



dpd

Spis treści

| | | |
|-------|--|----|
| 1 | Wstęp | 4 |
| 1.1 | Cel dokumentu | 4 |
| 1.2 | Słownik pojęć | 4 |
| 1.3 | Adresy usługi | 5 |
| 1.4 | Standard bezpieczeństwa oraz autoryzacja | 5 |
| 2 | Opis biznesowy | 6 |
| 3 | Opis struktury serwisu | 6 |
| 3.1 | Autoryzacja | 6 |
| 3.2 | Opis metod interfejsów | 7 |
| 3.2.1 | Interfejs DPDCRXmlServices | 7 |
| 3.3 | Kody błędów | 38 |
| 3.4 | Uwagi i sugestie | 39 |
| 4 | Załączniki | 39 |

Dokumentacja klienta

1 Wstęp

1.1 Cel dokumentu

Celem dokumentu jest opisanie sposobu działania i wykorzystania interfejsów typu WebService na użytek klientów firmy DPD. Stworzone rozwiązanie udostępnia uniwersalne (niezależne od typu systemu) narzędzie przesyłu informacji pomiędzy systemami klienckimi, a DPD.

Dokument zawiera opis metod Web Serwisu DPD AppServices służącego do przekazywania zleceń odbioru Collection Request (CR) do systemów DPD. Opis przygotowany dla wersji DPD AppServices 1.10.3.

Rodzaje zleceń odbioru CR:

- CR-IN - zlecenia CR, których adres nadania (odbioru paczki od nadawcy) jest w obrębie terytorium Polski;
- CR-OUT - zlecenia CR, których adres nadania jest poza granicami Polski;

Rodzaj zlecenia determinowany jest przez parametr kodu kraju podanego w parametrach nadawcy.

Każde wywołanie metody Web Serwisu jest autoryzowane loginem i hasłem oraz oznaczane jedną z trzech polityk zachowania się podczas napotkania błędu:

- przerwanie przetwarzania w momencie napotkania pierwszego błędu;
- ignorowanie przesyłek, które mają błędne dane;
- przerwanie przetwarzania w momencie napotkania pierwszego błędu i anulowanie przesyłek przetworzonych przed wystąpieniem błędu (opcja dostępna tylko w niektórych metodach Web Serwisu);

Klient zintegrowany tym rozwiązaniem z DPD ma pewność zgodności przygotowanych przesyłek z obowiązującym standardem, co implikuje poprawę wydajności i niezawodności usług oferowanych przez naszą firmę kurierską.

1.2 Słownik pojęć

Zbiór pojęć, które nie są w powszechnym użyciu a które są wykorzystywane w tej dokumentacji.

| Pojęcie | Definicja |
|----------|---|
| awizacja | Dostarczenie informacji o przekazaniu przesyłki |

| | |
|--------------------------|---|
| numer listu przewozowego | Numer identyfikujący daną paczkę; inaczej: waybill |
| numkat | Numer identyfikujący klienta DPD; inaczej: fid |
| OpenUMLF | Format danych wykorzystywany przy integracji systemów DPD. Format posiada wiele wersji, które oznaczane są sufiksem 'VX', gdzie 'X' odpowiada liczbie dla kolejnej wersji np. OpenUMLFV2. Format posiada również odmiany dedykowanych przeznaczeń, które są oznaczane sufiksem 'e' - zastosowanie zewnętrzne dla Klienta DPD np. OpenUMLFeV2 lub 'i' - zastosowanie wewnętrzne DPD np. OpenUMLFiV2. |

1.3 Adresy usługi

Interfejs DPDCRXmlServices

Wersja PROD:

<https://dpdappservices.dpd.com.pl/DPDCRXmlServicesService/DPDCRXmlServices?wsdl>

Wersja DEMO:

<https://dpdappservicesdemo.dpd.com.pl/DPDCRXmlServicesService/DPDCRXmlServices?wsdl>

Dane do logowania dla wersji DEMO

```
<login>DAS test</login>
<masterFid>1495</masterFid>
<password>123_DAS test_456</password>
```

1.4 Standard bezpieczeństwa oraz autoryzacja

Zabezpieczeniem przesyłanych danych jest użyty standard SSL o 128-bitowym niesymetrycznym kluczu. Każde wywołanie jest zabezpieczone loginem i hasłem przechowywanym po stronie DPD Polska w bazie danych opartej o LDAP. Dodatkowo warunkiem sprawdzanym podczas wywołania WebService jest konieczność posiadania przez użytkownika uprawnień do operacji dla danego numeru klienckiego.

Pełna konfiguracja konta użytkownika upoważnionego do korzystania z interfejsów webowych odbywa się w systemach DPD Polska po wcześniejszych ustaleniach z klientem. Autoryzacja odbywa się poprzez umieszczenie danych logowania oraz

numeru klienta wewnątrz wysłanego zapytania. Więcej na ten temat znajduje się wewnątrz opisu metod wywoływanych z interfejsów serwisu.

2 Opis biznesowy

Serwis DPD AppServices służy do rejestracji zleceń odbioru CR w systemach DPD oraz aktualizacji ich statusów. Zlecenia CR mogą być generowane na podstawie żądań z systemów Klientów DPD. W zależności od typu zlecenia CR-IN lub CR-OUT są one zapisywane przez AppServices w dedykowanych bazach. W ramach rejestracji zlecenia CR realizowana jest ich walidacja, zarówno pod kątem poprawnej składni jak również kontekstu biznesowego. W odpowiedzi na żądanie rejestracji zlecenia zwracany jest status wykonania rejestracji, w tym informacja o ewentualnych błędach występujących przy rejestracji.

3 Opis struktury serwisu

Metody opisane są według następującego porządku:

- Sygnatura – postać metody po stronie webserwisu
- Parametry – opisuje wszystkie parametry występujące w strukturze żądania
- Walidacja – dotyczy informacji zwrotnej w przypadku nieprawidłowego wypełnienia formularza tworzącego zapytanie do serwisu
- Wywołania metody – przykłady gotowych żądań w różnych konfiguracjach parametrów
- Wyniki zapytania – opis informacji zwrotnej w różnych wariantach

3.1 Autoryzacja

Wszystkie metody opisane w tym rozdziale posiadają w swej sygnaturze parametr `authDataV1` (typu `AuthDataV1`), przechowujący dane autoryzacji do usługi. Każdy odbiorca powinien posiadać swoje własne parametry logowania. `AuthDataV1` składa się z trzech elementów:

- `login` (`String`) – unikalny identyfikator przydzielany odbiorcy ze strony usługodawcy;
- `password` (`String`) – hasło (jawnym tekstem) przyporządkowane do loginu;
- `masterFid` (`Integer`) – unikalny numer katalogowy klienta przekazywany ze strony usługodawcy. Nazywany również numkatem;

Parametr ten jest wymagany, a jego brak skutkuje komunikatem „AuthData is null”. W takiej walidacji sprawdzane są następujące warunki jego struktury wewnętrznej:

| Nazwa pola | Warunek walidacyjny | Pole wymagane | Dodatkowe informacje |
|------------|----------------------------|---------------|--|
| login | Brak pola lub błędny login | Tak | Błąd skutkuje komunikatem "Login failed" |
| password | Brak pola lub błędny login | Tak | Błąd skutkuje komunikatem "Login failed" |
| masterFid | Brak walidacji | Nie | |

3.2 Opis metod interfejsów

Dla niektórych z metod z interfejsów DPDAppServices dostępne są dwie ich odmiany:

- zawierające w numerze wersji przedrostek „X”, które przyjmują oraz zwracają dane zakodowane w postaci Base64;
- zawierające w numerze wersji przedrostek „C”, które przyjmują oraz zwracają dane skompresowane za pomocą ZIP a następnie zakodowane w postaci Base64;

3.2.1 Interfejs DPDCRXmlServices

Interfejs DPDCRXmlServices udostępnia metody służące do przekazania zleceń CR z zewnętrznych systemów do systemu DPD. W ramach interfejsu udostępnione są dwie metody `importPackagesXV1` i `importPackagesCV1` oraz dwie metody `importPackagesXV2` i `importPackagesCV2`. W ramach parametrów wejściowych metod przekazywane są informacje o zleceniach CR a w tym informacje o przesyłkach, których dotyczą zlecenia. Szczegółowy opis metod znajduje się w dalszej części dokumentu.

3.2.1.1 Metoda `importPackagesXV1`

3.2.1.1.1 Sygnatura

| | |
|-------------------|--|
| Nazwa metody | <code>importPackagesXV1</code> |
| Sygnatura wejścia | <code>byte[] importPackagesXV1(byte[] openUMLFXV2, PkgImportPolicyV1 pkgNumsGenerationPolicyV1, AuthDataV1 authDataV1)</code> |

| | |
|----------------------------|--|
| Parametry wejściowe | openUMLFXV2 – dane (XML zakodowany jako Base64, w postaci tablicy bajtów) o przesyłkach w formacie OpenUMLFV2; |
| | pkgNumsGenerationPolicyV1 – polityka postępowania w razie wystąpienia błędów podczas importu danych, typ wyliczeniowy PkgImportPolicyV1; |
| | authDataV1 – dane używane do autoryzacji użytkownika, typ AuthDataV1; |
| Wyjście | byte[] – XML w formacie importPackagesXV1Response zakodowany jako Base64, w postaci tablicy bajtów; |

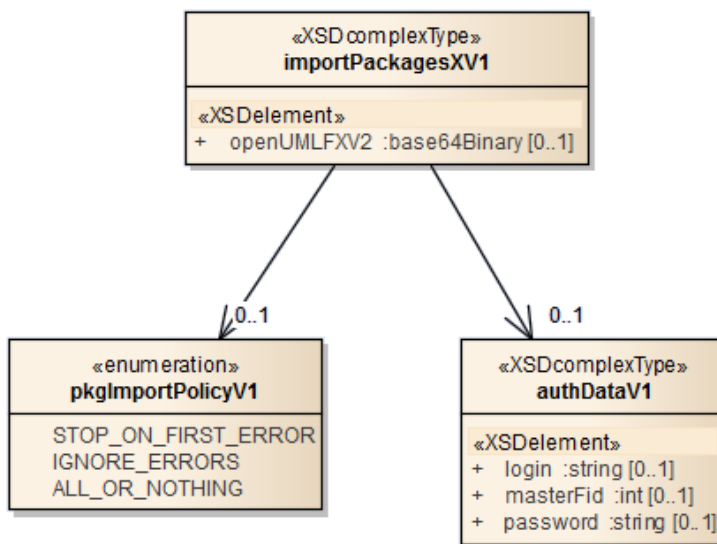
3.2.1.1.2 Parametry i walidacja danych

Dla importPackagesXV1:

| Nazwa pola | Typ | Warunek walidacyjny | Pole wymagane | Dodatkowe informacje |
|-------------|--------|---------------------|---------------|---|
| openUMLFXV2 | byte[] | Brak pola | Tak | Parametr, w którym przekazywane są dane dotyczące przesyłki, wybranych usług i innych kluczowych informacji, które wskazują na sposób przetwarzania informacji o przesyłce. Dane przekazywane są w postaci tablicy bajtów, która zawiera XML w formacie OpenUMLFV2 zakodowany Base64. Więcej informacji dot. formatu OpenUMLFV2 można znaleźć w załączniku opisującym wszystkie wersje formatu OpenUMLF. Załącznik, o którym mowa nosi nazwę „Opis OpenUMLF”. |

| | | | | |
|---------------------------|---------------------------|----------------|-----|--|
| pkgNumsGenerationPolicyV1 | PkgNumsGenerationPolicyV1 | Brak walidacji | Nie | <p>Parametr słownikowany określający politykę usługi dotyczącą sposobu reagowania na pojawiające się błędy podczas realizacji wywołania metody. W zależności od wybranej opcji realizacja usługi będzie przebiegała inaczej w przypadku napotkania na błędy. Możliwymi wartościami są:</p> <p>STOP_ON_FIRST_ERROR – oznacza, że usługa zatrzyma dalsze wykonywanie operacji w przypadku napotkania pierwszego błędu dla którejkolwiek z wystawianych paczek. Import pierwszych paczek nastąpi prawidłowo;</p> <p>IGNORE_ERRORS – powoduje ignorowanie ogólnego błędu określonego rekordu / paczki. Oznacza to, że ignorowana będzie paczka błędna, ale reszta paczek zawierająca prawidłowe dane zostanie prawidłowo wystawiona przez usługę;</p> <p>ALL_OR_NOTHING – zatrzymuje import paczek dla jakiegokolwiek błędu. Wszystkie paczki muszą zostać zweryfikowane prawidłowo, aby nastąpił import; Domyślną wartością jest ALL_OR_NOTHING.</p> |
| authDataV1 | AuthDataV1 | Brak pola | Tak | Brak pola skutkuje komunikatem "AuthData is null". |

3.2.1.1.3 Struktura zapytania



Poniżej zawarte jest przykładowe poprawne zapytanie XML dla metody `importPackagesXV1`. W elemencie `openUMLFXV2` zawarty jest XML w formacie `OpenUMLFV2` zakodowany jako `Base64`.

Zapytanie dla metody importPackagesXV1 interfejsu DPDCRXmlServices:

```
<?xml version='1.0' encoding='UTF-8'?><soapenv:Envelope xmlns:cr="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cr="http://cr.dpdappservices.dpd.com.pl/"><soapenv:Header/><soapenv:Body><cr:importPackagesXV1><openUMLFXV2>
```

PFBhY2thZ2VzPg0KCTxQYWNRyWdIPg0KICAgICAglCA8U2VuZGVyPg0KICAgICAglCAglCAgPENvbXBhbmk+PC9Db21wYW55Pg0KICAgICAglCAglCAglCAgPE5hbWU+cHJ6ZXN5bGthIHogYXBwlHNlcnpY2VzPC9OYW1lPg0KICAgICAglCAglCAglCAgPEFkZHJlc3M+VGVyZW5janVzemEgMTI8L0FkZHJlc3M+DQoglCAglCAglCAglCA8Q2l0eT5PbHN6dHluPC9DaXR5Pg0KICAgICAglCAglCAglCAgPENvdW50cnlDb2RIPIBMPC9Db3VudHJ5Q29kZT4NCiAgICAglCAglCAglCAglDxB3N0YXwDb2RlPjEwMTc4PC9Qb3N0YXwDb2RlPg0KICAgICAglCAglCAglCAgPFBob25lPjwvUGhvbmU+DQoglCAglCAglCAglCA8RW1haWw+PC9FbWFpbD4NCiAgICAglCAglCAgPC9TZW5kZXI+DQoglCAglCAglCAglDxSZWNlaXZlcj4NCiAgICAglCAglCAglCAglDxB21wYW55PjwvQ29tcGFueT4NCiAgICAglCAglCAglCAglDxOYW1lPIRlc3QgUGllY2R6aWVzaWF0eSBUCnplY2k8L05hbWU+DQoglCAglCAglCAglCA8QRWRkcmVzcz5DYXJza2EgMzI8L0FkZHJlc3M+DQoglCAglCAglCAglCA8Q2l0eT5XYXJzemF3YTwwQ2l0eT4NCiAgICAglCAglCAglCAglDxB3VudHJ5Q29kZT5QTdwwQ291bnRyeUNvZGU+DQoglCAglCAglCAglCA8UG9zdGFsQ29kZT4wMjlzMjwvUG9zd

```
</openUMLFXV2>
<pkgNumsGenerationPolicyV1>ALL_OR_NOTHING</pkgNumsGenerationPolicyV1>
<authDataV1>
  <login>pmarat</login>
  <masterFid>1495</masterFid>
  <password>****</password>
</authDataV1>
</cr:importPackagesXV1>
</soapenv:Body>
</soapenv:Envelope>
```

Struktura openUMLFXV2 dla zapytania metody importPackagesXV1 interfejsu DPDCRXmlServices:

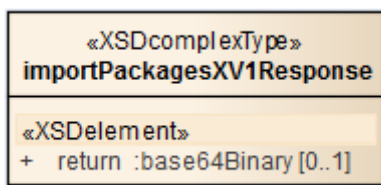
11

```

    <Phone></Phone>
    <Email></Email>
  </Receiver>
  <PayerType>THIRD_PARTY</PayerType>
  <ThirdPartyFID>1495</ThirdPartyFID>
  <Ref1>111111</Ref1>
  <Ref2></Ref2>
  <Customer>
    <FID>1495</FID>
  </Customer>
  <Services>
    <Guarantee type="TIME0930"/>
    <InPers/>
    <CarryIn/>
  </Services>
  <Parcels>
    <Parcel>
      <Weight>35</Weight>
      <SizeX>100</SizeX>
      <SizeY>40</SizeY>
      <SizeZ>40</SizeZ>
      <Content>test</Content>
      <CustomerData1>klienta dane 1</CustomerData1>
      <CustomerData2>klienta dane 2</CustomerData2>
      <CustomerData3>klienta dane 3</CustomerData3>
    </Parcel>
  </Parcels>
</Package>
</Packages>

```

3.2.1.1.4 Struktura odpowiedzi



Poniżej zawarta jest przykładowa odpowiedź dla poprawnego zapytania XML metody importPackagesXV1.

Odpowiedź dla poprawnego zapytania metody importPackagesXV1 interfejsu DPDCRXmlServices:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:importPackagesXV1Response xmlns:ns2="http://cr.dpdappservices.dpd.com.pl/">
      <return>
        PFBhY2thZ2VzSW1wb3J0UmVzcG9uc2U+PFN0YXR1c0luZm8+PFN0YXR1cz5PSzwwU3RhdHVzPjwv
        vU3RhdHVzSW5mbz48UGFja2FnZXM+PFBhY2thZ2U+PFBhY2thZ2VJZD45NzY5MDgzPC9QYWNRy
        WdISWQ+PFN0YXR1c0luZm8+PFN0YXR1cz5PSzwwU3RhdHVzPjwvU3RhdHVzSW5mbz48UGFyY2
        Vscz48UGFyY2VsPjxQYXJjZWxJZD45NjkyMTYyPC9QYXJjZWxJZD48V2F5MlIsbD4wMDAwMDA5
      </return>
    </ns2:importPackagesXV1Response>
  </S:Body>
</S:Envelope>

```

NjA5MzYyVDwvV2F5YmIsbD48L1BhcmNlbD48L1BhcmNlbHM+PE9yZGVyTnVtYmVyPkFQUC9DUk
IOLzk3NjkwODM8L09yZGVyTnVtYmVyPjwvUGFja2FnZT48L1BhY2thZ2VzPjwvUGFja2FnZXNJbXB
vcnRSZXNwb25zZT4=

```
</return>
</ns2:importPackagesXV1Response>
</S:Body>
</S:Envelope>
```

Poniżej zawarta jest postać odpowiedzi dla zapytania XML metody importPackagesXV1 interfejsu DPDCRXmlServices z błędnymi danymi autoryzacji użytkownika.

Odpowiedź dla zapytania z błędnymi danymi autoryzacji użytkownika:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <S:Fault xmlns:ns4="http://www.w3.org/2003/05/soap-envelope">
      <faultcode>S:Server</faultcode>
      <faultstring>Login failed</faultstring>
      <detail>
        <ns2:DPDAppAuthenticationException xmlns:ns2="http://cr.dpdappservices.dpd.com.pl/">
          <login>pmarat</login>
          <message>Login failed</message>
        </ns2:DPDAppAuthenticationException>
        <ns2:exception
class="pl.com.dpd.dpdappservices.exception.DPDAppAuthenticationException" note="To disable
this feature, set com.sun.xml.ws.fault.SOAPFaultBuilder.disableCaptureStackTrace
system property to false" xmlns:ns2="http://jax-ws.dev.java.net/">
          <message>Login failed</message>
          <ns2:stackTrace>
            <!--Zawartość StackTrace-->
          </ns2:stackTrace>
        </ns2:exception>
      </detail>
    </S:Fault>
  </S:Body>
</S:Envelope>
```

W elemencie return odpowiedzi dla importPackagesXV1 zawarty jest XML w formacie PackagesImportResponse zakodowany jako Base64. Zwracany jest w nim status importu paczek. Poniżej przykładowa postać dla poprawnie wykonanego importu.

Struktura PackagesImportResponse - Status OK:

```
<PackagesImportResponse>
  <StatusInfo>
    <Status>OK</Status>
  </StatusInfo>
  <Packages>
    <Package>
      <PackageId>9769083</PackageId>
      <StatusInfo>
        <Status>OK</Status>
      </StatusInfo>
    </Package>
  </Packages>
</PackagesImportResponse>
```

```

<Parcels>
  <Parcel>
    <ParcelId>9692162</ParcelId>
    <Waybill>0000009609362T</Waybill>
  </Parcel>
</Parcels>
<OrderNumber>APP/CRIN/9769083</OrderNumber>
</Package>
</Packages>
</PackagesImportResponse>

```

Przykład PackagesImportResponse dla zapytania XML metody importPackagesXV1 interfejsu DPDCRXmlServices z błędnym kodem pocztowym nadawcy.

Struktura PackagesImportResponse - Status INCORRECT_DATA:

```

<PackagesImportResponse>
  <StatusInfo>
    <Status>INCORRECT_DATA</Status>
  </StatusInfo>
  <Packages>
    <Package>
      <StatusInfo>
        <Status>INCORRECT_DATA</Status>
        <ErrorDetails>
          <Code>INCORRECT_SENDER_POSTAL_CODE</Code>
          <Description>Niepoprawny kod pocztowy nadawcy (10178)</Description>
          <Fields>SenderPostalCode</Fields>
        </ErrorDetails>
      </StatusInfo>
    <Parcels>
      <Parcel></Parcel>
    </Parcels>
    </Package>
  </Packages>
</PackagesImportResponse>

```

Przykład PackagesImportResponse dla zapytania XML metody importPackagesXV1 interfejsu DPDCRXmlServices z błędną konfiguracją usług dla przesyłki.

Struktura PackagesImportResponse - Status INCORRECT_DATA:

```

<PackagesImportResponse>
  <StatusInfo>
    <Status>INCORRECT_DATA</Status>
  </StatusInfo>
  <Packages>
    <Package>
      <StatusInfo>
        <Status>INCORRECT_DATA</Status>
        <ErrorDetails>
          <Code>ERROR_INCORRECT_WEIGHT_FOR_DOX</Code>

```

```

    <Description>Usługa 'DOX' dostępna dla paczek o wadze do 0,5 kg</Description>
    <Fields>Parcels/Parcel/Weight;Services/DOX;</Fields>
  </ErrorDetails>
</StatusInfo>
<Parcels>
  <Parcel></Parcel>
</Parcels>
</Package>
</Packages>
</PackagesImportResponse>

```

3.2.1.2 Metoda importPackagesCV1

3.2.1.2.1 Sygnatura

| | |
|----------------------------|---|
| Nazwa metody | importPackagesCV1 |
| Sygnatura wejścia | byte[] importPackagesCV1(byte[] openUMLFCV2, PkgImportPolicyV1 pkgNumsGenerationPolicyV1, AuthDataV1 authDataV1) |
| Parametry wejściowe | openUMLFCV2 – dane (tablica bajtów – XML skompresowany za pomocą ZIP a następnie zakodowany jako Base64) o przesyłkach w formacie OpenUMLFV2; |
| | pkgNumsGenerationPolicyV1 – polityka postępowania w razie wystąpienia błędów podczas importu danych, typ wyliczeniowy PkgImportPolicyV1; |
| | authDataV1 – dane używane do autoryzacji użytkownika, typ AuthDataV1; |
| Wyjście | byte[] – XML w formacie importPackagesCV1Response skompresowany za pomocą ZIP a następnie zakodowany jako Base64 w postaci tablicy bajtów; |

3.2.1.2.2 Parametry i walidacja danych

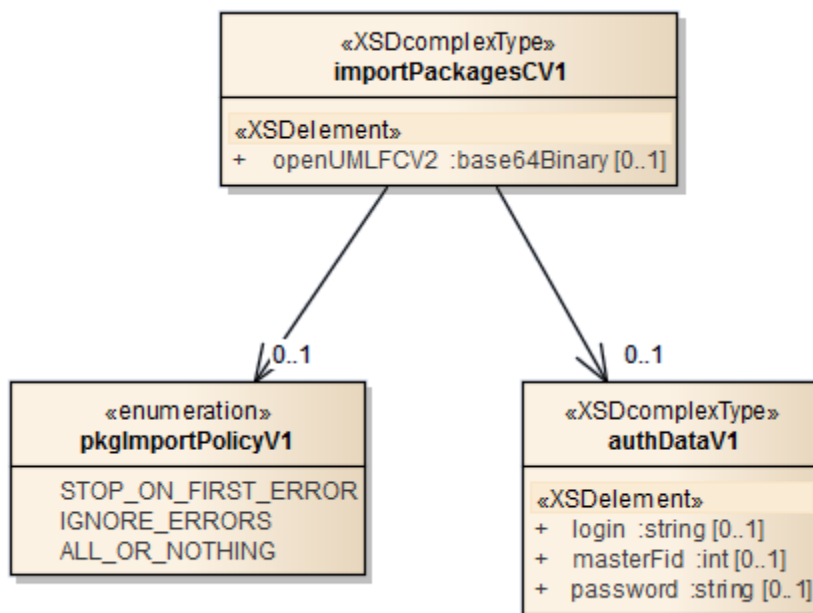
Dla importPackagesCV1:

| Nazwa pola | Typ | Warunek walidacyjny | Pole wymagane | Dodatkowe informacje |
|------------|-----|---------------------|---------------|----------------------|
|------------|-----|---------------------|---------------|----------------------|

| | | | | |
|-------------|--------|-----------|-----|---|
| openUMLFCV2 | byte[] | Brak pola | Tak | <p>Parametr, w którym przekazywane są dane dotyczące przesyłki, wybranych usług i innych kluczowych informacji, które wskazują na sposób przetwarzania informacji o przesyłce. Dane przekazywane są w postaci tablicy bajtów, która zawiera XML w formacie OpenUMLFV2 skompresowane za pomocą ZIP a następnie zakodowany Base64. Więcej informacji dot. formatu OpenUMLFV2 można znaleźć w załączniku opisującym wszystkie wersje formatu OpenUMLF. Załącznik, o którym mowa nosi nazwę „Opis OpenUMLF”</p> |
|-------------|--------|-----------|-----|---|

| | | | | |
|---------------------------|---------------------------|----------------|-----|--|
| pkgNumsGenerationPolicyV1 | PkgNumsGenerationPolicyV1 | Brak walidacji | Nie | <p>Parametr słownikowy określający politykę usługi dotyczącą sposobu reagowania na pojawiające się błędy podczas realizacji wywołania metody. W zależności od wybranej opcji realizacja usługi będzie przebiegała inaczej w przypadku napotkania na błędy. Możliwymi wartościami są:</p> <p>STOP_ON_FIRST_ERROR – oznacza, że usługa zatrzyma dalsze wykonywanie operacji w przypadku napotkania pierwszego błędu dla którejkolwiek z wystawianych paczek. Import pierwszych paczek nastąpi prawidłowo;</p> <p>IGNORE_ERRORS – powoduje ignorowanie ogólnego błędu określonego rekordu / paczki. Oznacza to, że ignorowana będzie paczka błędna, ale reszta paczek zawierająca prawidłowe dane zostanie prawidłowo wystawiona przez usługę;</p> <p>ALL_OR_NOTHING – zatrzymuje import paczek dla jakiegokolwiek błędu. Wszystkie paczki muszą zostać zweryfikowane prawidłowo, aby nastąpił import; Domyślną wartością jest ALL_OR_NOTHING.</p> |
| authDataV1 | AuthDataV1 | Brak pola | Tak | Brak pola skutkuje komunikatem "AuthData is null". |

3.2.1.2.3 Struktura zapytania



Poniżej zawarte jest przykładowe poprawne zapytanie XML dla metody importPackagesCV1. W elemencie openUMLFCV2 zawarty jest XML w formacie OpenUMLFV2, skompresowany ZIP a następnie zakodowany jako Base64. Postać tego elementu jest tożsama z opisem dla metody importPackagesXV1 interfejsu DPDCRXmlServices.

Zapytanie dla metody importPackagesCV1 interfejsu DPDCRXmlServices:

```

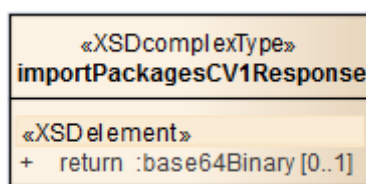
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cr="http://cr.dpdappservices.dpd.com.pl/">
  <soapenv:Header/>
  <soapenv:Body>
    <cr:importPackagesCV1 xmlns:ns2="http://cr.dpdappservices.dpd.com.pl/">
      <openUMLFCV2>
        UEsDBBQAAAAIAHtwbEwfZm5GqgIAAB4JAAAYAAAAQ1ltSU5fb3BibIVNTFhWMI9yZXEuG1s7V
        bbbtNAEH0uEv9g8h7WI5YWtLVU7BYiocZyTG8vaGVP2yWObE2uC/YP8BH8DvwX42vqpMUSKm/1
        Q7xn5szs7pyZyNRj4ZLdgLRfvtjpAK619qELSCIQraW3OukqY0lhU9KuNgmnBAV2JkqQRbxbkWqmx
        LNMkiDsegqSkcm+GHEWRACntAAQk4ddclkwzTEpa+9YRuCrseSxLVSSUVGj7kHmiROGkEdjeJ+S
        s8SbVS6Vice0xdGP/...+vacw785hZloKRZPcRZ8BlusLnr92OEy5Zw+RjhqiVcPURwUuxE1F/hXxgIH
        dom9pPjMsUMexlz5DEtYgloOI8D71i4OQzHEgy8Y+HWMNwahlubEpBGg0bNcUH2xgUx9FFJDH1U
        FEN/lqWVpQPNlw9Zf/r0a2n/AVBLAQI/ABQAAAAIAHtwbEwfZm5GqgIAAB4JAAAYACQAAAAAA
        AAIAAAAAAAAAABDui1JTl9vcGVuVU1MWFYyX3JlcS54bWwKACAAAAAAAAEAGADC96aSArrTAS
        +KF0YvttMBL6gSRi+20wFQSwUGAAAAAAEAAQBqAAAA4AIAAAAA
      </openUMLFCV2>
      <pkgNumsGenerationPolicyV1>ALL_OR_NOTHING</pkgNumsGenerationPolicyV1>
    </cr:importPackagesCV1>
  </soapenv:Body>
</soapenv:Envelope>
  
```

```

<authDataV1>
  <login>pmarat</login>
  <masterFid>1495</masterFid>
  <password>****</password>
</authDataV1>
</cr:importPackagesCV1>
</soapenv:Body>
</soapenv:Envelope>

```

3.2.1.2.4 Struktura odpowiedzi



Poniżej zawarta jest przykładowa odpowiedź dla poprawnego zapytania XML metody importPackagesCV1.

Odpowiedź dla poprawnego zapytania metody importPackagesCV1 interfejsu DPDCRXmlServices:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:importPackagesCV1Response xmlns:ns2="http://cr.dpdappservices.dpd.com.pl/">
      <return>
        UEsDBBQACAAIAMhzbEwAAAAAAAAAAAAAAAAAAAAAdGV4dJWRbwuCMBCHP9KZyZwwBtGr
        EahY0OupK8L5h22+6NtH5k2Jltqbe/jdeDjuWC6rRI6VFe3QG1coO/SdVZwdnXSjFd2IR+bZgcGMCK8
        +Ojx5EDVPYkljGjNYoj/lplLalmB9mkkSbqLJPCfsLO/ITWseTC8hAY1JeGKAOf79aqO/bNuPNvBTZq
        ZWJh3bUhm+y3PYFyIFv4N1129klbvCt4M8AFBLBwgjiqNKogAAAKkBAABQSwECFAAUAAgACAD
        lc2xMI4qjSqlAAACpAQAABAAAAAAAAAAAAAAAAAAAAAdGV4dFBLBQYAAAAAAAAQABADIA
        AADUAAAAAAAA=
      </return>
    </ns2:importPackagesCV1Response>
  </S:Body>
</S:Envelope>

```

W elemencie return powyższego komunikatu zawarty jest XML w formacie PackagesImportResponse skompresowany ZIP a następnie zakodowany jako Base64. Zwracany jest w nim status importu paczek. Postać formatu PackagesImportResponse jest taka sama jak dla odpowiedzi z metody importPackagesXV1 interfejsu DPDCRXmlServices.

3.2.1.3 Metoda importPackagesXV2

3.2.1.3.1 Sygnatura

| | |
|----------------------------|--|
| Nazwa metody | importPackagesXV2 |
| Sygnatura wejścia | byte[] importPackagesXV2(byte[] openUMLFXV2, PkgImportPolicyV1 pkgNumsGenerationPolicyV1, AuthDataV1 authDataV1) |
| Parametry wejściowe | openUMLFXV2 – dane (XML zakodowany jako Base64, w postaci tablicy bajtów) o przesyłkach w formacie OpenUMLFV2; |
| | pkgNumsGenerationPolicyV1 – polityka postępowania w razie wystąpienia błędów podczas importu danych, typ wyliczeniowy PkgImportPolicyV1; |
| | authDataV1 – dane używane do autoryzacji użytkownika, typ AuthDataV1; |
| Wyjście | byte[] – XML w formacie importPackagesXV2Response zakodowany jako Base64, w postaci tablicy bajtów; |

3.2.1.3.2 Parametry i walidacja danych

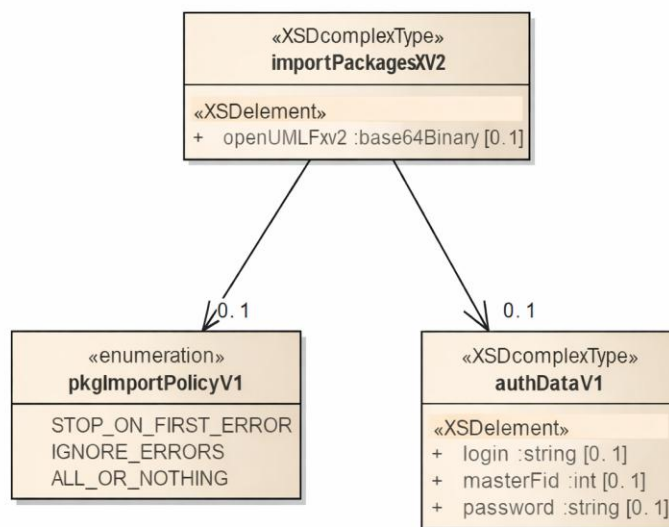
Dla importPackagesXV2:

| Nazwa pola | Typ | Warunek walidacyjny | Pole wymagane | Dodatkowe informacje |
|------------|-----|---------------------|---------------|----------------------|
|------------|-----|---------------------|---------------|----------------------|

| | | | | |
|-------------|--------|-----------|-----|--|
| openUMLFXV2 | byte[] | Brak pola | Tak | <p>Parametr, w którym przekazywane są dane dotyczące przesyłki, wybranych usług i innych kluczowych informacji, które wskazują na sposób przetwarzania informacji o przesyłce. Dane przekazywane są w postaci tablicy bajtów, która zawiera XML w formacie OpenUMLFV2 zakodowany Base64. Więcej informacji dot. formatu OpenUMLFV2 można znaleźć w załączniku opisującym wszystkie wersje formatu OpenUMLF. Załącznik, o którym mowa nosi nazwę „Opis OpenUMLF”.</p> |
|-------------|--------|-----------|-----|--|

| | | | | |
|---------------------------|---------------------------|----------------|-----|--|
| pkgNumsGenerationPolicyV1 | PkgNumsGenerationPolicyV1 | Brak walidacji | Nie | <p>Parametr słownikowany określający politykę usługi dotyczącą sposobu reagowania na pojawiające się błędy podczas realizacji wywołania metody. W zależności od wybranej opcji realizacja usługi będzie przebiegała inaczej w przypadku napotkania na błędy. Możliwymi wartościami są:</p> <p>STOP_ON_FIRST_ERROR – oznacza, że usługa zatrzyma dalsze wykonywanie operacji w przypadku napotkania pierwszego błędu dla którejkolwiek z wystawianych paczek. Import pierwszych paczek nastąpi prawidłowo;</p> <p>IGNORE_ERRORS – powoduje ignorowanie ogólnego błędu określonego rekordu / paczki. Oznacza to, że ignorowana będzie paczka błędna, ale reszta paczek zawierająca prawidłowe dane zostanie prawidłowo wystawiona przez usługę;</p> <p>ALL_OR_NOTHING – zatrzymuje import paczek dla jakiegokolwiek błędu. Wszystkie paczki muszą zostać zweryfikowane prawidłowo, aby nastąpił import; Domyślną wartością jest ALL_OR_NOTHING.</p> |
| authDataV1 | AuthDataV1 | Brak pola | Tak | Brak pola skutkuje komunikatem "AuthData is null". |

3.2.1.3.3 Struktura zapytania



Poniżej zawarte jest przykładowe poprawne zapytanie XML dla metody importPackagesXV2. W elemencie openUMLFXV2 zawarty jest XML w formacie OpenUMLFV2 zakodowany jako Base64.

Zapytanie dla metody importPackagesXV2 interfejsu DPDCRXmlServices:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cr="http://cr.dpdappservices.dpd.com.pl/">
  <soapenv:Header/>
  <soapenv:Body>
    <cr:importPackagesXV2>
      <openUMLFXV2>

```

```

PFBhY2thZ2VzPgogICAgPFBhY2thZ2U+CiAgICAgICAgPFNIbmRlcj4KICAgICAgICAgICAgPENVbXBhbmk+QX
V0b21hdGVkIFNIbmRlcjwvQ29tcGFueT4KICAgICAgICAgICAgPE5hbWU+QXV0b21hdGVkIFNIbmRlcjwvTmFtZ
T4KICAgICAgICAgICAgPEFkZHIc3M+TWluZXJhbG5hIDE1PC9BZGRyZXNzPgogICAgICAgICAgICAgCA8Q2I0eT5
YXJzZemF3YTwwQ2I0eT4KICAgICAgICAgICAgPENVdW50cnIDb2RlPBMPC9Db3VudHJ5Q29kZT4KICAgICAg
ICAgICAgPFBvc3RhbnRvZGU+MDIyNzQ8L1Bvc3RhbnRvZGU+CiAgICAgICAgICAgICAgICAgICAgICAgICAgIC
ODk8L1Bob25lPgogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
AgPC9TZW5kZXI+CiAgICAgICAgICAgPFJlY2VpdMvYpPgogICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
AgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
AgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
eXN6dG9mYSBLb2x1bWJhIDEyPC9BZGRyZXNzPgogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
AgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAg
ICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
2RlPjYyMDUyPC9Qb3N0YWwDb2RlPpogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
AgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAg

```



```

    <CountryCode>PL</CountryCode>
    <PostalCode>02272</PostalCode>
    <Phone></Phone>
    <Email></Email>
  </Receiver>
  <PayerType>THIRD_PARTY</PayerType>
  <ThirdPartyFID>1495</ThirdPartyFID>
  <Ref1>111111</Ref1>
  <Ref2></Ref2>
  <Customer>
    <FID>1495</FID>
  </Customer>
  <Services>
    <Guarantee type="TIME0930"/>
    <InPers/>
    <CarryIn/>
  </Services>
  <Parcels>
    <Parcel>
      <Weight>35</Weight>
      <SizeX>100</SizeX>
      <SizeY>40</SizeY>
      <SizeZ>40</SizeZ>
      <Content>test</Content>
      <CustomerData1>klienta dane 1</CustomerData1>
      <CustomerData2>klienta dane 2</CustomerData2>
      <CustomerData3>klienta dane 3</CustomerData3>
    </Parcel>
  </Parcels>
  <DeliveryDate>2026-02-11</DeliveryDate>
</Package>
</Packages>

```

3.2.1.3.4 Struktura odpowiedzi



Poniżej zawarta jest przykładowa odpowiedź dla poprawnego zapytania XML metody importPackagesXV2.

Odpowiedź dla poprawnego zapytania metody importPackagesXV2 interfejsu DPDCRXmlServices:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:importPackagesXV2Response xmlns:ns2="http://cr.dpdappservices.dpd.com.pl/">
      <return>

```

```

PFBhY2thZ2VzSW1wb3J0UmVzcG9uc2U+PFN0YXR1c0luZm8+PFN0YXR1cz5PSzwwU3RhdHVzPjw
vU3RhdHVzSW5mbz48UGFja2FnZXN+PFBhY2thZ2U+PFBhY2thZ2VJZD45ODczMDE5PC9QYWNRy
WdlSWQ+PFN0YXR1c0luZm8+PFN0YXR1cz5PSzwwU3RhdHVzPjwvU3RhdHVzSW5mbz48UGFyY2
Vscz48UGFyY2VsPjxQYXJjZWxJZD45NzQ0ODE4PC9QYXJjZWxJZD48V2F5YmlsbD4wMDAwMDA5
Njk0MDEwVDwwV2F5YmlsbD48L1BhcmNlbD48UGFyY2VsPjxQYXJjZWxJZD45NzQ0ODE5PC9QYX
JjZWxJZD48V2F5YmlsbD4wMDAwMDA5Njk0MDEwVDwwV2F5YmlsbD48L1BhcmNlbD48L1BhcmNI
bHM+PE9yZGVyTnVtYmVyPkFQUUC9DUKIOLzk4NzMTk8L09yZGVyTnVtYmVyPjwvUGFja2FnZT4
8L1BhY2thZ2VzPjwvUGFja2FnZXNjbXBvcnRSZXNwb25zZT4=
</return>
</ns2:importPackagesXV2Response>
</S:Body>
</S:Envelope>

```

Poniżej zawarta jest postać odpowiedzi dla zapytania XML metody importPackagesXV2 interfejsu DPDCRXmlServices z błędnymi danymi autoryzacji użytkownika.

Odpowiedź dla zapytania z błędnymi danymi autoryzacji użytkownika:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <S:Fault xmlns:ns4="http://www.w3.org/2003/05/soap-envelope">
      <faultcode>S:Server</faultcode>
      <faultstring>Login failed</faultstring>
      <detail>
        <ns2:DPDAppAuthenticationException xmlns:ns2="http://cr.dpdappservices.dpd.com.pl/">
          <login>pmarat</login>
          <message>Login failed</message>
        </ns2:DPDAppAuthenticationException>
        <ns2:exception
class="pl.com.dpd.dpdappservices.exception.DPDAppAuthenticationException" note="To disable
this feature, set com.sun.xml.ws.fault.SOAPFaultBuilder.disableCaptureStackTrace
system property to false" xmlns:ns2="http://jax-ws.dev.java.net/">
          <message>Login failed</message>
          <ns2:stackTrace>
            <!--Zawartość StackTrace-->
          </ns2:stackTrace>
        </ns2:exception>
      </detail>
    </S:Fault>
  </S:Body>
</S:Envelope>

```

W elemencie return odpowiedzi dla importPackagesXV2 zawarty jest XML w formacie PackagesImportResponse zakodowany jako Base64. Zwracany jest w nim status importu paczek. Poniżej przykładowa postać dla poprawnie wykonanego importu.

Struktura PackagesImportResponse - Status OK:

```

<PackagesImportResponse>
  <StatusInfo>
    <Status>OK</Status>
  </StatusInfo>
  <Packages>

```

```

<Package>
  <PackageId>9769083</PackageId>
  <StatusInfo>
    <Status>OK</Status>
  </StatusInfo>
  <Parcels>
    <Parcel>
      <ParcelId>9692162</ParcelId>
      <Waybill>0000009609362T</Waybill>
    </Parcel>
  </Parcels>
  <OrderNumber>APP/CRIN/9769083</OrderNumber>
</Package>
</Packages>
</PackagesImportResponse>

```

Przykład PackagesImportResponse dla zapytania XML metody importPackagesXV2 interfejsu DPDCRXmlServices z błędnym kodem pocztowym nadawcy.

Struktura PackagesImportResponse - Status INCORRECT_DATA:

```

<PackagesImportResponse>
  <StatusInfo>
    <Status>INCORRECT_DATA</Status>
  </StatusInfo>
  <Packages>
    <Package>
      <StatusInfo>
        <Status>INCORRECT_DATA</Status>
      <ErrorDetails>
        <Code>INCORRECT_SENDER_POSTAL_CODE</Code>
        <Description>Niepoprawny kod pocztowy nadawcy (10178)</Description>
        <Fields>SenderPostalCode</Fields>
      </ErrorDetails>
    </StatusInfo>
    <Parcels>
      <Parcel></Parcel>
    </Parcels>
  </Package>
</Packages>
</PackagesImportResponse>

```

Przykład PackagesImportResponse dla zapytania XML metody importPackagesXV2 interfejsu DPDCRXmlServices z błędną konfiguracją usług dla przesyłki.

Struktura PackagesImportResponse - Status INCORRECT_DATA:

```

<PackagesImportResponse>
  <StatusInfo>
    <Status>INCORRECT_DATA</Status>
  </StatusInfo>
  <Packages>

```

```

<Package>
  <StatusInfo>
    <Status>INCORRECT_DATA</Status>
    <ErrorDetails>
      <Code>ERROR_INCORRECT_WEIGHT_FOR_DOX</Code>
      <Description>Usługa 'DOX' dostępna dla paczek o wadze do 0,5 kg</Description>
      <Fields>Parcels/Parcel/Weight;Services/DOX;</Fields>
    </ErrorDetails>
  </StatusInfo>
  <Parcels>
    <Parcel></Parcel>
  </Parcels>
</Package>
</Packages>
</PackagesImportResponse>

```

3.2.1.4 Metoda importPackagesCV2

3.2.1.4.1 Sygnatura

| | |
|----------------------------|---|
| Nazwa metody | importPackagesCV2 |
| Sygnatura wejścia | byte[] importPackagesCV2(byte[] openUMLFCV2, PkgImportPolicyV1 pkgNumsGenerationPolicyV1, AuthDataV1 authDataV1) |
| Parametry wejściowe | openUMLFCV2 – dane (tablica bajtów – XML skompresowany za pomocą ZIP a następnie zakodowany jako Base64) o przesyłkach w formacie OpenUMLFV2; |
| | pkgNumsGenerationPolicyV1 – polityka postępowania w razie wystąpienia błędów podczas importu danych, typ wyliczeniowy PkgImportPolicyV1; |
| | authDataV1 – dane używane do autoryzacji użytkownika, typ AuthDataV1; |
| Wyjście | byte[] – XML w formacie importPackagesCV2Response skompresowany za pomocą ZIP a następnie zakodowany jako Base64 w postaci tablicy bajtów; |

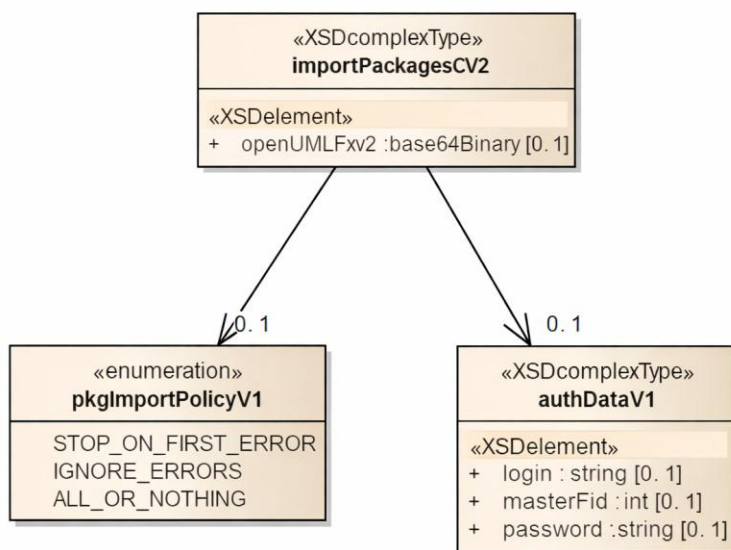
3.2.1.4.2 Parametry i walidacja danych

Dla importPackagesCV2:

| Nazwa pola | Typ | Warunek walidacyjny | Pole wymagane | Dodatkowe informacje |
|-------------|--------|---------------------|---------------|--|
| openUMLFCV2 | byte[] | Brak pola | Tak | Parametr, w którym przekazywane są dane dotyczące przesyłki, wybranych usług i innych kluczowych informacji, które wskazują na sposób przetwarzania informacji o przesyłce. Dane przekazywane są w postaci tablicy bajtów, która zawiera XML w formacie OpenUMLFV2 skompresowane za pomocą ZIP a następnie zakodowany Base64. Więcej informacji dot. formatu OpenUMLFV2 można znaleźć w załączniku opisującym wszystkie wersje formatu OpenUMLF. Załącznik, o którym mowa nosi nazwę „Opis OpenUMLF” |

| | | | | |
|---------------------------|---------------------------|----------------|-----|--|
| pkgNumsGenerationPolicyV1 | PkgNumsGenerationPolicyV1 | Brak walidacji | Nie | <p>Parametr słownikowy określający politykę usługi dotyczącą sposobu reagowania na pojawiające się błędy podczas realizacji wywołania metody. W zależności od wybranej opcji realizacja usługi będzie przebiegała inaczej w przypadku napotkania na błędy. Możliwymi wartościami są:</p> <p>STOP_ON_FIRST_ERROR – oznacza, że usługa zatrzyma dalsze wykonywanie operacji w przypadku napotkania pierwszego błędu dla którejkolwiek z wystawianych paczek. Import pierwszych paczek nastąpi prawidłowo;</p> <p>IGNORE_ERRORS – powoduje ignorowanie ogólnego błędu określonego rekordu / paczki. Oznacza to, że ignorowana będzie paczka błędna, ale reszta paczek zawierająca prawidłowe dane zostanie prawidłowo wystawiona przez usługę;</p> <p>ALL_OR_NOTHING – zatrzymuje import paczek dla jakiegokolwiek błędu. Wszystkie paczki muszą zostać zweryfikowane prawidłowo, aby nastąpił import; Domyślną wartością jest ALL_OR_NOTHING.</p> |
| authDataV1 | AuthDataV1 | Brak pola | Tak | Brak pola skutkuje komunikatem "AuthData is null". |

3.2.1.4.3 Struktura zapytania



Poniżej zawarte jest przykładowe poprawne zapytanie XML dla metody importPackagesCV2. W elemencie openUMLFCV2 zawarty jest XML w formacie OpenUMLFV2, skompresowany ZIP a następnie zakodowany jako Base64. Postać tego elementu jest tożsama z opisem dla metody importPackagesXV2 interfejsu DPDCRXmlServices.

Zapytanie dla metody importPackagesCV2 interfejsu DPDCRXmlServices:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cr="http://cr.dpdappservices.dpd.com.pl/">
  <soapenv:Header/>
  <soapenv:Body>
    <cr:importPackagesCV2 xmlns:ns2="http://cr.dpdappservices.dpd.com.pl/">
      <openUMLFCV2>
        UEsDBBQACAgIAE9NTFwAAAAAAAAAAAAAAAAAAAAAdGV4dN2V327aMBTG7yf1HfICJYIDoEiR
        NQsdVrFNEWXq6M3kJqdgkdjINq3C09f5AyQmNL3YxVQulvs7v5Nz9B2bBCGJNmQFEI99sfQvqPbV
        tpDugcUgakqhTni6JSzD453iKVEQWyUX2lelklCLpNBCF7KBjuNYgJT4J2UgSMKI5fqBfVDNRqjK8A
        MRck9eia6eb8963TElsgmPAYc/8g5PewMNUvQkKSIOQsN+YNcUk11zBthFXt8fDG9GGi0Eg7pNCU2
        wAqncr/E27kU87W2TwC71ms/2mdHBHCKgLf5n/6jZh/e8b/dM7DO5V/yZWDOe7Nln7Tt63/cZT7lgdE
        P/pfED5Pio2/jRzXDg9z3kdhrf4XuLy/oeZCAW2Rbw4vvdFpO3HM8XS13oKNfYxZqKOCRCZd/uptjtj/R
        hbWqNkT67OH/kdfWyGUJ5CBUhVA9NdILP8vwgnAoaZeyWDH2TxQuNQBqn7IzUjUeQmCNvzqBA
        WojgAehqrbDuqVq1MPd0D39ypFxcIJbYd0pkeQI5PCKPbciEMwVMYSc/e+Wyyjaq8mhJFXPz7layotU
        mopsnJyDLYkY1w8Wxmoa4sr8jymImeOWu71fIPDKL3X43C/ZSjOEj5nQmmkOT/J5nOBYwcNLh20L
      
```

```

WrX9oIVF9c+/TJPa4IfgNQSwcIXRTeyvMBAACZBwAAUEsBAhQAFAAICAgAT01MXF0U3srzAQAA
mQcAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAHRIeHRQSwUGAAAAAAEAAQAYAAAAJQIAAAAA
</openUMLFCV2>
<pkgNumsGenerationPolicyV1>ALL_OR_NOTHING</pkgNumsGenerationPolicyV1>
<authDataV1>
  <login>pmarat</login>
  <masterFid>1495</masterFid>
  <password>****</password>
</authDataV1>
</cr:importPackagesCV2>
</soapenv:Body>
</soapenv:Envelope>

```

3.2.1.4.4 Struktura odpowiedzi



Poniżej zawarta jest przykładowa odpowiedź dla poprawnego zapytania XML metody importPackagesCV2.

Odpowiedź dla poprawnego zapytania metody importPackagesCV2 interfejsu DPDCRXmlServices:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:importPackagesCV2Response xmlns:ns2="http://cr.dpdappservices.dpd.com.pl">
      <return>
        UEsDBBQACAgIAJxNTFwAAAAAAAAAAAAAAAAEAAAAdGV4dJWRUQuCMBDHP9JcLm0wBtHT
        CFQs6Hnqimg62eZD3z4sb0oU0V7ux//Gj+OOFbK+yYtyou2N9aVyvmc4uzgpR+c6M4GmOd7hiYEe
        PXBESiAaDjdpDHGK4bm6E+5rZV2AFBHc0olxfFonhJ2kvfqjWPno8mIOBofWQlcvj71UZ+2ZKPNh
        SmzG2jbDa0IbJ8WxRoV4oMhR0su2EjM7kFvh3kAVBLBwh378tpogAAAKkBAABQSwECFAAUAAgl
        CACcTUxcd+/LaalAAACpAQAAABAAAAAAAAAAAAAAAAAAAAAAAAAAAdGV4dFBLBQYAAAAAAQAB
        ADIAAADUAAAAAA=
      </return>
    </ns2:importPackagesCV2Response>
  </S:Body>
</S:Envelope>

```

W elemencie return powyższego komunikatu zawarty jest XML w formie PackagesImportResponse skompresowany ZIP a następnie zakodowany jako Base64. Zwracany jest w nim status importu paczek. Postać formatu PackagesImportResponse jest taka sama jak dla odpowiedzi z metody importPackagesXV2 interfejsu DPDCRXmlServices.

3.2.1.5 Usługi dla przesyłek

W serwisie DPDAppServices w ramach zapytań dla interfejsu DPDCRXmlServices jest możliwość wskazania usług związanych z dostarczaniem przesyłek. Usługi wskazywane są w elemencie Services jako nowe elementy. W zależności od tego czy przesyłka będzie dostarczana w obrębie kraju bądź dostarczana za granicą, lista usług będzie się różnić. Część usług wymaga określonych warunków, które muszą wystąpić, aby możliwe było ich użycie. Walidacja części usług może również wykluczać wzajemne ich użycie. Niedozwolone konfiguracje będą skutkowały zwróceniem odpowiedniego komunikatu w odpowiedzi na zapytanie (przykłady w podrozdziale dot. struktury odpowiedzi). W poniższych tabelach przedstawione są dostępne usługi względem tego czy nadanie i odbiór przesyłki występuje na terenie Polski (Serwis krajowy) oraz nadanie lub odbiór przesyłki występuje poza granicami Polski (Serwis międzynarodowy).

3.2.1.5.1 Serwis krajowy

Usługi serwisu krajowego:

- Guarantee – usługa dotyczy gwarancji dostarczenia przesyłki wg konfiguracji usługi. Wskazanie konfiguracji następuje poprzez ustawienie wybranej wartości dla atrybutu type oraz ewentualnych dodatkowych elementów. Dostępne konfiguracje:
 - TIME0930 – gwarancja podjęcia próby doręczenia przesyłki przed godziną 9:30 rano, pierwszego dnia roboczego po dniu nadania przesyłki. W przypadku zamówienia w okresie od 1 listopada do ostatniego dnia lutego, godzina dostarczenia przesyłki jest przesunięta do godziny 10:30, ustawienie poprzez wskazanie wartości TIME0930 dla atrybutu type;
 - TIME1200 – gwarancja podjęcia próby doręczenia przesyłki przed godziną 12:00 pierwszego dnia roboczego po dniu nadania przesyłki, ustawienie poprzez wskazanie wartości TIME1200 dla atrybutu type;
 - TIMEFIXED – gwarancja podjęcia próby dostarczenia przesyłki na określoną przez nadawcę godzinę, pierwszego dnia roboczego po dniu nadania przesyłki. Podana godzina będzie godziną dostarczenia przesyłki z tolerancją +/- 20 minut. W przypadku wybrania godziny dostarczenia w przedziale 10:30 do godziny 12:00, cena usługi dodatkowej jest powiększona o cenę usługi dostarczenia przesyłki przed godziną 12-tą (TIME1200), ustawienie poprzez wskazanie wartości TIMEFIXED dla atrybutu type oraz elementu potomnego Attr1, w którym wskazana jest konkretna godzina w formacie gg:mm. Podana godzina powinna się znajdować w przedziale od 10:30 do godziny 16:00;
 - B2C – usługa udostępniana przez DPD dla wybranych klientów – dla określonych kodów pocztowych możliwe jest wybranie precyzyjnych przedziałów czasowych, w których zagwarantowane będzie dostarczenie paczki, ustawienie poprzez wskazanie wartości B2C dla atrybutu type oraz elementu potomnego Attr1, w którym wskazany jest przedział godzinowy w formacie gg:mm-gg:mm, dostępne przedziały godzinowe to: 10:00-13:00, 13:00-16:00, 16:00-19:00;

- SATURDAY – gwarancja podjęcia próby doręczenia przesyłki w sobotę pomiędzy godziną 8:00 a 17:00, ustawienie poprzez wskazanie wartości SATURDAY dla atrybutu type;
- DPDNEXTDAY – gwarancja podjęcia próby doręczenia przesyłki następnego dnia roboczego po nadaniu, ustawienie poprzez wskazanie wartości DPDNEXTDAY dla atrybutu type;

Przykład użycia:

```
<Services>
  <Guarantee type="TIME0930"/>
</Services>
```

```
<Services>
  <Guarantee type="TIME1200"/>
</Services>
```

```
<Services>
  <Guarantee type="TIMEFIXED">
    <Attr1>15:45</Attr1>
  </Guarantee>
</Services>
```

```
<Services>
  <Guarantee type="B2C">
    <Attr1>10:00-13:00</Attr1>
  </Guarantee>
</Services>
```

```
<Services>
  <Guarantee type="SATURDAY"/>
</Services>
```

```
<Services>
  <Guarantee type="DPDNEXTDAY"/>
</Services>
```

- DeclaredValue – zadeklarowana wartość paczki, zawiera obowiązkowe elementy Amount – kwota podawana w formacie decimal 0.00 oraz Currency – waluta, w przypadku niepodania waluty system przyjmuje domyślną wartość PLN. Dostępne wartości dla waluty:
 - PLN
 - EUR
 - USD

Przykład użycia:

```
<Services>
  <DeclaredValue>
    <Amount>100.5</Amount>
    <Currency>PLN</Currency>
  </DeclaredValue>
</Services>
```

- COD – Płatność przy odbiorze przesyłki. Po dodaniu usługi, odbiorca przesyłki zobowiązany jest zapłacić za przesyłkę według kwoty zadeklarowanej w parametrach usługi. Kwota odebrana przez kuriera jest przekazywana do Zleceniodawcy. Maksymalna kwota jaką kurier może pobrać to 3.000,00 EUR w

walucie PLN obliczanej według średniego kursu ustalonego przez Narodowy Bank Polski ostatniego dnia roboczego poprzedzającego dzień nadanie przesyłki. Gdy mamy do czynienia z kwotą powyżej 1.000,00 PLN konieczne jest pobranie opłaty za ubezpieczenie przesyłki. Kwota ubezpieczenia jest ustalana na podstawie aktualnego cennika standardowego.

Usługa zawiera obowiązkowe elementy Amount – kwota pobrania w formacie decimal 0.00 oraz Currency – waluta, w przypadku niepodania waluty system przyjmuje domyślną wartość PLN. Dostępne wartości dla waluty:

- PLN
- EUR
- USD

W ramach usługi krajowej waluta używana w parametrze Currency musi być ustawiona na wartość „PLN”. Kwota usługi musi się zawierać w przedziale: 0.01-15000. Usługa COD jest dostępna dla Płatnika/Odbiorcy, który nie jest zablokowany dla tej usługi.

Przykład użycia:

```
<Services>
  <COD>
    <Amount>78.35</Amount>
    <Currency>EUR</Currency>
  </COD>
</Services>
```

- CUD – usługa przesyłki zwrotnej. Podczas doręczania, odbiorca przekazuje przesyłkę zwrotną wysyłaną na adres nadawcy przesyłki. Czas realizacji usługi wynosi 2 dni robocze od dnia doręczenia.

Przykład użycia:

```
<Services>
  <CUD/>
</Services>
```

- DOX – przesyłka kopertowa do 0.5 kg, dla tej usługi waga wskazana w elemencie Weight nie może przekroczyć wartości 0.5 kg. Nie można łączyć tej usługi z usługą Tires (Opony) oraz CarryIn (Wniesienie).

Przykład użycia:

```
<Services>
  <DOX/>
</Services>
```

- ROD – usługa polegająca na odebraniu podpisu odbiorcy na dokumencie dołączonym do przesyłki i zwrocie wskazanego dokumentu nadawcy.

Przykład użycia:

```
<Services>
  <ROD/>
</Services>
```

- InPers – doręczenie przesyłki do rąk własnych. Przesyłkę musi odebrać osoba wskazana na etykiecie. Kurier weryfikuje tożsamość na podstawie dokumentu tożsamości np. dowód osobisty. Wymagane jest podanie imienia i nazwiska osoby, dla której wysyłana jest paczka.

Przykład użycia:

```
<Services>
  <InPers/>
```

</Services>

- SelfCol – odbiór własny w oddziale DPD. Odbiór następuje po weryfikacji tożsamości odbiorcy za pomocą dokumentu tożsamości – osoba prywatna lub pieczęć firmowa dla odbiorcy nie będącego osobą fizyczną. W atrybucie receiver wskazywany jest rodzaj odbiorcy, dostępne wartości:

- PRIV – osoba prywatna;
- COMP – firma;

Nie jest możliwe łączenie tej usługi z usługą CarryIn (Wniesienie).

Przykład użycia:

<Services>

<SelfCol receiver="PRIV"/>

</Services>

- PrivPers – doręczenie do osoby prywatnej, na adres prywatny.

Przykład użycia:

<Services>

<PrivPers/>

</Services>

- CarryIn – Wniesienie paczki przez kuriera na miejsce wskazane przez klienta. Usługa udostępniana przez DPD dla wybranych klientów (wybrane numery klienta). Jest to usługa dostępna wyłącznie dla przesyłek składających się z jednej paczki. Waga rzeczywista paczki nie może być poza zakresem 31.5 – 150kg. Waga gabarytowa takiej paczki nie może przekraczać 312kg. Usługa jest dostępna wyłącznie dla wybranych kodów pocztowych odbiorcy.

Nie można łączyć tej usługi z usługą selfCol (Odbiór własny).

Przykład użycia:

<Services>

<CarryIn/>

</Services>

- Tires – przesyłka zawierająca opony.

Nie można łączyć tej usługi z usługą DOX (Przesyłka kopertowa).

Przykład użycia:

<Services>

<Tires/>

</Services>

3.2.1.5.2 Serwis międzynarodowy

Usługi serwisu międzynarodowego:

- Guarantee – usługa dotyczy gwarancji dostarczenia przesyłki wg konfiguracji usługi. Wskazanie konfiguracji następuje poprzez ustawienie wybranej wartości dla atrybutu type oraz ewentualnych dodatkowych elementów. Dostępne konfiguracje:
 - INTER – gwarantowany dzień doręczenia, usługa wyklucza się z usługami Palette (Przesyłka paletowa) oraz TiresExport (Opony międzynarodowe);"

Przykład użycia:

<Services>

<Guarantee type="INTER"/>

</Services>

- DeclaredValue – zadeklarowana wartość paczki, zawiera obowiązkowe elementy Amount – kwota podawana w formacie decimal 0.00 oraz Currency – waluta, w przypadku niepodania waluty system przyjmuje domyślną wartość PLN. Dostępne wartości dla waluty:

- PLN
- EUR
- USD

Przykład użycia:

```
<Services>
  <DeclaredValue>
    <Amount>100.5</Amount>
    <Currency>PLN</Currency>
  </DeclaredValue>
</Services>
```

- COD – płatność przy odbiorze przesyłki. Po dodaniu usługi, odbiorca przesyłki zobowiązany jest zapłacić za przesyłkę według kwoty zadeklarowanej w parametrach usługi. Kwota odebrana przez kuriera jest przekazywana do Zleceniodawcy. Maksymalna kwota jaką kurier może pobrać to 3.000,00 EUR w walucie PLN obliczanej według średniego kursu ustalonego przez Narodowy Bank Polski ostatniego dnia roboczego poprzedzającego dzień nadanie przesyłki. Gdy mamy do czynienia z kwotą powyżej 1.000,00 PLN konieczne jest pobranie opłaty za ubezpieczenie przesyłki. Kwota ubezpieczenia jest ustalana na podstawie aktualnego cennika standardowego.

Usługa zawiera obowiązkowe elementy Amount – kwota pobrania w formacie decimal 0.00 oraz Currency – waluta, w przypadku niepodania waluty system przyjmuje domyślną wartość PLN. Dostępne wartości dla waluty:

- PLN
- EUR
- USD

W ramach usługi krajowej waluta używana w parametrze Currency musi być ustawiona na wartość „PLN”. Usługa dostępna tylko dla odbiorców na terenie PL. Kwota usługi musi się zawierać w przedziale: 0.01-15000. Usługa COD jest dostępna dla Płatnika/Odbiorcy, który nie jest zablokowany dla tej usługi.

Przykład użycia:

```
<Services>
  <COD>
    <Amount>78.35</Amount>
    <Currency>EUR</Currency>
  </COD>
</Services>
```

- CUD – usługa przesyłki zwrotnej. Podczas doręczania, odbiorca przekazuje przesyłkę zwrotną wysyłaną na adres nadawcy przesyłki. Czas realizacji usługi wynosi 2 dni robocze od dnia doręczenia. Usługa dostępna tylko dla odbiorców na terenie PL.

Przykład użycia:

```
<Services>
  <CUD/>
</Services>
```

- DOX – przesyłka kopertowa do 0.5 kg, dla tej usługi waga wskazana w elemencie Weight nie może przekroczyć wartości 0.5 kg. Usługa dostępna tylko dla odbiorców na terenie PL.
Przykład użycia:

```
<Services>
  <DOX/>
</Services>
```
- ROD – usługa polegająca na odebraniu podpisu odbiorcy na dokumencie dołączonym do przesyłki i zwrocie wskazanego dokumentu nadawcy. Usługa dostępna tylko dla odbiorców na terenie PL. Przykład użycia:

```
<Services>
  <ROD/>
</Services>
```
- PrivPers – doręczenie do osoby prywatnej, na adres prywatny. Usługa dostępna tylko dla odbiorców na terenie PL.
Przykład użycia:

```
<Services>
  <PrivPers/>
</Services>
```
- Duty – odprawa celna dla przesyłek wysyłanych poza obszar uznany za Wspólny Obszar Celny Unii Europejskiej. Usługa dostępna dla wybranych krajów i kodów pocztowych.
Przykład użycia:

```
<Services>
  <Duty/>
</Services>
```
- Pallet – usługa umożliwia przesyłanie przesyłek paletowych. Maksymalna dopuszczalna waga paczki to 700 kg. Maksymalna wysokość paczki to 180cm. Maksymalna długość paczki to 120cm. Maksymalna szerokość to 80cm. Podane wymiary (kolejność wymiarów x, y, z, ma znaczenie) muszą być liczbą całkowitą dodatnią.
Nie można łączyć tej usługi z usługami: Guarantee INTER (Gwarant międzynarodowy) oraz TiresExport (Opony międzynarodowe).
Przykład użycia:

```
<Services>
  <Pallet/>
</Services>
```
- TiresExport – przewóz opon jako usługa eksportowa. Nie jest możliwe łączenie tej usługi z usługami Guarantee INTER (Gwarant międzynarodowy) oraz Palette (Przesyłka paletowa).
Przykład użycia:

```
<Services>
  <TiresExport/>
</Services>
```

3.3 Kody błędów

Pełna lista występujących błędów wraz z identyfikatorami i opisem błędu znajduje się w załączniku Kody błędów.

3.4 Uwagi i sugestie

Poziomy dostępu:

Gdy użytkownik będzie próbował wykonać operacje niedostępne dla uprawnień jakie posiada zostanie mu wyświetlony komunikat: „User has no privileges to execute this operation”.

Błąd formatu OpenUMLF:

Istnieje wiele przypadków, dla których może się pojawić błąd formatu OpenUMLF. Najczęściej oznacza to po prostu błędną interpretację składni parametru. Komunikat o treści: „Incorrect OpenUMLFe format: ” poprzedza dokładną treść błędu formatu.

Pole z datą realizacji:

Dla metod importPackagesV2, XV2 oraz CV2 zostało zdefiniowane pole <DeliveryDate>2026-02-11</DeliveryDate> które przyjmuje datę od 1 do 3 dni roboczych do przodu i jest wymagane tylko w przypadku gdy kraj nadawcy to PL. W innych przypadkach pole jest ignorowane.

Błąd nieznany:

W przypadku wewnętrznych problemów usługi możliwe jest pojawienie się błędu o treści „Unknown error”. Dotyczy on wszystkich niesklasyfikowanych błędów występujących w usłudze. W przypadku pojawienia się podobnego błędu konieczny będzie kontakt z usługodawcą.

4 Załączniki

1. Opis OpenUMLF
2. Kody błędów