



Get Documents Web Service 1.3.0

Confidential

Revision Notes

Revision	Date	Change Description
1.0.0	04/23/2009	Version 1.0 Created.
1.0.1	11/27/2009	URL for endpoints has been modified. Version allowed value has been updated.
1.1.0	04/20/2010	Addition of GetShipmentManifestDocument Method Update to the Introduction. Update to Implementation Considerations.
1.1.1	10/04/2010	Additional error codes.
1.2.0	07/07/2014	Update to the introduction. Additional Field in the request context.
1.3.0	04/10/2015	Additional Fields in document request and response and additional error codes

Table of Contents

1	GETTING STARTED.....	5
1.1	INTRODUCTION.....	5
1.2	DOCUMENT OVERVIEW	6
1.3	SPECIFICATION COMPLIANCE	6
1.4	DEVELOPMENT PLATFORMS	6
1.5	RELATED RESOURCES	7
1.6	PUROLATOR E-SHIP WEB SERVICES DEVELOPMENT PROCESS	7
2	WEB SERVICES	8
2.1	CHARACTERISTICS OF CALLS	9
2.2	DATA TYPES.....	9
2.3	SECURITY	9
2.4	SERVICE METHODS AND OBJECTS	9
2.4.1	<i>Get Documents</i>	9
2.4.2	<i>Get Shipment Manifest Document</i>	10
2.5	ERROR HANDLING.....	12
2.6	IMPLEMENTATION CONSIDERATIONS	13
2.7	ERROR MESSAGES / CODES.....	13
	APPENDIX A – COMPLETE FIELD LIST	17

Legal and Copyright Notices

This document, and the information contained herein, is confidential and proprietary to Purolator Inc., and its use is governed by the Purolator E-Ship® Resource Centre Agreement or the Purolator E-Ship® Web Services Soft Launch Agreement (as applicable). In the event of a conflict between this document and the Purolator E-Ship Resource Centre Agreement or the Purolator E-Ship Web Services Soft Launch Agreement (as applicable), the Purolator E-Ship Resource Centre Agreement or the Purolator E-Ship Web Services Soft Launch Agreement (as applicable) will govern to the extent of the inconsistency.

The information in this documentation may be changed at any time without notice. If you have gained access to this document without first agreeing to the Purolator E-Ship Resource Centre Agreement or the Purolator E-Ship Web Services Soft Launch Agreement (as applicable) you must cease using this document immediately and destroy any copies obtained.

© 2014 Purolator Inc., Purolator and the Purolator logo are trademarks of Purolator Inc. All rights reserved.

1 Getting Started

1.1 Introduction

To help you integrate Purolator E-Ship Web Services into your website or application, Purolator provides technical documentation and sample code for each Web Service. The documents include detailed technical information for each Web Service and detail Purolator's service offerings. Additional information and support is available on the Purolator E-Ship Developer Forum <http://www.purolatorwebservices.com>, on the E-Ship Web Services Wiki <http://www.purolatorwebservices.com/wiki2>, and through the Support section of the E-Ship Resource Centre <http://www.purolator.com/eship>.

The following documents can be downloaded in the Documentation and Sample Code section.

Introduction

Introduction to Purolator E-Ship Web Services

Service Availability Web Service

Validate origin and destination addresses and receive all available Purolator services, products, options and associated rules.

Estimates Web Service

Receive estimates based on origin and destination addresses as well as package and service inputs. Estimates are available as quick estimates using list prices or detailed estimates for customers with negotiated pricing.

Shipping Web Service

Create domestic, U.S. and International shipments. Please note that shipping labels and documents are created using the Get Documents Web Service.

Returns Management Web Service

Create domestic Returns Management shipments. Please note that shipping labels and documents are created using the Get Documents Web Service.

Get Documents Web Service

Retrieve shipment labels and documentation for shipments.

Tracking Web Service

Get detailed tracking information in real-time using a Purolator PIN / tracking number or shipment reference.

Pickup Web Service

Schedule, modify, validate and void pickups. View your pickup history and upcoming pickups.

Soap UI Project File

Example use cases and sample results for all Purolator E-Ship Web Services.

Certification

Guidelines to help ensure your Purolator E-Ship Web Services integration is ready to promote into Purolator's production environment.

Security Options (if applicable)

Security options available for your Purolator E-Ship Web Services implementation. This document will be provided when you are ready to take your website or application into Purolator's production environment. The availability of this document will depend on the type of web service integration you are executing.

1.2 Document Overview

This document is a technical specification that describes the Purolator Get Documents Web Service. It is geared towards developers who are building applications that require Purolator shipping documentation. This includes for example, shipping labels that need to be affixed to each shipment package, Commercial Invoices that must accompany U.S. and International Shipments, and NAFTA and FDA declaration forms for certain kinds of U.S. destined shipments.

The Get Documents Web Service is not a stand-alone web service but is used in conjunction with other web services. For example, once a shipment has been created using the Shipping Web Service or Returns Management Web Service then Get Documents would be used to obtain shipping labels. For a more thorough description on the shipping process please see the 'Purolator E-Ship Web Services - Introduction' document in the Documentation and Samples section of the E-Ship Resource Centre <http://www.purolator.com/eship>.

This document uses the following structure to explain our Get Documents Web Service:

Introduction (this chapter)

- Web Services overview
- Specification compliance
- Development platform compatibility
- Related resources
- Web services development process

Web Services

- Explanations of the characteristics of calls, as well as data types used
- Security information
- Service details for the Get Documents service
- Implementation considerations
- Error details and error codes and descriptions

Appendix

- Complete field listing
- Enumeration and expected values

1.3 Specification Compliance

The Web Services are designed to comply with the following specifications:

Specification Name	Website
Simple Object Access Protocol (SOAP) 1.1	http://www.w3.org/TR/2000/NOTE-SOAP-20000508/
Web Service Description Language (WSDL) 1.1	http://www.w3.org/TR/2001/NOTE-wsdl-20010315

Please visit the websites listed above for detailed information regarding SOAP and WSDL technologies.

1.4 Development Platforms

The Web Services work with any current SOAP/XML development environments.

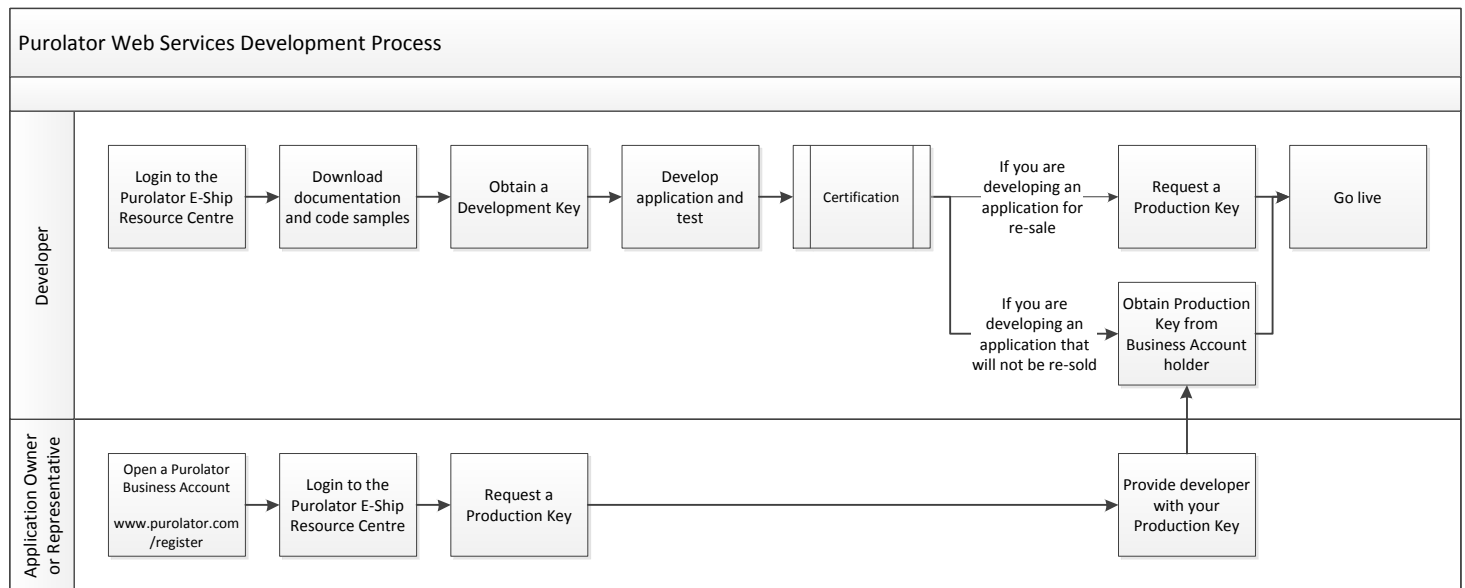
Note: Development platforms vary in their SOAP/XML implementations. Please refer to your platform's documentation for more information.

1.5 Related Resources

The Purolator E-Ship Resource Centre provides a full suite of documentation, sample code, sample SOAP messages, community-based support, and other resources to help you with your development projects. Be sure to visit www.purolator.com/eship to access all of the resources available to you and get support with your integration.

1.6 Purolator E-Ship Web Services Development Process

To integrate Purolator E-Ship Web Services within your application, follow the steps outlined below.



Web Services Developer Process

Follow these steps if you're a developer that will be integrating Purolator E-Ship Web Services.

Step 1: **Download Documentation and Code Samples**

View code samples and documentation for the Purolator E-Ship Web Service(s) you will be using.

Documentation and sample code, such as WSDL, is available in the Documentation and Sample Code section of the E-Ship Resource Centre at www.purolator.com/eship.

Step 2: **Obtain a Development Key**

Obtain a Development Key. This will allow you to test your code in the Purolator E-Ship Test Environment.

Note: Your Development Key, password, and test account number will be used to access the development environment.

Step 3: **Develop**

Develop your application using the code samples and documentation.

Step 4: Test

Test your application in the Purolator E-Ship Test Environment. Data in the Test Environment is sample data and should be used for testing purposes only.

Step 5: Certification

Based on qualification criteria, some applications may be required to go through Certification before moving to production. A Purolator representative will inform you of whether your application will be required to complete this step. Certification documentation is available in the Documentation and Sample Code section.

Step 6: Request a Production Key

Request a Production Key once your development and testing are complete. You will receive an inactive Production Key until we review your request.

If you're integrating Purolator E-Ship Web Services into an application that won't be re-sold, the Purolator Business Account holder must request the Production Key.

Replace the Development Key with the Production Key, and replace the development WSDL with the production WSDL. The Production Key will allow you to access account-specific shipping, tracking, estimates and returns information. Once activated, you will then be ready to use Purolator E-Ship Web Services.

Note: Your Production Key, password, and account number will be used to access the production environment.

Application Owner or Representative Process

Follow these steps if you have development resources and want to integrate Purolator E-Ship Web Services into your website or application that won't be re-sold.

Step 1: Open a Purolator Business Account

If you do not already have a Purolator Business Account, please visit www.purolator.com/register to get started.

Step 2: Request a Production Key

Login to the Purolator E-Ship Resource Centre and request a Production Key. The Production Key will allow you to access shipping, tracking, estimates and returns information specific to your Purolator Business Account.

You will receive an inactive Production Key until we review your request. A Purolator representative may contact you for security purposes before your Production Key is activated. Optional security features are available through consultation with Purolator.

Step 3: Provide your developer with the Production Key

Ask your developer to replace their Development Key with your Production Key. You will then be ready to use Purolator E-Ship Web Services.

2 Web Services

Web Services is a collection of programming technologies, including XML, Web Services Description Language (WSDL) and SOAP, which allow you to build programming solutions for specific messaging and application integration.

Web Services are, by definition, platform independent. Purolator Web Services allow developers to build custom applications that are independent of changes to the Purolator interface. This is achieved through backwards compatibility support for all versions of the web services.

2.1 Characteristics of Calls

All Web Service calls are:

- **Service Requests and Responses**—your client application prepares and submits a service request to Purolator E-Ship Web Services, which is processed and a response is returned, and the client application handles the response.
- **Synchronous**—once the API call is invoked, your client application waits until it receives a response from the service. Asynchronous calls are not supported.

2.2 Data Types

Value	Description
String	Character strings. Fields that are of data type string contain text and some have length restrictions. Please see the field reference list for format and restrictions.
Decimal	Decimal represents a subset of the real numbers, which can be represented by decimal numerals. The <u>·value space·</u> of decimal is the set of numbers that can be obtained by dividing an integer by a non-negative power of ten, i.e., expressible as $i / 10^n$ where i and n are integers and $n \geq 0$. Precision is not reflected in this value space; the number 2.0 is not distinct from the number 2.00. (The datatype <code>precisionDecimal</code> may be used for values in which precision is significant.) The order relation on decimal is the order relation on real numbers, restricted to this subset.
Int	The <code>precisionDecimal</code> datatype represents the numeric value and (arithmetic) precision of decimal numbers which retain precision; it also includes values for positive and negative infinity and for "not a number", and it differentiates between "positive zero" and "negative zero". This datatype is introduced to provide a variant of decimal that closely corresponds to the floating-point decimal datatypes described by the expected forthcoming revision of IEEE/ANSI 754. Precision of values is retained and values are included for two zeroes, two infinities, and not-a-number.
Enumeration	Enumeration constrains the <u>·value space·</u> to a specified set of values. Enumeration does not impose an order relation on the <u>·value space·</u> it creates; the value of the <u>·ordered·</u> property of the derived datatype remains that of the datatype from which it is derived.

2.3 Security

Purolator E-Ship Web Services authentication uses the standard HTTP Basic Authentication method allowing client programs to provide credentials in the form of a user name and password when making an HTTP request. Purolator E-Ship Web Services uses the standard HTTP protocol and in our case the username corresponds to the Development and Production Key that is issued to you by Purolator when you signed up for Purolator E-Ship Web Services. These credentials are sent from your client programs to our Purolator E-Ship Web Services servers securely using SSL.

2.4 Service Methods and Objects

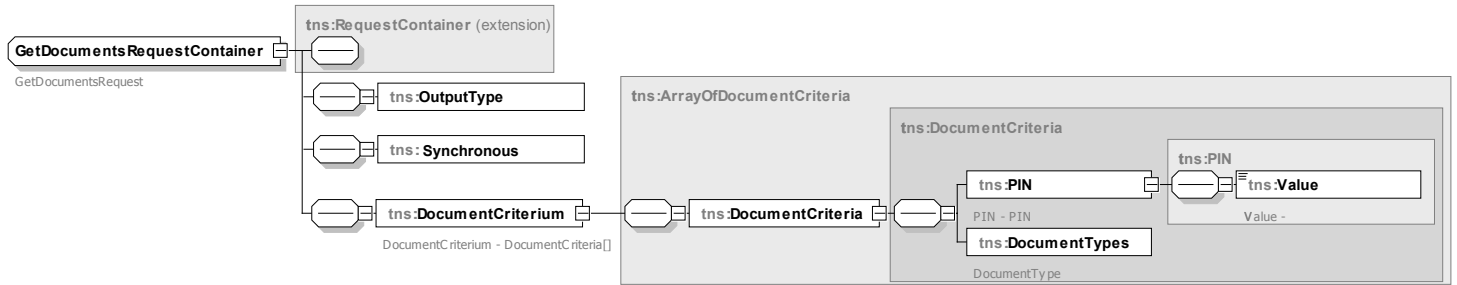
The following section will highlight the various methods associated with the shipment service. As well, visual representations will depict the XML schema of both the requests, and the responses of the service calls.

2.4.1 Get Documents

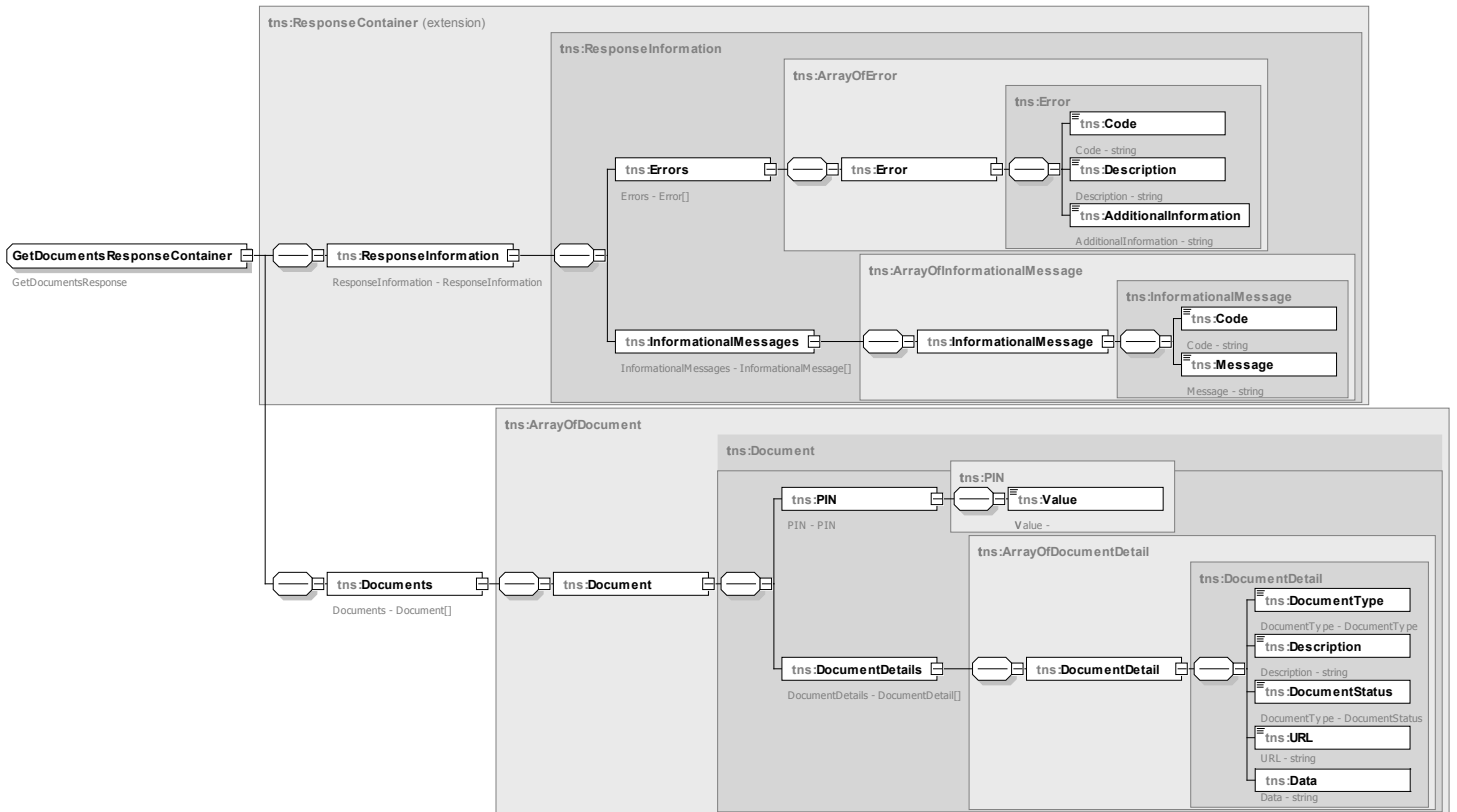
The `getdocuments` method is used to obtain shipment labels and/ or additional shipping documentation.

Note: The retention period for all shipping documents is same day. If you wish to reprint a shipping document, you must submit another `GetDocuments` request.

2.4.1.1 Request Diagrams – GetDocuments



2.4.1.2 Response Diagrams – GetDocuments

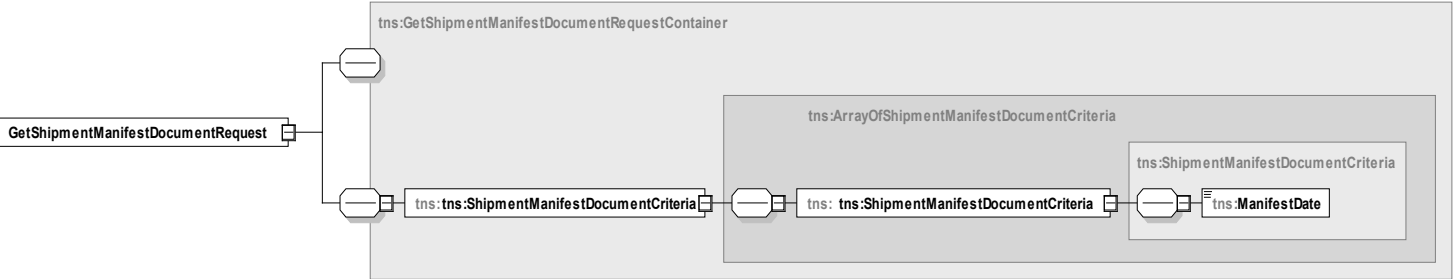


2.4.2 Get Shipment Manifest Document

The `GetShipmentManifestDocument` method is used to obtain the shipping manifest for the date requested.

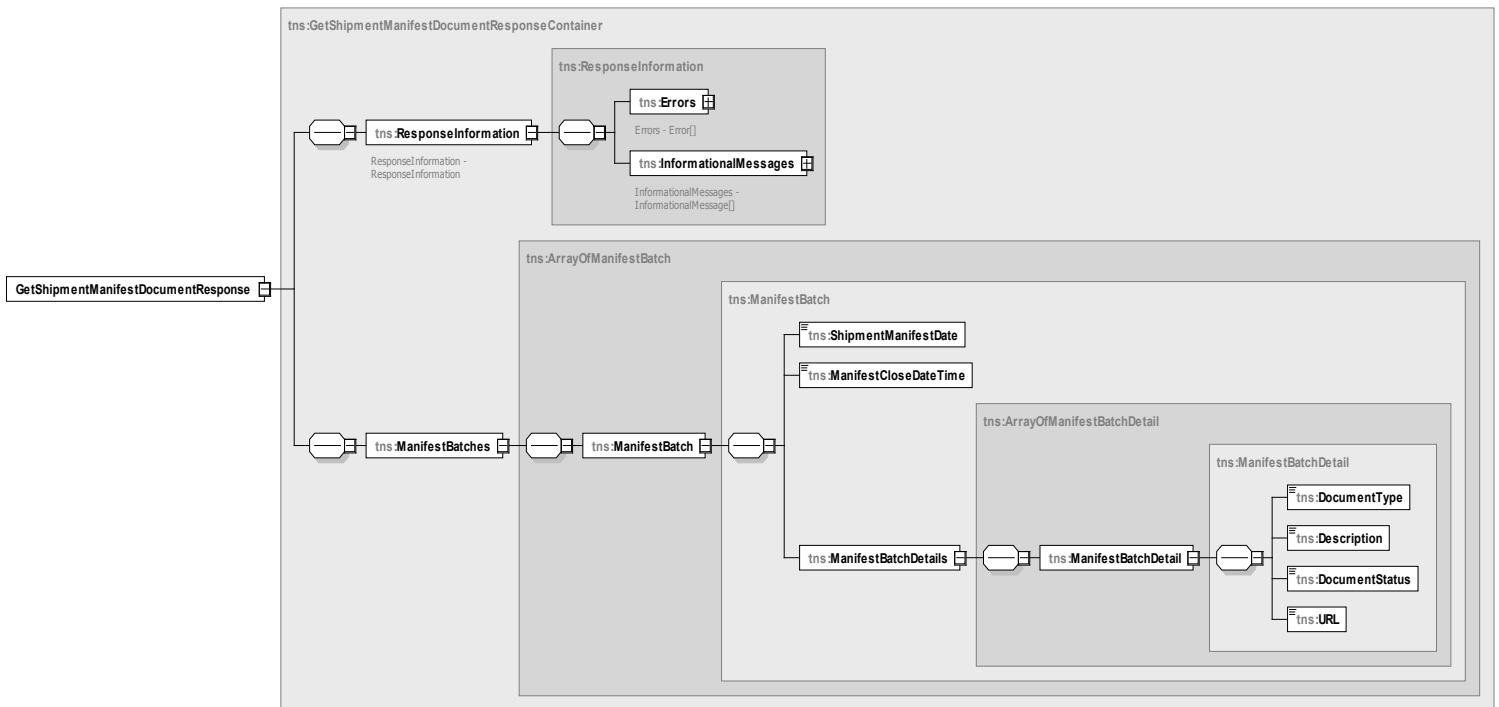
Note: The retention period for all shipping documents is same day. If you wish to reprint a shipping document, you must submit another `GetShipmentManifestDocument` request. Manifests can be reprinted up to 24 months after consolidated date.

2.4.2.1 Request Diagrams – GetShipmentManifestDocument



2.4.2.2

Response Diagrams – GetShipmentManifestDocument

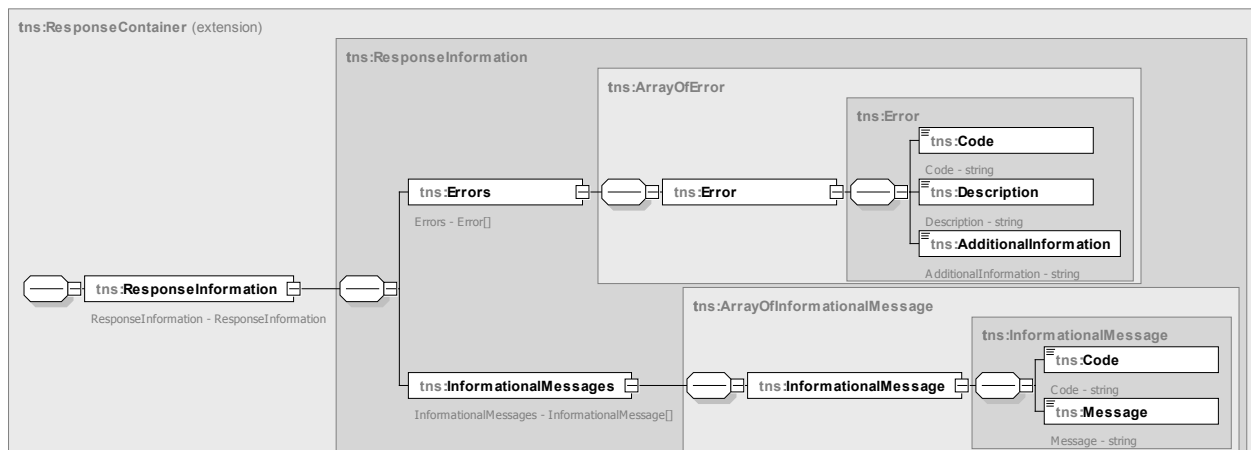


2.5 Error Handling

The Web Service calls return error data that your client application can use to identify and resolve runtime errors. If an error occurs during the invocation of most Web Services calls, then the API provides the following types of error handling:

- ? For errors resulting from badly formed messages, failed authentication, or similar problems, the Web Services returns a SOAP fault message.
- ? For errors resulting from a problem with the information sent to Purolator's systems, error codes and error descriptions are returned to the client application.

The following SOAP example shows the response to a request that contains an invalid postal code, and receiver phone number.



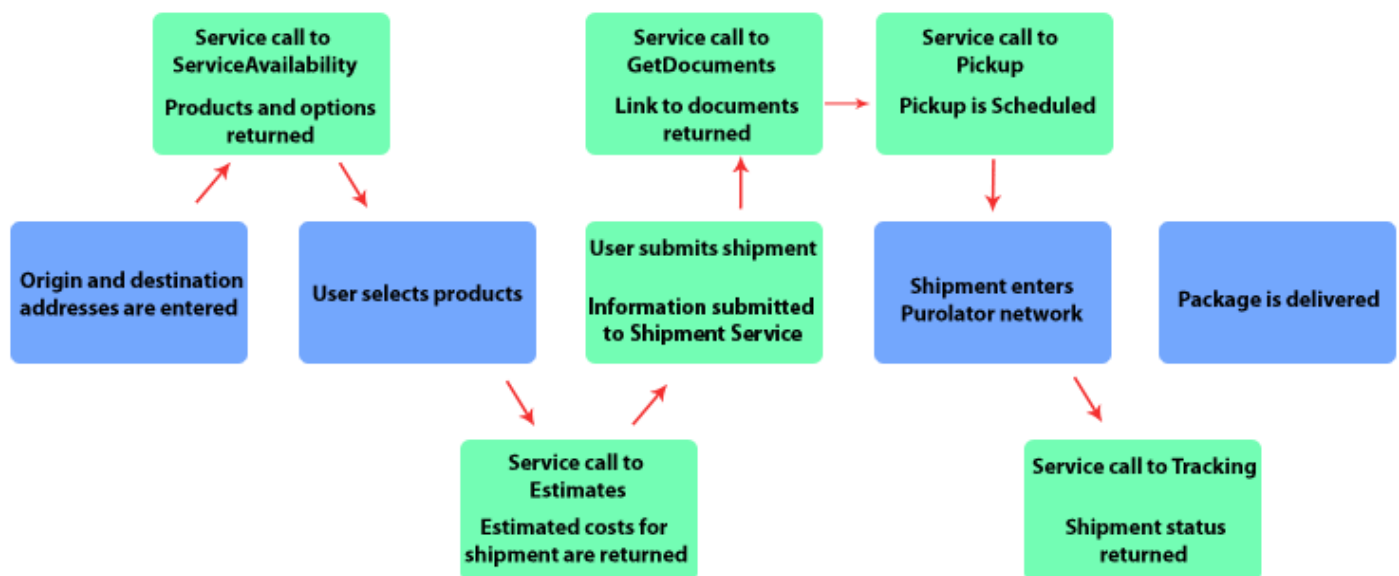
2.6 Implementation Considerations

As Purolator's business changes to meet the on-going needs of its customer, the business rules for certain products may change from time to time. As our business changes, your application doesn't have to. Purolator E-Ship Web Services offers customers an advanced Service Availability service that, when called, will return available products and services for a given origin and destination address. In addition to the available products and services, all business rules as well as product rules, product inclusions and exclusions, and products min/max dimensions and weights.

This can be used to offer your customers a greater experience, and offer all of Purolator's products and services, without needing to submit a request, only to discover that a particular product is not available for a particular address.

As well, additional Purolator E-Ship Web Services can be consumed to offer a richer customer experience. From estimates to tracking to retrieving necessary documentation, Purolator's E-Ship Web Services are designed to offer you the flexibility to consume the services required for your business.

Below is an example of how the all Purolator E-Ship Web Services may be utilized. (Green boxes denote service calls)



2.7 Error Messages / codes

Below is a list of all the error codes and descriptions.

Web Service Error

Code	Message
9001	Authorization failed -- Service Operation is inactive.
9000-1	Authorization failed -- Daily Threshold limit reached.
9000-2	Authorization failed -- Hourly Threshold limit reached.
9000-3	Authorization failed -- Minute Threshold limit reached.

Code	Message
1000100	System exception occurs.
1100000	One or more errors have occurred.
1100100	Field is missing.
1100101	The Account Number is a mandatory field.
1100102	The Account Name is a mandatory field.
1100103	The Account Number is a mandatory field.
1100200	Field is invalid.
1100411	Account Status is invalid.
1100424	PIN %pin% is invalid.
1100505	Undefined Account Type for Account %account%.
1100676	Invalid Printer Type.
3001099	Invalid PackageType Code: {0}. Valid values are: {1}.
3001100	Invalid TaxType Code: {0}. Valid values are: {1}.
3001101	Invalid DataType Code: {0}. Valid values are: {1}.
3001102	Invalid ScanType Code: {0}. Valid values are: {1}.
3001103	Invalid SignatureImageFormat Code: {0}. Valid values are: {1}.
3001104	Invalid ILLanguage Code: {0}. Valid values are: {1}.
3001105	Invalid ESOLanguage Code: {0}. Valid values are: {1}.
3001106	Invalid ESOWYesNo Code: {0}. Valid values are: {1}.
3001107	Invalid Option Code: {0}. Valid values are: {1}.
3001108	Invalid Product Code: {0}. Valid values are: {1}.

Code	Message
3001109	Invalid StreetType Code: {0}. Valid values are: {1}.
3001110	Invalid StreetDir Code: {0}. Valid values are: {1}.
3001111	Invalid StreetSuffix Code: {0}. Valid values are: {1}.
3001112	Invalid Country Code: {0}. Valid values are: {1}.
3001113	Invalid Province/State Code: {0}. Valid values are: {1}.
3001114	Translation Failed.
3001115	Service Call Failed.
3001116	Service Failed
3001117	A Web Service Error has occurred.
3001122	Invalid ChainOfSignature Code: {0}. Valid values are: {1}.
3001123	Invalid DangerousGoods Code: {0}. Valid values are: {1}.
3001124	Invalid DangerousGoodsClassification Code: {0}. Valid values are: {1}.
3001125	Invalid DangerousGoodsMode Code: {0}. Valid values are: {1}.
3001126	Invalid ExpressCheque Code: {0}. Valid values are: {1}.
3001127	Invalid ExpressChequeMethodOfPayment Code: {0}. Valid values are: {1}.
3001128	Invalid HoldForPickup Code: {0}. Valid values are: {1}.
3001129	Invalid SaturdayDelivery Code: {0}. Valid values are: {1}.
3001130	Invalid SaturdayPickup Code: {0}. Valid values are: {1}.
3001131	Invalid SpecialHandling Code: {0}. Valid values are: {1}.
3001132	Invalid TypesOfSpecialHandling Code: {0}. Valid values are: {1}.
3001133	Invalid DocumentType Code: {0}. Valid values are: {1}.

Code	Message
3001134	Invalid ExpressChequeAmount Code: {0}. Valid values are: {1}.
3001136	Service version is invalid
3001137	Invalid Declared Value Code: {0}. Valid values are: {1}.
3001138	Invalid OSNR Value Code: {0}. Valid values are: {1}.
3001139	Invalid registered shipping account specified in shipment
3001140	Invalid registered shipping account specified in return shipment
3001141	Postal / Zip code {0} code is not valid for Canada or U.S.
3001142	Country is not found
3001143	Suggested address is supplied
3001144	One or more addresses are invalid or suggestions are supplied
3001145	Invalid Piece Option Code: {0}. Valid values are: {1}.
3001146	Invalid Printer Type Option Code: {0}. Valid values are: {1}.
3001147	Invalid Address.
3001148	Postal / Zip Code is required to validate a Canadian / U.S. Address.
3001153	Service Error
3001180	Invalid Output Type for Asynchronous
3001181	Unsupport Document Type

APPENDIX A – Complete Field List

[ArrayOfDocument](#) | [ArrayOfDocumentCriteria](#) | [ArrayOfDocumentDetail](#) | [ArrayOfError](#) | [ArrayOfInformationalMessage](#) | [Document](#) | [DocumentCriteria](#) | [DocumentDetail](#) | [DocumentStatus](#) | [DocumentType](#) | [Error](#) | [GetDocumentsRequestContainer](#) | [GetDocumentsResponseContainer](#) | [GetManifestDocumentRequest](#) | [GetManifestDocumentResponse](#) | [InformationalMessage](#) | [Language](#) | [ManifestBatch](#) | [ManifestBatchDetails](#) | [PIN](#) | [RequestContext](#) | [ResponseContext](#) | [ResponseInformation](#) | [ShipmentManifestDocumentCriteria](#)

Input

Field Name	Description	Output Values
ComplexType RequestContext		
	Used by: RequestContext	
Version	Version Number of the Web Service Request.	1.0, 1.1, 1.2, or 1.3 (Currently)
Language	Used to define the Language of the Response text.	Enumeration. en fr
GroupID	For Future Use	
RequestReference	Reference Identifier for the Service Request. Used to aid in Customer Support.	String
UserToken	Activation Key belonging to a Purolator account and customer. The Activation Key is associated with the application that it will be used on and is only required for applications intended for resale.	String
ComplexType GetDocumentsRequestContainer		
OutputType	Specifies the output type of the document for various printing options.	Optional. Default is PDF Enumeration PDF, ZPL, DPL, EPL
Synchronous	When set to true, will return the binary data in the response. False will return a URL. False only applies to PDF.	Optional. Default is false. Boolean
DocumentCriterion		Complex Type DocumentCriterion
ComplexType DocumentCriterion		
	Used by: GetDocumentsRequestContainer	
ArrayOfDocumentCriteria		Complex Type ArrayOfDocumentCriteria
ComplexType ArrayOfDocumentCriteria		
	Used by: GetDocumentsRequestContainer/DocumentCriterion	
DocumentCriteria		Complex Type DocumentCriteria

Input

Field Name	Description	Output Values
ComplexType DocumentCriteria	Used by: GetDocumentsRequestContainer/DocumentCriteria	
PIN	PIN number for document request.	String Value
DocumentTypes	Array of document types	Complex Type DocumentType
ComplexType DocumentType	Used by: GetDocumentsRequestContainer/DocumentCriteria	
DocumentType		Enumeration: COSBillOfLading CustomsInvoice CustomsInvoiceThermal DangerousGoodsDeclaration DomesticBillOfLading DomesticBillOfLadingThermal ExpressChequeReceipt ExpressChequeReceiptThermal FCC740 FDA2877 InternationalBillOfLading InternationalBillOfLadingThermal NAFTA
ComplexType GetShipmentManifestDocumentRequest	Used by: GetShipmentManifestDocumentRequestContainer	
ShipmentManifestDocumentCriteria	Array to submit document request.	Complex type ShipmentManifestDocumentCriteria
ComplexType ShipmentManifestDocumentCriteria	Used by: GetShipmentManifestDocumentRequest	
ManifestDate	Required. Date of the shipping date.	Date. Format: YYYY-MM-DD

Output

Field Name	Description	Output Values
Element GetDocumentsResponseContainer		
ResponseInformation Documents		Complex Type ResponseInformation Complex Type ArrayOfDocuments
Element ResponseInformation	Used by: ResponseInformation	
ResponseReference		

Output

Field Name	Description	Output Values
Errors	Array of errors	Complex Type ArrayOfError
InformationalMessages		Complex Type ArrayOfInformationalMessages
ComplexType ArrayOfError	Used by: ArrayOfError, ResponseInformation/Errors	
Error	Error information returned by the service call.	Complex Type Error
ComplexType Error	Used by: Error, ArrayOfError/Error	
Code	Error Code	
Description	Error Description	
AdditionalInformation	Additional error information	
ComplexType ArrayOfInformationalMessage	Used by: ArrayOfInformationalMessage, ResponseInformation/InformationalMessages	
InformationalMessage		
Element ArrayOfPIN	Used by: ArrayOfInformationalMessage, ResponseInformation/InformationalMessages	
PIN	Array of PIN	
Element ArrayOfDocument	Used by: GetDocumentsResponseContainer/Documents	
Document		Complex Type Document
Element Document	Used by: GetDocumentsResponseContainer/Documents	
PIN		Complex Type PIN
DocumentDetails		Complex Type ArrayOfDocumentDetail
Element PIN	Used by: GetDocumentsResponseContainer/Documents	
Value	PIN number	String
Element ArrayOfDocumentDetail	Used by: GetDocumentsResponseContainer/Documents	
DocumentDetail		Complex Type DocumentDetail
Element DocumentDetail	Used by: GetDocumentsResponseContainer/Documents	
DocumentType		Complex Type DocumentType

Output

Field Name	Description	Output Values
Description	Description of document type	String
DocumentStatus	Status of document	Enumeration: Pending Queued Processing Completed Error
URL	URL of Documents	String
Data	Contains document binary.	String. Unbound.
ComplexType		
ResponseContext		
ResponseReference		
Element GetShipmentManifestDocument Response	Used by: GetShipmentManifestDocument	
ResponseInformation		Complex Type ResponseInformation
ManifestBatches		Complex Type ManifestBatches
Element ManifestBatches	Used by: GetDocumentsResponseContainer/Documents	
ManifestBatch		Complex Type ManifestBatch
Element ManifestBatch	Used by: GetDocumentsResponseContainer/Documents	
ShipmentManifestDate ManifestCloseDateTime		
ManifestBatchDetails		Complex Type ArrayOfManifestBatchDetails
Element ArrayOfManifestBatchDetails	Used by: GetDocumentsResponseContainer/Documents	
DocumentType Description DocumentStatus URL		