie) název nákupní (objednávací) strategie

QuantityGroups: TQuantityGroups

Availbaility: string(50) – (T_Katalog.B2CDodaciLhuta) dostupnost

ArrivalDate: TDateTime – (Polozky_dokladu.Datum_dodani, nebo Doklady.Datum_splatnosti) datum příští dodávky na sklad

B2BAvailbaility: string(5) – (T_Katalog.B2BDodaciLhuta) dostupnost pro B2B

TSupplies = array of TSupply

TState

StateName: string – název stavu

Quantity: double - množství

Date: TDateTime - datum

TStates = array of TState;

TAvailableSupply

Disponibilní zásoba

ID: string – (SkladDisponibilni.rowguid) ID disponibilní zásoby

ProductID: string – (SkladDisponibilni.IDprodukt) ID produktu

VariantID: string – (SkladDisponibilni.IDvarianta) ID varianty produktu

StoreID: string – (SkladDisponibilni.IDsklad) ID skladu

Quantity: double – (SkladDisponibilni.m) množství

QuantityMinimum: double – (Sklad.Mnozstvi_minimalni_skladu) množství minimální na skladové kartě pro objednávání

QuantityMaximum: double – (Sklad.Mnozstvi_maximalni_skladu) množství maximální na skladové kartě pro objednávání

QuantityReorderPoint: double – (Sklad.Mnozstvi_bodu_objednani) množství bodu objednání na skladové kartě pro objednávání

PurchasingStrategy: string(30) – (Sklad.Nakupni_strategie) název nákupní (objednávací) strategie

Availbaility: string(50) – (T_Katalog.B2CDodaciLhuta) dostupnost

ArrivalDate: TDateTime – (Polozky_dokladu.Datum_dodani, nebo Doklady.Datum_splatnosti) datum příští dodávky na sklad

B2BAvailbaility: string(50) – (T_Katalog.B2BDodaciLhuta) dostupnost pro B2B

States: TStates – doplňující informace o stavech skladu podle nastavení

TAvailableSupplies = array of TAvailableSupply

TStockLevel

Skladová karta – zásoba

ID: string – (Sklad.rowguid) ID skladové zásoby, při zápisu není nutné vyplnit, pokud je vyplněno ProduktID, VariantID a StoreID (doporučená varianta zápisu)

ProductID: string – (Katalog.rowguid) ID produktu, při zápisu nutné vyplnit

VariantID: string – (Katalog_varianty_produkту.rowguid) ID varianty produktu

StoreID: string – (Knihy.rowguid) ID skladu, při zápisu nutné vyplnit

ProductName: string(255) – (Sklad.Popis_skladove_polozky) popis produktu na skladové kartě, pokud není při zápisu poslán, nemění se, případně výchozí hodnota je popis produktu z katalogu

ProductNumber: string(30) – (Sklad.Katalogove_cislo_skl_polozky) katalogové číslo produktu na skladové kartě, pokud není při zápisu posláno, nemění se, případně výchozí hodnota je katalogové číslo produktu z katalogu

Quantity: double – (Sklad.Skladem) množství skladové zásoby

QuantityMinimum: double – (Sklad.Mnozstvi_minimalni_skladu) množství minimální na skladové kartě pro objednávání

QuantityMaximum: double – (Sklad.Mnozstvi_maximalni_skladu) množství maximální na skladové kartě pro objednávání

QuantityReorderPoint: double – (Sklad.Mnozstvi_bodu_objednani) množství bodu objednání na skladové kartě pro objednávání

PurchasingStrategy: string(30) – (Sklad.Nakupni_strategie) název nákupní (objednávací) strategie

Note: string(255) – (Sklad.Poznamka_sklad) poznámka

Data1: string(100) – (Sklad. Udaj_sklad_1) údaj 1

Data2: string(100) – (Sklad. Udaj_sklad_2) údaj 2

TStockLevels = array of TStockLevel

TBank

Bankovní spojení

ID: string – (Bankovni_spojzeni.rowguid) ID bankovního spojení

BankName: string(50) – (Bankovni_spojzeni.Banka) název banky

AccountNumber: string(35) – (Bankovni_spojzeni.Cislo_uctu) číslo účtu

BankCode: string(11) – (Bankovni_spojzeni.Kod_banky) kód banky

SWIFT: string(11) – (Bankovni_spojzeni.Kod_banky)

IBAN: string(35) – (Bankovni_spojzeni.Cislo_uctu)

SpecificSymbol: string(20) – (Bankovni_spojzeni.Specificky_symbol)

Default: boolean – (Bankovni_spojzeni.Vychozi) výchozí banka

TBanks = array of TBank

TAddressType = (atNone, atPrimary, atResidence, atDelivery, atAddress2, atOther)

Typy adres (primární, sídlo, doručovací, adresa 2, ostatní)

TAddress

Adresa

ID: string – (Firmy_adresy.rowguid) ID adresy (není nutno vyplňovat při založení adresy)

AddressType: TAddressType

AddressName: string – (Firmy_adresy.Nazev_adresy) název adresy

Address: string(255) – (Firmy_adresy.Adresa_rozpis) adresa ve formátu Ulice číslo<crlf>PSČ Město<crlf>Stát

Street: string(135) – (Firmy_adresy.Ulice_rozpis) ulice a číslo

City: string(50) – (Firmy_adresy.Mesto_rozpis) město

ZIP: string(10) – (Firmy_adresy.PSC_rozpis) PSČ

Country: string(50) – (Firmy_adresy.Stat_rozpis) Stát

CountryISO: string(2) – ISO země, doplňuje se podle Státu z číselníku

UseOnDocuments: boolean – (Firmy.Adresa_X_na_doklad, X – název adresy ve Variu) vkládat adresu na doklady

TAddresses = array of TAddress

TCustomer

Zákazník

ID: string – (Firmy.rowguid) ID zákazníka, pokud se nepošle při založení, doplní se, při aktualizaci povinné

Book: string(30) – (Doklady.Kniha) kniha, pokud se nepošle při zápisu, doplní se výchozí

CustomerType: string(2) – (Firmy.Typ_zaznamu) typ záznamu (OR – organizace, OP – pobočka, OZ – podnikající osoba, OS - osoba)

CustomerTypeName: string(30) – (Firmy.Typ_kontaktu) název typu zákazníka

MainCustomerID: string – (Firmy.rowguid) ID nadřazeného zákazníka (pro pobočky, osoby apod.)

Identificator: string(30) – (Firmy.Firma) jedinečný identifikátor, při zápisu se většinou zadává

CustomerName: string(255) – (Firmy.Nazev_firmy) obchodní název, případně Jméno a Příjmení

Email: string(100) – (Firmy.E_mail) e-mail

WWW: string(100) – (Firmy.Internet) webové stránky

IC: string(20) – (Firmy.ICO) IČ
DIC: string(20) – (Firmy.DIC) DIČ
CompanyRegister: string(100) – (Firmy.Obchodni_rejstrik) obchodní rejstřík
VATPayer: boolean – (Firmy.Platce_DPH) plátce DPH
CorporateBody: boolean – (Firmy.Pravnicka_osoba) právnická osoba
Telephone1: string(40) – (Firmy.Telefon_1) telefon 1
Telephone2: string(40) – (Firmy.Telefon_2) telefon 2
Fax1: string(40) – (Firmy.Fax_1) fax 1
Fax2: string(40) – (Firmy.Fax_2) fax 2
SalesAgent: string(30) – (Firmy.Referent) referent
InvoiceDue: integer – (Firmy.Splatnost_faktur) splatnost faktur
Interest: Single – (Firmy.Urok) úrok (z prodlení)
Credit: Currency – (Firmy.Kredit) kredit
Discount: string – (Firmy.Sleva) číselná hodnota slevy, text se ignoruje
PriceListID: string – (Ceniky.rowguid) ID ceníku
Delivery: string(30) – (Firmy.Doprava) doprava
SettlementType: string(30) – (Firmy.Zpusob_uhrady) způsob úhrady
Language: string(30) – (Firmy.Jazyk) jazyk
Note: string(8000) – (Firmy.Poznamky) poznámka
Category: string(255) – (Firmy.Kategorie) zařazení do kategorií
EAN: string(13) – (Firmy.EAN_firmy) čárový kód
Supplier: boolean – (Firmy.Dodavatel) je firma dodavatel?
SupplierVarNumber: string(30) – (Firmy.Variabilni_symbol_dodavatel) variabilní symbol dodavatele
Customer: boolean – (Firmy.Odberatel) je firma odběratel?
CustomerVarNumber: string(30) – (Firmy.Variabilni_symbol_odberatel) variabilní symbol odběratele
Data1: string(100) – (Firmy.Udaj_1) údaj 1
Data2: string(100) – (Firmy.Udaj_2) údaj 2
Number1: double – (Firmy.Pocet_1) počet 1
Number2: double – (Firmy.Pocet_2) počet 2
CountryISO: string(2) – (Firmy.ISO_zeme) ISO země
Addresses: TAddresses
Banks: TBanks
Title: string(20) – (Firmy.Titul) titul, pokud je zákazník osoba
FirstName: string(30) – (Firmy.Krestni) křestní jméno, pokud je zákazník osoba
SecondName: string(30) – (Firmy.Druhe_krestni) druhé jméno, pokud je zákazník osoba
Surname: string(30) – (Firmy.Prijmeni) příjmení, pokud je zákazník osoba
TitleAfter: string(20) – (Firmy.Za_prijmenim) titul za jménem, pokud je zákazník osoba
Vocative: string(30) – (Firmy.Osloveni) oslovení, pokud je zákazník osoba
SurnameVocative: string(30) – (Firmy.Osloveni_prijmeni) oslovení příjmení, pokud je zákazník osoba
PriceGroup: string(30) – (Firmy.Kategorie_cen) kategorie cen zákazníka (skupina ceníků, z těchto ceníků platí pro daný produkt nejvýhodnější cena)
Position: string(50) – (Firmy.Funkce) funkce, např. osoby ve firmě
Send: string(255) – (Firmy.Zasilat) klíčová slova, co zasílat osobě, seznam oddělený středníkem
Login: string(20) – (Firmy.Prihlasovaci_jmeno) přihlašovací jméno pro využití např. webem
Password: string(64) – (Firmy.Heslo) heslo pro využití např. webem
DateOfBirth: TXSDateTime – (Firmy.Datum_narozeni) datum narození
BirthNumber: string(20) – (Firmy.Rodne_cislo) rodné číslo
CameFrom:string(30) – (Firmy.Odkud_prisel) odkud byl kontakt získán
ContactObtainDate TXSDateTime – (Datum_ziskani_kontaktu) datum získání kontaktu
State: string(20) – (Firmy.Stav_kontaktu) stav
Active: boolean – (Firmy.Nabizet_v_dokladech) určuje, zda se má kontakt nabízet
UserFields: string – hodnoty uživatelských polí v XML formátu, např.
<pole1>hodnota</pole1><pole2>hodnota</pole2>

Actualize: TArray – pokud je alespoň jeden prvek pole, budou se při aktualizaci zákazníka aktualizovat jen zadaná pole, pokud je pole prázdné, aktualizuje se vše

TCustomers = array of TCustomer

TContact deprecated ,nahrazeno TCustomer‘

Osoba z firmy (zákazníka)

ID: string – (Firmy.rowguid) ID osoby, pokud se nepošle při založení, doplní se, při aktualizaci povinné

CustomerID: string – (Firmy.rowguid) ID zákazníka (firmy), ke kterému osoba patří

ContactType: string(2) – (Firmy.Typ_zaznamu) typ záznamu (OS – osoba)

ContactTypeName: string(30) – (Firmy.Typ_kontaktu) název typu osoby

Title: string(20) – (Firmy.Titul) titul

FirstName: string(30) – (Firmy.Krestni) křestní jméno

SecondName: string(30) – (Firmy.Druhe_krestni) druhé jméno

Surname: string(30) – (Firmy.Prijemni) příjmení

TitleAfter: string(20) – (Firmy.Za_prijmenim) titul za jménem

Telephone: string(40) – (Firmy.Telefon_1) telefon

Cellphone: string(40) – (Firmy.Telefon_2) mobilní telefon

Fax: string(40) – (Firmy.Fax) fax

Email: string(100) – (Firmy.E_mail) e-mail

Position: string(50) – (Firmy.Funkce) pozice

Data1: string(100) – (Firmy.Udaj_1) údaj 1

Data2: string(100) – (Firmy.Udaj_2) údaj 2

Number1: double – (Firmy.Pocet_1) počet 1

Number2: double – (Firmy.Pocet_2) počet 2

Send: string(255) – (Firmy.Zasilat) zasílat co

SalesAgent: string(30) – (Firmy.Referent) referent

Login: string(20) – (Firmy.Prihlasovaci_jmeno) přihlašovací jméno

Password: string(64) – (Firmy.Heslo) heslo

Orders: boolean – smí objednávat

Rmas: boolean – smí tvořit reklamace

Look: boolean – smí prohlížet

Vocative: string(30) – (Firmy.Osloveni) oslovení

SurnameVocative: string(30) – (Firmy.Osloveni_prijmeni) oslovení příjmení

UserFields: **string** – hodnoty uživatelských polí v XML formátu, např.
<pole1>hodnota</pole1><pole2>hodnota</pole2>

Actualize: TArray – pokud je alespoň jeden prvek pole, budou se při aktualizaci osoby aktualizovat jen zadaná pole, pokud je pole prázdné, aktualizuje se vše

TDocumentItem

Položka dokladu

ID: string – (Polozky_dokladu.rowguid) ID položky dokladu, pokud se nepošle při založení, doplní se, při aktualizaci povinné

DocumentID: string – (Doklady.rowguid) ID dokladu, pokud se položka zapisuje samostatně, nutno vyplnit

DocumentOrderNumber: integer – (Polozky_dokladu.Polozka_dokladu) číslo (pořadí) položky, v rámci dokladu musí být unikátní

Description: string(255) – (Polozky_dokladu.Popis) popis

ItemNumber: string(30) – (Polozky_dokladu.Cislo) (katalogové) číslo

Quantity: Double – (Polozky_dokladu.Mnozstvi) množství

QuantityUnit: string(30) – (Polozky_dokladu.Jednotky) jednotka

GPL: Currency – (Polozky_dokladu.Cena_zakladni) základní cena

PricePerUnit: Currency – (Polozky_dokladu.Cena_za_jednotku) cena za jednotku, při zápisu nutno vyplnit
PriceWithoutVAT: Currency – (Polozky_dokladu.Cena_bez_DPH) cena celkem bez DPH, při zápisu nutno vyplnit
TotalVAT: Currency – (Polozky_dokladu.DPH_celkem) celkem DPH, při zápisu nutno vyplnit
TotalPrice: Currency – (Polozky_dokladu.Cena_s_DPH) celková cena včetně DPH, při zápisu nutno vyplnit
VATRate: Double – (Polozky_dokladu.Sazba_DPH) sazba DPH, při zápisu nutno vyplnit
DiscountRate: Single – (Polozky_dokladu.Sleva_polozky) výše slevy
VATType: string(30) – (Polozky_dokladu.Zdanitelne_plneni) typ zdanitelného plnění (Základ daně, Z tuzemská, ...)
StoreID: string – (Knihy.rowguid) ID skladu
ProductID: string – (Katalog.rowguid) ID produktu
VariantID: string – (Katalog_varianty_produkту.rowguid) ID varianty
State: string(100) – (Polozky_dokladu.Stav) stav položky (DODAT, rezervovat, REZEROVÁNO, fakturovat lze zapsat)
OrderID: string – (Doklady.rowguid) ID související zakázky (např. u faktur)
DeliveryDate: TXSDateTime – (Polozky_dokladu.Datum_dodani) (požadovaný) datum dodání
QuantityGroups: TQuantityGroups
DeliveryNoteID: string – (Doklady.rowguid) ID dodacího listu (výdejky)
DeliveryNoteItemID: string – (Polozky_dokladu.rowguid) ID položky dodacího listu (výdejky)
CommissionID: string – (Doklady.rowguid) ID související zakázky
CommissionItemID: string – (Polozky_dokladu.rowguid) ID související položky zakázky
Note: string(255) – (Polozky_dokladu.Poznamka_polozky) poznámka
Data1: string(255) – (Polozky_dokladu.Udaj_polozky_1) Údaj 1
Data2: string(255) – (Polozky_dokladu.Udaj_polozky_2) Údaj 2
Number1: double – (Polozky_dokladu.Pocet_polozky_1) Počet 1
Number2: double – (Polozky_dokladu.Pocet_polozky_2) Počet 2
AccountCode: string(30) – (Polozky_dokladu.Predkontace) Textová předkontace položky, nebo číslo účtu
ExternID: string – volné pole pro případnou identifikaci položky shopem
UserFields: string – hodnoty uživatelských polí v XML formátu, např. <pole1>hodnota</pole1><pole2>hodnota</pole2>

TDocumentItems = array of TDocumentItem

TDocument

Doklad

ID: string – (Doklady.rowguid) ID dokladu, pokud se nepošle při založení, doplní se, při aktualizaci povinné
Number: string(30) – (Doklady.Cislo_dokladu) číslo dokladu, pokud se nepošle při zápisu, doplní se podle číselné řady
Book: string(30) – (Doklady.Kniha) kniha, pokud se nepošle při zápisu, doplní se výchozí
DocumentName: string(30) – (Doklady.Doklad) název dokladu (Zakázka, Zálohová faktura, Faktura, ...)
DocumentType: string(2) – (Doklady.Typ_dokladu) typ dokladu, při zápisu nutno vyplnit (ZZ – zakázka, ZV – záloha vydaná, FV – faktura vydaná, SV – skladová výdejka, VV – vratka výdejky, PP – pokladní doklad příjmový, PV – pokladní doklad výdajový, ...)
Currency: string(5) – (Doklady.Mena) měna
DontMakeInvoice: boolean – (Doklady.Fakturovat) pokud je true, doklad se nemá fakturovat
VarNumber: string(30) – (Doklady.Variabilni_symbol) variabilní symbol (u typu dokladu ZZ se použije pole OrderNumber, pokud není vyplněno)
Comment: string(255) – (Doklady.Komentar) komentář
Status: string(30) – (Doklady.Stav_dokladu) stav dokladu
Text: string(255) – (Doklady.Text) text (obvykle se tiskne před položkami)
Date: TXSDateTime – (Doklady.Datum) datum vzniku dokladu
TaxDate: TXSDateTime – (Doklady.Datum_zdanitelneho_plneni) DUZP
SettlementDate: TXSDateTime – (Doklady.Datum_splatnosti) datum splatnosti
SettlementMethod: string(30) – (Doklady.Zpusob_uhrady) způsob úhrady

IO: Integer – (Doklady.PV) směr toku peněz (+1 faktura, pokladní příjmový doklad, výdejka, ..., 0 stornovaný doklad, -1 dobropis, pokladní výdajový doklad, vratka výdejky, ...)

PlaceOfSupply: string(10) – (Doklady.Misto_plneni) místo plnění pro účely DPH

TotalWithoutVAT: Currency – (Doklady.Celkem_bez_DPH) celkem bez DPH, při zápisu nutno zadat

TotalWithVAT: Currency – (Doklady.Celkem_s_DPH) celkem s DPH, při zápisu nutno zadat

Rounding: Currency – (Doklady.Zaokrouhleni) velikost zaokrouhlení

RequestedAdvance: Currency – (Doklady.Pozadovana_zaloha) požadovaná záloha

AdvancePayed: Currency – (Doklady.Uhrazena_zaloha) zaplacená záloha

Total: Currency – (Doklady.Celkem) celkem, při zápisu nutno zadat

Payed: Currency – (Doklady.Uhrazeno) zaplacen

SettlementLeft: Currency – (Doklady.Zbyva_uhradit) zbývá uhradit, při zápisu faktur nutno zadat

VATRoundingPlace: Single – (Doklady.Zaokrouhlovat_DPH) zaokrouhlování DPH (např. 0,01 znamená na halíře)

SumRoundingPlace: Single – (Doklady.Zaokrouhlovat_soucet) zaokrouhlování součtu (např. 1,00 znamená na koruny)

Interest: Single – (Doklady.Urok) úrok z prodlení

CompanyID: string – (Firmy.rowguid) ID zákazníka (firmy), CustomerID

DeliveryCompanyID: string – (Firmy.Kontakt_zbozi_dorucit), CustomerID pro zboží doručit

CompanyName: string(255) – (Doklady.Nazev_firmy) obchodní jméno zákazníka (Customer)

PersonName: string(60) – (Doklady.Jmeno) jméno osoby

Addresses: TAddresses

IC: string(20) – (Doklady.ICO) IČ

DIC: string(20) – (Doklady.DIC) DIČ

Telephone: string(40) – (Doklady.Telefon) telefon

Email: string(100) – (Doklady.E_mail) e-mail

BankName: string(50) – (Doklady.Banka) jméno banky, bankovní spojení

BankBranch: string(50) – (Doklady.Pobočka) pobočka banky

AccountNumber: string(35) – (Doklady.Cislo_uctu) číslo účtu

BankCode: string(11) – (Doklady.Kod_banky) kód banky

SpecificSymbol: string(20) – (Doklady.Specificky_symbol) specifický symbol

IBAN: string(35) – (Doklady.IBAN)

SalesAgent: string(30) – (Doklady.Referent) referent

DueDateDays: integer – (Doklady.Splatnost_dni) počet dní splatnosti

Category: string(255) – (Doklady.Kategorie) kategorie dokladu (více hodnot oddělených středníkem)

PriceGroup: string(30) – (Doklady.Kategorie_cen) kategorie cen dokladu (skupina ceníků, z těchto ceníků platí pro daný produkt nejvýhodnější cena)

PricelistID: string – (Ceniky.rowguid) ID ceníku

PricelistName: string(30) – (Doklady.Cenik) název ceníku

Discount: string(30) – (Doklady.Sleva) název nebo výše slevy

Delivery: string(30) – (Doklady.Zpusob_dopravy) způsob dopravy

OrderNumber: string(30) – (Doklady.Objednavky) číslo objednávky (u typu dokladu ZZ, pokud není vyplněno, jde o Doklady.Variabilni_symbol) číslo objednávky

OneDelivery: boolean – (Doklady.Dodat_najednou) dodat najednou

Data1: string(100) – (Doklady.Udaj_1) údaj 1

Data2: string(100) – (Doklady.Udaj_2) údaj 2

Note: string(255) – (Doklady.poznamky) poznámka

UserFields: string – hodnoty uživatelských polí v XML formátu, např.
 <pole1>hodnota</pole1><pole2>hodnota</pole2>

DocumentItems: TDocumentItems

TProductExtType = (petNothing, petImage, petRef, petFile)

Typ rozšířených informací k produktu (obrázek, odkaz, soubor)

TProductExt

Rozšířené informace k produktu

Id: **string** – (Katalog_odkazy.rowguid, nebo T_Katalog_Obrázky.rowguid, nebo T_Katalog_Odkazy.rowguid, nebo T_Katalog_Soubory.rowguid) ID rozšířené informace

PEType: **TProductExtType**

ProductID: **string** – (Katalog.rowguid) ID produktu

VariantID: **string** – (Katalog_varianty_produkту.rowguid) ID varianty produktu

Language: **string** – (Katalog_odkazy.Jazyk, nebo T_Katalog_Obrázky.Jazyk_TKO, nebo T_Katalog_Odkazy.Jazyk_TKOD, nebo T_Katalog_Soubory.Jazyk_TKS) jazyk rozšířené informace

Order: **Byte** – (Katalog_odkazy.Poradi) pořadí

PEName: **string(50)** – (T_Katalog_Obrázky.Nazev_TKO, nebo T_Katalog_Odkazy.Nazev_TKOD, nebo T_Katalog_Soubory.Nazev_TKS) název rozšířené informace

Description: **string(255)** – (Katalog_odkazy.Popis, nebo T_Katalog_Obrázky.Popis_TKO, nebo T_Katalog_Odkazy.Popis_TKOD, nebo T_Katalog_Soubory.Popis_TKS) popis

FileName: **string(255)** – (Katalog_odkazy.URL, nebo T_Katalog_Obrázky.Soubor_TKO, nebo T_Katalog_Soubory.Soubor_TKS) jméno souboru

Data: **TByteArray** – binární data

TProductExts = array of TProductExt

TElectroFee

Elektropoplatek

ID: **string** – (Elektrozarizeni.rowguid) ID elektropoplatku

FeeName: **string(125)** – (Elektrozarizeni.Popis_skupiny) název poplatku

Fee: **Currency** – (Elektrozarizeni.Poplatek_bez_DPH) velikost poplatku

TAuthorshipFee

Autorský poplatek

ID: **string** – (T_Autorske_poplatky.rowguid) ID autorského poplatku

FeeName: **string(125)** – (T_Autorske_poplatky.Popis_skupiny) název poplatku

Fee: **Currency** – (T_Autorske_poplatky.Poplatek_bez_DPH) velikost poplatku

FeePercent: **Double** – (T_Autorske_poplatky.Poplatek_procent) velikost poplatku v procentech z ceny

TReturnItem

Položka reklamace

ID: **string** – ID položky reklamace, pokud se nezadá při založení, doplní se, při aktualizaci povinné

ReturnID: **string** – ID reklamace, pokud se zakládá samostatně, je povinné

ReturnItemOrder: **integer** – jedinečné pořadí v rámci reklamace

ProductIDOld: **string** – ID reklamovaného produktu

ProductIDNew: **string** – ID nového produktu

SerialNumberOld: **string** – sériové číslo reklamovaného produktu

SerialNumberNew: **string** – sériové číslo nového produktu

DefectDescription: **string** – popis závady

Accessories: **string** – příslušenství

Acknowledgement: **string** – vyjádření k reklamaci

InvoiceID: **string** – ID faktury

DeliveryNoteID: **string** – ID dodacího listu

TReturnItems = array of TReturnItem

TReturn

Reklamace (hlavička)

ID: string – ID reklamace, pokud je prázdné při založení, tak se doplní, při aktualizaci povinné

ReturnNumber: string – číslo reklamace, pokud je prázdné při zápisu, doplní se podle číselné řady

CustomerReturnNumber: string – číslo reklamace zákazníka

FiledDate: TXSDateTime – datum reklamace

ResolutionDate: TXSDateTime – datum ukončení

Agent: string - referent

CustomerID: string – ID zákazníka

CustomerName: string – obchodní jméno zákazníka, případně jméno a příjmení

Address: string - adresa

Telephone: string - telefon

Email: string – e-mail

ReturnState: string – stav reklamace

DeliveryFromCustomer: string – způsob přijetí

DeliveryToCustomer: string – způsob odeslání

Resolution: string – typ vyřízení

Note: string – poznámka

OrderID: string – ID zakázky

InvoiceID: string – ID faktury

ReturnItems: TReturnItems

TRelationType = (rtAdvanceOnOrder)

Typ relace mezi doklady (záloha k zakázce)

TRecordType = (rtDocument, rtCustomer)

Typ záznamu (doklad, firma)

TActivityLog

Záznam v deníku

ID: string – (Denik.rowguid) ID záznamu

RecordID: string – (Doklady.rowguid) ID dokladu, ke kterému se záznam vztahuje

RecordType: TRecordType

LogType: string(30) – (Denik.Typ_polozky) Typ položky deníku

Subject: string(150) – (Denik.Predmet) předmět

AutorName: string(60) – (Denik.Jmeno) jméno autora

Agent: string(30) – (Denik.Referent) referent

DateAndTime: TXSDateTime – (Denik.Zacatek) datum a čas vytvoření

Note: string(8000) – (Denik.Poznamka_deniku) poznámka

TActivityLogs = array of TActivityLog

TTranslationFieldType = (tftProduct, tftCategory, tftParameter)

Typ objektu pro překlad pole (produkt, kategorie, parametr)

TTranslation

Překlad pole

ID: string – (Katalog_cizi_nazvy.rowguid, nebo T_Preklad.rowguid, nebo T_Preklad_Kategorie.rowguid, nebo T_Preklad_Parametry.rowguid) ID překladu

RecordID: string – (Katalog.rowguid, nebo Kategorie.rowguid, nebo T_Parametry.rowguid) ID objektu, který se překládá

FieldType: TTranslationFieldType

FieldName: string(50) – (Katalog_cizi_nazvy.Pole, nebo T_Preklad.Pole_TP, nebo ‚CategoryName‘, nebo ‚ParameterName‘) jméno pole

Language: string(2) – (Katalog_cizi_nazvy.Jazyk, nebo T_Preklad.IDJazyk_TP, nebo T_Preklad_Kategorie.IDJazyk_TPK, nebo T_Preklad_Parametry.IDJazyk_TPP) jazyk

Value: string(8000) – (Katalog_cizi_nazvy.Popis_cizi, nebo T_Preklad.Preklad_TP, nebo T_Preklad_Kategorie.Preklad_TPK, nebo T_Preklad_Parametry.Preklad_TPP) hodnota, překlad

TTranslations = array of TTranslation

TCoupon

Slevový kupón

Price: Currency - hodnota

SerialNumber: string(30) – sériové číslo

TVATRate

Číselník sazeb DPH

ID: string – (Sazby_DPH.rowguid) ID sazby

VATName: string(30) – (Sazby_DPH.Oznaceni) název sazby DPH (použit v TProduct)

Country: string(2) – (Sazby_DPH.Zeme) země, pro kterou sazba platí

VATRate: double – (Sazby_DPH.Sazba_DPH) procentní výše sazby

ValidFrom: TXSDateTime – (Sazby_DPH.Datum_platnosti_od) datum platnosti od

ValidTo: TXSDateTime – (Sazby_DPH.Datum_platnosti_do) datum platnosti do

TVATRates = array of TVATRate

TPriceMatrixType = (pmtStandard, pmtSale);

Typ nastavení Matice slev (základní, akční)

TPriceMatrixSetting

Nastavení Matice slev

ID: string – (T_Matice_Slev.rowguid) ID nastavení

Order: integer – (T_Matice_Slev.Poradi_MS) pořadí nastavení

CustomerID: string – (T_Matice_Slev.Firma_MS) ID firmy, zákazníka

CustomersGroup: string – (T_Matice_Slev.SkupinaFirmy_MS) skupina firem

ProductID: string – (T_Matice_Slev.Produkt_MS) ID produktu

ProductsGroup: string – (T_Matice_Slev.SkupinaKatalog_MS) skupina produktů

Discount: double – (T_Matice_Slev.Sleva_MS) výše slevy

Price: Currency – (T_Matice_Slev.Cena_MS) cena

DefaultPriceList: string – (T_Matice_Slev.CenikVychozi_MS) ID výchozího ceníku

PriceList: string – (T_Matice_Slev.Cenik_MS) ID ceníku

Rounding: double – (T_Matice_Slev.Zaokrouhleni_MS) zaokrouhlení

Name: string – (T_Matice_Slev.Nazev_MS) název

ValidFrom: TDateTime – (T_Matice_Slev.Od_MS) platnost od

ValidTo: TDateTime – (T_Matice_Slev.Do_MS) platnost do

Currency: string – (T_Matice_Slev.Mena_MS) měna

Type: TPriceMatrixType – (T_Matice_Slev.Typ_MS) typ

TFile

Soubory k danému záznamu (např. dokladu)

ID: string – (Dokumenty.rowguid)

RecordID: string – (Doklady.rowguid) ID dokladu, ke kterému se soubor vztahuje

Subject: string(150) – (Dokumenty.Predmet) předmět

FileName: string(30) – (Dokumenty.Typ_dokumentu) typ souboru

Author: string(30) – (Dokumenty.Autor) referet, který soubor vložil

DateTime: TXSDateTime – (Dokumenty.Datum_a_cas) datum a čas vložení souboru

Description: string(255) – (Dokumenty.Popis) popis souboru

Category: string(255) – (Dokumenty.Kategorie_dokumentu) kategorie, seznam hodnot oddělený středníkem

Version: integer – (Dokumenty.Verze) – verze souboru

FilePath: string(255) – (Dokumenty.Soubor) cesta k souboru z hlediska serveru

TFiles = array of TFile

Přehled aktualizací

Zde se nachází popis novinek ve službě, opravy chyb, úpravy objektů, případně další změny.

verze 0.0.6.241 (15.03.2022)

- TDiscount.DiscountType
- + TDiscount.Name, TDiscount.Currency, TDiscount.ValidFrom, TDiscount.ValidTo, TDiscount.WithVAT, TDiscount.PriceGroup
- + TDiscountRate
- přesun TDiscount.PriceFrom a TDiscount.Value do TDiscountRate

verze 0.0.6.240 (24.01.2022)

- + TSupply.QuantityMinimum, TSupply.QuantityMaximum, TSupply.QuantityMaximum, TSupply.PurchasingStrategy
- + TAvailableSupply.QuantityMinimum, TAvailableSupply.QuantityMaximum, TAvailableSupply.QuantityMaximum, TAvailableSupply.PurchasingStrategy

verze 0.0.6.239 (03.11.2021)

- + CreateOrUpdateStockLevel a CreateOrUpdateStockLevels aktualizuje informace u všech skladových karet (skladovou zásobu jen u externích skladů)

verze 0.0.6.238 (06.10.2021)

- + TVariant.ProductID
- + metoda CreateOrUpdateProduct zakládá i varianty (Katalog variant)
- + CreateOrUpdateVariant

verze 0.0.6.237 (23.08.2021)

- + function GetVariants: TStringArray
- + function GetVariant(const ID: string): TVariant
- + TObjectType.otVariant
- + TVariant.Manufacturer, TVariant.Supplier, TVariant.Weight
- + TVariant.QuantityMinimum, TVariant.QuantityMaximum
- + TVariant.QuantityReorderPoint, TVariant.Warranty, TVariant.WWW
- + TVariant.Data1, TVariant.Data2, TVariant.Number1, TVariant.Number2
- + TVariant.Dimensions, TVariant.Width, TVariant.Height
- + TVariant.Length, TVariant.QualityStandard
- + TVariant.PurchasingStrategy, TVariant.New, TVariant.Action
- + TVariant.TopProduct, TVariant.Availability
- + TVariant.SellOff, TVariant.Annotation, TVariant.Sale
- + metoda GetJobsData zpracovává i otSupply a otAvailableSupply

verze 0.0.6.236 (11.08.2021)

- ! přesun typů TAuthHeader a TPropHeader z namespace "VarioSOAPServiceIntf" do "VarioSOAPTYPES"
- + rezervaci položky lze vytvořit pomocí SQL procedury

verze 0.0.6.235 (10.08.2021)

- + TProduct.QuantityMinimum, TProduct.QuantityMaximum, TProduct.PurchasingStrategy

verze 0.0.6.234 (05.08.2021)

+ TDocumentItem.AccountCode

verze 0.0.6.233 (02.08.2021)

+ TCustomer.Active

verze 0.0.6.232 (02.07.2021)

+ TVariant.Visible

verze 0.0.6.231 (30.06.2021)

+ TDocument.PlaceOfSupply

verze 0.0.6.230 (29.06.2021)

+ TProduct.CustomsTariff, TProduct.CustomsNumberOfUnits
+ TProductSimple.CustomsTariff, TProductSimple.CustomsNumberOfUnits

verze 0.0.6.229 (22.06.2021)

+ TVariant.ImageRef a TVariant.SmallImageRef
+ TProductExt.VariantID
+ GetSuppliesByIDs a GetAvailableSuppliesByIDs
+ GetProductExts a GetProductExtsArray vrací data i podle ID variant
- vypuštění vlastnosti TCustomer.DiscountID (již se dlouho nepoužívá)

verze 0.0.6.228 (09.03.2021)

+ GetJobsData zpracovává i otCategory

verze 0.0.6.227 (03.03.2021)

+ TProduct.PricePurchase, TProductSimple.PricePurchase

verze 0.0.6.225 (02.02.2021)

+ TPriceList.Supplier

verze 0.0.6.224 (15.01.2021)

+ TProduct.RecyclingCosts a TProductSimple.RecyclingCosts

verze 0.0.6.223 (31.12.2020)

+ metoda CreateOrUpdateDocumentItem počítá součet dokladu

verze 0.0.6.222 (09.09.2020)

+ metody CreatePayment a FinishPayment podporují úhrady přijatých dokladů
+ procedure CreateRequestAdvance(const ADocumentID: string; const ARequested, APaid: Currency)

verze 0.0.6.221 (12.08.2020)

+ function GetPricesByProduct(const AProductID: string): TPrices

verze 0.0.6.220 (06.08.2020)

+ function CreateOrUpdateProductExt(AProductExt: TProductExt): string
+ procedure CreateOrUpdateProductExts(AProductExts: TProductExts)

verze 0.0.6.219 (20.07.2020)

+ TProduct.DesignProductsID

verze 0.0.6.218 (29.06.2020)

+ function FinishPayment(const APaidDocumentID, APayingItemID: string; const AAmount: Currency): string
+ napojení na doplněk Parametry produktů
+ TParameter.ParentID, TParameter.Type, TParameter.Definition

verze 0.0.6.217 (10.06.2020)

+ TStore.StoreType, TStore.DoNotUse
+ TStockLevel, TStockLevels
+ CreateOrUpdateStockLevel
+ CreateOrUpdateStockLevels

verze 0.0.6.216 (31.05.2020)

+ funkce UpdateDocumentBodyFields(ADocument: TDocument; const Actualize: TStringArray): string

verze 0.0.6.215 (24.02.2020)

+ TCustomer.CameFrom
+ TCustomer.ContactObtainDate

verze 0.0.6.214 (24.01.2020)

+ Vyvolání výjimky, pokud nebude nalezen nějaký požadovaný záznam

verze 0.0.6.213 (11.11.2019)

+ TCustomer.CustomerVarNumber, TCustomer.SupplierVarNumber, TCustomer.State

verze 0.0.6.212 (25.07.2019)

+ TObjectType.otProductCategory
+ TProductCategory
+ function GetProductCategories: TStringArray;
+ function GetProductCategory(const ID: string): TProductCategory;

verze 0.0.6.211 (28.06.2019)

+ TProduct.DiscountCategory a TProductSimple.DiscountCategory

verze 0.0.6.210 (18.06.2019)

+ Pro překlady se primárně používá tabulka Data####.Katalog_cizi_nazvy

verze 0.0.6.209 (12.12.2018)

+ TProduct.BasicUnit a TProductSimple.BasicUnit

verze 0.0.6.208 (23.10.2018)

+ TAvailableSupply.States

verze 0.0.6.207 (18.10.2018)

- TAvailableSupply.ArrivalDate respektuje variantu

verze 0.0.6.206 (08.10.2018)

+ TProduct.EANs a TVariant.EANs typu TEANS (array of TEAN)

verze 0.0.6.205 (07.08.2018)

+ function GetProductsByCriteria(const Code, ProductNumber, EAN, EANInternal, ProductName: string; const UseOr: boolean): TProducts;

verze 0.0.6.204 (22.05.2018)

+ function CreateOrUpdateCategory(ACategory: TCategory): string
+ procedure DeleteCategory(const ID: string);

verze 0.0.6.203 (14.02.2018)

+ možnost v INI zadat Polozky_dokladu.Vlastnosti

verze 0.0.6.202 (02.02.2018)

+ function UpdateProduct(AProduct: TProduct; const Actualize: TStringArray): string;

verze 0.0.6.201 (19.01.2018)

+ procedure DeletePriceMatrixSetting(const ID: string)

verze 0.0.6.200 (05.12.2017)

+ automatické vyplnění pole Polozky_dokladu.Vlastnosti hodnotou 19 (avVlastnostiPolozkyCenaZadana + avVlastnostiPolozkyCenaZadanaCenaBD + avVlastnostiPolozkyCenaZakladniZadana)

verze 0.0.6.199 (06.11.2017)

+ TPropHeader

verze 0.0.6.198 (24.10.2017)

+ pokud metoda CreateOrUpdateDocument vystaví k zakázce zálohovou fakturu, tuto zálohu uhradí částkou v Zakazka.AdvancePayed

verze 0.0.6.197 (03.10.2017)

+ TDocumentItem.UserFields

verze 0.0.6.196 (29.08.2017)

+ TDocumentItem.Data1, TDocumentItem.Data2
+ TDocumentItem.Number1 a TDocumentItem.Number2

verze 0.0.6.195 (22.08.2017)

+ TProduct.WWW a TProductSimple.WWW
+ funkce CreateOrUpdatePriceMatrixSetting(const APriceMatrixSetting: TPriceMatrixSetting): string
+ TFile a TFiles
+ funkce GetDocumentFiles(const DocumentID: string): TFiles

verze 0.0.6.194 (22.03.2017)

+ TProductSimple.UserFields

verze 0.0.6.193 (13.03.2017)

+ při automatické tvorbě zálohové faktury se vyplňují následující pole podle zakázky:
Zaloha.VarNumber = Zakazka.OrderNumber, pokud je prázdné tak Zakazka.Number
Zaloha.PricelistID = Zakazka.PricelistID
Zaloha.OrderNumber = Zakazka.OrderNumber
Zaloha.Delivery = Zakazka.Delivery
Zaloha.DueDateDays = Zakazka.DueDateDays

verze 0.0.6.192 (03.03.2017)

+ TProduct.Book a TProductSimple.Book

verze 0.0.6.191 (22.02.2017)

+ TDocument.Category a TDocument.PriceGroup

verze 0.0.6.190 (02.02.2017)

+ function GetProductsPage(const PageSize, PageNumber: integer): TStringArray

verze 0.0.6.189 (10.01.2017)

- náhrada TStringDynArray na TStringArray

verze 0.0.6.188 (20.12.2016)

- explicitní zjištění verze Varia z registru pro předání VarioLib

verze 0.0.6.187 (06.12.2016)

+ možnost změnit pole Firmy.Firma v CreateOrUpdateCustomer, pokud existuje
sp_VarioSOAP_ZmenaPoleFirma

verze 0.0.6.186 (10.10.2016)

+ TAddress.UseOnDocuments

verze 0.0.6.185 (31.08.2016)

+ TExceptionCode = (ecInternal, ecDuplicated, ecMustHaveValue, ecUnknowValue, ecNotFound, ecNotAllowed)
+ EVarioSOAPException.ExceptionCode a EVarioSOAPException.ExceptionSource

verze 0.0.6.184 (02.08.2016)

- TProductSetPart.Count, TSupply.Quantity a TAvailableSupply.Quantity změněno z Cardinal na Double

verze 0.0.6.183 (27.06.2016)

+ TCategory.Shop

verze 0.0.6.182 (27.06.2016)

+ TProduct.ActionDescription a TProduct.MarketingAction

verze 0.0.6.181 (07.06.2016)

+ TDocument.UserFields

verze 0.0.6.180 (04.03.2016)

+ TBank.SpecificSymbol

verze 0.0.6.179 (26.02.2016)

+ TCustomer.Supplier a TCustomer.Customer
+ procedure DeleteCustomer(const ID: string; const DeleteSubordinate: boolean)
+ procedure DeleteBank(const ID: string)

verze 0.0.6.178 (13.11.2015)

+ TProductExt.Order
+ TCategory.Image: TByteDynArray
+ TCategory.ImageRef: string

verze 0.0.6.177 (06.11.2015)

+ TProductSetPart.Authorized

verze 0.0.6.176 (14.10.2015)

+ rozšíření ActivityLog o práci se záznamy deníku pro Customer (Firma)
+ TRecordType.rtCustomer
+ TActivityLog.LogType

verze 0.0.6.175 (12.10.2015)

+ změna v TQuantityGroup (odebráno GroupName, Description; přidáno LOT, SerialNumber, States)
+ šablona dokladu podle státu v adrese

verze 0.0.6.174 (13.08.2015)

+ TProduct.InPriceList

verze 0.0.6.173 (07.08.2015)

+ TDocument.DeliveryCompanyID

verze 0.0.6.172 (09.07.2015)

- Změny v pojetí OrderNumber a VarNumber v TDocument u typu dokladu ZZ, pokud jsou obě pole vyplněna, používají se samostatně, pokud VarNumber vyplněno není, bere se z OrderNumber jako v předchozích verzích

verze 0.0.6.171 (24.06.2015)

+ TPriceMatrixSetting
+ funkce GetPriceMatrixSettings: TStringArray;
+ funkce GetPriceMatrixSetting(const ID: string): TPriceMatrixSetting;
+ job otPriceMatrixSetting

verze 0.0.6.170 (10.06.2015)

+ TCustomer.EAN

verze 0.0.6.169 (08.06.2015)

+ TCategory.B2BVisible

verze 0.0.6.168 (01.06.2015)

+ změna parametrů v function GetDocumentsByCriteria(const CustomerIdentificator, DocumentType: string; const DateFrom, DateTo, SettlementDateFrom, SettlementDateTo: TXSDateTime; const Payed: TXSBoolean; const PageSize, PageNumber: integer): TDocuments;

verze 0.0.6.167 (25.05.2015)

+ TProduct.AllowOrderByCustomer

verze 0.0.6.166 (14.05.2015)

+ funkce GetFile(const AFileName: string): TByteDynArray;

verze 0.0.6.165 (10.04.2015)

+ šablona dokladu podle DIČ
+ TProductExt.Language

verze 0.0.6.164 (15.04.2015)

+ šablona dokladu podle DIČ

verze 0.0.6.163 (10.03.2015)

+ možnost změnit XML Parser (DOM) - "MSXML" (výchozí), nebo "Omni XML"
+ function GetDocumentsByCriteria(const CustomerIdentificator, DocumentType: string; const DateFrom, DateTo, SettlementDateFrom, SettlementDateTo: TXSDateTime; const Payed: TXSBoolean): TDocuments;

verze 0.0.6.162 (13.02.2015)

+ TCustomer.Book, TCustomer.Position, TCustomer.Send, TCustomer.Login, TCustomer.Password, TCustomer.DateOfBirth, TCustomer.BirthNumber

verze 0.0.6.161 (09.02.2015)

+ TPricelist.Currency

verze 0.0.6.160 (29.01.2015)

+ TAddress.ID: string a TAddress.AddressName: string
- odstranění Vario11

verze 0.0.6.159 (05.12.2014)

+ TPrice.QuantityFrom

verze 0.0.6.158 (07.11.2014)

+ funkce TVarioSOAPService.GetTranslationsArray(const IDList: TStringArray): TTranslations;

verze 0.0.6.157 (05.11.2014)

+ nastavení INI.[ZZ].zpusobyUhradyProVystaveniZalohy

verze 0.0.6.156 (27.10.2014)

+ TCustomer.CountryISO
+ TAddress.CountryISO (hodnota z číselníku zemí)

verze 0.0.6.155 (21.10.2014)

+ Delphi XE7
- oprava chyby v GetTranslation

verze 0.0.6.154 (09.09.2014)

+ funkce GetAvailableSuppliesArray(const ProductIDList: TStringArray): TAvailableSupplies;

verze 0.0.6.153 (29.08.2014)

+ TVATRate, TVATRates
+ metoda GetVATRatesData(const Country: string; const ValidFrom, ValidTo: TXSDateTime): TVATRates;

verze 0.0.6.152 (03.07.2014)

+ pole Addresses vrací u Adresy zboží doručit, Adresy sídlo a Adresy 2 i rozpis adresy
+ do položky dokladu lze vložit i variantu z číselníku variant

verze 0.0.6.151 (24.06.2014)

+ funkce CreateOrUpdateProduct vyplňuje i tabulku Související_produkty

verze 0.0.6.150 (29.05.2014)

+ TCustomer.Number1, TCustomer.Number2
+ TContact.Data1, TContact.Data2, TContact.Number1, TContact.Number2
- oprava aktualizace číselné řady pro jednotlivé záznamy
- oprava založení signalizační tabulky #VarioSynchroRunning při založení produktů

verze 0.0.6.149 (29.04.2014)

+ TVariants vrací i záznamy z číselníku variant

verze 0.0.6.148 (18.04.2014)

+ Delphi XE6
+ do TProductSimple přidáno Visible, New, Action, Height, Sirka, Hloubka

verze 0.0.6.147 (01.04.2014)

+ ořezání textového pole na počet znaků včetně změny #10 na #13#10 (AdjustLineBreaks)

verze 0.0.6.146 (29.01.2014)

+ odstranění vazeb na VarioLib mimo volání funkce ZaznamDokladPolozka.NastavitStav
+ obehnutí chyby VarioLib.ZaznamDokladPolozka.NastavitStav pomocí volání VarioLibNastavitStav.exe
+ optimalizace vytváření polí (SetLength)
+ výchozí měna dokladů se načítá z nastavení Varia

verze 0.0.6.145 (23.12.2013)

+ změna dbExpress na FireDAC

verze 0.0.5.144 (21.11.2013)

+ při požadavku na nastavení stavu VYDÁNO na faktuře se vytváří výdejka do související knihy
+ TProduct.EANInternal, TProductSimple.EANInternal (čárový kód vlastní)

verze 0.0.5.143 (12.11.2013)

+ TProduct.QualityStandard a TProductSimple.QualityStandard: string – jakostní norma

verze 0.0.5.142 (30.10.2013)

+ výchozí stav pro položky bez produktu

verze 0.0.5.141 (18.10.2013)

+ Variolib volá vždy jen jedno vlákno

verze 0.0.5.140 (11.09.2013)

+ funkční, i pokud není databáze Parametrů Plus
+ změna direktivy podmíněné kompilace z Vario12 na Vario11

verze 0.0.5.139 (29.08.2013)

+ TDocument.DontMakeInvoice
+ GetDocuments vrazí i doklady ZV

verze 0.0.5.138 (20.08.2013)

+ procedure ChangeDocumentItemsState(const IDs: TStringArray; const State: string);
+ procedure CreateOrUpdatePrices(const Prices: TPrices);

verze 0.0.5.137 (02.08.2013)

+ GetProductExt vrací i obrázky, odkazy a soubory z tabulky Data####.Katalog_odkazy

verze 0.0.5.136 (09.07.2013)

+ function GetCustomersByCriteria

verze 0.0.5.135 (04.07.2013)

+ multidoménovost pomocí TAuthHeader

verze 0.0.5.134 (27.05.2013)

+ TCustomer.CustomerType, TCustomer.CustomerTypeName, TCustomer.UserFields
+ TCustomer.MainCustomerID
+ TContact.CustomerType, TContact.CustomerTypeName, TContact.UserFields

verze 0.0.5.133 (01.05.2013)

+ funkce GetBalicky a GetBalicek

verze 0.0.5.132 (30.04.2013)

+ TDocument.Currency

verze 0.0.5.131 (09.04.2013)

+ funkce GetPriceMatrix(const CustomerID, Currency: string; const ProductIDList: TStringArray):
TPriceMatrixPrices;

verze 0.0.5.130 (04.04.2013)

+ přidání Cena_zakladni_s_DPH a Cena_zakladni_s_DPH_m do aktualizace a vložení řádku tabulky
Polozky_dokladu
+ InsertQuery nevkládá null hodnoty

verze 0.0.5.129 (20.03.2013)

+ zápis TDocument.Status

verze 0.0.5.128 (26.02.2013)

+ TCategory.HyperLink: string, TCategory.Description: string, TCategory.Visible: boolean

verze 0.0.5.127 (20.02.2013)

+ TCustomer.PriceGroup
+ job otPriceList
+ procedure ChangeID(const AObjectType: TObjectType; const AOldID, ANewID: string);

verze 0.0.5.126 (11.02.2013)

+ TPriceList.PriceGroup

verze 0.0.5.125 (12.12.2012)

+ Delphi XE3
+ TProduct.SellOff, TProductSimple.SellOff
+ TPrice.VariantID

verze 0.0.4.124 (03.12.2012)

- oprava vkládání kurzu dokladu u měn

verze 0.0.4.123 (16.11.2012)

+ TProduct.Annotacion, TProductSimple.Annotation

verze 0.0.4.122 (05.11.2012)

- ve Variu 12 se dávají měnové položky 0, pokud je měna dokladu CZK

verze 0.0.4.121 (16.10.2012)

+ metoda CreatePayment do založeného denního výpisu (případně do existujícího) vloží doklad a uhradí

verze 0.0.4.120 (04.10.2012)

- špatně se posílalo smazání záznamu v GetJobsData (vše bylo typ otPrice)

verze 0.0.4.119 (21.08.2012)

+ nezáleží na velikosti písmen u stavu
+ v ini lze nastavit u záznamu výchozí "sklad", "vychoziStavPolozky" a "vychoziStavPolozkySL"

verze 0.0.4.118 (26.07.2012)

+ funkce CreateOrUpdateProduct(AProduct: TProduct): string, jen pro tabulky Katalog, Cenik, Kategorie_produkty, T_Katalog

verze 0.0.4.117 (19.07.2012)

- vyplnění měny a kurzu podle souvisejícího dokladu v CreateRelationship

verze 0.0.4.116 (27.06.2012)

+ funkce GetSQLData

verze 0.0.4.115 (13.06.2012)

+ TAddress.Country
+ TCustomer.Category
- oprava vyplnění položek podle šablony (dříve nerespektovalo knihu dokladu)

verze 0.0.4.114 (17.05.2012)

+ pokud je produkt typu SL, tak se na položku nedává stav mimo fakturovat a nedává se sklad

verze 0.0.4.113 (02.04.2012)

+ TPrice.ID

verze 0.0.4.112 (27.02.2012)

+ umožňuje zapnout logování nastavením v ini, sekce [service], logovatprichozi=1 a logovatodchozi=1 (po restartu)

verze 0.0.4.111 (01.02.2012)

- hlídání platnosti množstevní skupiny pro Vario 12

verze 0.0.4.110 (15.12.2011)

+ TDocumentItem.ExternID pro účely e-shopu
+ zrychlení TSQLDataSet (GetMetadata:=false)

verze 0.0.4.109 (06.12.2011)

+ TProduct.AccountCodeSell
+ na položku ZZ, FV, DV a ZV se vkládá předkontace prodej z produktu

verze 0.0.4.108 (05.12.2011)

+ umožňuje zapnout logování nastavením v ini, sekce [service], logovatvse=1 (po restartu)

verze 0.0.4.107 (21.11.2011)

+ GetPricesByPricelistValid vrací dnes platné ceny daného ceníku

verze 0.0.4.106 (14.10.2011)

+ metoda GetProductExtsArray

verze 0.0.4.105 (11.10.2011)

+ pole TPricelist.Sale

verze 0.0.4.104 (10.10.2011)

+ pole TPrice.Data1

verze 0.0.4.103 (27.09.2011)

+ při založení dokladu, pokud se nepošle zdanitelné plnění a sazba DPH je větší, než 0, doplní se Základ daně
+ pole TProduct.DefaultStoreID, vyplnění skladu u položky dokladu podle výchozího skladu produktu
+ pole TProduct.UserFields, vyplňuje (formát XML-like) se hodnotami polí uvedených v ini v sekci Katalog, ident
UzivatelstvaPole

verze 0.0.4.102 (23.09.2011)

+ v TProduct a TProductSimple přidáno Manufacturer (Výrobce)

verze 0.0.4.101 (20.09.2011)

+ v TProduct a TProductSimple přidáno ProductType a ProductTypeName

verze 0.0.4.100 (13.09.2011)

+ TProduct.Dimensions, TProductSimple.Dimensions

verze 0.0.4.99 (08.09.2011)

- v aktualizacích (BuildUpdateQuery) a insertovacích (BuildInsertQuery) dotazech se názvy polí dávají do
uvozovek

verze 0.0.4.98 (18.08.2011)

+ podpora měny dokladu, vyplní se podle měny knihy

verze 0.0.4.97 (01.08.2011)

+ do TCustomer a TContact přidáno pole Actualize, pokud je alespoň jeden prvek, budou se aktualizovat jen
vyjmenovaná pole
+ TProduct.TopProduct
+ v ini v sekci vario atribut pouzivatpolev12mistoparametru

verze 0.0.4.96 (01.08.2011)

+ do TProduct přidáno pole B2BMinimumOrderQuantity (Katalog.u_Mnozstvi_minimalni_shop)

verze 0.0.4.95 (21.07.2011)

- funkce GetJobsData vrací i deleted objekty

verze 0.0.4.94 (28.06.2011)

+ přidána function GetJobsByType(const ACount: integer; const AObjectType: TObjectType): TJobs; stdcall;
+ upravena metoda SetJobsData tak, aby nastavovala jedním dotazem

verze 0.0.4.93 (22.06.2011)

+ function GetPricesByPriceList(const APriceListID: string): TPrices; stdcall;
+ do TPrice přidáno ProductID
+ function GetPrice(const ID: string): TPrice; stdcall;
+ TProductSimple, otProductSimple
+ function GetProductSimple(const ID: string): TProductSimple; stdcall;
+ předefinování GetJobsData na function GetJobsData(const ACount: integer; const AObjectType: TObjectType): TJobsData;
+ GetJobsData umí otProductSimple, otPrice
+ stav položky rezervovat