
Servant API

Technická dokumentace

Obsah

Úvod	4
Terminologie a základní fakta o datové výměně a zobrazených informacích.....	5
Testování datového napojení	6
Operace	8
Ověření spojení a přihlašovacích údajů.....	8
Načtení stavu objednávek na příjem.....	8
Načtení stavu objednávek na výdej.....	8
Načtení inventurního stavu	8
Načtení aktuálního seznamu dopravců	8
Odeslání seznamu partnerů	8
Odeslání sortimentu	9
Odeslání objednávky na příjem	9
Odeslání objednávky na výdej.....	9
Funkce	10
CanAccess.....	10
GetAllReceiptOrders.....	10
GetReceiptOrdersByDate	10
GetReceiptOrder	11
GetReceiptOrders.....	11
GetReceiptOrderByDocument.....	11
GetReceiptOrdersByDocument	12
GetAllIssueOrders.....	12
GetIssueOrdersByDate	12
GetIssueOrder	13
GetIssueOrders.....	13
GetIssueOrderByDocument	13
GetIssueOrdersByDocument	14
GetInventory	14

GetCarriers	14
GetImportResults	15
ImportPartners	15
ImportProducts	15
ImportReceipt.....	16
ImportIssue.....	16
ImportFiles.....	16
Struktury.....	17
ResultType	17
CredentialsType.....	18
DateTimeIntervalType	18
ActivityType	18
ItemType.....	19
OrderItemType	19
TransportItemType.....	19
CashOnDeliveryType	20
DimensionType.....	20
EanType	20
ReceiptOrderType	20
IssueOrderType	21
InventoryItemType	22
CarrierItemType	22
ImportRequestResultType.....	23
ImportResultType	24
InfoTextType.....	25
ImportBankType	25
ImportPartnerDefinitionType.....	25
ImportItemIdType	25
ImportProductItemDetailsType.....	26
ImportProductDimensionType	26
ImportProductEanType	26
ImportItemIdDetailsType	27
ImportLabelType	27
ImportActivityType	27
ImportProductActivityType	28
ImportProductType	28
ImportPartnerType.....	28
ImportDetailsType.....	29

ImportDeliveryType.....	29
ImportItemType	30
ImportReceiptType.....	30
ImportIssueType.....	30
ImportFileType	31
Použití.....	32
C#.....	32
PHP	33
SAPI Aplikace	33
Export z Infoportálu.....	34
Import na Infoportál.....	38
Řízení přes příkazovou řádku	39
Externí nástroje a aplikace	41
Ověření XML dotazu vůči WSDL předpisu	41

Úvod

Dokument obsahuje popis SW rozhraní webové služby pro získání stavu objednávek a inventurního stavu a odeslání importu sortimentu a objednávek na výdej a příjem. Je určen pro vývojáře SW informačního systému na straně zákazníka společnosti Servant a.s. Rozhraní je postavené na protokolu SOAP a popisu rozhraní webové služby WSDL. Toto rozhraní je možné jednoduše použít v rámci datového napojení psaného v C# nebo PHP. Data lze přes rozhraní získat také pomocí SAPI Aplikace, která následně umožňuje data exportovat v několika nejběžnějších formátech.

URL: <https://www.webskladservant.cz/new/impl/SAPI/V5/?wsdl>

Verze SW rozhraní webové služby: V5.9

Verze dokumentu: V5.2024-10-23

Datová komunikace je šifrovaná pomocí TLS 1.2. **Tato verze šifrování je na straně klienta vyžadována** (připojení zabezpečené nižší verzí šifrování selže). Všichni uživatelé Infoportálu dostupného přes webový server na adrese <https://www.webskladservant.cz/> získávají také automaticky přístup přes SW rozhraní webové služby. Pro přihlášení se používá stejných přihlašovacích údajů, jakých se používá pro přihlášení na Infoportál.

SW rozhraní webové služby nabízí následující operace:

- [Ověření spojení a přihlašovacích údajů](#)
- [Načtení stavu objednávek na příjem](#)
- [Načtení stavu objednávek na výdej](#)
- [Načtení inventurního stavu](#)
- [Načtení aktuálního seznamu dopravců](#)
- [Odeslání seznamu partnerů](#)
- [Odeslání sortimentu](#)
- [Odeslání objednávky na příjem](#)
- [Odeslání objednávky na výdej](#)

POZOR: Maximální denní počet API dotazů může být limitován. V případě blížícího se limitu dochází při API dotazech, společně s daty, k návratu varování. Při překročení nastaveného denního limitu dojde do dalšího dne k zastavení možnosti API komunikace. Oba tyto stavy jsou indikovány ve struktuře [ResultType](#) i na hlavní stránce Infoportálu po přihlášení. V případě, že chcete předcházet této situaci, obzvláště při vyšší počtu dotazovaných objednávek, je potřeba využívat agregační funkce, tj. ptát se jedním dotazem na více objednávek. Pro více informací kontaktuje operátora Servant.

Terminologie a základní fakta o datové výměně a zobrazených informacích

V rámci dokumentu i další komunikace se můžete setkat s následujícími termíny:

<u>Infoportál</u>	Webové stránky umožňující komunikaci (datovou výměnu) se středisky společnosti Servant
<u>Servant API</u>	API (programové rozhraní) pro přístup na Infoportál
<u>SAPI aplikace</u>	Windows aplikace využívající Servant API

Infoportál je určen pro uživatele, kteří chtějí mít nepřetržitý přístup k objednávkám, inventuře zboží, atd. přes webové stránky. Servant API poskytuje totožné možnosti jako Infoportál s tím rozdílem, že je primárně určený pro programové napojení na informační systém zákazníka. SAPI aplikace je nadstavba nad vlastní API (aplikace interně Servant API přímo používá) a může sloužit jak k testovacím účelům, tak i k uživatelskému přístupu z PC. Aplikaci lze rovněž ovládat z příkazové řádky bez zobrazení jakéhokoli okna a tím docílit napojení na další procesy na serveru zákazníka.

POZOR: Pro funkce importu, tj. funkce [Odeslání seznamu partnerů](#), [Odeslání sortimentu](#), [Odeslání objednávky na příjem](#) a [Odeslání objednávky na výdej](#) lze v současné době využít pouze Servant API, resp. přidružené SAPI aplikace. Infoportál tyto funkčnosti zatím přes webové stránky nezpřístupňuje.

- Od 17. 6. 2022 je možné zadat [Odeslání objednávky na výdej](#) i přes webový formulář z Infoportálu. Pokud na Infoportálu tuto volbu nevidíte a rádi byste formuláře využívali, kontaktujte operátora Servant.
- Od 25. 1. 2023 je možné zadat [Odeslání objednávky na příjem](#) i přes webový formulář z Infoportálu. Pokud na Infoportálu tuto volbu nevidíte a rádi byste formuláře využívali, kontaktujte operátora Servant.
- Od 23. 8. 2023 je možné zadat [Odeslání sortimentu](#) i přes webový formulář z Infoportálu. Pokud na Infoportálu tuto volbu nevidíte a rádi byste formuláře využívali, kontaktujte operátora Servant.

Synchronizace dat na Infoportál probíhá z několika zdrojů a k synchronizaci dochází průběžně tak, jak se vyřizují požadavky na import a zpracovávají se vlastní objednávky. Je tedy možné, že v určitých okamžik budou informace zobrazené na Infoportálu nekompletní či neaktuální. Typicky se jedná o následující situace:

~~Q Odešlu objednávku na výdej, ale nevidím její položky. Proč?~~

~~A K načtení položek dojde až ve chvíli potvrzeného výdeje položek. V tuto chvíli je také možné vidět požadované a reálně vydané množství, případně i výstupy VAS prováděných nad položkami.~~

- Od 30. 6. 2022 jsou kódy požadovaných položek a jejich množství zobrazeny ihned po přijetí objednávky

~~Q Odešlu objednávku na výdej, ale nevidím informace o požadavku na dopravu. Proč?~~

~~A K načtení informací o dopravě dojde až po zadání dopravy do systému zvoleného dopravce / předání dopravci. **POZNÁMKA:** Pro dopravy OSB, IND a PDF dochází k zobrazení už ve chvíli načtení objednávky.~~

- Od 27. 7. 2022 je požadovaná doprava zobrazena ihned po přijetí objednávky

Q Číslo přepravy již bylo přiděleno, nicméně stále nevidím informace o dopravě. Proč?

A Tato situace je indikována rovněž varováním „Doprava pro tuto objednávku nebyla nalezena, informace o dopravě nebudou zobrazeny.“ a může být způsobena buď nenačtením dat o dopravě, případně chybou. Prosím počkejte minimálně 1 hodinu. Pokud chyba přetrvá, kontaktujte operátora Servant.

Mezi další hojně řešené situace patří:

- Q** Servant API mi vrací chybu „Chyba přístupu k serveru“ s popisem „Chyba při komunikaci se serverem Servant, zkuste to, prosím, později“. Proč?
- A** Chyba může být způsobena nedostupností serveru Servant. Pokud chyba přetrvává, zkontrolujte validitu odesílaných dat proti WSDL předpisu API rozhraní. Stává se, obzvláště při implementaci v PHP kde SoapClient nevaliduje vstup a provádí implicitní konverze, že zadaný vstup není platný. Týká se to především zadání NULL do atributů, které nulové být nemohou.
- Q** Servant API mi vrací chybu „Chyba zadání importu (Servant server)“ s popisem „Failed - Neexistující ukladatel“. Proč?
- A** S největší pravděpodobností ještě nemáte plně aktivovaný účet (na cílovém středisku Servant není tento účet známý). Kontaktujte operátora Servant.
- Q** Co mám vyplnit do atributu „warehouse“?
- A** Jedná se o zkratku střediska. Zkratku skladu dostanete přidělenou podle toho, ve kterém skladě skladujete. Existují následující zkratky: **HU2** = Humpolec - Lnářská, **BR2** = Brunka, **SV2** = Svěmyslice, **PH2** = Praha, **PE2** = Humpolec - Pelhřimovská.

Testování datového napojení

Pro testování neexistuje žádné speciální testovací prostředí. Poté, co od nás obdržíte přístup na Infoportál (a nastavíte si heslo), je zároveň automaticky aktivované i Servant API. Ne všechny funkce, které jsou poskytované Servant API, jsou využitelné pro vaši společnost. Požadavky na funkční obsah přenášených dat a vhodnost jednotlivých funkcí pro vaši společnost prosím konzultujte s naším IT technikem. Ten vám také případně sdělí konstanty, které je nutné do komunikace zahrnout.

Doporučujeme začít s:

1. Sjednání přístupu na Infoportál a seznámení se s touto dokumentací
2. Vytvoření základů datové komunikace a [Ověření spojení a přihlašovacích údajů](#)
3. [Odeslání sortimentu](#) a sledování stavu importu sortimentu voláním [GetImportResults](#)
4. [Odeslání seznamu partnerů](#) a sledování stavu importu
5. [Odeslání objednávky na příjem](#) a sledování stavu importu
6. [Načtení stavu objednávek na příjem](#)
7. [Načtení inventurního stavu](#)
8. [Odeslání objednávky na výdej](#) a sledování stavu importu
9. [Načtení stavu objednávek na výdej](#)

V případě, že selže vlastní import nebo dojde k následné chybě při zpracování dat, dostanete informaci o chybě. Pokud selže vlastní import (např. neplatné přihlašovací údaje, chybný formát XML dat vůči WSDL definici, apod.), chyba je reportovaná synchronně jako výsledek SOAP dotazu. **V případě nemožnosti odeslat nějaký XML dotaz si prosím vždy ověřte, že jste validní vůči WSDL předpisu (viz kapitola [Ověření XML dotazu vůči WSDL předpisu](#)).** Pokud selže následné zpracování dat (např. duplicitní objednávka, snaha vyskladnit nenaskladněný nebo neexistující sortiment, apod.), chyba je reportovaná asynchronně dotazem na stav importu ([GetImportResults](#)).

Po bezchybném přenosu testovací databáze sortimentu ([Odeslání sortimentu](#)), seznamu partnerů ([Odeslání seznamu partnerů](#)) a objednávky na příjem ([Odeslání objednávky na příjem](#)), prosím kontaktujte našeho IT technika, abychom v našem skladovém systému zpracovali testovací objednávku (potvrdíme testovací příjem a tím zboží fiktivně naskladníme). V tuto chvíli již máte na Infoportále informace o stavu této objednávky i o skladové zásobě ([Načtení inventurního stavu](#)). Pokračovat je možné vytvořením objednávky na výdej ([Odeslání objednávky na výdej](#)). Po jejím bezchybném přenosu nás opět informujte, abychom objednávku zpracovali. Stejně jako v případě objednávky na příjem, je po

zpracování i této objednávky na Infoportále možné získat její stav. Stav se aktualizuje tak, jak je objednávka realizovaná a tedy je možné následnými dotazy například sledovat stav nakládky, doručení, atd. Aktualizovaný stav skladových zásob ([Načtení inventurního stavu](#)) si můžete stahovat vždy každý večer kolem 18 té hodiny.

Pokud budete potřebovat, fyzicky odešleme zkušební zásilky.

Z praxe máme vyzkoušeno, že pokud se držíte manuálu a respektujete normu a prostředí, na které se manuál odvolává, pak technickou část nastavení datové výměny lze provést bez naší spolupráce.

Před ostrým provozem všechna zkušební data smažeme a nebudou tak zatěžovat ostrý provoz, na který se všichni těšíme.

Operace

Ověření spojení a přihlašovacích údajů

Ověřit spojení a platnost přihlašovacích údajů lze přes funkci [CanAccess](#). Volání této funkce před voláním dalších funkcí pro načtení objednávek nebo inventurního stavu není vyžadováno. Volání lze využít například pro jednorázové ověření přístupu na SW rozhraní webové služby nebo pro ověření dostupnosti serveru.

Načtení stavu objednávek na příjem

Objednávky na příjem lze načíst voláním jedné z následujících funkcí:

- [GetAllReceiptOrders](#) – vrátí všechny objednávky
- [GetReceiptOrdersByDate](#) – vrátí všechny objednávky, které odpovídají intervalu
- [GetReceiptOrder](#) – vrátí **jednu** konkrétní objednávku identifikovanou číslem objednávky
- [GetReceiptOrders](#) – vrátí **více** konkrétních objednávek identifikovaných čísly objednávek
- [GetReceiptOrderByDocument](#) – vrátí **jednu** konkrétní objednávku identifikovanou dokladem
- [GetReceiptOrdersByDocument](#) – vrátí **více** konkrétních objednávek identifikovaných doklady

Načtení stavu objednávek na výdej

Objednávky na výdej lze načíst voláním jedné z následujících funkcí:

- [GetAllIssueOrders](#) – vrátí všechny objednávky
- [GetIssueOrdersByDate](#) – vrátí všechny objednávky, které odpovídají intervalu
- [GetIssueOrder](#) – vrátí **jednu** konkrétní objednávku identifikovanou číslem objednávky
- [GetIssueOrders](#) – vrátí **více** konkrétních objednávek identifikovaných čísly objednávek
- [GetIssueOrderByDocument](#) – vrátí **jednu** konkrétní objednávku identifikovanou dokladem
- [GetIssueOrdersByDocument](#) – vrátí **více** konkrétních objednávek identifikovaných doklady

Načtení inventurního stavu

Položky inventurního stavu lze získat voláním funkce [GetInventory](#). Funkce vrací vždy kompletní stav inventury (všechny položky). Funkce neumožňuje data filtrovat pomocí parametrů, a tedy lze případné filtrování provádět až na straně klienta.

Načtení aktuálního seznamu dopravců

Aktuální seznam dopravců platný pro daného ukladatele lze získat voláním funkce [GetCarriers](#).

Odeslání seznamu partnerů

Voláním funkce [ImportPartners](#) se importují nové informace o partnerech. Partnerem je myšlen dopravce, ukladatel nebo odběratel.

Aktuální stav importu je možné získat voláním funkce [GetImportResults](#). **POZOR:** Přestože je indikované dokončení zpracování importu, může na pozadí k reálnému dokončení dojít až později bez ohledu na indikovaný stav zpracování. Zpracování a pronesení informací může reálně trvat až **30 minut**.

Odeslání sortimentu

Voláním funkce [ImportProducts](#) se importují nové informace o sortimentu. Funkce umožňuje nastavit jen některé atributy za účelem importu jen některých informací. Je tedy možné provést jen import EAN nebo prováděných VAS nad již definovaným zbožím. Funkce odesílá data přes Infoportál přímo na server Servant daného střediska. Pokud funkce skončí úspěchem, je předáno ID transakce, pomocí kterého je možné následně stav importu sledovat (vlastní import a navázané operace trvají netriviálně dlouhou dobu). V opačném případě je vrácen důvod selhání.

Aktuální stav importu je možné získat voláním funkce [GetImportResults](#). Zpracování a pronesení informací může trvat až **30 minut**.

Odeslání objednávky na příjem

Voláním funkce [ImportReceipt](#) se provede import objednávky na příjem. Funkce odesílá data přes Infoportál přímo na server Servant daného střediska. Pokud funkce skončí úspěchem, je předáno ID transakce, pomocí kterého je možné následně stav importu sledovat (vlastní import a navázané operace trvají netriviálně dlouhou dobu). V opačném případě je vrácen důvod selhání.

Aktuální stav importu je možné získat voláním funkce [GetImportResults](#). Zpracování a pronesení informací může trvat až **30 minut**.

Odeslání objednávky na výdej

Voláním funkce [ImportIssue](#) se provede import objednávky na výdej. K importu je možné připojit i dokument, který se automaticky vytiskne po zabalení objednávky (např. vaše faktura, návod na použití, dodací list, atp.), jehož data jsou kódována standardem Base64. Je možno vytisknout jen jeden dokument a jen jednou. Pokud tedy potřebujete přiložit např. 3 kopie jedné faktury, je nutné, aby PDF soubor obsahoval 3 stránky se stejným obsahem. Stejným způsobem je možné připojit i vlastní přepravní štítek. Soubor(y) je možné připojit i později voláním funkce [ImportFiles](#). Funkce odesílají data přes Infoportál přímo na server Servant daného střediska. Pokud funkce skončí úspěchem, je předáno ID transakce, pomocí kterého je možné následně stav importu sledovat (vlastní import a navázané operace trvají netriviálně dlouhou dobu). V opačném případě je vrácen důvod selhání.

Aktuální stav importu je možné získat voláním funkce [GetImportResults](#). Zpracování a pronesení informací může trvat až **30 minut**.

Funkce

CanAccess

Funkce pro ověření dostupnosti serveru a přihlašovacích údajů. Funkce přijímá jako vstup strukturu **CanAccessParams** a jako výstup poskytuje strukturu **CanAccessParamsResult**.

Syntaxe: **CanAccessParamsResult** CanAccess (**CanAccessParams**)

Vstup (CanAccessParams)		Výstup (CanAccessParamsResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType

GetAllReceiptOrders

Funkce pro získání všech objednávek na PŘÍJEM, které jsou k dispozici pod zadanými přihlašovacími údaji. Funkce přijímá jako vstup strukturu **GetAllReceiptOrdersParams** a jako výstup poskytuje strukturu **GetAllReceiptOrdersResult**. **POZOR:** Funkce může v rámci jednoho dotazu vrátit maximálně 1000 objednávek. V případě, že by dotaz měl vrátit objednávek více, bude vráceno prvních 1000 (seřazeno sestupně dle požadované expedice). V případě, že si přejete načíst konkrétní objednávky, je možné použít volání funkcí [GetReceiptOrdersByDate](#), [GetReceiptOrder](#) nebo [GetReceiptOrderByDocument](#).

Syntaxe: **GetAllReceiptOrdersResult** GetAllReceiptOrders (**GetAllReceiptOrdersParams**)

Vstup (GetAllReceiptOrdersParams)		Výstup (GetAllReceiptOrdersResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
		orders	ReceiptOrderType[]

GetReceiptOrdersByDate

Funkce pro získání všech objednávek na PŘÍJEM, které jsou k dispozici pod zadanými přihlašovacími údaji a zároveň vyhovují nastavenému intervalu data. Interval je aplikován na položku „načteno do systému“. Funkce přijímá jako vstup strukturu **GetReceiptOrdersByDateParams** a jako výstup poskytuje strukturu **GetReceiptOrdersByDateResult**. **POZOR:** Funkce může v rámci jednoho dotazu vrátit maximálně 1000 objednávek. V případě, že by dotaz měl vrátit objednávek více, bude vráceno prvních 1000 (seřazeno sestupně dle požadované expedice). Pokud k tomu dojde, je možné zúžit nastavený interval data, nebo v případě, že si přejete načíst konkrétní objednávku, je možné použít volání funkcí [GetReceiptOrder](#) nebo [GetReceiptOrderByDocument](#).

Syntaxe: **GetReceiptOrdersByDateResult** GetReceiptOrdersByDate (**GetReceiptOrdersByDateParams**)

Vstup (GetReceiptOrdersByDateParams)		Výstup (GetReceiptOrdersByDateResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
interval	DateTimeIntervalType	orders	ReceiptOrderType[]

GetReceiptOrder

Funkce vrátí detaily právě jedné objednávky na PŘÍJEM identifikované pomocí čísla objednávky. Pokud nebude objednávka nalezena, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetReceiptOrderParams](#) a jako výstup poskytuje strukturu [GetReceiptOrderResult](#).

Syntaxe: [GetReceiptOrderResult](#) GetReceiptOrder ([GetReceiptOrderParams](#))

Vstup (GetReceiptOrderParams)		Výstup (GetReceiptOrderResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
order	string	order	ReceiptOrderType

GetReceiptOrders

Agregační funkce pro hromadné získání detailu až 1000 objednávek na PŘÍJEM identifikovaných pomocí čísel objednávek. Pokud nebude nalezena ani jedna dotazovaná objednávka, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetReceiptOrdersParams](#) a jako výstup poskytuje strukturu [GetReceiptOrdersResult](#).

Syntaxe: [GetReceiptOrdersResult](#) GetReceiptOrder ([GetReceiptOrdersParams](#))

Vstup (GetReceiptOrdersParams)		Výstup (GetReceiptOrdersResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
orders	string[]	orders	ReceiptOrderType[]

GetReceiptOrderByDocument

Funkce vrátí detaily právě jedné objednávky na PŘÍJEM identifikované pomocí dokladu. Pokud nebude objednávka nalezena, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetReceiptOrderByDocumentParams](#) a jako výstup poskytuje strukturu [GetReceiptOrderByDocumentResult](#).

Syntaxe: [GetReceiptOrderByDocumentResult](#) GetReceiptOrderByDocument ([GetReceiptOrderByDocumentParams](#))

Vstup (GetReceiptOrderByDocumentParams)		Výstup (GetReceiptOrderByDocumentResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
document	string	order	ReceiptOrderType

GetReceiptOrdersByDocument

Agregační funkce pro hromadné získání detailu až 1000 objednávek na PŘÍJEM identifikovaných pomocí dokladů. Pokud nebude nalezena ani jedna dotazovaná objednávka, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetReceiptOrdersByDocumentParams](#) a jako výstup poskytuje strukturu [GetReceiptOrdersByDocumentResult](#).

Syntaxe: [GetReceiptOrdersByDocumentResult](#) [GetReceiptOrderByDocument](#) ([GetReceiptOrdersByDocumentParams](#))

Vstup (GetReceiptOrdersByDocumentParams)		Výstup (GetReceiptOrdersByDocumentResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
documents	string[]	orders	ReceiptOrderType[]

GetAllIssueOrders

Funkce pro získání všech objednávek na VÝDEJ, které jsou k dispozici pod zadanými přihlašovacími údaji. Funkce přijímá jako vstup strukturu [GetAllIssueOrdersParams](#) a jako výstup poskytuje strukturu [GetAllIssueOrdersResult](#). **POZOR:** Funkce může v rámci jednoho dotazu vrátit maximálně 1000 objednávek. V případě, že by dotaz měl vrátit objednávek více, bude vráceno prvních 1000 (seřazeno sestupně dle požadované expedice). V případě, že si přejete načíst konkrétní objednávky, je možné použít volání funkcí [GetIssueOrdersByDate](#), [GetIssueOrder](#) nebo [GetIssueOrderByDocument](#).

Syntaxe: [GetAllIssueOrdersResult](#) [GetAllIssueOrders](#) ([GetAllIssueOrdersParams](#))

Vstup (GetAllIssueOrdersParams)		Výstup (GetAllIssueOrdersResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
		orders	IssueOrderType[]

GetIssueOrdersByDate

Funkce pro získání všech objednávek na VÝDEJ, které jsou k dispozici pod zadanými přihlašovacími údaji a zároveň vyhovují nastavenému intervalu data. Interval je aplikován na položku „načteno do systému“. Funkce přijímá jako vstup strukturu [GetIssueOrdersByDateParams](#) a jako výstup poskytuje strukturu [GetIssueOrdersByDateResult](#). **POZOR:** Funkce může v rámci jednoho dotazu vrátit maximálně 1000 objednávek. V případě, že by dotaz měl vrátit objednávek více, bude vráceno prvních 1000 (seřazeno sestupně dle požadované expedice). Pokud k tomu dojde, je možné zúžit nastavený interval data, nebo v případě, že si přejete načíst konkrétní objednávku, je možné použít volání funkcí [GetIssueOrder](#) nebo [GetIssueOrderByDocument](#).

Syntaxe: [GetIssueOrdersByDateResult](#) [GetIssueOrdersByDate](#) ([GetIssueOrdersByDateParams](#))

Vstup (GetIssueOrdersByDateParams)		Výstup (GetIssueOrdersByDateResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
interval	DateTimeIntervalType	orders	IssueOrderType[]

GetIssueOrder

Funkce vrátí detaily právě jedné objednávky na VÝDEJ identifikované pomocí čísla objednávky. Pokud nebude objednávka nalezena, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetIssueOrderParams](#) a jako výstup poskytuje strukturu [GetIssueOrderResult](#).

Syntaxe: [GetIssueOrderResult](#) GetIssueOrder([GetIssueOrderParams](#))

Vstup (GetIssueOrderParams)		Výstup (GetIssueOrderResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
order	string	order	IssueOrderType

GetIssueOrders

Agregační funkce pro hromadné získání detailu až 1000 objednávek na VÝDEJ identifikovaných pomocí čísel objednávek. Pokud nebude nalezena ani jedna dotazovaná objednávka, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetIssueOrdersParams](#) a jako výstup poskytuje strukturu [GetIssueOrdersResult](#).

Syntaxe: [GetIssueOrdersResult](#) GetIssueOrder([GetIssueOrdersParams](#))

Vstup (GetIssueOrdersParams)		Výstup (GetIssueOrdersResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
orders	string[]	orders	IssueOrderType[]

GetIssueOrderByDocument

Funkce vrátí detaily právě jedné objednávky na VÝDEJ identifikované pomocí dokladu. Pokud nebude objednávka nalezena, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetIssueOrderByDocumentParams](#) a jako výstup poskytuje strukturu [GetIssueOrderByDocumentResult](#).

Syntaxe: [GetIssueOrderByDocumentResult](#) GetIssueOrderByDocument([GetIssueOrderByDocumentParams](#))

Vstup (GetIssueOrderByDocumentParams)		Výstup (GetIssueOrderByDocumentResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
document	string	order	IssueOrderType

GetIssueOrdersByDocument

Agregační funkce pro hromadné získání detailu až 1000 objednávek na VÝDEJ identifikovaných pomocí dokladů. Pokud nebude nalezena ani jedna dotazovaná objednávka, je indikován návratový kód operace 3 = NO_DATA (více viz [ResultType](#)). Funkce přijímá jako vstup strukturu [GetIssueOrdersByDocumentParams](#) a jako výstup poskytuje strukturu [GetIssueOrdersByDocumentResult](#).

Syntaxe: [GetIssueOrdersByDocumentResult](#) [GetIssueOrderByDocument](#) ([GetIssueOrdersByDocumentParams](#))

Vstup (GetIssueOrdersByDocumentParams)		Výstup (GetIssueOrdersByDocumentResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
documents	string[]	orders	IssueOrderType[]

GetInventory

Funkce pro získání všech položek inventury, které jsou k dispozici pod zadanými přihlašovacími údaji. Funkce přijímá jako vstup strukturu [GetInventoryParams](#) a jako výstup poskytuje strukturu [GetInventoryResult](#).

Syntaxe: [GetInventoryResult](#) [GetInventory](#) ([GetInventoryParams](#))

Vstup (GetInventoryParams)		Výstup (GetInventoryResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
		inventoryItems	InventoryItemType[]

GetCarriers

Funkce pro získání všech aktuálních dopravců, které jsou k dispozici pod zadanými přihlašovacími údaji. Funkce přijímá jako vstup strukturu [GetCarriersParams](#) a jako výstup poskytuje strukturu [GetCarriersResult](#).

Syntaxe: [GetCarriersResult](#) [GetCarriers](#) ([GetCarriersParams](#))

Vstup (GetCarriersParams)		Výstup (GetCarriersResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
		carrierItems	CarrierItemType[]

GetImportResults

Funkce pro získání jednoho nebo i více výsledků importu, které jsou k dispozici pod zadanými přihlašovacími údaji. Funkce přijímá jako vstup strukturu `GetImportResultsParams` a jako výstup poskytuje strukturu `GetImportResultsResult`. V případě, že je nastaven `includePDF`, budou položky v odpovědi doplněny o PDF report daného importu. **POZOR:** Funkce může v rámci jednoho dotazu přijmout požadavek na získání maximálně 100 výsledků importu. V případě, že by dotaz obsahoval požadavek na získání více výsledků importu, bude zpracováno prvních 100 požadavků v seznamu `transactionIds`.

Syntaxe: `GetImportResultsResult GetImportResults (GetImportResultsParams)`

Vstup (<code>GetImportResultsParams</code>)		Výstup (<code>GetImportResultsResult</code>)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
transactionIds	string[]	importResults	ImportResultType[]
includePDF	bool		

ImportPartners

Funkce pro import seznamu partnerů. Informace budou přeneseny na server Servant zvoleného střediska pod ukladatelem dle zadaných přihlašovacích údajů. Funkce přijímá jako vstup strukturu `ImportPartnersParams` a jako výstup poskytuje strukturu `ImportPartnersResult`.

Syntaxe: `ImportPartnersResult ImportPartners (ImportPartnersParams)`

Vstup (<code>ImportPartnersParams</code>)		Výstup (<code>ImportPartnersResult</code>)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
warehouse	string	importRequestResult	ImportRequestResultType
partners	ImportPartnerDefinitionType[]		

ImportProducts

Funkce pro import sortimentu, jeho EAN a prováděných VAS. Informace budou přeneseny na server Servant zvoleného střediska pod ukladatelem dle zadaných přihlašovacích údajů. Funkce přijímá jako vstup strukturu `ImportProductsParams` a jako výstup poskytuje strukturu `ImportProductsResult`.

Syntaxe: `ImportProductsResult ImportProducts (ImportProductsParams)`

Vstup (<code>ImportProductsParams</code>)		Výstup (<code>ImportProductsResult</code>)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
warehouse	string	importRequestResult	ImportRequestResultType
products	ImportProductType[]		

ImportReceipt

Funkce pro import objednávky na příjem. Informace budou přeneseny na server Servant zvoleného střediska pod ukladatelem dle zadaných přihlašovacích údajů. Funkce přijímá jako vstup strukturu **ImportReceiptParams** a jako výstup poskytuje strukturu **ImportReceiptResult**. **POZNÁMKA:** Přestože vstupní struktura umožňuje zadat soubory, není v současné době žádný soubor určený pro objednávky na příjem podporovaný. Jakékoli nastavení souboru bude mít za následek chybu importu.

Syntaxe: `ImportReceiptResult ImportReceipt(ImportReceiptParams)`

Vstup (ImportReceiptParams)		Výstup (ImportReceiptResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
warehouse	string	importRequestResult	ImportRequestResultType
receipt	ImportReceiptType		
files	ImportFileType[]		

ImportIssue

Funkce pro import objednávky na výdej. Informace budou přeneseny na server Servant zvoleného střediska pod ukladatelem dle zadaných přihlašovacích údajů. Funkce přijímá jako vstup strukturu **ImportIssueParams** a jako výstup poskytuje strukturu **ImportIssueResult**. V rámci importu je možné připojit právě jeden soubor expedice a právě jeden soubor s vlastním přepravním štítkem.

Syntaxe: `ImportIssueResult ImportIssue(ImportIssueParams)`

Vstup (ImportIssueParams)		Výstup (ImportIssueResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
warehouse	string	importRequestResult	ImportRequestResultType
issue	ImportIssueType		
Files	ImportFileType[]		

ImportFiles

Funkce pro import souboru(ů) k již existující objednávce. Funkce přijímá jako vstup strukturu **ImportFilesParams** a jako výstup poskytuje strukturu **ImportFilesResult**.

Syntaxe: `ImportFilesResult ImportFiles(ImportFilesParams)`

Vstup (ImportFilesParams)		Výstup (ImportFilesResult)	
Jméno	Typ	Jméno	Typ
credentials	CredentialsType	result	ResultType
order	string	importRequestResult	ImportRequestResultType
files	ImportFileType[]		

Struktury

ResultType

Struktura obsahující výsledek operace.

Jméno	Typ	Význam	Platné hodnoty
resultCode	int	Návratový kód operace	0 = OK 1 = INTERNAL_EXCEPTION 2 = INVALID_CREDENTIALS 3 = NO_DATA 4 = LIMITED_DATA
resultString	string	Textová reprezentace návratového kódu operace	10 až 20 = API_REQUESTS_WARNING_REACHED 20 až 30 = API_ACCESS_TIME_WARNING_REACHED 30 = API_REQUESTS_LIMIT_REACHED 40 = API_ACCESS_TIME_LIMIT_REACHED

OK

– operace skončila bez chyby

INTERNAL_EXCEPTION

– při zpracování dotazu došlo k interní chybě serveru

INVALID_CREDENTIALS

přihlašovací údaje jsou neplatné, nebo nemá daný uživatel k informacím přístup

NO_DATA

– operace byla dokončena, nicméně server neodeslal žádná data (parametrům dotazu neodpovídají žádná data; zkuste změnit parametry dotazu)

LIMITED_DATA

– operace byla dokončena, nicméně server vrátil méně dat, než je na serveru k dispozici (parametrům dotazu odpovídá více dat, než je server schopný v rámci jednoho dotazu poskytnout; zkuste změnit parametry dotazu)

API_REQUESTS_WARNING_REACHED

– varování – denní limit API dotazů se blíží; kód varování je kombinován s kódem OK, NO_DATA, nebo LIMITED_DATA – popisek i kód varování se sčítají

API_ACCESS_TIME_WARNING_REACHED

– varování – denní limit přístupové času k serveru se blíží; kód varování je kombinován s kódem OK, NO_DATA, nebo LIMITED_DATA – popisek i kód varování se sčítají

API_REQUESTS_LIMIT_REACHED

– chyba – denní limit API dotazů byl dosažen; API komunikace je až do dalšího dne zastavena

API_ACCESS_TIME_LIMIT_REACHED

– chyba – denní limit přístupové času k serveru byl dosažen; API komunikace je až do dalšího dne zastavena

CredentialsType

Struktura obsahující informace o uživateli požadující provedení operace.

Jméno	Typ	Význam	Platné hodnoty
email	string	E-mailová adresa registrovaná pro přihlášení na Infoportál www.webskladservant.cz	Platná emailová adresa, maximální délka je 60 znaků
password	string	Heslo pro přihlášení pod zadanou e-mailovou adresou	
switchForUser	bool	Nastavte na <code>true</code> , pokud si přejete přihlásit jako jiný uživatel	<code>true</code> nebo <code>false</code> (default)
switchForUserId	int	Pokud k tomu má přihlašovaný uživatel právo, přepne se po přihlášení interně uživatel na uživatele identifikovaného pomocí tohoto ID	<0,1000> nebo -1

Hodnoty `switchForUser` a `switchForUserId` jsou mimo výjimky použity pro vývoj vlastního SW rozhraní webové služby. Pro běžný přístup, prosím, nastavte `switchForUser = false` a `switchForUserId = -1`.

DateTimeIntervalType

Struktura umožňující nastavení rozsahu data a času.

Jméno	Typ	Význam	Platné hodnoty
from	dateTime	Datum a čas „od“	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace dateTime formátu definovaného XSD
to	dateTime	Datum a čas „do“	

ActivityType

Struktura obsahující informace o VAS prováděných nad položkou objednávky.

Jméno	Typ	Význam	Platné hodnoty
id	int	Číselný identifikátor VAS	≥ 0
name	string	Textové pojmenování VAS	Maximální délka je 120 znaků
value	string	Výstup VAS	Maximální délka je 120 znaků
count	float	Množství zboží	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD, platnost je na 2 desetinná místa
servantNumber	string	Balíkový štítek - číslo přidělené společnosti Servant a.s.	Maximální délka je 16 znaků
transportNumber	string	Číslo přidělené dopravcem (číslo dopravy)	Maximální délka je 32 znaků

ItemType

Struktura obsahující informace sdílené pro položky objednávky a položky inventury.

Jméno	Typ	Význam	Platné hodnoty
code	string	Kód sortimentu	Maximální délka je 60 znaků
name	string	Jméno sortimentu	Maximální délka je 120 znaků
attributes	string[]	Atributy	Pole o statické velikosti 3 položky, maximální délka jednoho prvku je 60 znaků
expiration	date	Expirace	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace date formátu definovaného XSD
batch	string	Šarže	Maximální délka je 60 znaků
unit	string	Měrná jednotka	Maximální délka je 8 znaků

OrderItemType

Struktura obsahující informace o jedné položce objednávky.

Jméno	Typ	Význam	Platné hodnoty
item	ItemType	Struktura obsahující informace sdílené pro položky objednávky a položky inventury	
requiredCount	float	Požadovaný počet kusů	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD, platnost je na 2 desetinná místa
count	float	Reálně vyskladněný počet kusů	
activities	ActivityType[]	VAS prováděné nad položkami objednávky	null nebo ≥ 0 položek

TransportItemType

Struktura obsahující informace o jedné položce dopravy objednávky.

Jméno	Typ	Význam	Platné hodnoty
unit	string	Nákladová jednotka přepravy	Maximální délka je 30 znaků
servantNumber	string	Balíkový štítek - číslo přidělené společností Servant a.s.	Maximální délka je 60 znaků
transportNumber	string	Číslo přidělené dopravcem (číslo dopravy)	Maximální délka je 60 znaků
statusUrl	string	URL pro sledování stavu dopravy na stránkách dopravce	
packageType	string	Typ přepravy (pro Českou Poštu se může jednat např. o DR, NP, apod.)	
weight	float	Hmotnost	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD, platnost je na 2 desetinná místa

Při osobní (OSB), individuální (IND) nebo vlastní dopravě s přepravním štítkem (PDF) není URL pro sledování dopravy na stránkách dopravce k dispozici a tato položka je vždy nulová. Stav dopravy lze vyčíst také přímo ze struktury [IssueOrderType](#) z položky „state“ (není nutné stav zjišťovat ze stránek dopravce).

CashOnDeliveryType

Struktura obsahující informace o dobírce dopravy.

Jméno	Typ	Význam	Platné hodnoty
amount	float	Cena dobírky v nastavené měně v případě, že je dobírka nastavená. V opačném případě je položka nastavená na 0.	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD, platnost je na 2 desetinná místa
currency	string	Měna dobírky	Formát dle ISO 4217
variableSymbol	string	Variabilní symbol, případně další informace potřebné pro identifikaci platby (oddělené pomocí ^)	Maximální délka je 30 znaků

DimensionType

Definice je totožná s typem [ImportProductDimensionType](#).

EanType

Struktura obsahující definici čárového kódu (balení) zboží.

Jméno	Typ	Význam	Platné hodnoty
ean	string	EAN	Maximální délka je 64 znaků
count	float	Počet MJ k EAN	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD
weight	float	Hmotnost v kg	
dimension	DimensionType	Rozměry	
velocity	float	Objem v m ³	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD
primary	bool	Výchozí EAN pro výpočty a tiskové sestavy	true nebo false

ReceiptOrderType

Struktura obsahující informace o objednávce na PŘÍJEM.

Jméno	Typ	Význam	Platné hodnoty
order	string	Označení objednávky	Maximální délka je 110 znaků
document	string	Číslo dokladu (např. 20-HU2-0203814)	Maximální délka je 60 znaků
invoice	string	Faktura	Maximální délka je 60 znaků
orderItems	OrderItemType[]	Jednotlivé položky objednávky	null nebo ≥ 0 položek
activities	ImportActivityType[]	VAS prováděné nad celou objednávkou	null nebo ≥ 0 položek
partnerName	string	Jméno partnera	Maximální délka je 60 znaků
partnerCode	string	Kód partnera	Maximální délka je 60 znaků
loaded	dateTime	Datum a čas načtení do systému	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace dateTime formátu definovaného XSD
received	dateTime	Datum a čas fyzického přijetí objednávky	
checkState	string	Popis stavu kontroly zboží	Maximální délka je 120 znaků
finished	dateTime	Datum a čas dokončení příjmu	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace dateTime formátu definovaného XSD
error	dateTime	Datum a čas chyby	

Všechny položky typu „date“ nebo „dateTime“ mohou být nastavené nebo nenastavené. Pokud je položka nenastavená, znamená to, že k události ještě nedošlo. Nenastavenou položku lze identifikovat tak, že je hodnota < 1.1.2000 00:00:00.

Položky objednávky („orderItems“) jsou načteny a uvedeny v datové struktuře typicky (pokud ne zvolil ukladatel jinak) až po 18. hodině toho dne, ve kterém došlo k jejich fyzickému naskladnění. Do té doby je indikováno 0 položek objednávky.

IssueOrderType

Struktura obsahující informace o objednávce na VÝDEJ.

Jméno	Typ	Význam	Platné hodnoty
order	string	Označení objednávky	Maximální délka je 110 znaků
document	string	Číslo dokladu (např. 20-HU2-0203814)	Maximální délka je 60 znaků
invoice	string	Faktura	Maximální délka je 60 znaků
orderItems	OrderItemType[]	Jednotlivé položky objednávky	null nebo ≥ 0 položek
activities	ImportActivityType[]	VAS prováděné nad celou objednávkou	null nebo ≥ 0 položek
transportItems	TransportItemType[]	Jednotlivé položky dopravy (přepravní balíky, palety, atd.)	null nebo ≥ 0 položek
partnerName	string	Jméno partnera	Maximální délka je 60 znaků
partnerCode	string	Kód partnera	Maximální délka je 60 znaků
recipientAddress	string	Doručovací adresa	Maximální délka je 120 znaků
cashOnDelivery	CashOnDeliveryType	Informace o dobírce dopravy	
note	string	Poznámka	Maximální délka je 120 znaků
state	string	Poslední stav dopravy	Maximální délka je 60 znaků
warning	string	Varování signalizované API	Maximální délka je 120 znaků
error	dateTime	Objedávka obsahuje chybu (například, že sortiment objednávky nebylo možné vykrýt v požadovaném množství)	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace dateTime formátu definovaného XSD
requiredDispatch	date	Požadovaný datum expedice	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace date formátu definovaného XSD
dispatch	dateTime	Datum a čas expedice	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace dateTime formátu definovaného XSD
loaded	dateTime	Datum a čas načtení do systému	
entered	dateTime	Datum a čas zadání do výroby	
packaged	dateTime	Datum a čas zabalení	
transportCarrier	string	Dopravce	Maximální délka je 30 znaků
transportEntered	dateTime	Datum a čas přidělení čísla/čísel dopravy	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace dateTime formátu definovaného XSD
transportPicked	dateTime	Datum a čas vyzvednutí dopravcem	
transportDelivered	dateTime	Datum a čas doručení	

V případě, že objednávka obsahuje více položek dopravy, reflektuje položka „state“ stav všech těchto položek.

Dopravce může nabývat jedné z následujících hodnot:

GLS	GLS (https://gls-group.eu/CZ/cs/home)
CP	Česká Pošta (https://www.ceskaposta.cz/index)
FOFR	FOFR (https://www.fofrcz.cz/)
TOPT	TOPTRANS (https://www.toptrans.cz/)
GEIS	Geis (https://www.geis-group.cz/cs)
ZAS	Zásilkovna (https://www.zasilkovna.cz/)
DPD	DPD (https://www.dpd.com/cz/cs/)

IND	individuální doprava
OSB	osobní odběr
PDF	vlastní doprava s přepravním štítkem

Všechny položky typu „date“ nebo „dateTime“ mohou být nastavené nebo nenastavené. Pokud je položka nenastavená, znamená to, že k události ještě nedošlo. Nenastavenou položku lze identifikovat tak, že je hodnota < 1.1.2000 00:00:00.

Položky objednávky („orderItems“) jsou načteny a uvedeny v datové struktuře typicky (pokud ne zvolil ukladatel jinak) až po 18. hodině toho dne, ve kterém došlo k jejich fyzickému vyskladnění. Do té doby je indikováno 0 položek objednávky.

Informace o dopravě (všechny položky „transport“) je možné získat až ve chvíli, kdy dopravce převezme balíky, palety, atd. Do doby převzetí je indikováno 0 položek dopravy a veškeré další informace o dopravě nejsou nastavené.

Při individuální dopravě nebo osobním odběru je indikováno doručení ve chvíli, kdy dojde k převzetí zboží.

InventoryItemType

Struktura obsahující informace o položce inventury.

Jméno	Typ	Význam	Platné hodnoty
item	ItemType	Struktura obsahující informace sdílené pro položky objednávky a položky inventury	
count	float	Aktuální fyzické množství	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD, platnost je na 2 desetinná místa
requestedCount	float	Aktuální evidenční množství	
group	string	Skupina	Maximální délka je 60 znaků
eans	EanType[]	Seznam definic čárového kódu (balení) zboží	≥ 0 položek
movement	date	Datum posledního pohybu (změny aktuálního počtu kusů)	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace date formátu definovaného XSD

CarrierItemType

Struktura obsahující informace o dopravci.

Jméno	Typ	Význam	Platné hodnoty
id	string	ID dopravce	Viz seznam dopravců uvedený v definici struktury IssueOrderType
name	string	Jméno dopravce (lidsky čitelné jméno, nebo jméno dopravce dle interního pojmenování Servant)	

ImportRequestResultType

Struktura obsahující informace o výsledku operace vložení importu ke zpracování. Pokud je vložení úspěšné, vlastní výsledek importu obsahuje struktura [ImportResultType](#).

Jméno	Typ	Význam	Platné hodnoty
transactionId	string	ID transakce, pod kterým lze následně ověřit výsledek importu	SHA256
success	bool	Nastaveno na <code>true</code> , pokud bylo vložení importu ke zpracování úspěšné	<code>true</code> nebo <code>false</code>
resultCode	int	Návratový kód operace	0 = Success 1 = PrerequisiteCheckError 2 = ServerAccessError 3 = ImportRequestError 4 = PostrequestOperationError
result	string	Textová reprezentace návratového kódu operace	
infoText	string	Doplňující informace (typicky nastavené, pokud dojde k chybě vložení importu)	

Význam návratového kódu operace:

Success	import úspěšně zadán ke zpracování
PrerequisiteCheckError	chyba při kontrole vstupních dat (webový server)
ServerAccessError	chyba přístupu k serveru
ImportRequestError	chyba zadání importu (Servant server)
PostrequestOperationError	chyba operací vykonávaných po úspěšném zadání importu (interní chyba webového serveru)

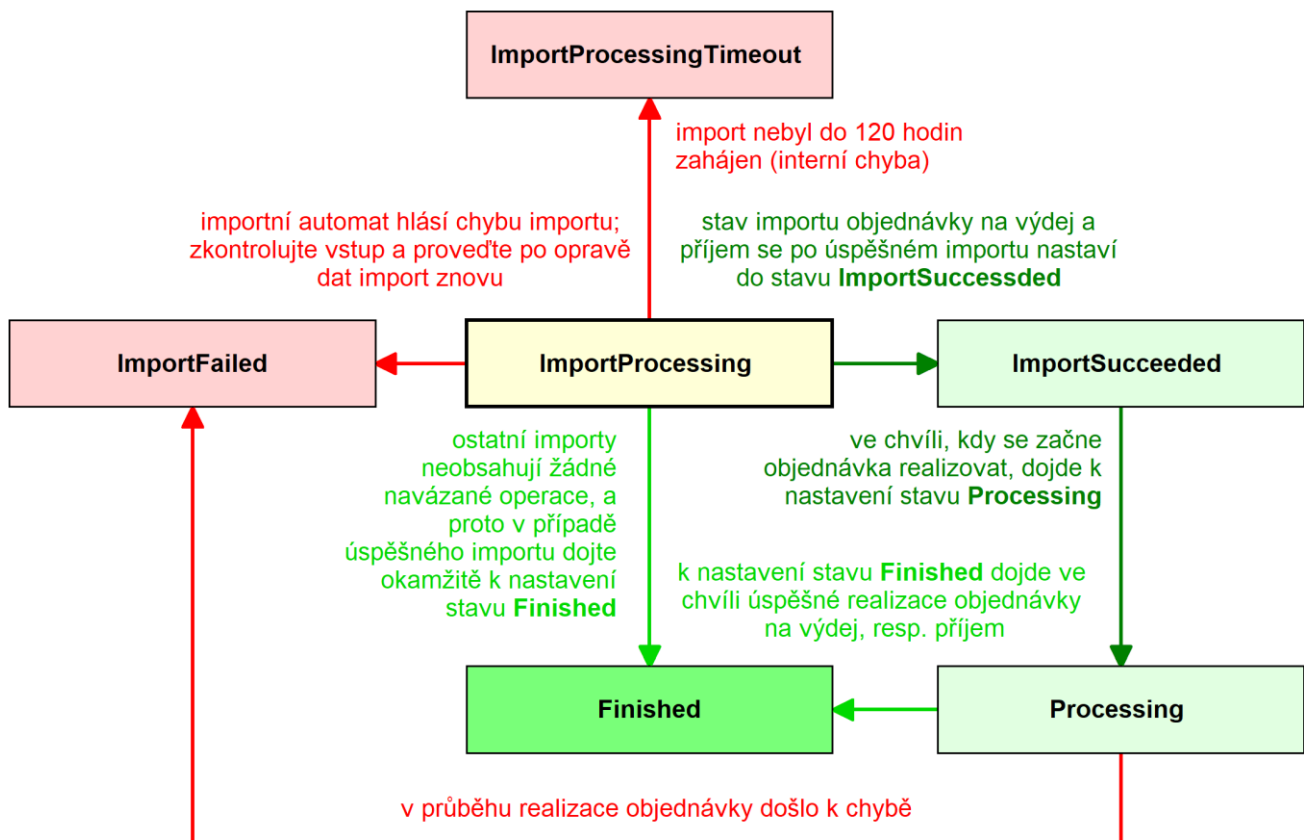
ImportResultType

Struktura obsahující informace o výsledku importu.

Jméno	Typ	Význam	Platné hodnoty
transactionId	string	ID transakce	SHA256
closed	bool	Nastaveno na <code>true</code> , pokud byl dokončen import i případné navázané operace	true nebo false
error	bool	Nastaveno na <code>true</code> , pokud v průběhu importu došlo k chybě	
state	string	Aktuální stav importu nebo případných navázaných operací	Platné hodnoty viz definice níže ...
infoTexts	InfoTextType[]	Doplňující informace k danému stavu	null nebo ≥ 0 položek
encodedContentPDF	string	PDF report importu	null nebo string obsahující PDF data zakódovaná dle Base64 standardu

Aktuální stav může nabývat jedné z následujících hodnot:

ImportProcessing	Import právě probíhá ...
ImportSucceeded	Import skončil úspěšně; očekávají se další navázané operace
Processing	Probíhají další operace (vlastní realizace objednávky) ...
Finished	Import i případné další navázané operace byly úspěšně dokončeny
ImportFailed	Import skončil s chybou
ImportProcessingTimeout	Došlo k timeoutu importu ze strany Servant serveru (interní chyba)



InfoTextType

Struktura obsahující rozšiřující textové informace, které mohou být doplněny o JSON data zakódovaná dle Base64 standardu.

Jméno	Typ	Význam	Platné hodnoty
text	string	Textová informace	
data	string	Data	null nebo string obsahující JSON data zakódovaná dle Base64 standardu

ImportBankType

Struktura obsahující informace o bankovním spojení.

Jméno	Typ	Význam	Platné hodnoty
name	string	Jméno banky	Maximální délka je 64 znaků
code	string	Kód banky	Právě 4 ASCII číslice (např. 0100)
accountNumber	string	Číslo účtu	Maximální délka je 24 znaků
specificSymbol	string	Specifický symbol	Maximální délka jsou 4 znaky

ImportPartnerDefinitionType

Struktura obsahující informace o partnerovi.

Jméno	Typ	Význam	Platné hodnoty
type	string	Typ partnera	
partner	ImportPartnerType	Definice partnera	
legalForm	string	Právní forma partnera	Maximální délka je 48 znaků
payingVAT	bool	Plátce DPH	true nebo false
website	string	Webové stránky	Maximální délka je 64 znaků
bank	ImportBankType	Bankovní spojení na partnera	
expirationDays	int	Neposílat zboží, jehož expirace končí za kratší počet dnů než zde uvedený	Kladná celočíselná hodnota

Typ partnera může nabývat jednu z následujících hodnot:

Carrier pro dopravce
Supplier pro dodavatele
Customer pro odběratele

ImportItemIdType

Struktura obsahující informace definující zboží. Společně se strukturou [ImportItemIdDetailsType](#) tvoří kompletní identifikátor zboží.

Jméno	Typ	Význam	Platné hodnoty
code	string	Jedinečný kód sortimentu. Pokud není vyplněn, přidělíme zboží vlastní kód	Povinné pole, které je zároveň jedinečné v rámci ukladatele. Maximální délka je 60 znaků
attributes	string[]	Atributy sortimentu. Velikost, barva, rozměr apod. Pokud je tento údaj vyplněný, je povinný, a musí se používat!	Pole o velikosti 0 až 3 prvků, maximální délka prvního prvku 60 znaků, maximální délku druhého a třetího prvku je 20 znaků

ImportProductItemDetailsType

Struktura obsahující podrobnosti o zboží.

Jméno	Typ	Význam	Platné hodnoty
name	string	Název zboží	Povinné pole. Maximální délka je 255 znaků
shortName	string	Krátký název zboží	Maximální délka je 32 znaků
description	string	Popis zboží	Maximální délka je 255 znaků
unit	string	Množstevní jednotka (MJ)	Povinné pole. Maximální délka je 8 znaků
type	string	Typ sortimentu	Maximální délka je 128 znaků
group	string	Skupina sortimentu	Maximální délka je 64 znaků
note	string	Poznámka ke zboží	Maximální délka je 255 znaků
notePrint	string	Poznámka, která se tiskne na některých štítcích a tiskových sestavách	Maximální délka je 64 znaků
fifo	bool	Má se pro tento sortiment používat FIFO?	true nebo false
supplierCode	string	Kód dodavatele nebo výrobce zboží	Maximální délka je 64 znaků

ImportProductDimensionType

Struktura obsahující fyzické rozměry zboží.

Jméno	Typ	Význam	Platné hodnoty
x	float	Rozměr X v metrech	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD
y	float	Rozměr Y v metrech	
z	float	Rozměr Z v metrech	

ImportProductEanType

Struktura obsahující definici čárového kódu (balení) zboží.

Jméno	Typ	Význam	Platné hodnoty
ean	string	EAN	Povinné pole, které je zároveň jedinečné v rámci ukladatele. Maximální délka je 64 znaků
count	float	Počet MJ k EAN	Povinná hodnota (hodnoty ≤ 0 jsou neplatné).
eanCount	float	Počet MJ v balení, které patří k tomuto EAN	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD
countToPlacement	float	MJ balení do umístění na jednu pozici (0,8 x 1,2 x 1,6 m)	
expandability	float	Na kolik částí lze balení rozdělit	
weight	float	Hmotnost v kg	
dimension	ImportProductDimensionType	Rozměry	
velocity	float	Objem v m ³	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD
primary	bool	Výchozí EAN pro výpočty a tiskové sestavy	true nebo false

Rozměr zboží musí být zadán buď formou fyzických rozměrů (*dimension*) nebo objemu (*velocity*). V případě nastavení formou fyzických rozměrů je možné nastavit *velocity* na 0. V případě nastavení formou objemu je možné nenastavovat *dimension*, případně všechny jeho rozměry nastavit na 0.

Smazat EAN nelze (pouze zásahem na naší straně). EAN lze tedy jen přidávat.

System automaticky kontroluje unikátnost EAN kódu v rámci již existujícího sortimentu a přiřazených EAN kódů. V rámci kontroly může dojít ke dvěma situacím:

1. EAN kód není dosud nikde evidován = EAN je přiřazen sortimentu
2. EAN kód již je někde evidován (přiřazen nějakému již zadanému sortimentu) = provede se kontrola aktuálního množství sortimentu, ke kterému je EAN přiřazen
 - a. V případě 0 (nulového množství) dojde k odstranění EAN kódu ze sortimentu s nulovým množstvím a přiřazení EAN kódu k právě zadávanému sortimentu
 - b. V případě nenulového množství dojde k chybě

ImportItemIdDetailsType

Struktura obsahující informace upřesňující zboží. Společně se strukturou [ImportItemType](#) tvoří kompletní identifikátor zboží.

Jméno	Typ	Význam	Platné hodnoty
expiration	date	Expirace	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace date formátu definovaného XSD
batch	string	Šarže	Maximální délka je 60 znaků

Pokud nechcete expiraci uvádět, je možné nastavit libovolný datum < 1.1.2000

Pokud nebude některé z polí `expiration` nebo/a `batch` nastaveno, vybere konkrétní zboží náš systém. Pokud se uvede, systém vybere podle uvedené hodnoty.

ImportLabelType

Struktura obsahující definici štítku.

Jméno	Typ	Význam	Platné hodnoty
type	int	Lepit štítek z číselníku štítku ukladatele	Celé číslo mezi hodnotami 10 a 20 včetně
specification	string	Jazyková mutace nebo jiné upřesnění štítku	Maximální délka jsou 3 znaky

ImportActivityType

Struktura obsahující definici VAS prováděných nad zbožím.

Jméno	Typ	Význam	Platné hodnoty
checkExpiration	bool	Nastaveno na <code>true</code> , pokud se má u zboží kontrolovat expirace	<code>true</code> , <code>false</code> nebo <code>null</code>
checkBatch	bool	Nastaveno na <code>true</code> , pokud se má u zboží kontrolovat šarže	
checkSerialNumbers	bool	Nastaveno na <code>true</code> , pokud se má u zboží sledovat výrobní čísla	
printLabel	bool	Nastaveno na <code>true</code> , pokud se má na zboží se lepit štítek	
labels	ImportLabelType[]	Definice štítků	<code>null</code> nebo $10 \geq \text{počet} \geq 0$
services	string[]	Provést služby dle číselníku VAS	<code>null</code> nebo $10 \geq \text{počet} \geq 0$, maximální délka jedné služby jsou 3 číslice v rozsahu 021-999
documentsType	string	K sortimentu přiložit dokumenty dle číselníků	Maximální délka je 64 znaků

Pokud chceme k nějaké službě VAS doplnit data, můžeme tak učinit doplněním dvojtečky, která je následovaná vlastními daty. Příklad: 96 : BCX123456OK

Seznam možných VAS služeb a očekávaných dat naleznete v [tomto](#) dokumentu.

ImportProductActivityType

Struktura obsahující definici VAS prováděných nad zbožím nebo celou objednávkou. Tato struktura je proti struktuře [ImportActivityType](#) použita výhradně pro definici VAS v rámci funkce pro import zboží.

Jméno	Typ	Význam	Platné hodnoty
idDetails	ImportItemIdDetailsType	Definice zboží	
activity	ImportActivityType	VAS prováděná nad zbožím	null nebo ≥ 0 položek

ImportProductType

TOP struktura obsahující definici právě jednoho zboží, jeho EAN a VAS, které se mají nad zbožím provádět.

Jméno	Typ	Význam	Platné hodnoty
id	ImportItemIdType	Definice zboží	
details	ImportProductItemDetailsType	Podrobnosti o zboží	null nebo vyplněná struktura s podrobnostmi o zboží
eans	ImportProductEanType[]	Definice EAN zboží	≥ 1 položek (minimálně jeden EAN k zboží je povinný)
activities	ImportProductActivityType[]	Definice VAS prováděných nad zbožím	null nebo ≥ 0 položek
files	ImportFileType[]	Obrázky zboží	null nebo ≥ 0 položek (jeden hlavní a až 2 další obrázky)

Definice zboží je povinná. Ostatní atributy je možné vyplnit nebo ponechat prázdné. Budou importovány jen ty části, které budou vyplněné. Při nevyplnění ani jednoho atributu (podrobnosti o zboží, definice EAN zboží nebo definice VAS prováděných nad zbožím), import selže.

ImportPartnerType

Struktura obsahující definici partnera k objednávce na příjem nebo výdej. Pokud se jedná o objednávku na příjem, je partnerem dodavatel. Pokud se jedná o objednávku na výdej, je partnerem odběratel.

Jméno	Typ	Význam	Platné hodnoty
code	string	Kód partnera	Povinný parametr. Nesmí být null, ani prázdný řetězec. Maximální délka je 64 znaků
title	string	Název partnera	
name	string	Jméno (zastupující osoby)	Maximální délka je 32 znaků
surname	string	Příjmení (zastupující osoby)	Maximální délka je 32 znaků
ic_rc	string	IČO, rodné číslo nebo jiná identifikace	Maximální délka je 10 znaků
dic	string	DIČ partnera	Maximální délka je 12 znaků
street	string	Ulice, ve které sídlí partner	Maximální délka je 64 znaků
city	string	Město / Obec, ve které sídlí partner	Maximální délka je 32 znaků
zipCode	string	PSČ bez mezery	Maximální délka je 5 znaků
countryCode	string	Kód státu	Kód státu dle ISO 3166-1
phone	string	Telefon	V případě dobírky je uvedení jednoho z parametrů povinné (telefon nebo mobilní telefon). Maximální délka je 16 znaků
mobile	string	Mobilní telefon	
email	string	Email	Maximální délka je 64 znaků
note	string	Poznámka	Maximální délka je 255 znaků

Lze zvolit mezi uloženým partnerem a manuálním zadáním informací. V případě, že chcete použít uloženého partnera, vyplňte pouze parametr `code` (ostatní nechte prázdné, tj. použijte prázdný řetězec). V případě, že chcete zadat informace manuálně, vyplňte všechna relevantní pole, minimálně však pole `street` (pole `code` se v takovém případě ignoruje).

ImportDetailsType

Struktura obsahující definici podrobností k objednávce na příjem nebo výdej.

Jméno	Typ	Význam	Platné hodnoty
order	string	Objednávka	Povinný parametr. Hodnota (objednávka) musí být unikátní. Maximální délka je 255 znaků
invoice	string	Faktura	Maximální délka je 32 znaků
note	string	Poznámka	Maximální délka je 255 znaků
notePrint	string	Tisknutelná poznámka	Maximální délka je 64 znaků
scannerCheck	bool	Nastaveno na <code>true</code> , pokud se má přijímané / vydávané zboží kontrolovat čtečkou	<code>true</code> nebo <code>false</code>
carrier	string	Dopravce	Jedna z hodnot viz Dopravce . Lze zvolit pouze dopravce viz Načtení aktuálního seznamu dopravců
carrierName	string	Jméno dopravce	Maximální délka je 64 znaků

ImportDeliveryType

Struktura obsahující podrobnosti k dopravě u objednávek na výdej.

Jméno	Typ	Význam	Platné hodnoty
dispatch	dateTime	Datum expedice	Povinný parametr. Musí být stejné nebo vyšší než datum zaslání dat.
delivery	dateTime	Datum dodání u zákazníka	Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace dateTime formátu definovaného XSD
leaveCustomsArea	bool	Nastaveno na <code>true</code> , pokud zboží opouští celní prostor	<code>true</code> nebo <code>false</code>
currency	string	Měna použitá pro <code>insurancePrice</code> a <code>price</code>	V případě dobírky se jedná o povinný parametr. Formát dle ISO 4217
insurancePrice	float	Cena balíku (pro potřeby pojištění)	Není potřeba nastavovat, tj. je možné ponechat na hodnotě 0. Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD
paymantMethod	string	Platební metoda	V případě dobírky se jedná o povinný parametr. Platné hodnoty viz definice níže ...
price	float	Cena k zaplacení požadovaná při předání dle platební metody	V případě dobírky se jedná o povinný parametr. Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD
variableSymbol	string	Variabilní symbol	V případě dobírky se jedná o povinný parametr. Maximální délka je 16 znaků

Možné platební metody jsou:

DOB Platba dobírkou při převzetí

V případě zvolení DOB jsou pole `price` a `variableSymbol` povinné.

ImportItemType

Struktura obsahující definici zboží pro objednávky na příjem nebo výdej.

Jméno	Typ	Význam	Platné hodnoty
id	ImportItemIdType	Definice zboží	
idDetails	ImportItemIdDetailsType	Informace upřesňující zboží	
activities	ImportActivityType[]	VAS prováděné nad zbožím	null nebo ≥ 0 položek
count	float	Počet kusů (MJ) zboží	Hodnota > 0. Při implementaci vlastního serializéru / deserializéru je potřeba dodržet specifikace decimal formátu definovaného XSD, platnost je na 2 desetinná místa

ImportReceiptType

TOP struktura obsahující informace o jedné objednávce na příjem.

Jméno	Typ	Význam	Platné hodnoty
partner	ImportPartnerType	Definice ukladatele	
details	ImportDetailsType	Podrobnosti k objednávce	
items	ImportItemType[]	Zboží objednávky	Hodnota > 0
activities	ImportActivityType[]	VAS prováděné nad celou objednávkou	null nebo ≥ 0 položek

ImportIssueType

TOP struktura obsahující informace o jedné objednávce na výdej.

Jméno	Typ	Význam	Platné hodnoty
partner	ImportPartnerType	Definice odběratele	
details	ImportDetailsType	Podrobnosti k objednávce	
delivery	ImportDeliveryType	Podrobnosti k dopravě	
items	ImportItemType[]	Zboží objednávky	Hodnota > 0
activities	ImportActivityType[]	VAS prováděné nad celou objednávkou	null nebo ≥ 0 položek

Při použití dopravce ZAS se očekává:

- Do parametru `street` ve struktuře [ImportPartnerType](#) vložit id z API Zásilkovny
- Do parametru `city` ve struktuře [ImportPartnerType](#) vložit jméno e-shopu (bez www)

Zdroje pro id z API Zásilkovny jsou na následujících webových stránkách

<http://www.zasilkovna.cz/api/v4/41494564a70d6de6/branch.xml>

V případě, že chcete v rámci dopravce CP využít služeb [Balíkovna](#), je potřeba:

- Do parametru `street` ve struktuře [ImportPartnerType](#) vložit „BALÍKOVNA“
- Do parametru `city` ve struktuře [ImportPartnerType](#) vložit NAZEV z API České pošty
- Do parametru `zipCode` ve struktuře [ImportPartnerType](#) vložit PSC z API České pošty

Zdroje pro NAZEV a PSC z API České pošty jsou na následujících webových stránkách

<http://napostu.cpost.cz/vystupy/balikovny.xml>

V případě, že chcete v rámci dopravce GLS využít služeb [Parcel Shop](#), je potřeba:

- Do parametru `street` ve struktuře [ImportPartnerType](#) vložit Parcel Shop ID z API GLS

Zdroje pro Parcel Shop ID z API GLS jsou na následujících webových stránkách

<https://datarequester.gls-hungary.com/glsconnect/getDropoffPoints.php?ctrancode=CZ>

ImportFileType

TOP struktura s definicí jednoho souboru.

Jméno	Typ	Význam	Platné hodnoty
type	string	Typ souboru	Platné hodnoty viz definice níže ...
name	string	Jméno souboru včetně přípony	
encodedContent	string	Obsah souboru	Kódovaný dle Base64 standardu RFC 4648

Tabulka platných typů souborů a jejich význam:

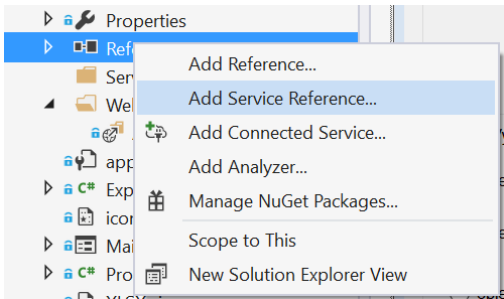
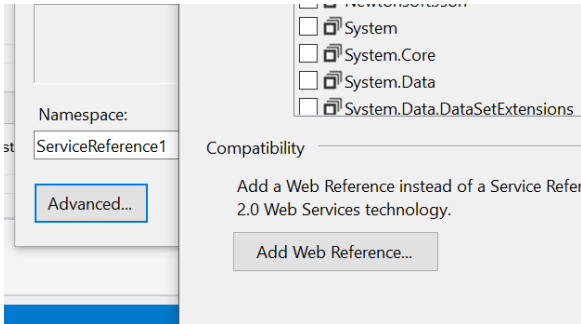
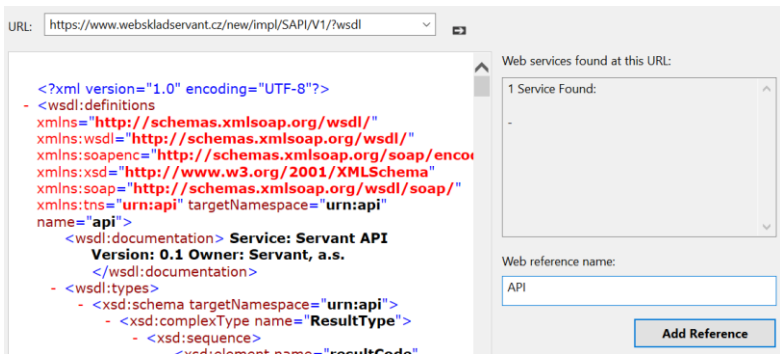
Typ	Použití	Očekávaný formát souboru
Expedition	Pojí se s objednávkou na výdej. Může obsahovat dokumenty příkládané ke zboží (faktura, apod.)	PDF
Transport_PDF PDF	Vlastní přepravní štítek	PDF
Sortiment_MainImage	Hlavní obrázek sortimentu	JPG: maximální rozlišení 1440 x 1080 px doporučené rozlišení 480 x 360 px (velikost kolem 300 kB, tj. u JPG komprese 50%)
Sortiment_Image	Obrázek sortimentu	JPG: maximální rozlišení 1440 x 1080 px doporučené rozlišení 160 x 120 px (velikost kolem 100 kB, tj. u JPG komprese 50%)

Použití

C#

Přístup k webovým službám popsaných WSDL protokolem SOAP pomocí Microsoft Visual Studio IDE a aplikací psaných v C# je velice snadný. Následující popis je pouze rychlým návodem, jak v Microsoft Visual Studio IDE 2015 k těmto službám přistoupit. Pro více informací je možné čerpat přímo z dokumentace na stránkách Microsoftu:

- <https://docs.microsoft.com/en-us/dotnet/api/system.web.services.description>
- <https://docs.microsoft.com/cs-cz/visualstudio/data-tools/windows-communication-foundation-services-and-wcf-data-services-in-visual-studio>

- 
 - V projektu aplikace otevřete kontextovou nabídku nad položkou „Reference“ (klikněte pravým tlačítkem)
 - Z nabídky vyberte „Add Service Reference“ (nebo totožnou volbu s lokalizovaným textem)
- 
 - V levém dolním rohu dialogu, co se zobrazí, vyberte „Advanced...“
 - Otevře se další dialog, v jehož opět levém dolním rohu klikněte na „Add Web Reference...“
- 
 - Do pole URL zadejte URL webové služby a výběr potvrďte tlačítkem enter
 - Pro přístup k webové službě z kódu aplikace zvolte její název (na obrázku zvoleno „API“)
 - Přidejte službu stiskem tlačítka „Add Reference“
- `ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls12;`

```
API.CanAccessParams CanAccessParams = new API.CanAccessParams();
CanAccessParams.credentials = new API.CredentialsType();
CanAccessParams.credentials.email = "schematic@chematic.cz";
CanAccessParams.credentials.password = "password";
try
{
    API.CanAccessResult CanAccessResult = _API.CanAccess(CanAccessParams);
    _bConnected = CanAccessResult.result.resultCode == 0;
}
catch
{
    _bConnected = false;
}
```

 - Před prvním voláním webové služby, v kódu aplikace, nastavte výchozí bezpečnostní protokol na TLS1.2
 - Příklad volání služby [CanAccess](#) je zobrazen na obrázku

PHP

Přístup k webovým službám popsaných WSDL protokolem SOAP pomocí aplikací psaných v PHP je velice snadný. Následující popis je pouze rychlým návodem, jak v PHP k těmto službám přistoupit. Pro více informací je možné čerpat přímo z dokumentace:

- <https://www.php.net/manual/en/book.soap.php>
- <https://www.php.net/manual/en/class.soapclient.php>
- <https://www.php.net/manual/en/soapclient.soapclient.php>

Vzorová implementace volání služby [CanAccess](#):

```
<?php

ini_set('display_errors', 1);
ini_set('display_startup_errors', 1);
error_reporting(E_ALL);

$url = 'https://www.webskladservant.cz/new/impl/SAPI/V1/?wsdl';
$soap = new SoapClient($url, Array(
    'soap_version' => SOAP_1_2,
    'stream_context' => stream_context_create(Array(
        'ssl' => Array(
            'crypto_method' => STREAM_CRYPTO_METHOD_TLSv1_2_CLIENT
        ),
    )),
));
print_r($soap->__getFunctions());

$scanAccessParams = Array(
    'credentials' => Array(
        'email' => 'schematic@schematic.cz',
        'password' => 'password',
        'switchForUser' => false,
        'switchForUserId' => -1
    )
);
$scanAccessResponse = $soap->CanAccess($scanAccessParams);
print_r($scanAccessResponse);

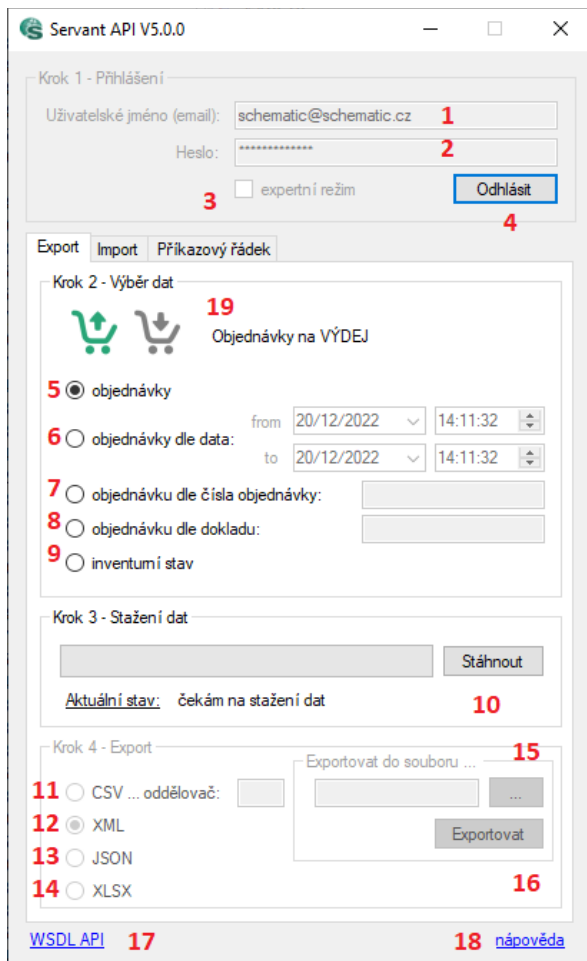
?>
```

SAPI Aplikace

SAPI je jméno desktopové aplikace určené pro Microsoft Windows. Aplikace slouží zákazníkům společnosti Servant a.s., kteří mají sjednaný přístup na Infoportál, k ověření přístupu, ke stažení a exportu objednávek nebo inventurní stavu a k importu sortimentu a objednávek na příjem nebo výdej zboží. SAPI přistupuje k datům přes webové služby definované rozhraním WSDL na adrese <https://www.webskladservant.cz/new/impl/SAPI/V5/?wsdl>.

Export z Infoportálu

Aplikace umožňuje stažená data exportovat v jednom z následujících formátů: CSV, XML, JSON a XLSX.



1. E-mail registrovaný do Infoportálu
2. Hesla přidružené k e-mailu
3. Zapne expertní režim přístupu (pouze pro role Admin a Operator)
4. Akce Přihlásit/Odhlásit
5. Stáhne posledních 1000 objednávek daného ukladatele (seřazeno dle požadované expedice)
6. Stáhne všechny objednávky daného ukladatele, které odpovídají vybranému časovému intervalu (maximálně 1000)
7. Stáhne konkrétní objednávku definovanou číslem objednávky (více objednávek lze oddělit čárkou)
8. Stáhne konkrétní objednávku definovanou dokladem (více objednávek lze oddělit čárkou)
9. Stáhne položky inventurního stavu daného ukladatele (seřazeno dle posledního pohybu)
10. Provede stažení dat/Zruší stažení dat (umožní výběr jiných dat)
11. ~~Exportuje data ve formátu CSV s nastaveným oddělovačem (více viz [Detaily CSV a XLSX exportu](#)) - ZRUŠENO~~
12. Exportuje data ve formátu XML (více viz [Detaily XML exportu](#))
13. Exportuje data ve formátu JSON (více viz [Detaily JSON exportu](#))
14. ~~Exportuje data ve formátu XLSX (více viz [Detaily CSV a XLSX exportu](#)) - ZRUŠENO~~
15. Vybere umístění pro export
16. Provede export (průběh je signalizován procenty)
17. Otevře WSDL rozhraní používané aplikací
18. Otevře tuto nápovědu
19. Přepínač mezi objednávkami na PŘÍJEM a objednávkami na VÝDEJ

Detaily CSV a XLSX exportu

Pokud zvolíte řádkový formát CSV nebo XLSX, budou řádky duplikované pro každou položku objednávky a každou položku dopravy. Tedy jedna objednávka bude na jednom až $X * Y$ řádcích, kde X je počet položek objednávky a Y je počet položek dopravy.

Jména sloupců objednávek jsou:

```
"order", "document", "orderActivity_id", "orderActivity_name", "orderActivity_value",
"orderActivity_count", "orderActivity_servantNumber", "orderActivity_transportNumber", "item_code",
"orderItem_name", "orderItem_attribute1", "orderItem_attribute2", "orderItem_attribute3",
"orderItem_expiration", "orderItem_batch", "orderItem_requiredCount", "orderItem_count",
```

```
"orderItemActivity_id", "orderItemActivity_name", "orderItemActivity_value",  
"orderItemActivity_count", "orderItemActivity_servantNumber", "orderItemActivity_transportNumber",  
"transportItem_unit", "transportItem_servantNumber", "transportItem_transportNumber",  
"transportItem_statusUrl", "transportItem_packageType", "transportItem_weight",  
"recipientAddress", "note", "state", "warning", "requiredDispatch", "dispatch", "loaded",  
"entered", "packaged", "transportCarrier", "transportEntered", "transportPicked",  
"transportDelivered"
```

Jména sloupců položek inventury jsou:

```
"code", "name", "attribute1", "attribute2", "attribute3", "expiration", "batch", "count",  
"movement"
```

Všechny položky typu „datum“ nebo „datum a čas“ mohou být nastavené nebo nenastavené. Pokud je položka nenastavená, znamená to, že k události ještě nedošlo. Nenastavenou položku lze identifikovat tak, že je daná hodnota prázdná.

U CSV formátu může nastavený oddělovač kolidovat s vlastními daty. Pokud k tomu dojde, bude vzor (znak nebo textový řetězec) použitý pro oddělení jednotlivých sloupců automaticky odstraněn ze všech dat a to bez jakéhokoli upozornění.

U CSV formátu jsou:

- desetinná čísla ukládána ve formátu 0.00
- položky datumu ukládány ve formátu yyyy-MM-dd
- položky datumu a času ukládány ve formátu yyyy-MM-dd hh:mm:ss

Kódování u CSV formátu je UTF-8.

Detaily JSON exportu

JSON formát odpovídá přesně (strukturou, typovostí i pojmenováním jednotlivých datových položek) definici [ReceiptOrderType\[\]](#) pro objednávky na PŘÍJEM, [IssueOrderType\[\]](#) pro objednávky na VÝDEJ a [InventoryItemType\[\]](#) pro položky inventury.

Všechny položky typu „datum“ nebo „datum a čas“ mohou být nastavené nebo nenastavené. Pokud je položka nenastavená, znamená to, že k události ještě nedošlo. Nenastavenou položku lze identifikovat tak, že je hodnota < 1.1.2000 00:00:00.

Kódování je UTF-8.

Detaily XML exportu

Pokud zvolíte datový formát XML, jsou data uložena dle struktury zobrazené na následujících stránkách.

Pro objednávky jsou XML elementy:

- <order> v rámci nadřazeného elementu <root>,
- <orderItem> v rámci nadřazeného elementu <orderItems>,
- <attribute> v rámci nadřazeného elementu <attributes>,
- <activity> v rámci nadřazeného elementu <activities>,
- <transportItem> v rámci nadřazeného elementu <transportItems>

... v exportu X krát podle počtu objednávek, počtu položek objednávky, počtu atributů položky objednávky (počet atributů je pevně nastaven na 3), počtu aktivit (označováno také jako VAS)

objednávky, resp. položky objednávky a počtu položek dopravy objednávky (pouze u objednávek na VÝDEJ).

Pro položky inventory jsou XML elementy:

- `<item>` v rámci nadřazeného elementu `<root>`,
- `<attribute>` v rámci nadřazeného elementu `<attributes>`

... v exportu X krát podle počtu položek inventory a počtu atributů položky inventory (počet atributů je pevně nastaven na 3).

U XML formátu jsou:

- desetinná čísla ukládána ve formátu `0.00`
- položky datumu ukládány ve formátu `yyyy-MM-dd`
- položky datumu a času ukládány ve formátu `yyyy-MM-dd hh:mm:ss`

Kódování je UTF-8.

Struktura exportovaného souboru viz následující příklady:

Položky inventory

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <item>
    <code></code>
    <name></name>
    <attributes>
      <attribute></attribute>
      <attribute></attribute>
      <attribute></attribute>
    </attributes>
    <expiration></expiration>
    <batch></batch>
    <count></count>
    <requestedCount></requestedCount>
    <unit></unit>
    <group></group>
    <eans>
      <ean>
        <ean></ean>
        <count></count>
        <weight></weight>
        <dimension>
          <x></x>
          <y></y>
          <z></z>
        </dimension>
        <velocity></velocity>
        <primary></primary>
      </ean>
    </eans>
    <movement></movement>
  </item>
</root>
```

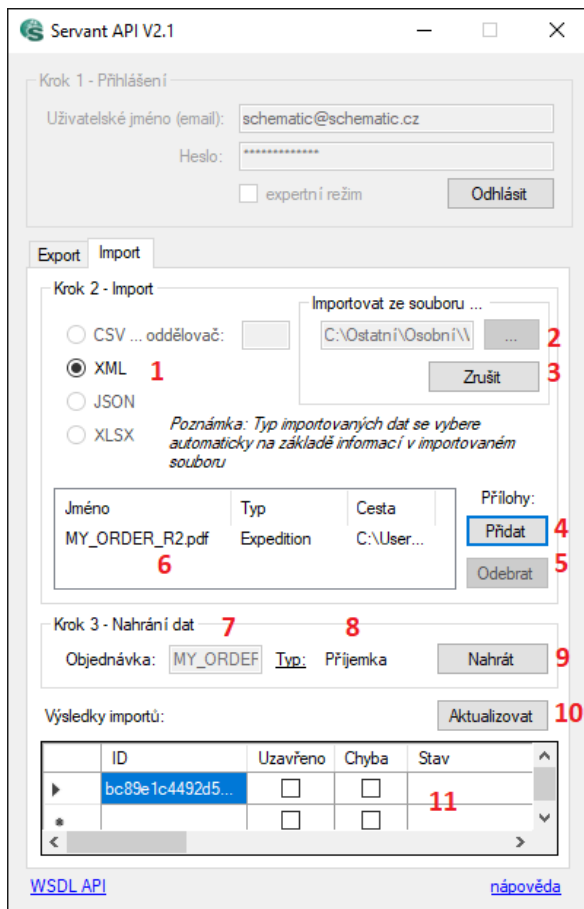
Objednávky na PŘÍJEM

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <order>
    <order></order>
    <document></document>
    <invoice></invoice>
    <activities>
      <activity>
        <id></id>
        <name></name>
        <value></value>
        <count></count>
        <servantNumber></servantNumber>
        <transportNumber></transportNumber>
      </activity>
    </activities>
    <orderItems>
      <orderItem>
        <code></code>
        <name></name>
        <attributtes>
          <attributte></attributte>
        </attributtes>
        <expiration></expiration>
        <batch></batch>
        <requiredCount></requiredCount>
        <count></count>
        <unit></unit>
        <activities>
          <activity>
            <id></id>
            <name></name>
            <value></value>
            <count></count>
            <servantNumber></servantNumber>
            <transportNumber></transportNumber>
          </activity>
        </activities>
      </orderItem>
    </orderItems>
    <partnerName></partnerName>
    <partnerCode></partnerCode>
    <loaded></loaded>
    <received></received>
    <checkState></checkState>
    <finished></finished>
    <error></error>
  </order>
</root>
```

Objednávky na VÝDEJ

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <order>
    <order></order>
    <document></document>
    <invoice></invoice>
    <activities>
      <activity>
        <id></id>
        <name></name>
        <value></value>
        <count></count>
        <servantNumber></servantNumber>
        <transportNumber></transportNumber>
      </activity>
    </activities>
    <orderItems>
      <orderItem>
        <code></code>
        <name></name>
        <attributtes>
          <attributte></attributte>
        </attributtes>
        <expiration></expiration>
        <batch></batch>
        <requiredCount></requiredCount>
        <count></count>
        <unit></unit>
        <activities>
          <activity>
            <id></id>
            <name></name>
            <value></value>
            <count></count>
            <servantNumber></servantNumber>
            <transportNumber></transportNumber>
          </activity>
        </activities>
      </orderItem>
    </orderItems>
    <transportItems>
      <transportItem>
        <unit></unit>
        <servantNumber></servantNumber>
        <transportNumber></transportNumber>
        <statusUrl></statusUrl>
        <packageType></packageType>
        <weight></weight>
      </transportItem>
    </transportItems>
    <partnerName></partnerName>
    <partnerCode></partnerCode>
    <recipientAddress></recipientAddress>
    <note></note>
    <state></state>
    <warning></warning>
    <requiredDispatch></requiredDispatch>
    <dispatch></dispatch>
    <loaded></loaded>
    <entered></entered>
    <packaged></packaged>
    <transportCarrier></transportCarrier>
    <transportEntered></transportEntered>
    <transportPicked></transportPicked>
    <transportDelivered></transportDelivered>
  </order>
</root>
```

Import na Infoportál



Krok 1 - Přihlášení

Uživatelské jméno (email): schematic@schematic.cz
 Heslo: *****
 expertní režim Odhlásit

Export Import

Krok 2 - Import

Importovat ze souboru ...
 C:\Ostatní\Osobní\ [Browse] Zrušit

CSV ... oddělovač: []
 XML **1**
 JSON
 XLSX *Poznámka: Typ importovaných dat se vybere automaticky na základě informací v importovaném souboru*

Jméno	Typ	Cesta	Přílohy:
MY_ORDER_R2.pdf 6	Expedition	C:\User...	Přidat 4 Odebrat 5

Krok 3 - Nahrání dat **7** **8**

Objednávka: MY_ORDEF **9** Typ: Příjemka Nahrát

Výsledky importů: Aktualizovat **10**

	ID	Uzavřeno	Chyba	Stav
▶	bc89e1c4492d5...	<input type="checkbox"/>	<input type="checkbox"/>	11
*		<input type="checkbox"/>	<input type="checkbox"/>	

[WSDL API](#) [nápověda](#)

- Formát vstupních dat importu. V současné době je podporovaný pouze **XML** formát. Typ importovaných dat se zvolí automaticky na základě dat obsažených v souboru importu. Viz vzorové XML pro SAPI aplikaci pro [import partnerů](#), [import sortimentu](#), [objednávky na příjem](#) a [objednávky na výdej](#).
- Tlačítko umožňující výběr dat k importu
- Tlačítko pro načtení vybraných dat k importu. Pokud jsou již data načtena, je možné načtení zrušit opětovným stiskem tlačítka. Po zrušení načtení je možné vybrat jiný soubor.
- Přidání souboru odesílaného spolu s importem. V případě, že nejsou načtena žádná data importu, i tak je možné odeslat soubor. V takovém případě je jen potřeba manuálně nastavit objednávku, ke které se soubor přiřadí.
- Odebrání souboru vybraného v seznamu připojených souborů z výčtu souborů odesílaných spolu s importem.
- Seznam připojených souborů
- Zobrazení buď automaticky načtené objednávky nebo umožnění objednávku zadat manuálně
- Typ importu, který může nabývat hodnot: **partneři** (proběhne import partnerů), **sortiment** (proběhne import sortimentu), **příjemka** (proběhne import objednávky na příjem), **výdejka** (proběhne import objednávky na výdej) nebo **soubory** (proběhne pouze upload souborů)
- Provede import na server Servant. Pokud dojde v průběhu k chybě, informace o chybě jsou zobrazené formou dialogu. Pokud vše proběhne správně, je ID transakce importu doplněno za začátek seznamu s výsledky importů.
- Aktualizuje tabulku s výsledky importů. **POZOR:** Maximální počet výsledků importů získaných v rámci aktualizace může být 100. Pokud bude v tabulce více řádků, dojde k aktualizaci pouze prvních 100 řádků.
- Seznam importů, u nichž chceme získat jejich výsledky. Záznamy lze ručně editovat (upravovat, mazat staré a přidávat nové)

Řízení přes příkazovou řádku

Aplikaci je možné spustit bez UI v režimu, ve kterém se veškeré vstupní informace zadají formou argumentu.

Argument	Možné hodnoty a popis	Vyžadováno pro
-modeX	X = <u>UI</u> , <u>Console</u> <u>UI</u> = spuštění aplikace s grafickým rozhraním (výchozí hodnota) <u>Console</u> = spuštění aplikace bez grafického rozhraní	
-actionX	X = <u>CheckCredentials</u> , <u>Export</u> , <u>Import</u> <u>CheckCredentials</u> = kontrola přístupu na Infoportál <u>Export</u> = export informací z Infoportálu do PC <u>Import</u> = import informací z PC na Infoportál	-modeConsole
-emailX	X = přihlašovací jméno (email)	-modeConsole
-passwordX	X = přihlašovací heslo	-modeConsole
-userIdX	X = ID uživatele, za kterého se chci přihlásit	
-ordersTypeX	X = <u>Issue</u> , <u>Receipt</u> <u>Issue</u> = objednávky na VÝDEJ <u>Receipt</u> = objednávky na PŘÍJEM	-modeConsole -actionExport
-exportTypeX	X = <u>AllOrders</u> , <u>OrdersByDate</u> , <u>Order</u> , <u>OrderByDocument</u> , <u>Inventory</u> <u>AllOrders</u> = stažení všech objednávek <u>OrdersByDate</u> = stažení všech objednávek v daném časovém intervalu <u>Order</u> = stažení konkrétní objednávky <u>OrderByDocument</u> = stažení konkrétní objednávky specifikované číslem dokladu <u>Inventory</u> = stažení inventury (aktuálního stavu skladu)	-modeConsole -actionExport
-ordersFromX	X = ISO 8601 datum a čas pro objednávky „od“	-modeConsole -actionExport -exportTypeOrdersByDate
-ordersToX	X = ISO 8601 datum a čas pro objednávky „do“	-modeConsole -actionExport -exportTypeOrdersByDate
-exportOrderX	X = číslo objednávky	-modeConsole -actionExport -exportTypeOrder
-documentX	X = číslo dokladu objednávky	-modeConsole -actionExport -exportTypeOrderByDocument
-exportFormatX	X = <u>CSV</u> , <u>XML</u> , <u>JSON</u> , <u>XLSX</u> Zvolený formát výstupních dat	-modeConsole -actionExport
-csvSeparatorX	X = oddělovač pro CSV formát	-modeConsole -actionExport -exportFormatCSV
-exportPathX	X = celá cesta k souboru, kam se mají data uložit	-modeConsole -actionExport
-importTypeX	X = <u>Import</u> , <u>Status</u> <u>Import</u> = požadavek na import <u>Status</u> = požadavek na získání stavu importu	-modeConsole -actionImport
-importPathX	X = celá cesta k souboru k importu ¹⁾	-modeConsole

Argument	Možné hodnoty a popis	Vyžadováno pro
		-actionImport -importTypeImport
-attPathX	X = celá cesta k příloze importu ¹⁾²⁾	-modeConsole -actionImport -importTypeImport
-importOrderX	X = objednávka ²⁾	-modeConsole -actionImport -importTypeImport
-statusX	X = ID importu nebo seznam ID importů pro zjištění statusu (jednotlivé ID jsou v případě seznamu od sebe odděleny pomocí lomítka)	-modeConsole -actionImport -importTypeStatus

1) Pro `-importTypeImport` je vyžadován buď argument `-importPathX` v případě importu, nebo `-attPathX` v případě, že se nahrává pouze příloha k již zadanému importu. Argumenty `-importPathX` a `-attPathX` mohou být rovněž zadány zároveň a to pokud k zadávanému importu chceme rovnou nahrát i přílohu.

2) Pro `-importTypeImport` je vyžadována objednávka ve chvíli, kdy se nahrává pouze příloha (bez vlastního importu) k již zadanému importu. Tedy je nastaven pouze argument `-attPathX`.

Návratové kódy

Aplikace může skončit s dvěma návratovými kódy:

- 0 = operace skončila úspěšně (pokud má smysl, výstup je v souboru **result.txt***)
- 1 = operace skončila s chybou, která je blíže specifikovaná v **error.txt** souboru

* Některé operace mohou poskytovat datové výstupy, které jsou následně snadno programově zpracovatelné (typicky požadavek na získání stavu importu). V takovém případě se kromě souboru **result.txt** vytvoří také jeden nebo i více souborů **data_file_X_Y.json** a **data_file_X_Y.xml**, kde X je inkrementální číslo pro identifikátor operace Y.

Příklad: Pokud skončí import 8cb75b0c7a88cb131f39d0d8c4cfa6e2137bef2aab27d5d22947d71099f1f886 s chybou, budou při požadavku na získání stavu tohoto importu vytvořeny soubory:

- `data_file_0_8cb75b0c7a88cb131f39d0d8c4cfa6e2137bef2aab27d5d22947d71099f1f886.json`,
- `data_file_0_8cb75b0c7a88cb131f39d0d8c4cfa6e2137bef2aab27d5d22947d71099f1f886.xml`,

obsahující programově zpracovatelné informace o chybě.

Příklady

Zkontroluje bez UI, zdali má uživatel `muj@email.cz` s heslem `mojeheslo` přístup k serverům Servant a.s.; vrátí 0, pokud ano.

```
-modeConsole -actionCheckCredentials -emailmuj@email.cz -passwordmojeheslo
```

Stáhne bez UI kompletní stav inventury uživatele `muj@email.cz` a uloží ho do souboru `C:\inventura.json` ve formátu JSON; pokud operace selže, vrátí 1 a popis chyby je v souboru `error.txt`.

```
-modeConsole -actionExport -emailmuj@email.cz -passwordmojeheslo  
-exportTypeInventory -exportFormatJSON -exportPathC:\inventura.json
```


Nahraje bez UI zboží / objednávku uživatele muj@email.cz ze souboru C:\import.xml; pokud operace selže, vrací 1 a popis chyby je v souboru error.txt; v opačném případě se vrací 0 a ID transakce je v souboru result.txt.

```
-modeConsole -actionImport -emailmuj@email.cz -passwordmojeheslo  
-importTypeImport -importPathC:\import.xml
```

Načte bez UI stav importu 8cb75b0c7a88cb131f39d0d8c4cfa6e2137bef2aab27d5d22947d71099f1f886 a f18ada56243048cf44f804a825396e734b6447776cf354fff6d4853920d89920; pokud operace selže, vrací 1 a popis chyby je v souboru error.txt; v opačném případě se vrací 0 a stavy pro jednotlivý ID transakce jsou v souboru result.txt.

```
-modeConsole -actionImport -emailmuj@email.cz -passwordmojeheslo  
-importTypeStatus -status8cb75b0c7a88cb131f39d0d8c4cfa6e2137bef2aab27d5d22947d71  
099f1f886/f18ada56243048cf44f804a825396e734b6447776cf354fff6d4853920d89920
```

Externí nástroje a aplikace

Webové služby definované pomocí WSDL a protokol SOAP je standardem a tedy je možné pro přístup využít také nástroje třetích stran.

Z těch webových je možné použít například:

- <http://www.soapclient.com/soaptest.html>
- <https://wsdlbrowser.com/>
- <https://extendsclass.com/soap-client-online.html>

Pro přímé volání z konzole (bash) je možné se inspirovat například:

- <https://stackoverflow.com/questions/1222607/how-to-do-a-soap-wsdl-web-services-call-from-the-command-line>

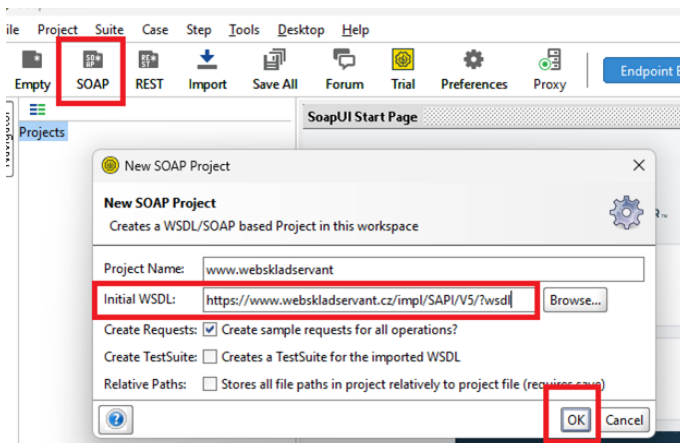
Ověření XML dotazu vůči WSDL předpisu

Při sestavování XML dotazu je potřeba si uvědomit, že se neřídíme pouze pravidly pro validní XML, ale také pravidly platnými pro SOAP a popis rozhraní webové služby WSDL. Tedy například i změna pořadí atributů v XML, absence nějakého atributu nebo porušení typovosti proti WSDL předpisu, to vše může vést k neplatnosti XML dotazu. Různé implementace SOAP/WSDL/XML pak dovolují určité deviace a i nevalidní dotaz si nechají líbit, nicméně nejedná se o správný přístup a vždy bychom se měli snažit dosáhnout validního dotazu.

Ověření je možné provést mnoho nástroji. Zde popsany příklad využívá open-source nástroje [SoupUI](#). Postup je následující:

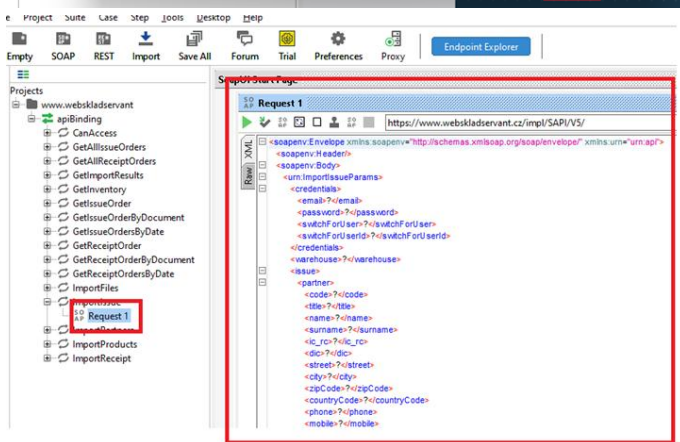
1)

Načíst WSDL API



2)

Otevřít funkci, kterou chceme kontrolovat (nástroj po otevření přímo ukáže předpis, jak má dotaz vypadat)



3)

Nakopírovat testovaný dotaz (ideálně jen obsah mezi elementy <urn:*Params> a </urn:*Params>, pravým tlačítkem vyvolat kontextovou nabídku a dát validovat

