

Интеграционный гид DPD

Версия: 1.7

Дата выпуска: 18.10.2013

Оглавление

| | |
|---|----|
| Интеграционный гид DPD | 1 |
| Оглавление..... | 2 |
| I. Введение | 6 |
| II. Области интеграции | 7 |
| 1. Запрос географии обслуживания | 7 |
| 2. Расчёт стоимости доставки | 7 |
| 3. Создание заказа на доставку..... | 7 |
| 4. Изменение заказа на доставку..... | 8 |
| 5. Отслеживание статуса доставки..... | 8 |
| 6. Отчеты..... | 8 |
| 7. Печать наклейки | 9 |
| III. Пример схемы взаимодействия..... | 10 |
| IV. Этапы интеграции..... | 12 |
| V. Руководство разработчика | 13 |
| 1. Веб-служба «География DPD» | 13 |
| 1.1. Описание | 13 |
| 1.2. Расположение..... | 13 |
| 1.3. Доступные методы | 13 |
| 1.4. Алгоритм | 13 |
| 1.5. Сообщения | 13 |
| 1.5.1. Параметры входного сообщения..... | 13 |
| 1.5.2. Параметры ответа getCitiesCashPay | 14 |
| 1.5.3. Параметры ответа getTerminalsSelfDelivery | 14 |
| 1.5.4. Параметры входного сообщения getParcelShops | 14 |
| 1.5.5. Параметры ответа getParcelShops..... | 15 |
| 2. Веб-служба «Расчёт стоимости» | 16 |
| 2.1. Описание | 16 |
| 2.2. Расположение..... | 16 |
| 2.3. Доступные методы | 16 |
| 2.4. Алгоритмы..... | 17 |
| 2.4.1. Рассчитать общую стоимость доставки | 17 |
| 2.4.2. Рассчитать стоимость доставки по параметрам посылок..... | 17 |
| 2.5. Сообщения | 17 |
| 2.5.1. Параметры входного сообщения для getServiceCost и getServiceCost2..... | 17 |

| | | |
|--------|--|----|
| 2.5.2. | Параметры входного сообщения для getServiceCostByParcels и getServiceCostByParcels2 | 19 |
| 2.5.3. | Параметры ответа при ошибке | 20 |
| 2.5.4. | Параметры ответа при ошибке (расширенный набор) | 21 |
| 2.5.5. | Параметры ответа при успешном запросе | 22 |
| 2.6. | Типы | 22 |
| 2.6.1. | Описание типа parcel | 22 |
| 3. | Веб-служба «Создание заказа» | 22 |
| 3.1. | Описание | 22 |
| 3.2. | Расположение | 22 |
| 3.3. | Доступные методы | 22 |
| 3.4. | Алгоритмы | 23 |
| 3.4.1. | Создание заказа на доставку | 23 |
| 3.4.2. | Создание адреса с кодом | 23 |
| 3.4.3. | Изменение адреса с кодом | 23 |
| 3.5. | Сообщения | 24 |
| 3.5.1. | Параметры входного сообщения createOrder | 24 |
| 3.5.2. | Параметры входного сообщения getOrderStatus | 26 |
| 3.5.3. | Параметры ответного сообщения createOrder \ getOrdersStatus | 26 |
| 3.5.4. | Параметры входного сообщения createAddress \ updateAddress | 27 |
| 3.5.5. | Параметры ответного сообщения createAddress \ updateAddress | 27 |
| 3.5.6. | Параметры входного сообщения getInvoiceFile | 27 |
| 3.5.7. | Параметры ответного сообщения getInvoiceFile | 28 |
| 3.5.8. | Параметры ответа при ошибке | 28 |
| 3.6. | Типы | 28 |
| 3.6.1. | Описание типа address | 28 |
| 3.6.2. | Описание типа dataInternational | 29 |
| 3.6.3. | Описание типа extraService | 30 |
| 3.6.4. | Описание типа parameter | 30 |
| 3.6.5. | Описание типа parcel | 30 |
| 3.7. | Варианты доставки | 30 |
| 3.8. | Интервалы времени приёма | 30 |
| 3.9. | Интервалы времени доставки | 31 |
| 3.10. | Опции | 31 |
| 3.11. | Статусы создания заказа | 32 |
| 3.12. | Статусы создания и изменения адреса | 33 |

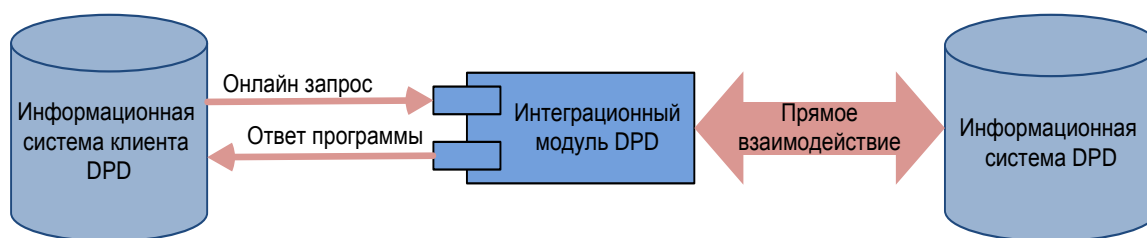
| | |
|--|----|
| 4. Веб-служба «Изменение заказа» | 33 |
| 4.1. Описание | 33 |
| 4.2. Расположение | 33 |
| 4.3. Доступные методы | 33 |
| 4.4. Алгоритмы | 33 |
| 4.4.1. Изменение заказа с добавлением посылок | 33 |
| 4.4.2. Изменение заказа с удалением посылок | 34 |
| 4.5. Сообщения | 34 |
| 4.5.1. Параметры входного сообщения addParcels \ removeParcels | 34 |
| 4.5.2. Параметры ответного сообщения для addParcels \ removeParcels | 34 |
| 4.6. Типы | 35 |
| 4.6.1. Описание типа parcel | 35 |
| 4.6.2. Описание типа parcelStatus | 35 |
| 4.7. Статусы | 35 |
| 4.7.1. Статусы изменения заказа | 35 |
| 4.7.2. Статусы добавления \ удаления посылки | 35 |
| 5. Веб-служба «Отслеживание статуса» | 36 |
| 5.1. Описание | 36 |
| 5.2. Расположение | 36 |
| 5.3. Доступные методы | 36 |
| 5.4. Алгоритмы | 37 |
| 5.4.1. Алгоритм работы с методом getStatesByClient | 37 |
| 5.4.2. Алгоритм работы с методами getStatesByClientOrder, getStatesByDPDOrder, getStatesByClientParcel | 37 |
| 5.5. Входящие сообщения | 38 |
| 5.5.1. getStatesByClient | 38 |
| 5.5.2. confirm | 38 |
| 5.5.3. getStatesByClientOrder | 38 |
| 5.5.4. getStatesByClientParcel | 39 |
| 5.5.5. getStatesByDPDOrder | 39 |
| 5.6. Исходящие сообщения | 39 |
| 5.6.1. getStatesByClientResponse | 39 |
| 5.6.2. confirmResponse | 41 |
| 5.6.3. getStatesByClientOrderResponse | 41 |
| 5.6.4. getStatesByClientParcelResponse | 42 |
| 5.6.5. getStatesByDPDOrderResponse | 42 |

| | | |
|--------|---|----|
| 5.7. | Состояния | 42 |
| 5.7.1. | Посылка | 42 |
| 6. | Веб-служба «Отчеты» | 42 |
| 6.1. | Описание | 42 |
| 6.2. | Расположение | 42 |
| 6.3. | Доступные методы | 42 |
| 6.4. | Алгоритмы | 43 |
| 6.5. | Сообщения | 43 |
| 6.5.1. | Параметры входного сообщения метода | 43 |
| 6.5.2. | Параметры ответного сообщения метода getNLAmount | 43 |
| 6.5.3. | Параметры ответного сообщения метода getNLInvoice | 43 |
| 7. | Веб-служба «Печать Наклейки» | 43 |
| 7.1. | Описание | 43 |
| 7.2. | Расположение | 43 |
| 7.3. | Доступные методы | 44 |
| 7.4. | Алгоритмы | 44 |
| 7.4.1. | Формат файла | 44 |
| 7.4.2. | Формат области печати | 44 |
| 7.5. | Входящие сообщения | 44 |
| 7.5.1. | createLabelFile | 44 |
| 7.6. | Исходящие сообщения | 45 |
| 7.6.1. | createLabelFileResponse | 45 |
| VI. | Справочник ошибок | 48 |
| 1. | Общие ошибки | 48 |
| 2. | Ошибки службы «Расчёт стоимости» | 48 |
| 3. | Ошибки службы «Создание заказа» | 48 |
| 4. | Ошибки службы «Отслеживание статуса» | 49 |
| 5. | Ошибки службы «Создание адреса с кодом» | 49 |
| 6. | Ошибки службы «Печать Наклейки» | 49 |
| VII. | Интеграционные примеры | 50 |
| 1. | 1С. Трейдинг | 50 |
| 2. | PHP. Расчет стоимости | 52 |
| 3. | Java. Создание заказа | 57 |
| 4. | PHP. Создание заказа и печать наклейки | 58 |
| VIII. | Контакты | 63 |

Введение

Компания DPD предоставляет ряд стандартных решений для интеграции своей информационной системы с информационными системами клиентов. Такого рода интеграция позволяет нашим компаниям обмениваться необходимыми данными в реальном времени и исключает обработку данных вручную, что значительно ускоряет и оптимизирует бизнес-процесс. Мы предлагаем воспользоваться простым и быстрым доступом к функциям нашей системы, которые включают в себя расчет стоимости доставки, отслеживание посылок, создание заказов на доставку и получение информации о географии обслуживания.

Базовый принцип взаимодействия представлен на следующем рисунке:



Интеграционный модуль DPD представляет собой набор веб-служб, построенных на базе технологии SOAP и развернутые на сервере приложений Tomcat.

Подробнее про веб-службы и про технологию SOAP можно прочитать, например, здесь:

<http://www.webmascon.com/topics/technologies/8a.asp>

I. Области интеграции

На данный момент существуют семь областей интеграции с информационной системой DPD:

1. Запрос географии обслуживания
2. Расчёт стоимости доставки
3. Создание заказа на доставку
4. Изменение заказа
5. Отслеживание статуса доставки
6. Отчеты
7. Печать Наклейки
8. Отмена заказа

1. Запрос географии обслуживания

Запрос географии обслуживания подразумевает получение информации о географии доставки DPD в реальном времени. Эта область интеграции поддерживает два метода:

- Получение списка городов, в которых расположены терминалы DPD.
- Получение списка городов, в которые осуществляется доставка с наложенным платежом.

2. Расчёт стоимости доставки

Интеграция в области расчёта стоимости нужна для того, чтобы ваша информационная система в любой момент могла получить актуальную стоимость доставки и при необходимости предоставить эту стоимость своему клиенту. Входящие в состав области методы позволяют:

- Выполнить расчёт стоимости доставки по выбранному направлению и услуге. Это актуально, если вы постоянно пользуетесь одной услугой для доставки. Для расчета стоимости необходимо выбирать лишь направление.
- Получить список доступных услуг и их стоимость для выбранного направления.
- Выполнить более точный расчет стоимости доставки на основе данных посылок. Это актуально, если вы точно знаете количество, габариты и веса ваших посылок.

3. Создание заказа на доставку

Создание заказа на доставку с помощью интеграционного модуля позволяет максимально быстро передать в нашу систему необходимую информацию для формирования отправки. Например, как только пользователь интернет-магазина подтверждает свой заказ, информационная система магазина генерирует онлайн запрос, в котором передает все необходимые сведения: дату вызова курьера, адрес приёма груза, адрес доставки и т.п. Интеграционный модуль DPD примет заявку на создание заказа и в ответном запросе сообщит статус создания заказа, ошибки создания заказа (если заказ не создан) и номер заказа в информационной системе DPD (если заказ создан корректно).

Статус создания заказа может быть следующим: заказ создан корректно, заказ отклонен из-за ошибок, или заказ принят на доработку. Последний статус означает, что мы не смогли автоматически принять заказ (например, потому что наша система не смогла распознать адрес

доставки), и ваш запрос будет обработан нашим сотрудником вручную, который уточнит и поправит всю необходимую информацию и завершит процесс создания заказа.

Данная область поддерживает методы:

- Создание заказов на доставку.
- Проверка статуса создания заказа и получение его номера. Этот метод необходим для тех случаев, когда заказ не был создан автоматически и был позднее доработан сотрудником DPD.
- Создание адресов с кодом. Данный метод необходим в случае, если есть адрес, который часто используется в заказе и вместо него проще передавать код адреса.
- Изменение адрес с кодом. Данный метод необходим в случае, когда закрепленный за кодом адрес изменился.

4. Изменение заказа на доставку

Изменение заказа на доставку с помощью интеграционного модуля позволяет максимально оперативно передать в нашу систему изменившуюся информацию для корректировки данных заказа. Изменения данных заказа возможно до момента получения груза на терминале компании.

Статус изменения заказ может быть следующим: заказ изменен успешно, состояние заказа не позволяет выполнять его корректировку. Последний статус означает, что груз уже был получен от клиента компанией.

Данная область поддерживает два метода:

- Создание заказов на доставку.
- Проверка статуса создания заказа и получение его номера. Этот метод необходим для тех случаев, когда заказ не был создан автоматически и был позднее доработан сотрудником DPD.

5. Отслеживание статуса доставки

Область отслеживания статуса доставки предоставляет две основные возможности: вы можете позволить своему клиенту отслеживать статус его доставки в реальном времени, и вы можете в любое время получать актуальные сведения о статусе всех своих доставок.

Данная область поддерживает три метода:

- Получение статусов ваших посылок, изменившихся с момента предыдущего запроса. Необходимо указать ваш клиентский номер.
- Получение статусов посылок по указанному заказу. Информация предоставляется как по номеру вашего заказа, так и по номеру заказа DPD.
- Получение статусов конкретной посылки. Информация предоставляется по номеру/штрих-коду вашей посылки.

6. Отчеты

Данная область позволяет получать необходимые Вам отчеты по запросу.

Входящие в состав области методы позволяют получить следующую отчеты:

- О предварительной стоимости перевозки отправок за период времени.
- Об окончательной стоимости перевозки отправок за период времени с номерами счетов.

7. Печать наклейки

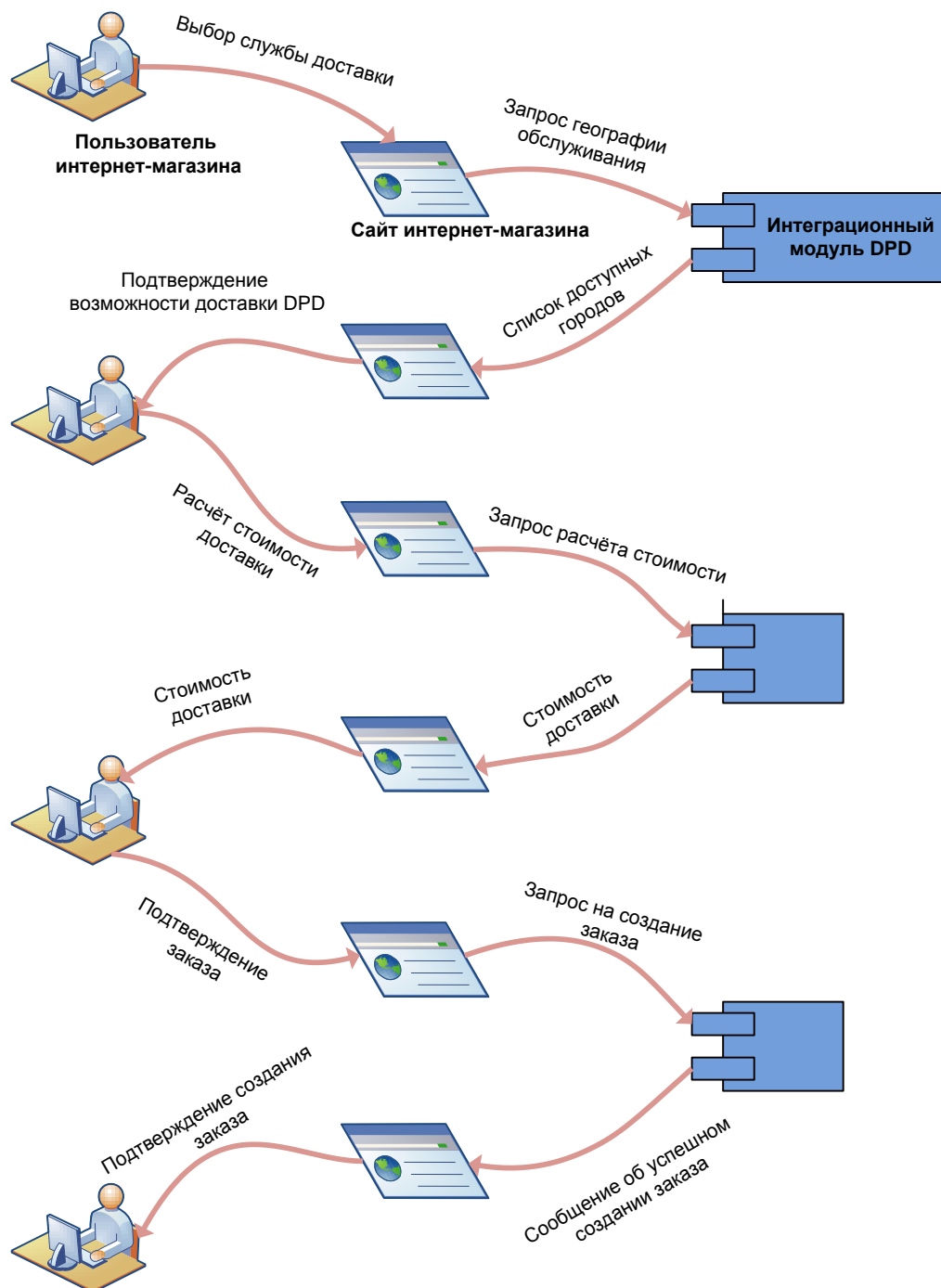
Данная область предоставляет вам возможность посредством API формировать наклейки DPD по ранее созданному заказу.

8. Отмена заказа

Данная область предоставляет вам возможность посредством API отменять заказы без звонка в Call центр DPD.

II. Пример схемы взаимодействия

Следующая схема описывает взаимодействие с использованием всех областей интеграции на примере клиента - интернет-магазина.





III. Этапы интеграции

Для запуска интеграционного процесса с DPD вам нужно совершить следующие действия:

1. Передать данное руководство в ваш IT-департамент.
2. Разработать программу-обработчик для взаимодействия с выбранными веб-службами.
3. Получить уникальный ключ клиента для аутентификации в интеграционном модуле DPD.
Для получения данного ключа вам нужно иметь клиентский номер DPD (номер вашего договора с DPD). Вы можете получить ключ у своего менеджера DPD или обратившись на адрес нашей информационной поддержки: itcustomers@dpd.ru.
4. Протестировать работу веб-служб совместно с сотрудниками информационной поддержки DPD, используя тестовый сервер DPD. Адрес тестового сервера приведен ниже в руководстве разработчика.
5. Настроить вашу программу на промышленный сервер DPD и начать работать в реальном режиме. Адрес промышленного сервера приведен ниже в руководстве разработчика.

IV. Руководство разработчика

В данном разделе дано подробное описание методов имеющихся в веб-службе.

В разделе «[Интеграционные примеры](#)» приведены различные примеры интеграции.

1. Веб-служба «География DPD»

1.1. Описание

Веб-служба «География DPD» предназначена для получения актуального списка городов, в которые возможна доставка наложенным платежом, списка терминалов DPD и списка пунктов выдачи посылок (ПВП).

1.2. Расположение

Тестовая версия сервиса доступна по адресу:

<http://wstest.dpd.ru/services/geography?wsdl>

Промышленная версия сервиса доступна по адресу:

<http://ws.dpd.ru/services/geography?wsdl>

1.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|--------------------------|---|--|---|
| getCitiesCashPay | Получить список городов с возможностью доставки с наложенным платежом | Параметры входного сообщения | Параметры ответа getCitiesCashPay |
| getTerminalsSelfDelivery | Получить список терминалов DPD | Параметры входного сообщения | Параметры ответа getTerminalsSelfDelivery |
| getParcelShops | Получить список пунктов выдачи посылок (ПВП) | Параметры входного сообщения | Параметры ответа |

1.4. Алгоритм

1. Подготавливаем данные для аутентификации входного запроса и вызываем нужный метод.
2. Обрабатываем результат в соответствии со спецификацией ответных сообщений выбранного метода.

1.5. Сообщения

1.5.1. Параметры входного сообщения

| Параметр | Описание | Тип | Обязательный | Пример |
|--------------|--|--------|--------------|-------------|
| Auth | | | | |
| clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Строка | Да | 1000000000 |
| clientKey | Ваш уникальный ключ для авторизации, | Строка | Да | 1FD890C3556 |

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|-----------------------------|-----|--------------|--------|
| | полученный у сотрудника DPD | | | |

1.5.2. Параметры ответа getCitiesCashPay

| Параметр | | Описание | Тип | Пример |
|----------|-------------|--|--------|-----------------|
| city | | Массив городов с поддержкой доставки с наложенным платежом | | |
| | cityId | Идентификатор города | Число | 49694102 |
| | countryCode | Код страны | Строка | RU |
| | countryName | Страна | Строка | Россия |
| | regionCode | Код региона | Строка | 50 |
| | regionName | Регион | Строка | Московская обл. |
| | cityName | Город | Строка | Люберцы |

1.5.3. Параметры ответа getTerminalsSelfDelivery

| Параметр | | Описание | Тип | Пример |
|----------|-------------------|--|--------|---|
| terminal | | Массив терминалов DPD | | |
| | terminalCode | Код терминала DPD | Строка | LED |
| | terminalName | Название терминала DPD | Строка | Санкт-Петербург |
| | terminalAddresses | Адрес терминала DPD | Строка | пр. Обуховской Обороны, д.295, лит БД 192012 г. Санкт-Петербург |
| city | | Информация о городе, в котором расположен терминал DPD | | |
| | cityId | Идентификатор города | Число | 49694103 |
| | countryCode | Код страны | Строка | RU |
| | countryName | Страна | Строка | Россия |
| | regionCode | Код региона | Строка | 78 |
| | regionName | Регион | Строка | г. Санкт-Петербург |
| | cityName | Город | Строка | Санкт-Петербург |

1.5.4. Параметры входного сообщения getParcelShops

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|--------------|--|--------|--------------|-------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Строка | Да | 1000000000 |
| | clientKey | Ваш уникальный | Строка | Да | 1FD890C3556 |

| Параметр | | Описание | Тип | Обязательный | Пример |
|-------------|--|--|---|--------------|----------|
| | | ключ для авторизации, полученный у сотрудника DPD | | | |
| countryCode | | Код страны | Строка | Нет | RU |
| regionCode | | Код региона | Строка | Нет | 77 |
| cityCode | | Код города | | | |
| cityName | | Наименование города | Строка | Нет | Москва |
| serviceCode | | Список кодов услуг DPD. Если параметр задан, то сервис возвращает сроки бесплатного хранения по заданным у. Если он не задан, сервис не возвращает сроки бесплатного хранения. | Строка Список кодов услуг через запятую | Нет | BZP, ECN |

ПРИМЕЧАНИЯ. 1. Поля countryCode, regionCode и cityCode задаются для фильтрации списка. Если коды не заданы возвращается полный список. Поле cityName анализируется только если не задан cityCode.

2. Сроки бесплатного хранения зависят от услуги DPD и одинаковы для всех пунктов. Для получения сроков бесплатного хранения необходимо задать поле serviceCode.

1.5.5. Параметры ответа getParcelShops

| Параметр | | | Описание | Тип | Пример |
|------------|----------------|---------------|---|--------|---|
| parcelShop | | | Массив ПВП | | |
| | code | | Код подразделения DPD | Строка | LED |
| | parcelShopType | | Тип подразделения: Т–терминал, СД–сити-депо, ПВП–отдельный пункт выдачи посылок | Строка | ПВП |
| | address | | Адрес пункта | | |
| | | countryCode | Код страны | Строка | RU |
| | | regionCode | Код региона | Строка | 78 |
| | | regionName | Регион | Строка | г. Санкт-Петербург |
| | | cityCode | Код города | Строка | 78000000001 |
| | | cityName | Город | Строка | Санкт-Петербург |
| | | addressString | Строка с адресом | Строка | пр. Обуховской Обороны, д.295, лит БД 192012 г. Санкт-Петербург |

| Параметр | | | Описание | Тип | Пример |
|---------------|----------------|-----------|---|--------|----------------|
| | geoCoordinates | | Географические координаты по карте Яндекс | | |
| | | geoX | | Число | 53.322927 |
| | | geoY | | Число | 83.638803 |
| | limits | | Ограничения параметров посылки | | |
| | | maxWeight | Макс. вес в кг | Число | 30.5 |
| | | maxLength | Макс. длина в см | Число | 70 |
| | | maxWidth | Макс. ширина в см | Число | 70 |
| | | maxHeight | Макс. высота в см | Число | 50 |
| | workingTime | | Массив расписаний работы по дням | | |
| | | weekDays | Список дней недели | Строка | пн,вт,ср,чт,пт |
| | | workTime | | Строка | 09:00–20:00 |
| storagePeriod | | | Массив сроков бесплатного хранения | | |
| | serviceCode | | Код услуги | Строка | CSM |
| | days | | Количество дней | Число | 14 |

2. Веб-служба «Расчёт стоимости»

2.1. Описание

Веб-служба «Расчёт стоимости» предназначена для расчёта стоимости доставки по тарифам DPD.

2.2. Расположение

Тестовая версия сервиса доступна по адресу:

<http://wstest.dpd.ru/services/calculator2?wsdl>

Промышленная версия сервиса доступна по адресу:

<http://ws.dpd.ru/services/calculator2?wsdl>

2.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|-------------------------|---|--|---|
| getServiceCost | Рассчитать общую стоимость доставки | Параметры входного сообщения | Параметры ответа при ошибке / Параметры ответа при успешном запросе |
| getServiceCostByParcels | Рассчитать стоимость доставки по параметрам посылок | Параметры входного сообщения | Параметры ответа при ошибке / Параметры ответа при успешном запросе |
| getServiceCost2 | Рассчитать общую стоимость доставки (с расширенным | Параметры входного сообщения | Параметры ответа при ошибке / Параметры ответа при успешном |

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|--------------------------|---|--|---|
| | набором параметров при ошибке) | | запросе |
| getServiceCostByParcels2 | Рассчитать стоимость доставки по параметрам посылок (с расширенным набором параметров при ошибке) | Параметры входного сообщения | Параметры ответа при ошибке / Параметры ответа при успешном запросе |

2.4. Алгоритмы

2.4.1. Рассчитать общую стоимость доставки

1. Подготавливаем параметры вызова в соответствии с [описанием входного сообщения](#) метода **getServiceCost**: пункт отправления и назначения, дату приёма груза, объявленную ценность. Если вы работаете с DPD только по одной услуге, рекомендуется указать код этой услуги во входном сообщении.
2. Вызываем метод **getServiceCost**.
3. Получаем ответное сообщение с кодом ошибки или с массивом доступных услуг для выбранной доставки и их стоимости. Если услуга была задана во входном сообщении, то при успешном выполнении запроса массив ответного сообщения будет состоять из одного элемента.
4. Если вы получили код ошибки *TooManyRows*, то проанализируйте массивы *pickupDups* и *DeliveryDups* во входном сообщении. Выберите ID нужного населенного пункта и повторите запрос.

2.4.2. Рассчитать стоимость доставки по параметрам посылок

1. Подготавливаем параметры вызова в соответствии с [описанием входного сообщения](#) метода **getServiceCostByParcels**: пункт отправления и назначения, дату приёма груза, объявленную ценность. Если вы работаете с DPD только по одной услуге, рекомендуется указать код этой услуги во входном сообщении.
2. Вызываем метод **getServiceCostByParcels**.
3. Получаем ответное сообщение с кодом ошибки или с массивом доступных услуг для выбранной доставки и их стоимости. Если услуга была задана во входном сообщении, то при успешном выполнении запроса массив ответного сообщения будет состоять из одного элемента.
4. Если вы получили код ошибки *TooManyRows*, то проанализируйте массивы *pickupDups* и *DeliveryDups* во входном сообщении. Выберите ID нужного населенного пункта и повторите запрос.

2.5. Сообщения

2.5.1. Параметры входного сообщения для getServiceCost и getServiceCost2

| Параметр | Описание | Тип | Обязательный | Пример |
|--------------|--|--------|--------------|-------------|
| auth | | | | |
| clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Строка | Да | 1000000000 |
| clientKey | Ваш уникальный | Строка | Да | 1FD890C3556 |

| Параметр | | Описание | Тип | Обязательный | Пример |
|--------------|-------------|--|--|--------------|------------|
| | | ключ для авторизации, полученный у сотрудника DPD | | | |
| pickup | | | | | |
| | cityId | Идентификатор города отправления | Число | Нет | 49694102 |
| | cityName | Город отправления | Строка | Нет | Москва |
| | regionCode | Код региона отправления | Строка | Нет | 77 |
| | countryCode | Код страны отправления | Строка | Нет | RU |
| delivery | | | | | |
| | cityId | Идентификатор города доставки | Число | Нет | 49265227 |
| | cityName | Город доставки | Строка | Нет | Челябинск |
| | regionCode | Код региона доставки | Строка | Нет | 74 |
| | countryCode | Код страны доставки | Строка | Нет | RU |
| selfPickup | | Самопривоз на терминал. | boolean | Да | false |
| selfDelivery | | Доставка до терминала. Самовывоз с терминала. | boolean | Да | true |
| weight | | Вес отправки, кг | Число | Да | 5 |
| volume | | Объём, м ³ | Число | Нет | 0.05 |
| serviceCode | | Список кодов услуг DPD. Если параметр задан, то сервис возвращает стоимость только заданных услуг. Если он не задан – всех доступных услуг. | Строка Список кодов услуг через запятую | Нет | BZP,ECN |
| pickupDate | | Предполагаемая дата приёма груза. Стоимость будет считаться на заданную дату. Если параметр не задан – будет считаться на текущую дату. | Дата | Нет | 2014-02-07 |
| maxDays | | Максимально допустимый срок. Если параметр задан, то все услуги с большим сроком не будут показываться. Параметр не имеет смысла, если задавать конкретную | Целое число | Нет | 2 |

| Параметр | | Описание | Тип | Обязательный | Пример |
|---------------|--|---|-------|--------------|--------|
| | | услугу. | | | |
| maxCost | | Максимально допустимая стоимость. Если параметр задан, то все услуги с большей стоимостью не будут показываться. Параметр не имеет смысла, если задавать конкретную услугу. | Число | Нет | 1000 |
| declaredValue | | Объявленная ценность груза | Число | Нет | 1000 |

Важно: нужно обязательно задать либо cityID, либо cityName как в случае города отправления, так и в случае города доставки.

2.5.2. Параметры входного сообщения для getServiceCostByParcels и getServiceCostByParcels2

| Параметр | | Описание | Тип | Обязательный | Пример |
|--------------|--------------|--|---------|--------------|-------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Строка | Да | 1000000000 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| pickup | | | | | |
| | cityId | Идентификатор города отправления | Число | Нет | 49694102 |
| | cityName | Город отправления | Строка | Нет | Москва |
| | regionCode | Код региона отправления | Строка | Нет | 77 |
| | countryCode | Код страны отправления | Строка | Нет | RU |
| delivery | | | | | |
| | cityId | Идентификатор города доставки | Число | Нет | 49265227 |
| | cityName | Город доставки | Строка | Нет | Челябинск |
| | regionCode | Код региона доставки | Строка | Нет | 74 |
| | countryCode | Код страны доставки | Строка | Нет | RU |
| selfPickup | | Доставка от терминала | boolean | Да | false |
| selfDelivery | | Доставка до терминала | boolean | Да | true |

| Параметр | | Описание | Тип | Обязательный | Пример |
|---------------|--|---|--|--------------|------------|
| serviceCode | | Список кодов услуг DPD. Если параметр задан, то сервис возвращает стоимость только заданных услуг. Если он не задан – всех доступных услуг. | Строка Список кодов услуг через запятую | Нет | BZP, ECN |
| pickupDate | | Предполагаемая дата приёма груза. Стоимость будет считаться на заданную дату. Если параметр не задан – будет считаться на текущую дату. | Дата | Нет | 2014-02-07 |
| maxDays | | Максимально допустимый срок. Если параметр задан, то все услуги с большим сроком не будут показываться. Параметр не имеет смысла, если задавать конкретную услугу. | Целое число | Нет | 2 |
| maxCost | | Максимально допустимая стоимость. Если параметр задан, то все услуги с большей стоимостью не будут показываться. Параметр не имеет смысла, если задавать конкретную услугу. | Число | Нет | 1000 |
| declaredValue | | Объявленная ценность груза | Число | Нет | 1000 |
| parcel | | Список посылок с параметрами для расчета | parcel | Да | |

2.5.3. Параметры ответа при ошибке

| Параметр | | Описание | Тип | Пример |
|----------|--|---------------------|--------|--------------------|
| code | | Код ошибки | Строка | too-many-rows |
| message | | Сообщение об ошибке | Строка | Неоднозначно задан |

| Параметр | | Описание | Тип | Пример |
|--------------|-------------|---------------------------------------|--------|------------------|
| | | | | населенный пункт |
| deliveryDups | | Массив дублирующихся городов доставки | | |
| | cityId | Идентификатор города | Число | 195981361 |
| | countryCode | Код страны | Строка | RU |
| | countryName | Страна | Строка | Россия |
| | regionCode | Код региона | Строка | 37 |
| | regionName | Регион | Строка | Ивановская обл. |
| | cityName | Город | Строка | Иваново |
| pickupDups | | Массив дублирующихся городов забора | | |
| | cityId | Идентификатор города | Число | 195638871 |
| | countryCode | Код страны | Строка | RU |
| | countryName | Страна | Строка | Россия |
| | regionCode | Код региона | Строка | 50 |
| | regionName | Регион | Строка | Московская обл. |
| | cityName | Наименование города | Строка | Пушкино |

2.5.4. Параметры ответа при ошибке (расширенный набор)

| Параметр | | Описание | Тип | Пример |
|--------------|--------------|---------------------------------------|--------|-------------------------------------|
| code | | Код ошибки | Строка | too-many-rows |
| message | | Сообщение об ошибке | Строка | Неоднозначно задан населенный пункт |
| deliveryDups | | Массив дублирующихся городов доставки | | |
| | cityId | Идентификатор города | Число | 195901073 |
| | countryCode | Код страны | Строка | RU |
| | countryName | Страна | Строка | Россия |
| | regionCode | Код региона | Строка | 71 |
| | regionName | Регион | Строка | Тульская |
| | cityCode | Код населенного пункта | Строка | RU71017001000 |
| | cityName | Населенный пункт | Строка | Новомосковск |
| | abbreviation | Аббревиатура | Строка | г |
| | indexMin | Минимальный индекс | Строка | 301650 |
| | indexMax | Максимальный индекс | Строка | 301650 |
| pickupDups | | Массив дублирующихся городов забора | | |
| | cityId | Идентификатор города | Число | 232636063 |
| | countryCode | Код страны | Строка | RU |
| | countryName | Страна | Строка | Россия |
| | regionCode | Код региона | Строка | 50 |
| | regionName | Регион | Строка | Московская |
| | cityCode | Код населенного пункта | Строка | RU50026000002 |
| | cityName | Населенный пункт | Строка | Каменка |
| | abbreviation | Аббревиатура | Строка | д |

| Параметр | Описание | Тип | Пример |
|----------|---------------------|--------|--------|
| indexMin | Минимальный индекс | Строка | 142167 |
| indexMax | Максимальный индекс | Строка | 142167 |

2.5.5. Параметры ответа при успешном запросе

| Параметр | Описание | Тип | Пример |
|-------------|---------------------|----------------|----------------|
| serviceCode | Код услуги DPD | Строка | ECN |
| serviceName | Название услуги | Строка | DPD ECONOMY |
| cost | Стоимость услуги | Число | 2,651.46 |
| days | Срок доставки, дней | Целое число | 2 |

2.6. Типы

2.6.1. Описание типа parcel

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|--------------------|-------|--------------|--------|
| weight | Вес посылки | Число | Да | 0.5 |
| length | Длина посылки | Число | Да | 0.5 |
| width | Ширина посылки | Число | Да | 0.2 |
| height | Высота посылки | Число | Да | 0.2 |
| quantity | Количество посылок | Число | Нет | 3 |

3. Веб-служба «Создание заказа»

3.1. Описание

Веб-служба «Создание заказа» предназначена для создания заказов на доставку в информационной системе DPD.

3.2. Расположение

Тестовая версия сервиса доступна по адресу:

<http://wstest.dpd.ru/services/order2?wsdl>

Промышленная версия сервиса доступна по адресу:

<http://ws.dpd.ru/services/order2?wsdl>

3.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|----------------|---------------------------------|---|---|
| createOrder | Создать заказ на доставку | Параметры входного сообщения CreateOrder | Параметры ответа при ошибке / Параметры ответного сообщения |
| getOrderStatus | Получить статус создания заказа | Параметры входного сообщения getOrderStatus | Параметры ответа при ошибке / Параметры ответного сообщения |
| createAddress | Создать адрес с кодом | Параметры | Параметры ответа |

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|----------------|--|--|---|
| | | входного сообщения | при ошибке / Параметры ответа при успешном запросе |
| updateAddress | Изменить адреса с кодом | Параметры входного сообщения | Параметры ответа при ошибке / Параметры ответа при успешном запросе |
| getInvoiceFile | Получить файл накладной (формат файла – PDF) | Параметры входного сообщения | Параметры ответа при ошибке/ Параметры ответа при успешном запросе |

3.4. Алгоритмы

3.4.1. Создание заказа на доставку

1. Подготавливаем параметры вызова: массив заказов, объединенных общим адресом приёма груза и датой приема отправки, в соответствии с [описанием входного сообщения](#) метода **createOrder**.
2. Вызываем метод **createOrder**.
3. Получаем ответное сообщение с результатом выполнения функции по каждому заказу.
4. Анализируем статус создания заказа (см. [«Статусы создания заказа»](#)) и при необходимости – сообщение об ошибке.
5. Если статус заказа – *OrderPending*, то через какое-то время можно вызвать метод **getOrderStatus**, чтобы убедиться в том, что заказ доработан и создан в информационной системе DPD.

3.4.2. Создание адреса с кодом

1. Подготавливаем параметры вызова в соответствии с [описанием входного сообщения](#) метода **createAddress**.
2. Вызываем метод **createAddress**.
3. Получаем ответное сообщение с результатом создания адреса.
4. Анализируем статус создания адреса (см. [«Статусы создания и изменения адреса»](#)) и при необходимости – сообщение об ошибке.
5. Если статус адреса - *address-error*, необходимо внести изменения в параметры и повторно вызвать метод.

3.4.3. Изменение адреса с кодом

1. Подготавливаем параметры вызова в соответствии с [описанием входного сообщения](#) метода **updateAddress**.
2. Вызываем метод **updateAddress**.
3. Получаем ответное сообщение с результатом создания адреса.
4. Анализируем статус создания адреса (см. [«Статусы создания и изменения адреса»](#)) и при необходимости – сообщение об ошибке.
 - a. Если статус адреса - *code-not-found* – в системе отсутствует адрес с таким кодом, соответственно для создания адреса необходимо воспользоваться методом **createAddress**.
 - b. Если статус адреса - *address-error*, необходимо внести изменения в параметры или добавить недостающие и повторно вызвать метод.

3.5. Сообщения

3.5.1. Параметры входного сообщения createOrder

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|---------------------|--|-------------------------|--------------|-------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| header | | Информация, содержащаяся в этом параметре, является общей для всех заказов в запросе | | | |
| | datePickup | Дата приёма груза | Дата | Да | |
| | payer | Клиентский номер плательщика в системе DPD (номер договора с DPD). Если этот параметр не заполнен, то плательщиком будет считаться заказчик (номер из параметра auth). | Число | Нет | |
| | senderAddress | Адрес приёма груза | address | Да | |
| | pickupTimePeriod | Интервал времени приёма груза. Доступные для выбора интервалы приёма см. в разделе «Интервалы времени приёма» . | Строка | Нет | 9-18 |
| | regularNum | Номер регулярного заказа DPD. Если вы используете доставку на регулярной основе, уточните этот номер у своего менеджера. | Строка | Нет | 1000 |
| order | | Массив данных, относящихся к каждому конкретному заказу | | | |
| | orderNumberInternal | Номер заказа в информационной системе клиента | Строка | Да | 123456 |
| | serviceCode | Код услуги DPD. | Строка | Да | CUR |

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|--------------------|---|-------------|--------------|--------|
| | | Уточните код нужной Вам услуги у своего менеджера или используйте код услуги, полученный из веб-сервиса «Калькулятор стоимости» | | | |
| | serviceVariant | Вариант доставки. Доступно 4 варианта: ДД, ДТ, ТД и ТТ. Расшифровку вариантов см. в разделе «Варианты доставки» . | Строка | Да | ДД |
| | cargoNumPack | Количество грузомест (посылок) в отправке | Целое число | Да | 5 |
| | cargoWeight | Вес отправки, кг | Число | Да | 5 |
| | cargoVolume | Объём, м ³ | Число | Нет | 0.05 |
| | cargoRegistered | <p>Ценный груз. Внутреннее вложение, включенное в перечень товаров, требующих дополнительных мер безопасности, снижающих риск его утери или повреждения при перевозке.</p> <p>Перечень товаров, относимых к категории «Ценный груз»:</p> <ol style="list-style-type: none"> 1. Мобильные телефоны 2. Ноутбуки, планшеты | boolean | Да | false |
| | cargoValue | Сумма объявленной ценности, руб. | Число | Нет | 1000 |
| | cargoCategory | Содержимое отправки | Строка | Да | Одежда |
| | deliveryTimePeriod | Интервал времени доставки груза. Доступные для выбора интервалы доставки см. в разделе «Интервалы времени доставки» . | Строка | Нет | 9-18 |

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|-----------------|---|-----------------------------------|--------------|--------|
| | extraParam | Зарезервированный параметр для ввода новых параметров без изменения схемы сервиса | parameter | Нет | |
| | dataInt | Данные для международных отправок | dataInternational | Нет | |
| | receiverAddress | Адрес доставки | address | Да | |
| | extraService | Массив опций доставки | extraService | Нет | |
| | parcel | Массив посылок отправки | parcel | Нет | |

3.5.2. Параметры входного сообщения getOrderStatus

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|---------------------|---|--------|--------------|-------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| order | | Информация о заказе, статус которого вы проверяете | | | |
| | orderNumberInternal | Номер заказа в информационной системе клиента | Строка | Да | 123456 |
| | datePickup | Дата приёма груза (на тот случай, если номер в вашей информационной системе не является уникальным) | Дата | Нет | |

3.5.3. Параметры ответного сообщения createOrder \ getOrderStatus

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|---------------------|--|--------|--------------|-------------|
| | orderNumberInternal | Номер заказа в информационной системе клиента | Строка | Да | 123456 |
| | orderNum | Номер заказа DPD. Возвращается в ответном сообщении. | Строка | Нет | 01010001MOW |
| | status | Статус создания | Строка | Да | OK |

| Параметр | Описание | Тип | Обязательный | Пример |
|--------------|--|--------|--------------|------------------------------|
| | заказа. Возвращается в ответном сообщении. Возможные статусы перечислены в разделе «Статусы создания заказа» . | | | |
| errorMessage | Текст ошибки | Строка | Нет | Не заполнен параметр «Улица» |

3.5.4. Параметры входного сообщения createAddress \ updateAddress

| Параметр | Описание | Тип | Обязательный | Пример |
|---------------|--|-------------------------|--------------|-------------|
| auth | | | | |
| clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| clientAddress | Адрес | address | Да | |

3.5.5. Параметры ответного сообщения createAddress \ updateAddress

| Параметр | Описание | Тип | Обязательный | Пример |
|--------------|---|--------|--------------|------------------|
| code | Код адреса | Строка | Да | X333 |
| status | Статус. Возможные статусы описаны в «Ошибки службы «Создания адреса с кодом»» | Строка | Да | address-error |
| errorMessage | Описание ошибки | Строка | Нет | Не указана улица |

3.5.6. Параметры входного сообщения getInvoiceFile

| Параметр | Описание | Тип | Обязательный | Пример |
|--------------|--|--------|--------------|-------------|
| auth | | | | |
| clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| clientKey | Ваш уникальный ключ для авторизации, полученный у | Строка | Да | 1FD890C3556 |

| Параметр | Описание | Тип | Обязательный | Пример |
|-------------|----------------------------------|--------|--------------|-----------------|
| | сотрудника DPD | | | |
| orderNum | Номер заказа DPD | Строка | Да | 01010001 MOW |
| parcelCount | Количество мест в заказе | Число | Нет | 3 |
| cargoValue | Сумма объявленной ценности, руб. | Число | Нет | 1000 |

ПРИМЕЧАНИЕ. Если не заданы parcelCount или cargoValue, то при формировании файла выводятся параметры из заказа.

3.5.7. Параметры ответного сообщения getInvoiceFile

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|----------|---------------|--------------|--------|
| file | Файл | Массив байтов | Да | |

3.5.8. Параметры ответа при ошибке

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|--|--------|--------------|-------------------|
| code | Код ошибки. Возможные коды описаны в « Ошибки службы «создания заказа» ». | Строка | Да | no-data-found |
| message | Описание ошибки | Строка | Нет | Данные не найдены |

3.6. Типы

3.6.1. Описание типа address

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|---|--------|-----------------|------------------------------|
| code | Код адреса в информационных системах заказчика и DPD. Адрес с кодом должен быть передан в DPD отдельно. | Строка | Нет | 1234 |
| name | Название отправителя/получателя. В случае, когда адрес приёма/доставки – это магазин, филиал компании, дилерский центр и т.п., в эту строку пишется его название. Если доставка | Строка | Да ¹ | Иванов Сергей Петрович |

¹ Данное поле не является обязательным при условии, что заполнено поле **code**

| Параметр | Описание | Тип | Обязательный | Пример |
|----------------|---|--------|------------------|-------------------------------------|
| | осуществляется физическому лицу, то пишется Ф.И.О получателя. | | | |
| terminalCode | Код терминала. | Строка | Да ² | M91 |
| addressString | Строка адреса для международных отправок | Строка | Нет | 16, Grafton Street, Dublin, Ireland |
| countryName | Название страны | Строка | Нет | Россия |
| index | Индекс | Строка | Нет | 140012 |
| region | Регион | Строка | Нет | Московская обл. |
| city | Город | Строка | Да ³ | Люберцы |
| street | Улица | Строка | Да ³ | Авиаторов |
| streetAbbr | Сокращения типа улицы (ул, пр-т, б-р и т.д.) | Строка | Да ³ | ул |
| house | Дом | Строка | Да ³ | 1 |
| houseKorpus | Корпус | Строка | Нет | A |
| str | Строение | Строка | Нет | 1 |
| vlad | Владение | Строка | Нет ⁴ | 1 |
| extraInfo | Доп. информация | Строка | Нет | Пав. 1 |
| office | Офис | Строка | Нет | 12 |
| flat | Квартира | Строка | Нет | 144 |
| workTimeFrom | Время работы от | Строка | Нет | 9:00 |
| workTimeTo | Время работы до | Строка | Нет | 21:00 |
| dinnerTimeFrom | Время обеда от | Строка | Нет | 14:00 |
| dinnerTimeTo | Время обеда до | Строка | Нет | 15:00 |
| contactFio | Контактное лицо | Строка | Да ⁵ | Смирнов Игорь Николаевич |
| contactPhone | Контактный телефон | Строка | Да ⁷ | 89165555555 |
| contactEmail | Контактный e-mail | Строка | Нет | smirnov@megashop.ru |
| instructions | Инструкции для курьера | Строка | Нет | Подъезд со стороны ул. Кирова |

3.6.2. Описание типа dataInternational

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|----------|-----|--------------|--------|
|----------|----------|-----|--------------|--------|

² Данное поле является обязательным для вариантов перевозки «ДТ», «ТД» или «ТТ» в рамках создания заказа. При простом создании адреса (не в рамках заказа) не должно указываться.

³ Данное поле не является обязательным при условии, что заполнено поле **code** или поле **vlad**.

⁴ Данное поле не является обязательным при условии, что заполнено поле **code** или поле **house**.

⁵ Данное поле не является обязательным при условии, что заполнено поле **code**

| Параметр | Описание | Тип | Обязательный | Пример |
|-----------------------|-------------------------------|--------|--------------|--------|
| currency | Валюта | Строка | Нет | USD |
| currencyDeclaredValue | Объявленная ценность в валюте | Строка | Нет | 100 |

3.6.3. Описание типа extraService

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|---|---------------------------|--------------|--------|
| esCode | Код опции. См описания опций в разделе «Опции». | Строка | Да | SMS |
| param | Параметр опции. | parameter | Нет | |

3.6.4. Описание типа parameter

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|--------------------|--------|--------------|------------|
| name | Имя параметра | Строка | Да | phone |
| value | Значение параметра | Строка | Да | 8916555555 |

3.6.5. Описание типа parcel

| Параметр | Описание | Тип | Обязательный | Пример |
|------------------|---|--------|--------------|------------|
| number | Номер посылки в информационной системе клиента (номер штрих-кода посылки) | Строка | Да | 1001000998 |
| weight | Вес посылки | Число | Нет | 0.5 |
| length | Длина посылки | Число | Нет | 0.5 |
| width | Ширина посылки | Число | Нет | 0.2 |
| height | Высота посылки | Число | Нет | 0.2 |
| insuranceCost | Стоимость посылки | Число | Нет | 100 |
| insuranceCostVat | НДС стоимости посылки | Число | Нет | 18 |
| codAmount | Сумма оплаты содержимого получателем | Число | Нет | 128 |

3.7. Варианты доставки

Доступны следующие варианты:

- ДД – от двери отправителя до двери получателя;
- ДТ – от двери отправителя до терминала DPD;
- ТД – от терминала DPD до двери получателя;
- ТТ – от терминала DPD до терминала DPD.

3.8. Интервалы времени приёма

Доступны следующие варианты выбора времени приёма груза:

- 9-18 – в любое время с 09:00 до 18:00 (вариант по умолчанию);
- 9-13 – с 09:00 до 13:00;

- 13-18 – с 13:00 до 18:00.

3.9. Интервалы времени доставки

Доступны следующие варианты выбора времени доставки груза:

- 9-18 – в любое время с 09:00 до 18:00 (вариант по умолчанию);
- 9-13 – с 09:00 до 13:00;
- 13-18 – с 13:00 до 18:00;
- 18-21 – с 18:00 до 21:00. (внимание, данный интервал оплачивается дополнительно!)

3.10. Опции

| Код опции | Описание опции | Опция платная | Имя параметра для передачи сопутствующих данных | Значение параметра |
|-----------|--|---------------|---|---|
| SMS | SMS уведомление получателя о приёме посылки у отправителя и о выходе посылки на доставку | Нет | phone | Телефон получателя для отправки уведомления |
| EML | E-mail уведомление получателя о приёме посылки у отправителя и о выходе посылки на доставку | Нет | email | E-mail получателя для отправки уведомления |
| ЭСД | Электронное сообщение о доставке груза получателю | Нет | email | E-mail, на который нужно отправить сообщение о доставке груза |
| ЭСЗ | Электронное сообщение о приёме заказа | Нет | email | E-mail, на который нужно отправить сообщение о приёме заказа |
| ВДО | Возврат документов отправителю. Если клиенту необходимо вернуть сопроводительные документы на груз (товарную накладную, акты приема-передачи), заверенные получателем. | Да | - | |
| ДВД | Доставка в выходные дни. Эта опция, позволяющая осуществить доставку отправки получателю в соответствии со сроком доставки ТК, но раньше даты, рассчитанной по стандартным правилам, при условии, что итоговая дата доставки придется на выходной день. | Да | - | |

| Код опции | Описание опции | Опция платная | Имя параметра для передачи сопутствующих данных | Значение параметра |
|-----------|--|---------------|---|--|
| НПП | Наложенный платеж | Да | sum_npp | сумма наложенного платежа ¹ |
| ПОД | Подтверждение о доставке | Да | email/fax | Е-mail или факс, на который нужно отправить подтверждение о доставке |
| ПРД | Погрузо-разгрузочные работы при доставке. Если необходимо организовать разгрузку и доставку поступивших посылок в то или иное помещение на территории, независимо от того, на каком этаже оно расположено. | Да | - | |
| ОЦ | Объявленная ценность отправки. Для обеспечения дополнительных гарантий сохранности отправок и для увеличения размера компенсации. | Да | - | |
| ТРМ | Температурный режим | Да | - | |
| ОЖД | Ожидание на адресе | Да | reason_delay | причина ожидания на адресе из Справочника причин ожидания на адресе: ВНЕШ Проверка на внешние повреждения ПРИМ Примерка РАБТ Проверка работоспособности СООТ Проверка на соответствие |

1. Указывать обязательно, при условии, что сумма за Наложенный платеж будет отличаться от суммы Объявленной стоимости.

3.11. Статусы создания заказа

В ответном сообщении о создании заказа может быть возвращен один из следующих статусов:

- OK – заказ на доставку успешно создан с номером, указанным в поле orderNum.
- OrderPending – заказ на доставку принят, но нуждается в ручной доработке сотрудником DPD, (например, по причине того, что адрес доставки не распознан автоматически). Номер заказа будет присвоен ему, когда это доработка будет произведена.
- OrderError – заказ на доставку не может быть создан по причине, указанной в поле errorMessage.

3.12. Статусы создания и изменения адреса

В ответном сообщении о результатах создания адреса может быть возвращен один из следующих статусов:

- OK – адрес с кодом создан.
- code-already-exists – адрес с указанным кодом уже существует.
- address-error – адрес не может быть создан. Причина при этом указывается в поле message.
- code-not-found- адрес с указанным кодом не найден.

4. Веб-служба «Изменение заказа»

4.1. Описание

Веб-служба «Изменения заказа» предназначена для внесения изменений в уже созданный заказ на доставку в информационной системе DPD. При этом внесение изменений в заказ возможно до физического получения груза на терминале DPD.

4.2. Расположение

Тестовая версия сервиса доступна по адресу:

<http://wstest.dpd.ru/services/order2?wsdl>

Промышленная версия сервиса доступна по адресу:

<http://ws.dpd.ru/services/order2?wsdl>

4.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|---------------|--------------------------------------|--|-------------------------------|
| addParcels | Изменить заказ с добавлением посылок | Параметры входного сообщения addParcels | Параметры ответного сообщения |
| removeParcels | Изменить заказ с удалением посылок | Параметры входного сообщения removeParcels | Параметры ответного сообщения |

4.4. Алгоритмы

4.4.1. Изменение заказа с добавлением посылок

1. Подготавливаем параметры вызова: массив данных изменяемых заказов и список добавляемых посылок в соответствии с описанием входного сообщения метода **addParcels**.
2. Вызываем метод **addParcels**.

3. Получаем ответное сообщение с результатом выполнения функции по каждому заказу.

4.4.2. Изменение заказа с удалением посылок

1. Подготавливаем параметры вызова: массив данных изменяемых заказов и список удаляемых посылок в соответствии с описанием входного сообщения метода **removeParcels**.
2. Вызываем метод **removeParcels**.
3. Получаем ответное сообщение с результатом выполнения функции по каждому заказу.

4.5. Сообщения

4.5.1. Параметры входного сообщения addParcels \ removeParcels

| Параметр | Описание | Тип | Обязательный | Пример |
|---------------------|--|------------------------|--------------|-------------|
| auth | | | | |
| clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| orderNum | Номер заказа | Строка | Да | 01010001MOW |
| orderNumberInternal | Номер заказа в информационной системе клиента | Строка | Нет | 123456 |
| cargoNumPack | Количество грузомест (посылок) в отправке | Целое число | Да | 5 |
| cargoWeight | Вес отправки, кг | Число | Да | 5 |
| cargoVolume | Объем, м ³ | Число | Нет | 0.05 |
| cargoValue | Сумма объявленной ценности, руб. | Число | Нет | 1000 |
| cargoCategory | Содержимое отправки | Строка | Да | Одежда |
| parcel | Посылки для добавления в заказ | parcel | | |

4.5.2. Параметры ответного сообщения для addParcels \ removeParcels

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|--|--------|--------------|-------------|
| orderNum | Номер заказа DPD. Возвращается в ответном сообщении. | Строка | Нет | 01010001MOW |
| status | Статус изменения заказа. Возвращается в ответном сообщении. Возможные статусы перечислены в разделе « Статусы изменения заказа » | Строка | Да | OK |

| Параметр | Описание | Тип | Обязательный | Пример |
|--------------|----------------|------------------------------|--------------|--------|
| parcelStatus | Список посылок | parcelStatus | Нет | |

4.6. Типы

4.6.1. Описание типа parcel

| Параметр | Описание | Тип | Обязательный | Пример |
|------------------|---|--------|--------------|--------|
| number | Номер посылки в информационной системе клиента (номер штрих-кода посылки) | Строка | Да | WS121 |
| weight | Вес посылки | Число | Нет | 0.5 |
| length | Длина посылки | Число | Нет | 0.5 |
| width | Ширина посылки | Число | Нет | 0.2 |
| height | Высота посылки | Число | Нет | 0.2 |
| insuranceCost | Стоимость посылки | Число | Нет | 100 |
| insuranceCostVat | НДС стоимости посылки | Число | Нет | 18 |
| codAmount | Сумма оплаты содержимого получателем | Число | Нет | 128 |

4.6.2. Описание типа parcelStatus

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|--|--------|--------------|--------|
| number | Номер посылки | Строка | Да | WS121 |
| status | Статус выполнения операции. Возможные статусы перечислены в разделе « Статусы добавления \ удаления посылки ». | Строка | Да | OK |

4.7. Статусы

4.7.1. Статусы изменения заказа

В ответном сообщении об изменении заказа может быть возвращен один из следующих статусов:

- OK – заказ успешно изменен.
- illegal-state - состояние заказа не позволяет выполнять его корректировку.
- no-data-found - состояние заказа не позволяет выполнять его корректировку.

4.7.2. Статусы добавления \ удаления посылки

В ответном сообщении об изменении состава посылок в заказе может быть возвращен один из следующих статусов:

- no-data-found - при удалении, посылка с указанным номером в заказе не найдена.
- too-many-rows - при удалении, найдено более одной посылки с указанным номером.
- illegal-state - состояние заказа не позволяет выполнять добавление или удаление посылки.

5. Веб-служба «Отслеживание статуса»

5.1. Описание

Веб-служба «Отслеживание статуса» предоставляет API для отслеживания состояния посылок клиента DPD. Можно отслеживать изменения всех статусов посылок своего клиентского номера, либо запрашивать историю состояний по конкретному заказу или посылке.

Для использования данной веб – службы Вам необходимо обратиться в it_customers и запросить одну из настроек – «с подтверждением» или «без подтверждения».

5.2. Расположение

Тестовая версия сервиса доступна по адресам:

<http://wstest.dpd.ru:80/services/tracing?wsdl>

Промышленная версия сервиса доступна по адресам:

<http://ws.dpd.ru:80/services/tracing?wsdl>

5.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Исходящее сообщение |
|-------------------------|---|---|---|
| getStatesByClient | Получить все состояния посылок клиента, изменившиеся с момента последнего вызова данного метода | getStatesByClient | getStatesByClientResponse |
| Confirm | Подтвердить получение состояний, переданных методом getStatesByClient | confirm | confirmResponse |
| getStatesByClientOrder | Получить историю состояний всех посылок заданного заказа. Заказ идентифицируется по номеру заказа в информационной системе клиента. | getStatesByClientOrder | getStatesByClientOrderResponse |
| getStatesByClientParcel | Получить историю состояний заданной посылки. Посылка идентифицируется по номеру посылки в информационной системе клиента. | getStatesByClientParcel | getStatesByClientParcelResponse |
| getStatesByDPDOrder | Получить историю состояний всех посылок заданного заказа. Заказ идентифицируется по | getStatesByDPDOrder | getStatesByDPDOrderResponse |

| Метод | Назначение | Входящее сообщение | Исходящее сообщение |
|-------|---|--------------------|---------------------|
| | номеру заказа в информационной системе DPD. | | |

5.4. Алгоритмы

5.4.1. Алгоритм работы с методом `getStatesByClient`

С методом `getStatesByClient` можно работать двумя способами: *с подтверждением* и *без подтверждения*. Вам необходимо сообщить желаемый метод сотруднику DPD при подключении сервиса. По умолчанию используется метод работы *с подтверждением*.

Если вы работаете *с подтверждением*, то метод **`getStatesByClient`** не выдаст вам новых состояний, пока не получит от вас подтверждения прочтения переданных состояний через метод **`confirm`**. Если наша система не получит такого подтверждения, то при следующем запросе она выдаст те же состояния, что и в предыдущем. Таким образом, гарантируется защита от сбоя при передаче данных - если сбой произошел, данные не потеряются, а будут переданы при повторном запросе. Вызов метода **`confirm`** говорит нам о том, что процесс передачи данных прошел успешно, и все данные закачаны в информационную систему клиента.

Если вы работаете *без подтверждения*, то метод **`getStatesByClient`** будет всегда выдавать только новые состояния. При формировании новой порции состояний система учитывает только то, были ли состояния уже переданы через веб-сервис и не принимает в расчёт то, были ли они загружены в систему клиента.

Алгоритм работы:

1. Подготавливаем параметры аутентификации.
2. Вызываем метод **`getStatesByClient`**.
3. Получаем ответное сообщение с идентификатором документа и массивом состояний посылок.
4. Если выбран способ работы *с подтверждением* - вызываем метод **`confirm`**, передавая в качестве параметра полученный идентификатор документа.

5.4.2. Алгоритм работы с методами `getStatesByClientOrder`, `getStatesByDPDOrder`, `getStatesByClientParcel`

Данные методы позволяют получить полную историю состояний нужного заказа или посылки. По запрошенному заказу или посылке передаются все состояния, независимо от того, были ли они уже переданы или нет. Никаких подтверждений не требуется.

Алгоритм работы:

1. Подготавливаем параметры аутентификации и номер для идентификации заказа или посылки.
2. Вызываем нужный метод.
3. Получаем ответное сообщение с массивом состояний посылок.
4. Чтобы узнать текущее состояние посылки, выбираем последнее по параметру `transitionTime`.

7.5. Входящие сообщения

5.5.1. getStatesByClient

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|--------------|--|--------|--------------|-------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |

5.5.2. confirm

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|--------------|---|--------|--------------|-----------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 0 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C35 56 |
| docId | | Идентификатор документа, полученного в запросе getStatesByClient. Все сообщения, переданные по данному документу, получают статус «Получение подтверждено». | Число | Да | 12346897 |

5.5.3. getStatesByClientOrder

| Параметр | | Описание | Тип | Обязательный | Пример |
|---------------|--------------|---|--------|--------------|-------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| clientOrderNr | | Номер заказа в информационной системе клиента | Строка | Да | 12346DPD |
| pickupDate | | Дата приёма груза (на случай, если номер заказа не уникален, и требуется уточнение по дате) | Дата | Нет | 2014-02-28 |

5.5.4. getStatesByClientParcel

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------------|--------------|--|--------|--------------|-------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| clientParcelNr | | Номер посылки в информационной системе клиента | Строка | Да | 12346897 |
| pickupDate | | Дата приёма груза (на случай, если номер посылки не уникален, и требуется уточнение по дате) | Дата | Нет | 2014-02-28 |

5.5.5. getStatesByDPDOrder

| Параметр | | Описание | Тип | Обязательный | Пример |
|------------|--------------|--|-------------|--------------|-----------------|
| auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 0 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C35 56 |
| dpdOrderNr | | Номер заказа в информационной системе DPD | Строка | Да | 04040001 MOW |
| pickupYear | | Год заказа (т.к. номера заказов DPD уникальные в пределах года, требуется уточнение, чтобы получить однозначный результат) | Целое число | Нет | 2012 |

5.6. Исходящие сообщения

5.6.1. getStatesByClientResponse

| Параметр | Описание | Тип | Пример |
|----------|---|-------|----------|
| docId | Идентификатор документа. Данный идентификатор | Число | 12346897 |

| Параметр | | Описание | Тип | Пример |
|----------------|---------------------|--|---------|-----------------|
| | | используется для подтверждения получения статусов | | |
| docDate | | Дата формирования документа | Дата | 2014-02-28 |
| clientNumber | | Ваш клиентский номер в системе DPD | Число | 1000000000 |
| resultComplete | | Показывает, выбраны ли в текущем запросе все новые состояния по клиенту (значение true), или был достигнут лимит записей в одном запросе и для продолжения необходим ещё один запрос (значение false). | boolean | true |
| states | | Массив состояний посылок | | |
| | clientOrderNr | Номер заказа в информационной системе клиента | Строка | 12346DPD |
| | clientParcelNr | Номер посылки в информационной системе клиента | Строка | 12346897 |
| | dpdOrderNr | Номер заказа в информационной системе DPD | Строка | 04040001MO W |
| | dpdParcelNr | Номер посылки в информационной системе DPD | Строка | 12346897 |
| | pickupDate | Дата приёма груза | Дата | 2014-02-28 |
| | dpdOrderReNr | Номер повторного заказа в системе DPD (заполняется в том случае, если по одному и тому же клиентскому номеру посылки в системе DPD существует два заказа – например, при заказе на возврат посылки) | Строка | 04040002MO W |
| | dpdParcelReNr | Номер посылки при повторном заказе в системе DPD (заполняется в том случае, если по одному и тому же клиентскому номеру посылки в системе DPD существует два заказа – например, при заказе на возврат посылки) | Строка | 12346899 |
| | planDeliveryDate | Планируемая дата доставки посылки | Дата | 2014-03-01 |
| | orderPhysicalWeight | Физический вес (кг) отправки | Число | 2.08 |
| | orderVolume | Объем (м ³) отправки | Число | 0.023 |

| Параметр | Описание | Тип | Пример |
|----------------------|--|------------|------------------------|
| orderVolumeWeight | Объемный вес (кг.)отправки | Число | 4.6 |
| orderPayWeight | Платный вес (кг.) отправки | Число | 5 |
| orderCost | Объявленная ценность | Число | 1200 |
| parcelPhysicalWeight | Физический вес (кг.) | Число | 1.5 |
| parcelVolume | Объем (м ³) | Число | 0.008 |
| parcelVolumeWeight | Объемный вес (кг.) посылки | Число | 1.5 |
| parcelPayWeight | Платный вес (кг.) посылки | Число | 1.5 |
| parcelLength | Длина (см.) посылки | Число | 157 |
| parcelWidth | Ширина (см.) посылки | Число | 7 |
| parcelHeight | Высота (см.) посылки | Число | 7 |
| newState | Состояние посылки после перехода. См. список возможных состояний в разделе «Состояния» пункт « Посылка ». | Строка | Delivering |
| transitionTime | Время перехода состояния | Дата/время | 2012-04-04T17:10:15 |
| terminalCode | Код терминала DPD, на котором произошел переход состояния | Строка | LED |
| terminalCity | Город терминала DPD, на котором произошел переход состояния | Строка | LED |
| incidentCode | Код инцидента, произошедшего при переходе состояния. Список возможных кодов инцидентов и их расшифровок вы можете получить у своего менеджера. | Строка | 90 |
| incidentName | Наименование инцидента, произошедшего при переходе состояния | Строка | Возвращено отправителю |
| consignee | Фактический получатель посылки (передается только со статусом Delivered) | Строка | Иванов И.И. |

5.6.2. confirmResponse

Структура сообщения аналогична входящему «[confirm](#)».

5.6.3. getStatesByClientOrderResponse

Параметры сообщения совпадают с параметрами ответного сообщения «[getStatesByClientResponse](#)».

5.6.4. [getStatesByClientParcelResponse](#)

Параметры сообщения совпадают с параметрами ответного сообщения «[getStatesByClientResponse](#)».

5.6.5. [getStatesByDPDOrderResponse](#)

Параметры сообщения совпадают с параметрами ответного сообщения «[getStatesByClientResponse](#)».

5.7. Состояния

5.7.1. Псылка

Посылка может принимать одно из следующих состояний:

- NewOrderByClient – оформлен новый заказ по инициативе клиента
- NotDone– заказ отменен
- OnTerminalPickup– посылка находится на терминале приема отправления
- OnRoad– посылка находится в пути (внутренняя перевозка DPD)
- OnTerminal– посылка находится на транзитном терминале
- OnTerminalDelivery– посылка находится на терминале доставки
- Delivering – посылка выведена на доставку
- Delivered – посылка доставлена получателю
- Lost – посылка утеряна
- Problem– с посылкой возникла проблемная ситуация
- ReturnedFromDelivery– посылка возвращена с доставки
- NewOrderByDPD – оформлен новый заказ по инициативе DPD

6. Веб-служба «Отчеты»

6.1. Описание

Веб-служба «Отчеты» предоставляет возможность получать отчеты из информационной системы компании.

6.2. Расположение

Тестовая версия сервиса доступна по адресу:

<http://wstest.dpd.ru/services/nl?wsdl>

Промышленная версия сервиса доступна по адресу:

<http://ws.dpd.ru/services/nl?wsdl>

6.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|--------------|--|--|---|
| getNLAmount | Отчет о предварительной стоимости перевозки за указанный период. | Параметры входного сообщения | Параметры ответного сообщения |
| getNLInvoice | Отчет об окончательной стоимости перевозки за | Параметры входного | Параметры ответного |

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|-------|-----------------|---------------------------|---------------------------|
| | указный период. | сообщения | сообщения |

6.4. Алгоритмы

6.5. Сообщения

6.5.1. Параметры входного сообщения метода

| Параметр | Описание | Тип | Обязательный | Пример |
|--------------|--|--------|--------------|-------------|
| auth | | | | |
| clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| dateFrom | Начальная дата периода отчета | Дата | Да | 12.11.2012 |
| dateTo | Конечная дата периода отчета | Дата | Да | 16.11.2012 |

6.5.2. Параметры ответного сообщения метода getNLAmount

| Параметр | Описание | Тип | Обязательный | Пример |
|----------|---------------------|--------|--------------|--------|
| amount | Стоимость перевозки | Число | Нет | 100 |
| ordernum | Номер заказа | Строка | Нет | |

6.5.3. Параметры ответного сообщения метода getNLInvoice

| Параметр | Описание | Тип | Обязательный | Пример |
|------------|--------------|--------|--------------|-------------|
| amount | Сумма счета | Число | Нет | 100 |
| invoicenum | Номер счета | Строка | Нет | |
| ordernum | Номер заказа | Строка | Нет | 02200119MOW |

7. Веб-служба «Печать Наклейки»

7.1. Описание

Веб-служба «Печать Наклейки» предоставляет API для формирования наклейки DPD по ранее созданному заказу.

7.2. Расположение

Тестовая версия сервиса доступна по адресу:
<http://wstest.dpd.ru/services/label-print?wsdl>

Промышленная версия сервиса доступна по адресу:
<http://ws.dpd.ru/services/label-print?wsdl>

7.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|-----------------|----------------------------------|---------------------------------|---|
| createLabelFile | Формирует файл с наклейками DPD. | createLabelFile | createLabelFileResponse |

7.4. Алгоритмы

Перед вызовом метода в ИС DPD заказы должны быть сформированы, им присвоены номера DPD (например, 05120002MOW). Далее необходимо в метод передать список номеров заказов DPD и для каждого заказа количество необходимых наклеек (как правило, оно равно количеству мест в заказе). В ответ сервис вернет файл с наклейками DPD.

7.4.1. Формат файла

Возвращаемый файл имеет один из двух форматов, который задается во входящем сообщении:

- **PDF**;
- **FP3** – формат генератора отчетов FastReport.

7.4.2. Формат области печати

Размер наклейки – А6 (четвертушка стандартного листа А4). Возвращаемый файл формируется в одном из двух вариантов расположения наклеек на листах бумаги: в одну колонку (для узких принтеров) и две наклейки на ширине листа. Входящее сообщение содержит строковое поле Формат области печати, которое может содержать значения:

- **А6** – в одну колонку;
- **А5** – в две колонки.

Примечание. Если задан формат А5, то для каждого заказа первая наклейка печатается в левой части листа. То есть, если заказы включают по одной посылке, то наклейки разместятся в одну колонку.

7.5. Входящие сообщения

7.5.1. createLabelFile

| Параметр | | Описание | Тип | Обязательный | Пример |
|------------|--------------|--|--------|--------------|-------------|
| Auth | | | | | |
| | clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| | clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| fileFormat | | Формат файла. Возможные значения: PDF, FP3 | Срока | Да | PDF |
| pageSize | | Формат области печати. | | Да | A5 |

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|---------------|---|--------|--------------|-------------|
| | | Возможные значения: A5, A6. | | | |
| order | | Массив данных, относящихся к каждому конкретному заказу | | | |
| | orderNum | Номер заказа DPD | Строка | Да | 05120002MOW |
| | parcelsNumber | Кол-во наклеек для формирования | Число | Да | 3 |

7.6. Исходящие сообщения

7.6.1. createLabelFileResponse

| Параметр | | Описание | Тип | Обязательный | Пример |
|----------|--------------|---|---------------|--------------|------------------------------|
| file | | Файл | Массив байтов | | |
| order | | Массив данных, относящихся к каждому конкретному заказу | | | |
| | orderNum | Номер заказа DPD | Строка | | 05120002MOW |
| | status | Статус | Строка | | OrderPending |
| | errorMessage | Сообщение об ошибке | Строка | | Не заполнен параметр «Улица» |

8. Веб-служба «Отмена заказа»

8.1. Описание

Веб-служба «Отмена заказа» предоставляет API клиентам сервис, который позволяет без звонка в call-центр DPD:

- Отменять оформленные заказы
- Инициировать возврат груза уже переданного курьеру

8.2. Расположение

Тестовая версия сервиса доступна по адресу:

<http://wstest.dpd.ru/services/order2?wsdl>

Промышленная версия сервиса доступна по адресу:

<http://ws.dpd.ru/services/order2?wsdl>

8.3. Доступные методы

| Метод | Назначение | Входящее сообщение | Ответное сообщение |
|-------------|---|------------------------------------|-------------------------------------|
| CancelOrder | Отмена заказа. Создание заказа на возврат | Входящие сообщения | Исходящие сообщения |

8.4. Алгоритмы

Перед вызовом метода в ИС DPD должна быть зарегистрирована заявка на ввод заказа или заказы должны быть сформированы, им присвоены номера DPD (например, 05120002MOW). Заказ не должен иметь признака «возврат». Заявка на ввод заказа считается разовым заказом, если нет номера регулярного заказа, иначе – фактическим регулярным заказом. Далее необходимо в метод передать список номеров заказов из системы клиента; дату приема заказа (не обязательно); номер клиента.

Условие успешной отмены заказа – не забранный груз. Условие успешного оформления возврата отправления – груз забран и нет незавершенного звонка в Call-центр.

Далее система вернет статус результата операции «отмена заказа».

8.5. Входящие сообщения

| Параметр | Описание | Тип | Обязательный | Пример |
|---------------------|--|--------|--------------|-------------|
| Auth | | | | |
| clientNumber | Ваш клиентский номер в системе DPD (номер вашего договора с DPD) | Число | Да | 1000000000 |
| clientKey | Ваш уникальный ключ для авторизации, полученный у сотрудника DPD | Строка | Да | 1FD890C3556 |
| Cancel | | | | |
| orderNumberInternal | Номер заказа в информационной системе клиента | Строка | Да* | 123456 |
| orderNum | Номер заказа DPD | Строка | Да* | 05120002MOW |
| pickupdate | Дата приема груза | Дата | Нет | 2014-12-05 |

*Обязательно должен быть указан один из параметров «Номер заказа клиента» или «Номер заказа DPD»

8.6. Параметры ответного сообщения

| Параметр | Описание | Тип | Обязательный | Пример |
|---------------------|---|--------|--------------|-------------|
| orderNumberInternal | Список посылок | Строка | Нет | |
| orderNum | Номер заказа DPD. Возвращается в ответном сообщении. | Строка | Нет | 01010001MOW |
| status | Статус изменения заказа. Возвращается в ответном сообщении. Возможные статусы перечислены в разделе « Статусы » | Строка | Да | Cancelled |

| <u>Параметр</u> | <u>Описание</u> | <u>Тип</u> | <u>Обязательный</u> | <u>Пример</u> |
|-----------------|---|------------|---------------------|---------------|
| errorMessage | Текст сообщения об ошибке | Строка | | |

8.7. Статусы

| Код Status | Описание |
|----------------------|---|
| Cancelled | Операция выполнена успешно |
| Cancelled_previously | Отменено ранее |
| Error | Текст сообщения об ошибке |

8.8. Текст сообщений об ошибке

Отмена заказа невозможна, возникла ошибка: <Описание проблемы>

Описание проблемы:

- Получены не все данные на вход
- Полученные данные некорректны
- Не найден заказ \ заявка \ отправка

V. Справочник ошибок

1. Общие ошибки

| Код ошибки | Текст ошибки | Примечание |
|------------------|--|--|
| system-error | Системная ошибка | Не типизированная ошибка. Обратитесь на itcustomers@dpd.ru . |
| too-many-calls | Превышен лимит одновременных вызовов сервиса | На сервере DPD установлено ограничение на частоту вызовов сервиса с одного клиентского номера. Попробуйте уменьшить частоту вызова сервиса. |
| call-client-twin | Повторный вызов сервиса по одному коду клиента | На сервере DPD установлено ограничение на количество одновременных вызовов сервиса. Попробуйте уменьшить частоту вызова сервиса. |
| client-num-error | Не найден номер клиента | На сервере DPD установлено ограничение, не допускающее более одного вызова сервиса с одного клиентского номера. Попробуйте уменьшить частоту вызова сервиса. |
| auth-error | Ошибка в параметрах аутентификации | Уникальный ключ клиента не соответствует номеру клиента. Обратитесь на itcustomers@dpd.ru . |
| required-value | Не заданы значения обязательных полей | Проверьте алгоритм заполнения всех обязательных полей входного сообщения. |
| error-value | Недопустимое значение поля | Проверьте алгоритм заполнения полей, имеющих ограниченный набор значений («Услуга», «Вариант доставки» и пр.) |

2. Ошибки службы «Расчёт стоимости»

| Код ошибки | Текст ошибки | Примечание |
|----------------------|--|---|
| no-service-available | Невозможна услуга, удовлетворяющая запросу | Доставка груза по данной услуге невозможна. Попробуйте задать другое направление, услугу или дату доставки. |
| no-data-found | Данные не найдены | По заданным параметрам не найден пункт отправления и пункт назначения. |
| too-many-rows | Найдено более одной записи данных | По заданным параметрам найдено более одного пункта отправления и пункта назначения. |

3. Ошибки службы «Создание заказа»

| Код ошибки | Текст ошибки | Примечание |
|---------------|-----------------------------------|---|
| no-data-found | Данные не найдены | По заданным параметрам не найдено ни одного заказа. Проверьте корректность указанного номера или попробуйте изменить дату приёма груза. |
| too-many-rows | Найдено более одной записи данных | По заданным параметрам найдено более одного заказа. Попробуйте указать дату приёма груза, чтобы сузить область поиска. |

4. Ошибки службы «Отслеживание статуса»

| Код ошибки | Текст ошибки | Примечание |
|----------------------|---|---|
| no-data-found | Данные не найдены | По заданным параметрам не найдено ни одного заказа. Проверьте корректность указанного номера или попробуйте изменить дату приёма груза. |
| too-many-rows | Найдено более одной записи данных | По заданным параметрам найдено более одного заказа. Попробуйте указать дату приёма груза, чтобы сузить область поиска. |
| no-setting | Не предусмотрены настройки | Для указанного клиента не указаны настройки службы отслеживания. Обратитесь к вашему менеджеру или напишите на itcustomers@dpd.ru . |
| date-before-start | Дата запрошенного объекта ранее подключения клиента к сервису | Вы можете получать статусы заказов, созданных в системе после вашего подключения к сервису. Выдача статусов «задним числом» не поддерживается. |
| confirm-not-required | Запрос не предполагает подтверждения | Вы попытались вызвать запрос подтверждения, но ваши настройки не предполагают подтверждения. Обратитесь на itcustomers@dpd.ru , чтобы изменить эти настройки. |

5. Ошибки службы «Создание адреса с кодом»

| Код ошибки | Текст ошибки | Примечание |
|---------------------|--|--|
| code-already-exists | Адрес с указанным кодом уже существует | Адрес с указанным кодом уже существует. Обратитесь к вашему менеджеру или напишите на itcustomers@dpd.ru . |
| address-error | Адрес не может быть создан. | Не для всех параметров адреса заданы значения. Необходимо указать недостающие значения. |
| code-not-found | Адрес с указанным кодом не найден. | По указанному коду не найден адрес. Необходимо создать адрес. |

6. Ошибки службы «Печать Наклейки»

| Код ошибки | Текст ошибки | Примечание |
|------------------|--|---|
| no-data-found | Заказ № для клиента № не найден | Ошибка в номере заказа |
| input-data-error | У заказа № дата забора ранее текущей даты | Заказ устарел |
| illegal-state | Состояние заказа № не позволяет создавать наклейки | Заказ уже переведен в состояние, не позволяющее формировать наклейки. Например, посылки уже получены у клиента. |

VI. Интеграционные примеры

В данном разделе даны примеры интеграции с различными системами.

1. 1С. Трейсинг.

Функция ПолучитьТекстЗапросПоЗаказамORDERNUM(Заказы, Заказчик, Ключ)

```
ЗаписьXML = Новый ЗаписьXML;
ЗаписьXML.УстановитьСтроку();
//ЗаписьXML.ЗаписатьОбъявлениеXML();

ЗаписьXML.ЗаписатьНачалоЭлемента("soap:Envelope");
ЗаписьXML.ЗаписатьАтрибут("xmlns:soap", "http://schemas.xmlsoap.org/soap/envelope/");
ЗаписьXML.ЗаписатьАтрибут("xmlns:ns0", "http://dpd.ru/ws/order2/2012-04-04");

ЗаписьXML.ЗаписатьНачалоЭлемента("soap:Header");
ЗаписьXML.ЗаписатьКонецЭлемента();

ЗаписьXML.ЗаписатьНачалоЭлемента("soap:Body");

    ЗаписьXML.ЗаписатьНачалоЭлемента("ns0:getOrderStatus");
    ЗаписьXML.ЗаписатьАтрибут("xmlns:xs", "http://www.w3.org/2001/XMLSchema");
    ЗаписьXML.ЗаписатьАтрибут("xmlns:xsi", "http://www.w3.org/2001/XMLSchema-
instance");

        ЗаписьXML.ЗаписатьНачалоЭлемента("orderStatus");

            ЗаписьXML.ЗаписатьНачалоЭлемента("auth");

                ЗаписьXML.ЗаписатьНачалоЭлемента("clientNumber");
                ЗаписьXML.ЗаписатьТекст(Заказчик);
            ЗаписьXML.ЗаписатьКонецЭлемента();

                ЗаписьXML.ЗаписатьНачалоЭлемента("clientKey");
                ЗаписьXML.ЗаписатьТекст(Ключ);
            ЗаписьXML.ЗаписатьКонецЭлемента();

        ЗаписьXML.ЗаписатьКонецЭлемента();

        Для Каждого Заказ Из Заказы Цикл
            ЗаписьXML.ЗаписатьНачалоЭлемента("order");

                ЗаписьXML.ЗаписатьНачалоЭлемента("orderNumberInternal");
                ЗаписьXML.ЗаписатьТекст(Заказ);
                ЗаписьXML.ЗаписатьКонецЭлемента();

                ЗаписьXML.ЗаписатьКонецЭлемента();
            КонецЦикла;

        ЗаписьXML.ЗаписатьКонецЭлемента();

    ЗаписьXML.ЗаписатьКонецЭлемента();
```

```
ЗаписьXML.ЗаписатьКонецЭлемента();
ЗаписьXML.ЗаписатьКонецЭлемента();
```

```
ТекстXML = ЗаписьXML.Закрыть();
```

```
Возврат ТекстXML;
```

КонецФункции

Функция ОтправитьЗапросWEBCервису(URL,ТекстXML,Таймаут = 10000)

```
Если ПустаяСтрока(ТекстXML) Тогда
```

```
    Возврат "";
```

```
КонецЕсли;
```

```
WinHttp = Новый СОМОбъект("WinHttp.WinHttpRequest.5.1");
```

```
WinHttp.SetTimeouts(Таймаут, Таймаут, Таймаут, Таймаут);
```

```
WinHttp.Option(2,"utf-8");
```

```
WinHttp.Open("POST",URL);
```

```
WinHttp.setRequestHeader("Content-Type","text/xml; charset=utf-8");
```

```
Сообщить("Начало - " + ТекущаяДата());
```

```
Сообщить("URL: " + URL);
```

```
//Сообщить("ТекстXML запроса: " + ТекстXML);
```

```
Попытка
```

```
    WinHttp.Send(ТекстXML);
```

```
    Сообщить("Завершение - " + ТекущаяДата());
```

```
    Если WinHttp.Status = 200 Тогда
```

```
        Сообщить("Ответ: " + WinHttp.ResponseText);
```

```
        Возврат(WinHttp.ResponseText);
```

```
    Иначе
```

```
        Сообщить("Ошибка сервера: " + WinHttp.Status + " " + WinHttp.ResponseText  
+ " " + WinHttp.StatusText, СтатусСообщения.Важное);
```

```
        Возврат("");
```

```
    КонецЕсли;
```

```
Исключение
```

```
    Сообщить(ОписаниеОшибки(), СтатусСообщения.Важное);
```

```
    Возврат("");
```

```
КонецПопытки;
```

КонецФункции

2. PHP. Расчет стоимости

```
<?
/*
```

```
@access public
```

DPD_service - класс работы с DPD-сервисом

ВАЖНО!

1. Перед началом работы необходимо прописать свои значения в свойства класса \$MY_NUMBER и \$MY_KEY
2. Для переключения на рабочий хост DPD-сервиса нужно обнулить свойство класса \$IS_TEST
3. Данный класс заточен на работу в кодировке windows-1251

ПРИМЕР РАБОТЫ:

```
// подключение класса (указать свой путь к классу)
```

```
require_once($_SERVER['DOCUMENT_ROOT'].'/system/classes/dpd_service.class.php');
```

```
$DPD = new DPD_service;
```

```
// ЗАПРОС СПИСКА ГОРОДОВ -----
```

```
$arCityList = $DPD->getCityList();
```

```
    формат $arCityList:
```

```
    Array
```

```
        0 => Array[6]
```

```
            cityId => "48951627"
```

```
            countryCode => "RU"
```

```
            countryName => "Россия"
```

```
            regionCode => "42"
```

```
            regionName => "Кемеровская обл."
```

```
            cityName => "Кемерово"
```

```
        ...
```

```
// ЗАПРОС СТОИМОСТИ ДОСТАВКИ -----
```

```
// массив входных данных: город доставки и вес отправки
```

```
// остальные параметры запроса к сервису прописаны в методе
```

```
$arData = array(
```

```
    'delivery' => array(                                     // город доставки
```

```
        'cityId' => 49172112,
```

```
        'cityName' => 'Пермь',
```

```
    ),
```

```
    'weight' => 5,                                           // вес отправки
```

```
);
```

```
$arCost = $DPD->getServiceCost($arData);
```

```
    формат $arCost:
```

```
    Array
```

```
        0 => Array[4]
```

```
            serviceCode => "TEN"
```

```

        serviceName => "DPD 10:00"
        cost => "1887.59"
        days => "1"

        ...

    */

class DPD_service
{
    public $arMSG = array();          // массив-сообщение ('str' => текст_сообщения, 'type'
=> тип_сообщения (по дефолту: 0 - ошибка)

    private $IS_ACTIVE    = 1;        // флаг активности сервиса (0 - отключен, 1 - включен)
    private $IS_TEST      = 1;        // флаг тестирования (0 - работа, 1 - тест)
    private $SOAP_CLIENT;              // SOAP-клиент

    private $MY_NUMBER = '1111111111'; // ЗАМЕНИТЬ НА СВОЙ!!! - клиентский номер
в системе DPD (номер договора с DPD)
    private $MY_KEY      = '2222222222'; // ЗАМЕНИТЬ НА СВОЙ!!! - уникальный ключ
для авторизации

    private $arDPD_HOST = array(
                                0 => 'ws.dpd.ru/services/',          //
рабочий хост
                                1 => 'appl.dpd.ru:8080/services/',    //
тестовый хост
                                );

    private $arSERVICE = array(                                     //
сервисы: название => адрес
                                'getCitiesCashPay'    => 'geography',    //
География DPD (города доставки)
                                'getTerminalsSelfDelivery' => 'geography', // список
терминалов DPD (TODO)
                                'getServiceCost'      => 'calculator2',  // Расчёт
стоимости
                                'createOrder'         => 'order2',        //
Создать заказ на доставку (TODO)
                                'getOrderStatus'      => 'order2',        //
Получить статус создания заказа (TODO)
                                );

    /**
     * Конструктор
     *
     * @access public
     * @return void
     */
    public function __construct()
    {
        $this->IS_TEST = $this->IS_TEST ? 1 : 0;
    }
}

```

```

/**
 * Список городов доставки
 *
 * @access public
 * @return
 */
public function getCityList()
{
    $obj = $this->_getDpdData('getCitiesCashPay');

    // конверт $obj --> $arr
    $res = $this->_parceObj2Arr($obj->return);

    return $res;
}

/**
 * Определение стоимости доставки
 *
 * @access public
 * @param array $arData           // массив входных параметров*
 * @return
 */
public function getServiceCost($arData)
{
    // куда
    if($arData['delivery']['cityName'])
    {
        $arData['delivery']['cityName'] = iconv('windows-1251','utf-
8',$arData['delivery']['cityName']);
    }

    // откуда
    $arData['pickup'] = array(
        'cityId' => 48994107,
        'cityName' => iconv('windows-1251','utf-
8','Екатеринбург'),
        'regionCode' => '66',
        'countryCode' => 'RU',
    );

    // что делать с терминалом
    $arData['selfPickup'] = true; // Доставка ОТ терминала
    $arData['selfDelivery'] = true; // Доставка ДО терминала

    // третий параметр - флаг упаковки запроса в общее поле "request"
    $obj = $this->_getDpdData('getServiceCost', $arData, 1);

    // конверт $obj --> $arr
    $res = $this->_parceObj2Arr($obj->return);

    return $res;
}

```

```
// PRIVATE -----
```

```

/**
 * Коннект с соответствующим сервисом
 *
 * @access private
 * @param string $method_name Запрашиваемый метод сервиса (см. ключ
свойства класса $this->arSERVICE)
 * @return bool Результат инициализации (если
положительный - появится свойство $this->SOAP_CLIENT, иначе $this->arMSG)
 */
private function _connect2Dpd($method_name)
{
    if(!$this->IS_ACTIVE) return false;

    if(!$service = $this->arSERVICE[$method_name])
    {
        $this->arMSG['str'] = 'В свойствах класса нет сервиса "'.$method_name.'";
        return false;
    }
    $host = $this->arDPD_HOST[$this->IS_TEST].$service.'?WSDL';

    try
    {
        // Soap-подключение к сервису
        $this->SOAP_CLIENT = new SoapClient('http://'.$host);
        if(!$this->SOAP_CLIENT) throw new Exception('Ошибка');
    } catch (Exception $ex) {
        $this->arMSG['str'] = 'Не удалось подключиться к сервисам DPD '.$service;
        return false;
    }

    return true;
}

/**
 * Запрос данных в методе сервиса
 *
 * @access private
 * @param string $method_name Название метода Dpd-сервиса (см.
$arSERVICE)
 * @param array $arData Массив параметров, передаваемых в
метод
 * @param integer $is_request флаг упаковки запроса в поле 'request'
 * @return XZ_obj Объект, полученный от сервиса
 */
private function _getDpdData($method_name, $arData=array(), $is_request=0)
{
    if(!$this->_connect2Dpd($method_name)) return false;

    // параметр запроса для аутентификации
    $arData['auth'] = array(
        'clientNumber' => $this->MY_NUMBER,

```

```

        'clientKey'          => $this->MY_KEY,
    );

    // упаковка запроса в поле 'request'
    if($is_request) $arResult['request'] = $arResult;
    else $arResult = $arResult;

    try
    {
        //eval("\$obj = \$this->SOAP_CLIENT->\$method_name(\$arResult);");
        $obj = $this->SOAP_CLIENT->$method_name($arResult);

        if(!$obj) throw new Exception('Ошибка');
    } catch (Exception $ex) {
        $this->arResult['str'] = 'Не удалось вызвать метод '.$method_name.' / '.$ex;
    }

    return $obj ? $obj : false;
}

/**
 * Парсер объекта в массив (рекурсия)
 *
 * @access private
 * @param object      $obj      Объект
 * @param integer     $isUTF     Флаг необходимости конвертирования строк из UTF в
WIN (0|1), по-дефолту "1" - конвертировать
 * @param array       $arr      Внутренний служебный массив для обеспечения
рекурсии
 * @return array
 */
private function _parseObj2Arr($obj,$isUTF=1,$arr=array())
{
    $isUTF = $isUTF ? 1 : 0;

    if(is_object($obj) || is_array($obj) )
    {
        $arr = array();
        for(reset($obj); list($k, $v) = each($obj);)
        {
            if($k === "GLOBALS") continue;
            $arr[$k] = $this->_parseObj2Arr($v, $isUTF, $arr);
        }
        return $arr;
    }
    elseif(gettype($obj) == 'boolean')
    {
        return $obj ? 'true' : 'false';
    }
    else
    {
        // конверт строк: utf-8 --> windows-1251
        if($isUTF && gettype($obj)=='string') $obj = iconv('utf-8','windows-1251',$obj);
        return $obj;
    }
}

```



```

    }
}

?>

```

3. Java. Создание заказа.

Сначала генерируем классы прокси клиента. Пример build.xml:

```

<?xml version="1.0" encoding="UTF-8"?>

<project basedir=".">

    <target name="properties">
        <property name="host.url" value="http:// . . ." />
        <property name="src.generated.dir" value="src/main/generated" />
        <property name="bin.dir" value="bin" />
    </target>

    <target name="codegen-Wsclient" depends="properties"
description="Генерация java-классов WS-клиента из WSDL">

        <property name="package" value="ru.dpd.b2b.order.wsdlgen" />
        <exec executable="wsimport">
            <arg value="-p"/>
            <arg value="${package}"/>
            <arg value="-d"/>
            <arg value="${bin.dir}"/>
            <arg value="-keep"/>
            <arg value="-s"/>
            <arg value="${src.generated.dir}"/>
            <arg value="-verbose"/>
            <arg value="${host.url}/services/order2?wsdl"/>
        </exec>
    </target>

</project>

```

Примеры написанного кода:

Открываем порт сервиса:

DPDOrder и DPDOrderService – сгенерированные классы

```

import ru.dpd.b2b.order.wsdlgen.DPDOrder;
import ru.dpd.b2b.order.wsdlgen.DPDOrderService;

private DPDOrder getService() throws MalformedURLException{
    Utility.proxySetting(appParam);
    URL url = new URL(appParam.getWsUrl() + "?wsdl");
}

```

```

        DPDOrder service = new DPDOrderService(url, Q_NAME).getDPDOrderPort();
        return service;
    }

    // настройки нашего прокси

    public static void proxySetting(AppParameters appParam){
        String hostProxy = appParam.getWsProxyUrl();
        Integer portProxy = appParam.getWsProxyPort();
        if(hostProxy != null && hostProxy.trim().length() > 0 &&
portProxy != null) {
            System.setProperty("http.proxyHost", hostProxy);
            System.setProperty("proxyPort", portProxy.toString());
        }
    }
}

```

Вызываем сервис и получаем результат:

DpdOrdersData (данные заявки) и DpdOrderStatus (статус заявки) – сгенерированные классы.

```

import ru.dpd.b2b.order.wsdlgen.DPDOrder;
import ru.dpd.b2b.order.wsdlgen.DpdOrderStatus;
import ru.dpd.b2b.order.wsdlgen.DpdOrdersData;
import ru.dpd.b2b.order.wsdlgen.WSFault_Exception;

    public List<DpdOrderStatus> getResult(DPDOrder service, DpdOrdersData
ordersData) {
        try {
            final List<DpdOrderStatus>
orderStatusList=service.createOrder(ordersData);
            return orderStatusList;
        } catch (WSFault_Exception e) {
            System.err.println("WSFault: " + e.getFaultInfo().getCode() + " /
" + e.getFaultInfo().getMessage());
        } catch (Throwable e) {
            System.err.println ("Fatal error: ", e);
        }
        return null;
    }
}

```

4. PHP. Создание заказа и печать наклейки

Dpd_class.php

<?

```

class DPD_class
{
    public $arMSG = array();

    private $IS_ACTIVE = 1;
    private $IS_TEST = 0;
    private $SOAP_CLIENT;

    private $MY_NUMBER = '18181818181818';
    private $MY_KEY = '1123AB873233ED871';

    private $arDPD_HOST = array(0 => 'ws.dpd.ru/services/', 1 => 'wstest.dpd.ru/services/');

    private $arSERVICE = array(
        'getServiceCostByParcels' => 'calculator2',
        'getServiceCost' => 'calculator2',
        'getTerminalsSelfDelivery' => 'geography',
        'getCitiesCashPay' => 'geography',
        'createOrder' => 'order2', // Создание заявки в DPD
        'getOrderStatus' => 'order2', // Получение статуса заказа
        'createLabelFile' => 'label-print', // Получить наклейки
    );

    public function __construct() {
        $this->IS_TEST = $this->IS_TEST ? 1 : 0;
    }

    public function createOrder($arData) {
        $obj = $this->_getDpdData('createOrder', $arData, 'orders');
        $res = $this->_parceObj2Arr($obj->return, 0);
        return $res;
    }

    public function getCityList() {
        $obj = $this->_getDpdData('getCitiesCashPay');
        $res = $this->_parceObj2Arr($obj->return);
        return $res;
    }

    public function getTerminalsSelfDelivery($arData) {
        $obj = $this->_getDpdData('getTerminalsSelfDelivery');
        $res = $this->_parceObj2Arr($obj->return, 0);
        return $res;
    }

    public function getServiceCostByParcels($arData) {
        $obj = $this->_getDpdData('getServiceCostByParcels');
        return $obj;
    }

    public function getServiceCost($arData) {
        $obj = $this->_getDpdData('getServiceCost', $arData, 'request');
        $res = $this->_parceObj2Arr($obj->return);
    }
}

```

```

        return $res;
    }

    public function createLabelFile($arData) {
        $obj = $this->_getDpdData('createLabelFile', $arData, 'getLabelFile');
        return $obj;
    }

    public function getOrderStatus($arData) {
        $obj = $this->_getDpdData('getOrderStatus', $arData, 'orderStatus');
        return $obj;
    }

    private function _connect2Dpd($method_name) {
        if(!$this->IS_ACTIVE) return false;

        if(!$service = $this->arSERVICE[$method_name]) {
            $this->arMSG['str'] = 'В свойствах класса нет сервиса "'.$method_name.'"';
            return false;
        }
        $host = $this->arDPD_HOST[$this->IS_TEST].$service.'?WSDL';

        try {
            // Soap-подключение к сервису
            $this->SOAP_CLIENT = new SoapClient('http://'.$host);
            if(!$this->SOAP_CLIENT) throw new Exception('Error');
        } catch (Exception $ex) {
            $this->arMSG['str'] = 'Не удалось подключиться к сервисам DPD '.$service;
            return false;
        }

        return true;
    }

    private function _getDpdData($method_name, $arData=array(), $is_request=0) {
        if(!$this->_connect2Dpd($method_name)) return false;

        $arData['auth'] = array(
            'clientNumber' => $this->MY_NUMBER,
            'clientKey' => $this->MY_KEY,
        );

        if($is_request) $arRequest[$is_request] = $arData;
        else $arRequest = $arData;

        try {
            $obj = $this->SOAP_CLIENT->$method_name($arRequest);
            if(!$obj) throw new Exception('Error');

        } catch (Exception $ex) {
            $this->arMSG['str'] = 'Не удалось вызвать метод '.$method_name.' / '.$ex;
        }

        return $obj ? $obj : false;
    }

```

```

    }

    private function _parceObj2Arr($obj,$isUTF=1,$arr=array()) {
        $isUTF = $isUTF ? 1 : 0;

        if(is_object($obj) || is_array($obj) ) {
            $arr = array();
            for(reset($obj); list($k, $v) = each($obj);) {
                if($k === "GLOBALS") continue;
                $arr[$k] = $this->_parceObj2Arr($v, $isUTF, $arr);
            }
            return $arr;
        }
        elseif (gettype($obj) == 'boolean') {
            return $obj ? 'true' : 'false';
        }
        else {
            if($isUTF && gettype($obj)=='string') $obj = iconv('utf-8','windows-1251',$obj);
            return $obj;
        }
    }
}
}

```

?>

Dpd.php

<?

```

include_once 'dpd_class.php';
$DPD = new DPD_class;

```

```

$order = array();
$ORDER_ID = intval($_REQUEST['order']);

```

```

$arOrder_DPD = array(
    'header' => array(
        'datePickup'    => '2013-06-19',
        'senderAddress' => array(
            'name'                => 'ООО ...',
            'terminalCode' => 'КJA',
            'city'                => 'Город',
            'street'              => 'Улица',
            'streetAbbr'         => 'ул',
            'house'               => 'Номер дома',
            'houseKorpus'        => 'А',
            'contactFio'         => 'ФИО',
            'contactPhone'       => '+79000000000',
        ),
    ),
    'order' => array(
        'orderNumberInternal' => $ORDER_ID,
        'serviceCode' => 'ECN',
        'serviceVariant' => 'ТД',
        'cargoNumPack' => '6',
        'cargoWeight' => '62',
    ),
);

```

```

        'cargoRegistered' => false,
        'cargoCategory' => 'Диски',
        'receiverAddress' => array(
            'name' => 'ФИО',
            'terminalCode' => 'PKV',
            'city' => 'Псков',
            'street' => 'Герцена',
            'streetAbbr' => 'ул',
            'house' => '3',
            'houseKorpus' => '',
            'contactFio' => 'ФИО',
            'contactPhone' => '+7 (900) 000-00-00',
        ),
    ),
);

```

```

$order = $DPD->createOrder($arOrder_DPD);
?>

```

```

<span style="font-size: 20px;">
    <?if($order['errorMessage']):?>
        <?=$order['errorMessage']?>
    <?endif;?>
    <?if($order['orderNum']):?>
        Номер заказа DPD: <?=$order['orderNum']?>
    <?endif;?>
</span>

```

VII. Контакты

По всем вопросам, связанным с использованием нашего интеграционного модуля, обращайтесь в службу информационной поддержки DPD: itcustomers@dpd.ru.