PROGRAMME & DEVELOPMENT SERVICES

XML Services

Version R3.4.1

Toolkit

This document is published by, and remains the property of, DHL Express. It contains confidential and proprietary information. The customer shall hold this information in confidence and may not make any copies or disseminate this document to any third party. The customer may not disclose this information internally to its employees or agents except on a need-to-know basis and shall instruct all such employees or agents to comply with the DHL confidentiality requirement implied by this statement.

This document is furnished for informational use only and is subject to change without notice. DHL Express assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

DHL Express
All Rights Reserved

Published: Nov 2026

Table of Contents

Revision History 4
1 Introduction 5
1.1 XML Services 5
1.2 XML Services Toolkit 6
2 Structure of Tool Kit 7
2.1 Overview 8
2.2 clientSoftware 8
2.2.1 lib folder 8
2.2.2 logs folder 9
2.2.3 TransformXMLtoPDF 9
2.3 Documents folder – Reference Documents 9
2.4 Documents folder – Unit Test Plan 10
2.4.1 Unit Test Plan - Pickup 10
2.4.2 Unit Test Plan - Routing 11
2.4.3 Unit Test Plan - Shipment Validation 11
2.4.4 Unit Test Plan - Tracking 11
2.4.5 Unit Test Plan - Capability and Quote Service 12
2.4.6 Unit Test Plan - Image Upload Service 12
2.5 xsd 12
3 Scripts usage 12
3.1 DHLClient software to send request to XML Services 13
3.1.1 Pre requisites 13
3.1.2 Settings in runDHLClient script 13
3.1.3 runDHLClient script Retry Function 15
3.1.4 runDHLClient script UTF-8 Functionality 16
3.1.5 runDHLClient script DNS Flusher 17
3.1.6 runDHLClient script with additional logging details 17

Revision History

XML Services version	Release Date	Comments
R3.4.0	25th Oct 2026	1) Shipment Validation Service a) New Shipment Validation XML schema version 10.1 request and response. b) New ExportDeclarations array in XML schema version 10.1 2) Shipment Preparation Guide a) Update to use ExportDeclarations array instead of ExportDeclaration 3) Toolkit document a) xsd folder i) xsd folder for all services Remove <Global> sub-folder and ii) added new 10.1 datatypes.xsd and shipment request and response schema files. b) ShipmentValidation i) Updated sample request and response XMLs file with 10.1 schema c) ImageUpload i) Updated sample request and response XMLs file with 10.1 schema
R3.4.1	1st Nov 2026	Toolkit document 1. Reference_Data.xlsx: a. updated <RegistrationNumberTypeCode> for NID, PAS and EOR enablement for MU b. updated VAT regex updates for AE, JP and MU c. added <ItemReferenceType> for PSR, IUC, IUD, AXO, PRV and PCI d. added <CustomsDocumentType> for POA e. added <CustomsPaperworkType> for POA

Note: Only Major changes will increase the version number.

1 Introduction

1.1 XML Services

DHL offers XML based online integration platform that provides Product Availability, Transit time and Rating, Shipment Booking, Courier Pickup Booking and Tracking services for DHL Express customers globally from over 140 countries.

Here is how XML services can help DHL customers and vendor partners:

- They can use their shipping and warehouse management systems and still provide access to complete set of DHL shipping services
- They can provide DHL capabilities from their internal shipping sites, Shopping carts and Commerce web sites
- No DHL hardware or software deployed on their site and they can still get visibility to DHL's latest products and service offerings
- They have a standard (XML) and secured (HTTPS and User Authentication) platform to integrate with DHL and they can control the implementation timelines

Below are list of available services or operations:

- Capability Service – Provides list of available product and services for given shipment criteria along with Booking cut-off, Pickup cut-off and Transit times
- Quote Service – Provides total cost along with breakdown of charges for requested product and service combinations
- Routing Service - Validates the origin and destination city and postcodes. Note that these validations are now better performed in Capability Service and hence we recommend users to stop using Routing Service and instead use Capability Service.
- Shipment Validation - Validates shipment data and return Waybill, License Plate and data for label printing
- Image Upload - to upload and re-upload of PLT images to a previously created shipment. The PLT images for the shipment can be uploaded before the shipment has been physically collected by DHL courier.
- Label Image utility – Generates DHL compliant label image using the XML response from Shipment Validation
- Pickup Booking – Can request, modify or delete courier pick-up
- Tracking - Obtain shipment details and event visibility filtered by Waybill, License Plate, or Reference Number and Account Number

Here is the process of implementing XML integration with DHL

1. Register via [DHL Express - MyDHL API On-boarding request | DHL API Developer Portal] to download the latest XML Toolkit.
   Make sure that you have the programming resources experienced with Web Services and XML to work with this toolkit

2. Test Environment Access: Contact DHL Sales to request access to Testing and Certification environment
3. Development: Complete development and XML integration work
4. Certification: Contact DHL to request certification by providing Label image and Request and Response XML messages as recommended in Test Plan in the toolkit. These are validated by the DHL eCommerce team and then customer is certified as production-ready.
5. Go-Live: After passing certification you will receive access to production environment and then can start using DHL Web Services
6. Production Support: Contact DHL Technical Support team for production support
7. For installing or renewing certificate for XML Services, please refer to Reference_Data.xls – XML Services Certificate tab sheet.

1.2 XML Services Toolkit

This toolkit is provided to DHL clients to help them understand and implement integration with XML Services application.

It contains a detailed description for the format of the request XML messages and response XML messages for each service. Sample request and response XML messages that help understand the structure of the request and response messages are also included. These messages also help to understand the conditions/data values that might result in an error response from XML-PI server.

A Java tool, detailed below.

1) DHLClient: To send an XML request message to the XML Services application.
   This can be directly integrated with the client application or it can be used to understand the method to establish a connection with XML Services.

XML Services is available in three regions and is slightly customized for each region. All three regions are supported by a central application and accessible by same web server address.

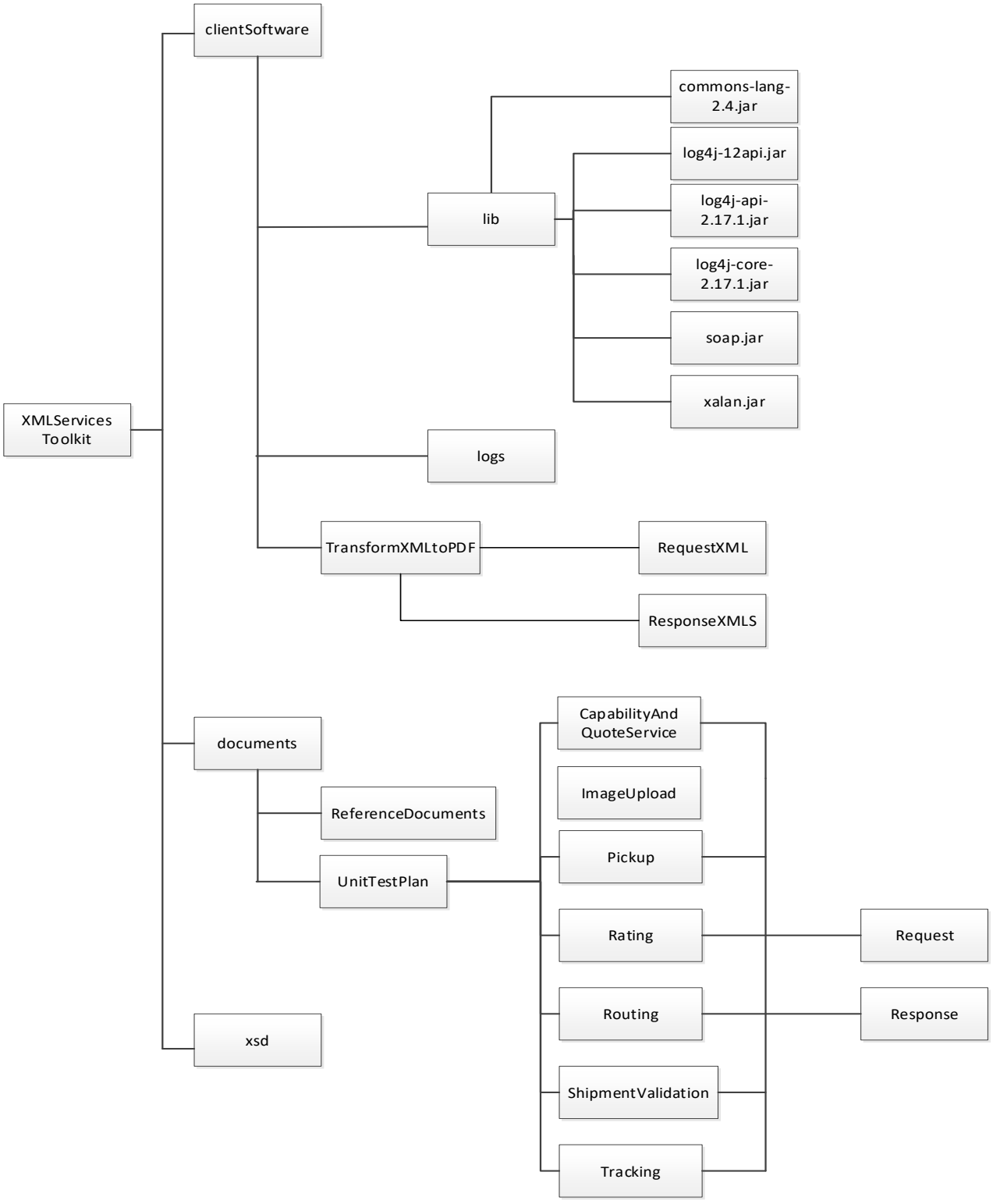
- AP-EM Region - Supports countries in Asia, Africa, Australia and Pacific.
- EU Region - Supports countries in Europe.
- AM Region - Supports USA and other countries in North and South Americas.

Please refer to Reference_Data.xls for the list of countries in each region of supported country deployment. XML Services offers Global XSD schema which available for Pickup, Shipment Validation, Image Upload, Routing, Tracking, DCT Capability & Quote and Image Upload services.

Note: Image Upload service is leveraging the same Shipment Validation schema, refer to Shipment Validation documentation for details.

2 Structure of Tool Kit

The toolkit will have the directory structure as shown below: The contents and description of each folder will be explained in the subsequent sections.



2.1 Overview

The following section will detail the content of each folder and give a brief description of the contents.

Along with this ToolKit_R3.4.1.pdf document, there is a Reference Data excel sheet (Reference_Data.xls) consisting of reference data for each country, currency code, product lists, services list, error messages for each service and other information relevant to the XML Services.

2.2 clientSoftware

This folder has the following files:

1) DHLClient.java – Simulates a condition where an XML message is sent and received to XML-PI application over HTTP or HTTPS source file

2) DHLClient.class – Compiled class file of DHLClient.java source file.

3) runDHLClient.cmd – Used to run the DHLClient class to send request messages and receive response messages.

4) setClassPath.cmd – Used to set the class path.

5) The TransformXMLtoPDF directory - The contents of this folder are detailed in section 2.2.3.

6) log4j2.properties – log4j2 setting to include additional logging at client application level.

A detail of the scripts to execute the tools is included in section 3 – Scripts usage.

2.2.1 lib folder

This folder has the following files:

1) xalan.jar: XML Libraries
2) soap.jar : SOAP / base64 library.
3) log4j-12api.jar, log4j-api-2.17.1.jar, log4j-core-2.17.1.jar: log4j logging library
4) commons-lang-2.4.jar: Java utility class

2.2.2 logs folder

This folder contains the log file that DHLClient.log that generated via toolkit runDHLClient.cmd sample program with the designated log level.

DHLClient.log filename convention is DHLClient.logMMDD, example: DHLClient.log0623

Note: DHLClient.log filename without MMDD indicates current day log file.

2.2.3 TransformXMLtoPDF

This folder has the following files/directories:

1) RequestXML
   This directory has the request XML that needs to be sent to the XML Services application. This is used only when the runDHLClient.cmd script is used. If the client is using some other software to send a request XML than this may not be used.
2) ResponseXMLS

   This directory stores the response XML's that are to be processed for transformation to PDF's. If runDHLClient.cmd script is used, the response XML is directly stored under this directory.

2.3 Documents folder – Reference Documents

This folder has the following files/directories:

ReferenceDocuments – This has the reference documents for all the services offered by XML Services.

1) Pickup Service Guide (XMLServices10.0.4_Pickup.pdf):
   This would help the XML Services client to understand the pickup service field elements definition and its usage offered in XML Services Pickup service.

2) Routing Service Guide (XMLServices10.0.1_Routing.pdf):
   This would help the XML Services client to understand the routing service field elements definition and its usage offered in XML Services Routing service.

3) Shipment Validation Service Guide (XMLServices_R3.4.0_ShipmentValidationService.pdf):
   This would help the XML Services client to understand the Shipment Validation service and Image Upload service field elements definition and its usage offered in XML Services Shipment Validation & Image Upload service.

4) Tracking Service Guide (XMLServices10.0.4_Tracking.pdf):
   This would help the XML Services client to understand the tracking service field

elements definition and its usage offered in XML Services Tracking service.

5) Capability and Quote Service Guide (XMLServices10.0.7_CapabilityAndQuoteService.pdf):
This would help the XML Services client to understand the capability and quote service field elements definition and its usage offered in XML Services Capability and Quote service.

6) Shipment Preparation Guide (XMLServices_R3.4.0_ShipmentPreparationGuide.pdf):
This would help the XML Services client to understand the preparation guide for new BBX shipment, PLT shipment, regular shipment, Label-Free and Dangerous Goods shipment offered in XML Services Shipment Validation service.

2.4 Documents folder – Unit Test Plan

This folder has the following files/directories:

1) UnitTestPlan – This would have the unit test plans and the sample request and response XMLs for each type of messages supported. The details of each directory are described in the subsequent section.

2.4.1 Unit Test Plan - Pickup

This directory will have the following files/directories:

For Pickup service, please refer to below documentations:

XMLServices10.0.2_Pickup_UTP.pdf - This is the unit test plan for testing the pickup service. The XML messages that are to be used for testing are in the directories below.

The Request and Response directory has further sub directories respectively"

i. \Request\ BookPickup – Contains sample request XML messages that can be used to test the book pickup functionality.
ii. \Request\ CancelPickup – Contains sample request XML messages to test cancel pickup functionality. (Not applicable for EA region)
iii. \Request\ ModifyPickup – Contains sample request XML messages to test modify pickup functionality. (Not applicable for EA region)
iv. \Response\ BookPickup– contains sample response XML messages corresponding to each request message for book pickup.
v. \Response\ CancelPickup – contains sample response XML messages corresponding to each request message for cancel pickup.
vi. \Response\ ModifyPickup– contains sample response XML messages corresponding to each request message for modify pickup. (Not applicable for EU region).

2.4.2 Unit Test Plan - Routing

This directory will have the following files/directories:

XMLServices10.0_Routing_UTP.pdf - This is the unit test plan for testing the routing service. The XML messages that are to be used for testing are in the directories below.

The Request and Response directory has further sub directories respectively:

i. \Request – Contains sample request XML messages to test the routing functionality.

ii. \Response – Contains sample response XML messages for routing service request messages in above directory.

2.4.3 Unit Test Plan - Shipment Validation

This directory will have the following files/directories:

XMLServices_R3.4.0_ShipVal_UTP.pdf - This is the unit test plan for testing the shipment validation service. The XML messages that are to be used for testing are in the directories below.

The Request and Response directory has further sub directories respectively:

i. \Request – Contains sample request XML messages to test the shipment validation functionality.

ii. \Response – Contains sample response XML messages for shipment validation service request messages in above directory.

2.4.4 Unit Test Plan - Tracking

This directory will have the following files/directories:

XMLServices10.0.1_Tracking_UTP.pdf - This is the unit test plan for testing the tracking service.
The XML messages that are to be used for testing are in the directories below.

i. Request - Contains sample request XML messages to test the tracking functionality.

ii. Response – Contains sample response XML messages for tracking service request messages in above directory.

2.4.5 Unit Test Plan - Capability and Quote Service

This directory will have the following files/directories:

XMLServices10.0.3_CapabilityAndQuoteService_UTP.pdf - This is the unit test plan for testing the Capability and Quote service.

The XML messages that are to be used for testing are in the directories below.

i. Request - Contains sample request XML messages to test the capability and quote functionality.

ii. Response – Contains sample response XML messages for capability and quote service request messages in above directory.

2.4.6 Unit Test Plan - Image Upload Service

XMLServices_R3.4.0_ImageUpload_UTP.pdf - This is the unit test plan for testing the image upload service.
The XML messages that are to be used for testing are in the directories below.

i. \Request – Contains sample request XML messages to test the image upload functionality.

ii. \Response – Contains sample response XML messages for image upload service request messages in above directory.

2.5 xsd

This will include the schema for Pickup, Routing, Shipment Validation, Pickup, Capability & Quote and Routing service.

i. xsd – Contains xsd schema files for above mentioned services.

ii. readme.txt - Note for developer Usage on the schema version

3 Scripts usage

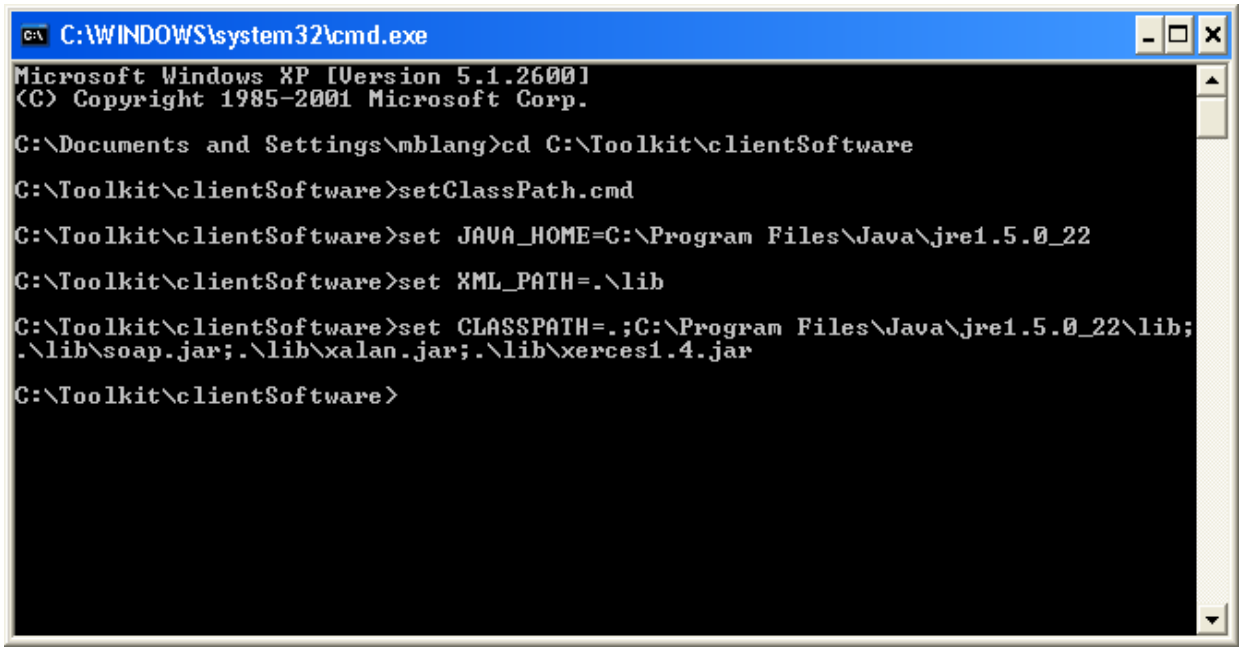
This section gives a detailed description for using the scripts for sending the request XML message to XML Services and to transform the response shipment XML to Transport label and Archive document.

3.1 DHLClient software to send request to XML Services

This script should be run from the directory where the DHLClient.class file is kept. If this script is run from the same directory as provided in the tool kit then no changes need to be done for the directory paths. If the name of the shipment validation request XML file is not 'Request_UStoDE_PLTShipment.xml', change the script to reflect the correct request xml filename.

3.1.1 Pre requisites

1) Java environment must be there in PATH variable.
2) Make sure following folders are present inside Customer Tool Kit-
   a) TransformXMLtoPDF/ResponseXMLS
   b) TransformXMLtoPDF/ProcessedXMLS
   c) TransformXMLtoPDF/PDFReports
3) Set the JAVA_HOME directory path in setClassPath.cmd
4) Check and update the path for soap.jar, xalan.jar, xerces.jar, commons-lang-2.4.jar in SetClassPath.cmd
5) Run the setClassPath.cmd script.



3.1.2 Settings in runDHLClient script

1) These variable needs to be set before running the script.

a) RESPONSE_PATH=TransformXMLtoPDF\ResponseXMLS\

This path points to the directory where the response XMLs are to be stored. This by default is set to the folder "..\TransformXMLtoPDF\ResponseXMLS\" present in the toolkit. If you need the response XMLs to be stored at some other location please change this path to point to destination accordingly. The directory path may be complete/ relative path.

b) SERVER_URL=https://xmlpitest-ea.dhl.com/XMLShippingServlet

This path points to the XML-PI URL which the request must be submitted. This by default points to the XMLShippingTest installation. This can be changed to point to some other XML installation if required.

c) INPUT_FILE=TransformXMLtoPDF\RequestXML\ShipmentValidation_v10.0_US-DE_request.xml

This path points to the input XML file location. This by default points to "..\TransformXMLtoPDF\RequestXML\ ShipmentValidation_v10.1_US-DE_request.xml" present in the toolkit. This can be changed to point to some other request file location if required. The file path may be complete/ relative path.

d) FUTURE_DAY=false

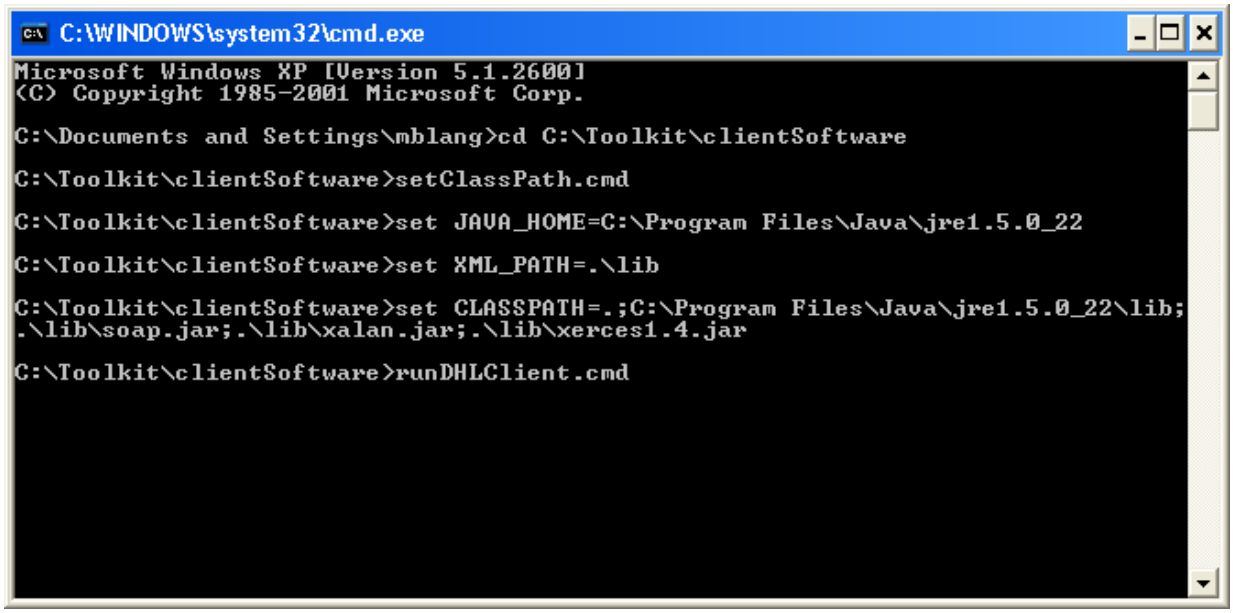
This parameter indicates to replace the XML request with '*date' element with future day or remaining as per the request XML value.

i) With default value of false, it means the '*date' element value will be remaining same as per request XML value.

ii) With value of true, it means the '*date' element will be overwritten with current date + 1 days.
Example:
<Date>2020-12-20</Date> in request XML
Current date is 2020-12-21

Overwritten value will be: <Date>2020-12-22</Date>

2) Execute the runDHLClient.cmd by double click in Window Explorer or in command prompt as below.



3) If XML Services Servlet URL is down, runDHLClient.cmd will initiate retry function.

3.1.3 runDHLClient script Retry Function

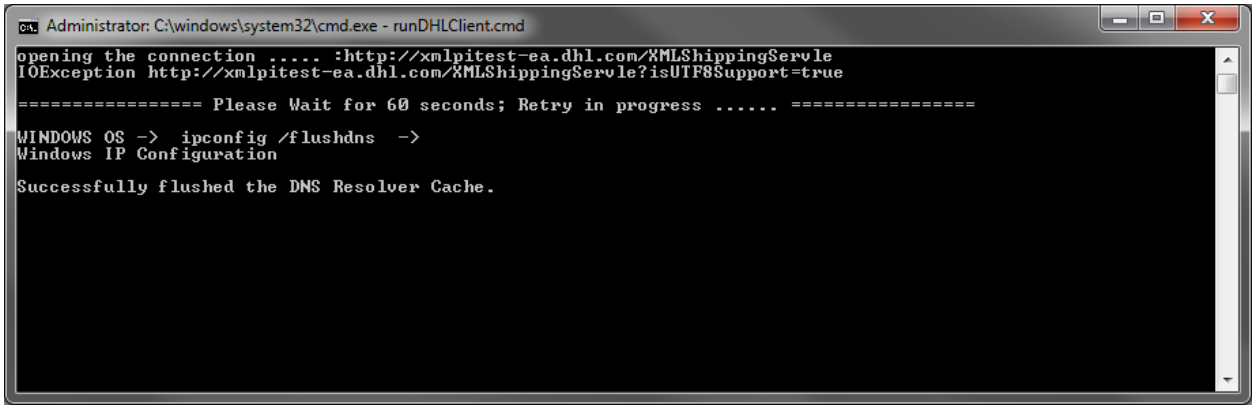
If XML Services Servlet is down, runDHLClient.cmd will initiate retry function. Since 3DNS/ Big IP take about 60 sec approximately to understand the unavailability of XML Services Servlet. Hence, DHLClient is enhanced to auto-retry in the same transaction after 60 seconds to handle IOException. There will be 3 retries for re-connection.
Client's integration application that does not use dhlclient program is recommended to build this function in order to cater server failover/ fallback scenario.

The sample code may refer to DHLClient.java which published in Toolkit.

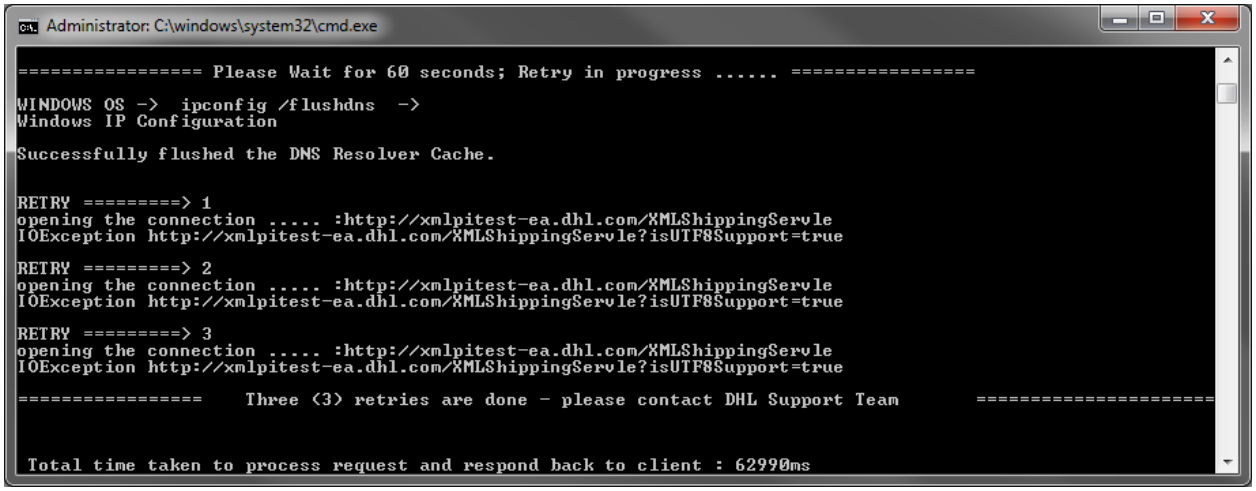
```
try {
    System.out.println("================= Please Wait for " + (timeToSleep / 1000L) + " seconds; Retry
in progress ..... ================= \n");

    Thread.sleep(timeToSleep);
  } catch (InterruptedException e1) {
    System.out.println("Exception while suspending execution of current thread for" + (timeToSleep / 1000L)
+ " seconds :" + e1);
    e1.printStackTrace(); }
```

First retry when connection exception occurred.



There will be 3 times re-connect to XML Services Servlet URL.



3.1.4 runDHLClient script UTF-8 Functionality

To enable the UTF-8 functionality, you may add “isUTF8Support=true” on the servlet URL.
For example:
http://xmlpitest-ea.dhl.com/XMLShippingServlet?isUTF8Support=true

Please note that UTF-8 functionality only applicable for Shipments Validation and Tracking services.
Extended characters that being sent manually (Not via Toolkitv6.1) in all other service requests may experience unexpected result.

User may refer to DHLClient.java file for more information on source coding.

```
if (isUTF8Support) {
  String query = "isUTF8Support=true";
  servletURL = new URL(httpURL + "?" + query);
 } else {
  servletURL = new URL(httpURL);
 }
```

Please note that in this release, the DHLClient is enabled to send UTF-8 characters to XML application for below Services:

1. Shipment Validation
2. Tracking
3. Capability and Quote
4. Pickup
5. Routing

XML Services application behavior will be vary based on which Toolkit is used, which contains isUTF8Support=true parameter in the request header.

3.1.5 runDHLClient script DNS Flusher

When XML Services Servlet is down, DHLClient has been updated to flush DNS explicitly in underlying user's machine OS (Operating System). It supports Windows, Unix, Linux and MAC OS. If required utility (which trigger OS command) is not installed in client machine, DNS flushing might not work as expected.

DHLClient is using different command for each OS respectively. When encounter "IO Exception" after 3 times retry, please execute below commands from local terminal and retry:

MAC OS: dscacheutil –flushcache (or) killall -HUP mDNSResponder
WINDOWS OS: ipconfig /flushdns
Linux/Unix OS: nscd -I hosts (or) dnsmasq restart (or) rndc restart

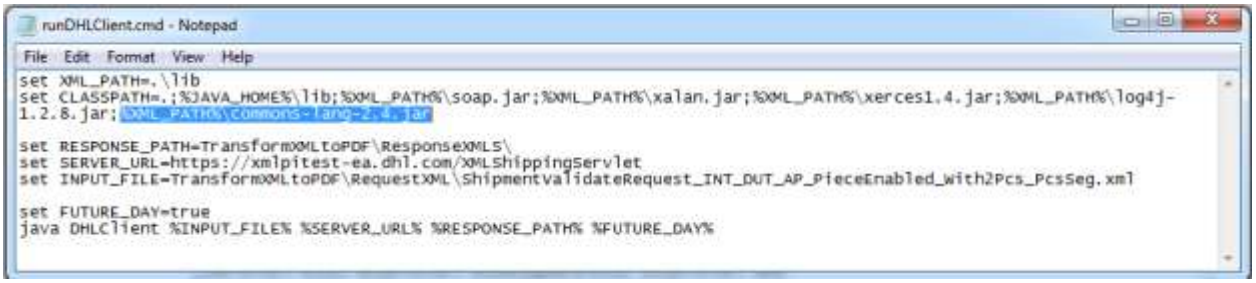
Note: If encounter any issue while executing above commands, please contact DHL XML Services support team.

3.1.6 runDHLClient script with additional logging details

In XML Services v10.1 toolkit, **log4j2.properties** with additional logging feature is published in toolkit.

This will allowed the XML Services client to further analyze the response time from sending the XML request to XML Services application and receiving the XML response from XML Services application.

Pre-requisite:
"%XML_PATH%\commons-lang-2.4.jar" must be added on runDHLClient.cmd classpath.



Example of log4j2.properties file:



1) Only rootLogger.level = debug logging parameter can be used.

   This log level designates additional informational messages for debug the application. This is the value in log4j2.properties.

   Example:
   rootLogger.level = debug

   Sample log file generated:

2) Sample log file contents with debug log level:



3) Do not make any changes on log4j2 .appender.file parameter.