



TNT Express

ExpressConnect

**ExpressConnect Pricing
Integration Guide
v3 Schema**

Revision History

Date	Version	Description
26/07/2012	1.0	First revision of technical user guide for ExpressConnect Pricing
30/07/2012	1.1	Minor changes following review
31/07/2012	1.2	Addition of error messages and sample client request
17/08/2012	1.3	Correction made following UAT
18/09/2012	1.4	Removed invalid elements from example UK Domestic request.
11/03/2013	2.0	Updated to use ExpressConnect Pricing 3.0
02/07/2013	2.1	Removed the redundant insuranceCurrency element from the schema

Table of Contents

1. Introduction	5
1.1 Legend	6
2. Registration	7
3. Making a request to the TNT server	8
4. Example XML Price Request Document	9
5. Input XML format	10
5.1 Header	10
5.2 The priceRequest element	11
5.3 The priceCheck element	12
5.4 The rateId element	13
5.5 The sender element	13
5.6 The delivery element	13
5.7 The collectionDateTime element	14
5.8 The account element	14
5.9 The product element	15
5.10 The insurance elements	15
5.11 The currency element	16
5.12 The termsOfPayment element	16
5.13 The priceBreakDown element	16
5.14 The consignmentDetails elements	17
5.15 The pieceLine element	17
6. Processing the XML Response	19
6.1 High level description of an XML Response Document	19
6.2 Header and Root Element	20
6.3 The priceResponse Section	20
6.4 The ratedService Section	21
6.5 The product Section	21
6.6 The chargeElements Section	22
7. Errors	23
7.1 The runtimeError section	24
7.2 The parseError section	24
7.3 The brokenRule section	25
7.4 Table of application generated error codes, messages and resolutions	26
8. Connecting to ExpressConnect Pricing	34
8.1.1 XML	34
9. Appendix A: XML elements definition (input)	35
10. Appendix B: XSD Data Types	41

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

1. Introduction

ExpressConnect Pricing is part of the ExpressConnect family, providing B2B interfaces into TNT's operational and financial systems. The ExpressConnect Pricing interface is used to provide an estimated cost for the delivery of the consignment.

This manual provides a technical guide to the ExpressConnect Pricing interface. It is designed to help developers understand the interface sufficiently to program an application to request an estimated cost for the consignment. With the use of XML (extensible mark-up language) technology, this will provide price checking functionality for multiple PRICECHECK submissions. The majority of examples in the document will be XML based as it is easier to understand the data involved.

Return data from TNT will be formatted as XML, again in a predetermined structure. When this data is returned to a client, they may parse it for storage or presentation, as they require.

An ExpressConnect Login Id will be arranged by your TNT representative. The customer must supply a list of valid TNT accounts numbers to be used with the system. A secure connection is thus set up, using both authentication and secure protocols, to submit requests and to receive processed priceCheck labels based on the published URL:

<https://express.tnt.com/expressconnect/pricing/getprice>

Sample scripts are provided to show how a connection can be achieved together with example requests and responses to illustrate the data required; see [Connecting to ExpressConnect Pricing](#). Consideration is given below to the networking and security requirements to ensure that this is successful.

This document is structured as follows:

- Registration
- Making a request to the TNT server
- Example XML Price Request Document
- Input XML Format - the structure and content of the request
- Processing the response - information about the data that will be returned
- Errors - possible error messages and the steps you can take to resolve them
- Connecting to ExpressConnect Pricing
- Appendices

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

1.1 Legend

The following conventions have been used throughout this document.

Normal	The majority of text in this document is in this style. Section in this style are part of the narrative of the document
Code	Sections or words in this text indicate a section of XML, XML element, or section of code.
[0..1]	Digits within square brackets indicate the number of times an element may occur in an XML document. Examples include: [1] The element must appear once in the document [0..*]. The element may appear once, many times (unlimited) or not at all. [0..1] The element is optional. If it appears, it must appear only once. [1..5] The element may appear any number of times between 1 and 5 times
xsd:string	This indicates one of the schema types, in this case a string. More information on defined schema data types can be found at http://www.w3.org/TR/xmlschema-2/#built-in-datatypes
...	Means that the section has been omitted for the sake of clarity. This usually means that the omitted elements are described elsewhere or that the section where they appear is a repetition of a previous stanza. For example: <pre> <house> <room> <width unitOfMeasure="m">12</width> </room> <!-- the next room contains the same dimensions sections as the one above. --> <room>...</room> </house> </pre>

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

2.Registration

Each customer is set up with a username and password, required for all communications with the system. The same username and password is used for ExpressConnect Shipping Pricing and Tracking.

Please contact your local TNT representative to complete a registration request.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

3. Making a request to the TNT server

To make a Price request, you must construct an XML file which conforms to the standard set out in this document. The submission will be validated to check for any problems with the structure of the XML. This facility is provided to allow you to self-diagnose problems with the XML.

Before sending an XML document over the internet to TNT, you should verify that you understand the XML format by successfully using the Express Web Services Pricing site provided by TNT, contact your TNT representative for further information. The audience for the website is intended to be developers who can use it as a tool to test their XML and to analyse results, prior to and during the development of client applications. It includes a Test Harness that allows the developer to submit sample XML to the service.

Please note that the site requires you to enter your user id and password before displaying the test harness.

- ExpressConnect Pricing does NOT support non-ASCII characters.
- The demonstration site links to a production like environment so that it replicates exactly the results that the customer will achieve with their completed application.

Having made a successful submission of an XML document via the test page, you are ready to set up a socket connection and make a programmatic submission using HTTP POST to the following URL:

<https://express.tnt.com/expressconnect/pricing/getprice>

Please be aware that all submissions to the aforementioned URL will the require you to supply your user id and password in the HTTP header of the request using Basic authentication.

More detailed information on connecting to the ExpressConnect servers can be found in [Connecting to ExpressConnect Pricing](#) section

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

4. Example XML Price Request Document

An example is provided below of a typical price request. The request specifies a single consignment, with no specific service.

There can be multiple `priceCheck` elements in a request, so each one contains the `rateId` element to act as a key to uniquely identify it. This is then used in responses to identify which one each returned price, error message, etc relates to.

```
<?xml version="1.0" encoding="UTF-8"?>
<priceRequest>
  <appId>PC</appId>
  <appVersion>3.0</appVersion>
  <priceCheck>
    <rateId>rate2</rateId>
    <sender>
      <country>GB</country>
      <town>Atherstone</town>
      <postcode>CV9 2RY</postcode>
    </sender>
    <delivery>
      <country>ES</country>
      <town>Alicante</town>
      <postcode>03006</postcode>
    </delivery>
    <collectionDateTime>2013-03-11T15:01:00</collectionDateTime>
    <product>
      <type>N</type>
    </product>
    <currency>GBP</currency>
    <priceBreakDown>true</priceBreakDown>
    <consignmentDetails>
      <totalWeight>1.25</totalWeight>
      <totalVolume>0.1</totalVolume>
      <totalNumberOfPieces>1</totalNumberOfPieces>
    </consignmentDetails>
  </priceCheck>
</priceRequest>
```

Figure 1

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

5.Input XML format

The input XML format for ExpressConnect Pricing is a list of price requests.

The structure for a Price Request is described below in detail. Alternatively, refer to Appendix A which contains a summary of the points below.

Note

It should be noted that XML defines a number of characters which are reserved. These include the greater-than (>), less-than (<), ampersand (&), and percent (%) characters. Where these appear in the data which is being submitted to ExpressLabel, the characters must be escaped or the content surrounded with a CDATA section.

A common requirement is to submit an address which includes a company name such as: "Andrews & Plummer". The ampersand must therefore be escaped as per the XML rules (&#amp;) or alternatively the whole or part of the text must be wrapped in a CDATA section as follows:

```
...
<name><![CDATA[Andrews & Plummer]]></name>
```

The request has the following structure:

- Header – always required, this defines the XML document
- Price request– A list of `priceCheck` elements for which a price is required

5.1 Header

The header section will begin every ExpressConnect Pricing request XML document submitted to TNT.

This contains the XML declaration, which contains the character encoding used for the document and the standalone attribute, which should be set to “no”:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

Figure 2

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

5.2 The priceRequest element

A Price Request at a high level is structured as shown in Figure 3:

```
<priceRequest>
  <appId>PC</appId>
  <appVersion>3.0</appVersion>
  <priceCheck>
    ...
  </priceCheck>
  <priceCheck>
    ...
  </priceCheck>
</priceRequest>
```

Figure 3

A priceRequest contains 1 or more priceCheck elements. This allows batching of pricing requests. Each priceCheck element contains the set of information needed to generate an estimate cost for the priceCheck referenced.

A priceCheck element contains a rateId key attribute that identifies it uniquely *within the request*. The response will associate pricing data and validation errors with their priceCheck requests through this key. The value of this key must be of type string and is only valid in ExpressConnect Pricing for a single transaction - i.e. ExpressConnect Pricing does not retain any history of the keys used once a request has been processed and a response sent to the client system.

Additionally, the priceRequest also contains a few elements with details of the pricing interface being used. These are:

- The **appId** element should be defaulted to “PC”. This element is used by various source applications to highlight where the request was made; it has no bearing on the estimated cost returned.
- The **appVersion** denotes the version of ExpressConnect Pricing that will be used. If a value of 3.0 is supplied then the new interface functionality will be evoked. If no appVersion is supplied it will be assumed that the existing interface is used. Only 3.0 will be considered valid, if any other value is supplied then an error will be returned.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

5.3 The priceCheck element

Each price request must contain at least one priceCheck element, which is made up of the following parts:

```

<priceCheck>
  <rateId>rate1</rateId>
  <sender>
    <country>GB</country>
    <town>Atherstone</town>
    <postcode>CV9 2RY</postcode>
  </sender>
  <delivery>
    <country>GB</country>
    <town>Hinckley</town>
    <postcode>LE10 1YG</postcode>
  </delivery>
  <collectionDateTime>2013-03-11T13:58:00</collectionDateTime>
  <product>
    <id>AM</id>
    <type>N</type>
    <options>
      <option>
        <optionCode>CF</optionCode>
      </option>
    </options>
  </product>
  <insurance>
    <insuranceValue>110.00</insuranceValue>
    <goodsValue>100.00</goodsValue>
  </insurance>
  <currency>GBP</currency>
  <priceBreakDown>true</priceBreakDown>
  <consignmentDetails>
    <totalWeight>2.5</totalWeight>
    <totalVolume>0.1</totalVolume>
    <totalNumberOfPieces>2</totalNumberOfPieces>
  </consignmentDetails>
  <pieceLine>
    <numberOfPieces>2</numberOfPieces>
    <pieceMeasurements>
      <length>1</length>
      <width>2.5</width>
      <height>0.5</height>
      <weight>1.25</weight>
    </pieceMeasurements>
    <pallet>true</pallet>
  </pieceLine>
</priceCheck>

```

Figure 5

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

5.4 The **rateId** element

The **rateId** is a mandatory element which identifies the priceCheck uniquely within the request.

```
<rateId>rate1</rateId >
```

Figure 6

It is recommended that the value is unique within the priceRequest to allow easy identification of results to requests.

5.5 The **sender** element

The **sender** is a mandatory element and contains information on the origin of the consignment.

```
<sender>  
  <country>GB</country>  
  <town>Atherstone</town>  
  <postcode>CV9 2RY</postcode>  
</sender>
```

Figure 7

The **country** is the country where the shipment will be collected and sent from. It is represented by the 2 digit ISO 3166-1 Alpha-2 country code. It is a mandatory element.

The **town** and the **postcode** should be used to more accurately locate where the shipment will be collected and sent from. The postcode and town combination must match and offers the greatest level of accuracy for the estimated cost.

If neither the postcode / town are specified, then the default collection depot for the country will be used. This may reduce the accuracy of the returned price.

If all three are supplied then they will be validated but the postcode and town will be used in the cost estimation process.

5.6The **delivery** element

The **delivery** is a mandatory element and contains information on the destination of the consignment.

```
<delivery>  
  <country>GB</country>  
  <town>Hinckley</town>  
  <postcode>LE10 1YG</postcode>  
</delivery>
```

Figure 8

The **country** is the country where the shipment will be collected and sent from. It is represented by the 2 digit ISO 3166-1 Alpha-2 country code. It is a mandatory element.

The **town** and the **postcode** should be used to more accurately locate where the shipment will be collected and sent from. The postcode and town combination must match and offers the greatest level of accuracy for the estimated cost.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

If neither the postcode / town are specified, then the default collection depot for the country will be used. This may reduce the accuracy of the returned price.

If all three are supplied then they will be validated but the postcode and town will be used in the cost estimation process.

5.7 The **collectionDateTime** element

The **collectionDateTime** identifies when the consignment will be collected

```
<collectionDateTime>2013-03-11T13:58:00</collectionDateTime>
```

Figure 9

This is an optional element that will default today's date and time. The date is the date the consignment will be collected. The format of a date time is `yyyy-mmddThh:MM:ss`

Format Date Component Description

yyyy - The year in four digits. e.g. 2013

mm - The month in digits, January is 01, December is 12

dd - Day of the month. Valid range is 1 to 31

hh - The collection hour in 24 hour notation (Please note that the 24:00:00 notation for midnight is not currently supported and 00:00:00 should be used instead)

MM - The minutes from 00 to 60. If the hour is 24, the minute value must be 00

ss - The seconds from 00 to 60. If the hour is 24, the second value must be 00

All other characters - the dashes, colons, and the capital T which separates the day from the hour - are literals which must appear as they are shown above.

5.8 The **account** element

The **account** element contains information on the customers account. This is an optional element, but must be supplied to obtain accurate price estimates.

```
<account>
  <accountNumber>1234567890</accountNumber>
  <accountCountry>NL</accountCountry>
</account>
```

Figure 10

The **account** element consists of an **accountNumber** and **accountCountry** element which are both mandatory.

The **accountNumber** must be a valid TNT account number as supplied by your TNT representative.

The **accountCountry** is the country where the TNT Account is held. It is represented by the 2 digit ISO 3166-1 Alpha-2 country code.

The account number must be authorised to be used by the userid and password supplied in the `login` element.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

5.9 The product element

This contains information on the product and product type.

```

<product>
  <id>15N</id>
  <division>G</ division >
  <type>N</type>
  <!--Element option is optional, maxOccurs=4-->
  <options>
    <option>PR</option>
  </options>
</product>

```

Figure 11

The **id** is an optional element, and if none is supplied then all possible products that are valid for the origin/destination combination will be used for pricing purposes up to a maximum of 10 products. If you wish to determine the estimate cost for a shipment using a particular service then the service should be populated within the element. It is recommended that this element is populated if known, to offer a more rapid response to the request. If this element is not provided, then an estimated cost will be provided for all feasible services between the origin and destination based on the shipment details.

The **division** code is an optional element that will default to G for international shipments, D for Global link domestic, H for German domestic and 010 for UK domestic.

The **type** is a mandatory element which identifies whether you are shipping a “D” Document (paper/manuals/reports) or “N” Non-document (packages). Document services are not offered for a number of TNT domestic shipments. For domestic shipment within these countries please use a type of “N” for Non-document.

The **options** element can contain up to 4 child **option** elements, which each must contain a valid code for the services requested. For example, PR (Priority).

5.10The insurance elements

In order to retrieve a price for your requested services it is necessary to include the following additional elements in your request.

```

<insurance>
  <insuranceValue>110.00</insuranceValue>
  <goodsValue>100.00</goodsValue>
</insurance>

```

Figure 12

The **goodsValue** must be in the same currency as the **insuranceValue** Validation is enforced which states that the **insuranceValue** cannot be more than 110% of the **goodsValue**.

If an option code of ‘IN’ is selected, then the following elements become mandatory.

Please refer to the TNT terms and conditions for further information on TNT’s liability and coverage.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

5.11 The **currency** element

This element identifies the **currency** that the estimate costs should be returned in.

```
<currency>GBP</currency>
```

Figure 13

It is represented by the 3 digit ISO 4217 Alpha-3 currency codes.

5.12 The **termsOfPayment** element

This identifies whether this is an import (receiver pays) or export (sender pays) shipment.

```
<termsOfPayment>S</termsOfPayment>
```

Figure 14

A **termsOfPayment** type of “S” represents a sender pays shipments, “R” represents a receiver. If this element is not supplied then the payment type is defaulted to sender.

5.13 The **priceBreakDown** element

This is an optional element that determines if a full price breakdown is returned or just the high level rate. The default value is false.

```
< priceBreakDown>>true</ priceBreakDown>
```

Figure 16

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

5.14 The **consignmentDetails** elements

The **consignmentDetails** element contains summary piece information for the consignment you wish to request a price for.

```
<consignmentDetails>
  <totalWeight>1.25</totalWeight>
  <totalVolume>0.1</totalVolume>
  <totalNumberOfPieces>1</totalNumberOfPieces>
</consignmentDetails>
```

Figure 17

Key elements in determining the price are the **totalWeight** and **totalVolume** of the shipment. These data can be supplied at piece level using the **pieceLine** or consignment level using the **consignmentDetails**. If both element types are supplied then they must match, if only one is supplied then the other is defaulted.

The **totalWeight** element contains the total weight of the shipment in Kilograms and is a mandatory element.

The **totalVolume** element contains the total volume of the shipment in cubic metres and is a mandatory element for Non-documents, **type** = 'N'.

The **totalNumberOfPieces** element contains the number of items within the shipment. This is a mandatory element.

If no **pieceLine** is specified then a default **pieceLine** is created using the values specified for the **consignmentDetails**. The calculation is as follows:

- **weight** - This is calculated by dividing the **totalWeight** by the **totalNumberOfPieces**.
- **length/width/height** - These are calculated by firstly dividing the **totalVolume** by the **totalNumberOfPieces**. This provides the volume of an individual piece which can then be further broken down to an arbitrary length, width and height by taking the cube root.

5.15 The **pieceLine** element

The **pieceLine** element is used to describe the individual piece lines which make up the consignment in greater detail. It is an optional element and can occur multiple times up to a maximum of 99 piece lines.

```
<pieceLine>
  <numberOfPieces>1</numberOfPieces>
  <pieceMeasurements>
    <length>1</length>
    <width>2</width>
    <height>3</height>
    <weight>1.25</weight>
  </pieceMeasurements>
  <pallet>true</pallet>
</pieceLine>
```

Figure 18

If both the **pieceLine** and **consignmentDetails** are specified then the total number of pieces and weight must match or an error will be returned.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

The **numberOfPieces** element is mandatory and contains the number of items within the `pieceLine`.

The **pieceMeasurements** element is mandatory and contains the precise measurements for the pieces within the line. This consists of the:

- The **length** of the piece in metres.
- The **width** of the piece in metres.
- The **height** of the piece in metres.
- The **weight** of the piece in kilograms.

The **pallet** element can be set to true or false and indicates whether the piece line will be contained on a pallet. It is currently only used for UK Domestic pricing requests. This will default to false if not specified.

If no **consignmentDetails** are specified the attributes are calculated using the values specified for the `pieceLine`. The calculation is as follows:

- **totalWeight** - This is calculated by first multiplying each piece line weight by the `numberOfPieces` in the line. The `totalWeight` is then equal to the sum of the weight from each `pieceLine`.
- **totalNumberOfPieces** - This is calculated by adding the number of pieces on each `pieceLine` together.
- **totalVolume** - This is calculated for the `pieceLine` by multiplying the length x width x height x `numberOfPieces`. The volumes from each `pieceLine` can then be added together to get the `totalVolume`.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

6.Processing the XML Response

The document that is returned by the system is a standard XML document, which contains a generic pricing response that will apply to all types of pricing requests.

6.1 High level description of an XML Response Document

The first section contains the usual XML header; see [Header and Root Element](#). The ROOT node contains all the responses to the price requests. Within the root node there are 2 main sections

- `<priceResponse>` - This element provides the pricing information that is returned per service
- `<errors>` - This provides a breakdown of any errors or warning generated by the request. See the section on [Errors](#) for more details.

The sections that follow provide a more detailed explanation of each part of the response.

Below is a simple example of the new generic response XML:

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<document>
  <requestId>28817</requestId>
  <errors>
    <brokenRule>
      <rateId>rate2</rateId>
      <messageType>W</messageType>
      <code>P13</code>
      <description>Standard Rates</description>
    </brokenRule>
  </errors>
  <priceResponse>
    <ratedServices>
      <rateId>rate2</rateId>
      <currency>GBP</currency>
      <ratedService>
        <product>
          <id>09N</id>
          <productDesc>9:00 Express</productDesc>
        </product>
        <totalPrice>288.47</totalPrice>
        <totalPriceExclVat>238.40</totalPriceExclVat>
        <vatAmount>50.07</vatAmount>
      </ratedService>
    </ratedServices>
  </priceResponse>
</document>
```

Figure 19

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

6.2 Header and Root Element

As explained above, the header section will begin every ExpressConnect Pricing response XML document sent back by TNT. This contains the XML declaration, which defines the encoding of the document. The root element, `Document`

If the system has encountered a fatal error, this will appear as a `fault`. This is discussed in more details in the section on [Errors](#).

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<document>
<requestId>28817</requestId>
...
</document>
```

Figure 20

The `requestId` is a unique identifier for your price request which should be provided to the agent when contacting the service centre with any queries you have about the ExpressConnect pricing service. It will allow quick recovery of the request and response XML and enable a more timely response to any enquiry.

6.3 The `priceResponse` Section

A `priceResponse` section contains estimated cost information for a service, which may be used for the shipment request details defined in the `priceCheck` element.

There may be up to a maximum of 10 `priceResponse` sections per `priceCheck`.

```
<priceResponse>
  <ratedServices>
    <rateId>rate2</rateId>
    <currency>GBP</currency>
    <ratedService>
      <product>
        <id>09N</id>
        <productDesc>9:00 Express</productDesc>
      </product>
      <totalPrice>288.47</totalPrice>
      <totalPriceExclVat>238.40</totalPriceExclVat>
      <vatAmount>50.07</vatAmount>
    </ratedService>
  </ratedServices>
</priceResponse>
```

Figure 21

Each `priceResponse` contains a `ratedServices` element. This is made up of one or more `ratedService` which contain a price for a particular service option combination from the original request.

At the `ratedServices` level the following high level information is returned:

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

- The **rateId** is used to associate the `priceResponse` with the corresponding `priceCheck`.
- The **currency** element describes the currency that the price is displayed in.

6.4 The `ratedService` Section

For each `ratedService` the following information is returned:

- The **totalPrice** element contains the total estimated price for the service including VAT.
- The **totalPriceExclVat** element contains the total estimated price for the service excluding VAT.
- The **vatAmount** element contains the estimated amount of VAT for the service requested.
- The **product** element contains a summary of the service & option information to which the price request relates. See `product` section for more information.
- The **chargeElements** element contains an optional breakdown of any charges which are included within the price such as VAT and fuel supplements. See the `chargeElements` section for more information.

6.5 The `product` Section

The **product** element within the `ratedService` contains a summary of the service & option information to which the price request relates.

```

<product>
  <id>15N</id>
  <productDesc>Express</productDesc>
  <options>
    <option>
      <optionCode>PR</optionCode>
      <optionDesc>Priority</optionDesc>
    </option>
  </options>
</product>

```

Figure 22

The following information is returned:

- The **id** is the identifier for the service to which the price relates e.g. 15N
- The **productDesc** is the description of the service to which the price relates e.g. Express.
- The **options** element can contain up to 4 options that relate to the price. Each **option** is made up of an **optionCode**, which is a unique identifier and an **optionDesc** which describes the option.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

6.6 The chargeElements Section

The **chargeElements** section contains an optional breakdown of any charges (**chargeElement**) which are included within the price such as VAT and fuel supplements. This additional information will only be displayed if the **priceBreakDown** element is set to true in the request. An example of a charge breakdown is shown below:

```
<chargeElements>
  <chargeElement>
    <chargeItem>1</chargeItem>
    <chargeCategory>SURCHARGE</chargeCategory>
    <chargeCode>ESS00</chargeCode>
    <description>ENHANCED SECURITY SURCHARGE</description>
    <chargeValue>0.86</chargeValue>
    <vatIndicator>true</vatIndicator>
  </chargeElement>
  <chargeElement>
    <chargeItem>2</chargeItem>
    <chargeCategory>SURCHARGE</chargeCategory>
    <chargeCode>FSI00</chargeCode>
    <description>FUEL SURCHARGE REFERENCIADO</description>
    <chargeValue>43.22</chargeValue>
    <vatIndicator>true</vatIndicator>
  </chargeElement>
  <chargeElement>
    <chargeItem>3</chargeItem>
    <chargeCategory>VAT</chargeCategory>
    <chargeCode>VAT02</chargeCode>
    <description>IVA 21%</description>
    <chargeValue>55.35</chargeValue>
    <vatIndicator>false</vatIndicator>
  </chargeElement>
</chargeElements>
```

Figure 23

For each **chargeElement** the following information is returned:

- The **chargeItem** is an identifier for the **chargeElement** within the list of **chargeElements** returned.
- The **chargeCategory** is a descriptor of the type of charge element being returned. This will either be SURCHARGE or VAT.
- The **chargeCode** is a unique identifier for the charge being applied.
- The **description** provides a more detailed explanation of the charge being applied e.g. Fuel Surcharge.
- The **chargeValue** is the cost of the charge in the currency.
- The **vatIndicator** defines whether the VAT is applied to the charge.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

7.Errors

There are a number of different errors that may occur when using ExpressConnect Pricing. Many of these are likely to be encountered in the initial development phase and are concerned with the format of the XML message and the presence of data items.

The remaining messages are concerned with validation of the data items and the availability of the service. The error messages are shown below:

It would be sensible to ensure that your code is capable of handling all of the potential error messages returned by ExpressConnect Pricing.

All errors are contained with the **errors** element as shown in the example below:

```
<errors>
  <brokenRule>
    <rateId>rate2</rateId>
    <messageType>W</messageType>
    <code>P13</code>
    <description>Standard Rates</description>
  </brokenRule>
  <brokenRule>
    <rateId>rate3</rateId>
    <messageType>E</messageType>
    <code>P42</code>
    <description>The country must be entered</description>
  </brokenRule>
</errors>
```

There are three error types which can be returned within the `errors` element

- `runtimeError`
- `parseError`
- `brokenRule`

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

7.1 The runtimeError section

The **runtimeError** section is for errors which are not resolvable by the customer e.g. database unavailable. The customer would be advised to contact their administrator for assistance.

The following excerpt shows an example of a runtime error:

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<document>
  <errors>
    <runtimeError>
      <errorReason>ExpressConnect Pricing request has failed. If you continue to receive this error, please
contact your local service centre for further assistance. </errorReason>
    </runtimeError>
  </errors>
```

The key information in the `runtimeError` error is as follows:

- **errorReason** – This element contains a description indicating that the pricing request has failed and what action the customer should take. Normally the only course of action would be to resubmit the request or contact the local service centre for further assistance.
- **errorSrcText** – This element might contain some additional information regarding the source of the error.

7.2 The parseError section

The **parseError** section is for errors with the validity of the xml passed in e.g. missing elements, incomplete tags. These errors should only occur during your initial integration work with the pricing web service.

The following excerpt shows an example of a parse error caused by not including the mandatory `sender` element in the request.

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<document>
  <errors>
    <parseError>
      <errorReason>cvc-complex-type.2.4.a: Invalid content was found starting with element 'country'. One of
'{'sender}' is expected.</errorReason>
      <errorLine>11</errorLine>
      <errorLinepos>16</errorLinepos>
      <errorSrcText></errorSrcText>
    </parseError>
  </errors>
</document>
```

The key information in the `parseError` error is as follows:

- **errorLine** – This element indicates the line number where the error occurred in the request XML.
- **errorLinepos** – This element indicates the position on the line where the error occurred in the request XML.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

- **errorSrcText** – This element is occasionally used to provide additional information related to the error.

7.3 The brokenRule section

The **brokenRule** section is for application errors which the customer can resolve such as invalid postcode, login credentials.

The following excerpt shows an example of a broken rule where the customer has specified an invalid address in the **delivery** element

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<document>
  <requestId>28821</requestId>
  <errors>
    <brokenRule>
      <rateId>rate2</rateId>
      <messageType>E</messageType>
      <code>P203</code>
      <description>Destination address town not found</description>
    </brokenRule>
  </errors>
```

The key information in the **brokenRule** error is as follows:

- **rateId** – A unique identifier for the `priceRequest` to which the error relates
- **messageType** – This can be used to determine the severity of the error returned. The three categories are:
 - 'I' – Information
 - 'W' – Warning
 - 'E' – Error
- **code** – This is the unique identifier for the error.
- **description** – This element provides a descriptive explanation of the error message.

The **requestId** can be used when contacting the customer service centre to speed up the process of investigating the cause of the issue.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

7.4 Table of application generated error codes, messages and resolutions

The Default Message column contains the English language message that is returned with each code. This value is intended to be used by systems integrators. For content management you should use the error code.

An element is considered empty if it contains no value or only whitespace. Therefore all of the following elements are empty:

```
<emptyElement/>

<emptyElement></emptyElement>

<emptyElement>          </emptyElement>
```

Error Code	Error Description	Default Message	Resolution
1	The XML in the request is not formatted correctly.	Failure to format incoming XML string	Ensure the XML in the request is of the correct structure. Refer to the XSD and available documentation to rectify.
2	The services returned in the response may not all be valid due to a failure to validate the possible services.	<i>Deprecated</i>	
3	The options returned in the response may not all be valid due to a failure to validate the possible options.	<i>Deprecated</i>	
4	Failure to retrieve details of the service or options.	<i>Deprecated</i>	
5	A timeout occurred when making a call to the database.	<i>Deprecated</i>	
6	An error occurred on the TNT server.	ExpressConnect Pricing request has failed. If you continue to receive this error, please contact your local service centre for further assistance	Resubmit the request. If the problem persists contact your local service centre
7	No rates available for the request	ExpressConnect Pricing request has returned no rates. If you continue to receive no rates, please contact your local service centre	Resubmit the request. If the problem persists contact your local service centre

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

Error Code	Error Description	Default Message	Resolution
		for further assistance	
8	Mandatory fields missing or invalid in the request XML.	<i>Deprecated</i>	
9	No price requests found in the XML	No Price Requests Found	Refer to the XSD and available documentation to rectify.
10	The account number entered is not associated with this customer	Invalid Account Number	Enter a valid account number.
11	The country code in the request XML is invalid.	Invalid Country Code	Ensure the country code matches one of those in the TNT dataset.
12	The town group specified in the request XML is invalid.	Invalid Town Group	Enter a town and/or postcode as the default depot address for the country is not valid.
13	The rates returned are standard rates and do not include any account specific prices.	Standard Rates	If account specific rates are required then account details should be populated in the request XML.
14	Domestic price requests are not supported in the country specified.	Invalid Domestic Consignment	Domestic shipments should be within supported TNT countries.
15	The value of the Insurance element should be a number.	Insurance value must be numeric	Ensure the value of the Insurance element is a number.
16	Options have been specified in the request XML without an associated service code.	Service required with options	Ensure a service code is specified for the requested options.
17	The system was unable to retrieve the service and option for the request.	<i>Deprecated</i>	
18	The service specified in the request is invalid.	Service is invalid	Ensure the service in the request matches a code in the TNT dataset.
19	The system was unable to parse the request XML.	<i>Deprecated</i>	
20	The request XML was not recognised as an Express Connect request.	XML supplied is not recognised as an ExpressConnect request.	Refer to the XSD and available documentation to rectify.
21	No data was received from the client.	No data received at gateway from client error	Resubmit the request.
22	The maximum number of options per service has been exceeded.	Exceeded maximum number of allowed options	A maximum of 4 options per service are allowed.
23	The {invalidfield.name} specified must be numeric	The {invalidfield.name} specified must be numeric	Ensure that the value of the element specified in the error is numeric
24	ExpressConnect Pricing version not recognised	ExpressConnect Pricing version not recognised	Check that the correct version is specified between the appVersion

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

Error Code	Error Description	Default Message	Resolution
			elements
25	The goods value must be greater than zero	The goods value must be greater than zero	Amend the goodsValue to be greater than zero
26	The goods value currency must be entered	The goods value currency must be entered	Enter a currency in the currency element
27	The goods value currency is invalid	The goods value currency is invalid	Amend the currency to use a valid value from the TNT Dataset
28	The Insurance value must be greater than zero	The Insurance value must be greater than zero	Amend the insuranceValue to be greater than zero
31	The goods value must be entered if insurance value supplied	The goods value must be entered if insurance value supplied	Enter a goodsValue if the insuranceValue is supplied
32	The Insurance value must not exceed 110 % of goods value	The Insurance value must not exceed 110 % of goods value	Ensure that insuranceValue is not more than the specified percentage greater than the goodsValue
33	The insurance details must be entered for options specified	The insurance details must be entered for options specified	If the option IN is specified then the insurance details must also be populated.
34	Volume must be entered for NON DOCS	Volume must be entered for NON DOCS	The totalVolume must be entered for NON DOCS. This may not be necessary if piece lines are supplied and the volume is derivable
35	Volume must be greater than zero	Volume must be greater than zero	The totalVolume must be greater than zero
36	Weight must be entered	Weight must be entered	The totalWeight must be entered
37	Weight must be greater than zero	Weight must be greater than zero	The totalWeight must be greater than zero
38	Volume-weight must be greater than zero	<i>deprecated</i>	
39	The total number of pieces must be entered	The total number of pieces must be entered	The totalNumberOfPieces must be specified. This may not be necessary if piece lines are supplied and the number of pieces is derivable
40	The total number of pieces must be greater than zero	The total number of pieces must be greater than zero	The totalNumberOfPieces must be greater than zero
41	The product type must be entered	The product type must be entered	The product type must be entered
42	The country must be entered	<i>deprecated</i>	
43	The goods value must be numeric	The goods value must be numeric	The goodsValue must be numeric
44	The piece weight must be numeric	The piece weight must be numeric	The piece line weight must be numeric
45	The total volume must be numeric	The total volume must be numeric	The totalVolume must be numeric
46	The total number of pieces must be	The total number of pieces must be	The totalNumberOfPieces must be numeric

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

Error Code	Error Description	Default Message	Resolution
	numeric	numeric	
47	The number of pieces on the piece line must be numeric	The number of pieces on the piece line must be numeric	The numberOfPieces on the piece line must be numeric
48	The piece length must be numeric	The piece length must be numeric	The piece length must be numeric
49	The piece width must be numeric	The piece width must be numeric	The piece width must be numeric
50	The piece height must be numeric	The piece height must be numeric	The piece height must be numeric
51	Total number of pieces does not match sum of piece line items	Total number of pieces does not match sum of piece line items	The totalNumberOfPieces does not match sum of numberOfPieces
52	Total weight does not match sum of piece line weight to within acceptable error margin	Total weight does not match sum of piece line weight to within acceptable error margin	The totalWeight does not match sum of piece line weight to within acceptable error margin, currently 5%
53	Consignment summary details or piece lines must be specified	Consignment summary details or piece lines must be specified	consignmentDetails or pieceLine must be specified
54	The total weight must be numeric	The total weight must be numeric	The totalWeight must be numeric
55	The account number must be specified	The account number must be specified	The account number must be specified
56	The collection date must be specified	The collection date must be specified	The collectionDateTime date must be specified
57	No guaranteed services	No guaranteed services	
58	No divisional default residential surcharges	No divisional default residential surcharges	
59	Error rounding weight specified for rate	Error rounding weight specified for rate	
60	Exceptional volume weight calculated	Exceptional volume weight calculated	Review the ratio of volume to weight in request
61	The weight and volume were derived from the piece lines	The weight and volume were derived from the piece lines	No need to do anything as this is just a warning that some of the data has had to be derived as its not included in the request.
62	Piece line dimension unit is not supported	Piece line dimension unit is not supported	Contact your local service centre
63	Piece line weight unit is not supported	Piece line weight unit is not supported	Contact your local service centre
64	Option is not valid for chosen service	One or more options are not valid for the chosen service	Resubmit request without the option code and contact your local service centre for more information about the option.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

Error Code	Error Description	Default Message	Resolution
65	Option code is not recognised	Option code is not recognised	Resubmit request without the option code and contact your local service centre for more information about the option.
66	Total volume will be used for rating	Total volume will be used for rating	This is an information message to indicate the volume that will be used to generate the price
67	Total weight will be used for rating	Total weight will be used for rating	This is an information message to indicate the weight that will be used to generate the price
68	Total Volume unit is not supported	Total Volume unit is not supported	Contact your local service centre
69	Volume weight is bigger than effective weight	Volume weight is bigger than effective weight	Review the ratio of volume to weight in request
70	Total Weight unit is not supported	Total Weight unit is not supported	Contact your local service centre
71	Goods value must be entered	Goods value must be entered	Ensure that goods value is entered
72	Piece width must be entered	Piece width must be entered	Ensure that piece width is entered
73	Piece height must be entered	Piece height must be entered	Ensure that piece height is entered
74	Piece length must be entered	Piece length must be entered	Ensure that piece length is entered
75	Piece weight must be entered	Piece weight must be entered	Supply a piece weight where not total weight exists or a piece line is missing the weight
101	Origin address town not supplied	Origin address town not supplied	Supply origin address town
102	Origin address town and postcode not supplied	Origin address town and postcode not supplied	Supply origin address town and postcode.
103	Origin address town has not been recognized.	Origin address town not found	Ensure that the specified origin town is a valid town.
104	Origin address town results in more than one match	Origin address town not unique	Ensure the origin town is entered correctly. Do not use partial town names.
105	The origin address province maximum length has been exceeded.	Origin address province length exceeded	Maximum 30 characters.
106	Origin address postcode has been entered for a country that does not accept postcodes as part of the address.	Origin address postcode not allowed for selected country	Remove the origin postcode element from the request.
107	Origin address province not found	Origin address province not found	Ensure origin province is entered correctly or remove the origin province element.
108	The origin postcode is invalid in the request.	Origin address postcode invalid	Ensure the origin postcode element is entered correctly.
109	Origin address postcode has an invalid	Origin address postcode invalid	Ensure the origin postcode element is entered correctly and is no longer

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

Error Code	Error Description	Default Message	Resolution
	format. For example CV9 XTT is incorrect for a UK postcode.	format	than 9 characters.
110	Origin address postcode matches more than one town. This may occur if a partial postcode is entered.	Origin address postcode not unique	Ensure the origin postcode is entered correctly.
111	Origin address has an invalid town / postcode combination. For example if you entered a London postcode for Birmingham.	Origin address town/postcode combination invalid	Ensure origin postcode and/or town are entered correctly.
112	Origin address postcode is not within the valid range of postcodes for the town.	Origin address postcode not in valid range	Ensure origin postcode and/or town are entered correctly.
113	Origin postcode is missing from the request XML.	Origin address postcode not found	Ensure origin postcode is entered correctly
114	Origin country not found in the XML	Origin address country not supplied	Country is mandatory in the request.
115	Origin address country is not valid ISO 3166-1 ALPHA-2. (2 character alpha).	Origin address country not valid iso.	Ensure origin country is entered correctly.
116	Origin address town results in more than one match.	Origin address not unique	Ensure origin town and/or postcode are entered correctly. Do not use partial town or postcodes.
117	Origin address postcode is below minimum length for selected country	Origin address postcode is below minimum length for selected country	Ensure origin town and/or postcode are entered correctly. Do not use partial town or postcodes.
201	Destination address town not supplied	Destination address town not supplied	Supply destination address town
202	Destination address town and postcode not supplied	Destination address town and postcode not supplied	Supply destination address town and postcode.
203	Destination address town has not been recognized.	Destination address town not found	Ensure that the specified destination town is a valid town.
204	Destination address town results in more than one match	Destination address town not unique	Ensure the destination town is entered correctly. Do not use partial town names.
205	The Destination address province maximum length has been exceeded.	Destination address province length exceeded	Maximum 30 characters.
206	Destination address postcode has been entered for a country that does not	Destination address postcode not allowed for selected country	Remove the destination postcode element from the request.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

Error Code	Error Description	Default Message	Resolution
	accept postcodes as part of the address.		
207	Destination address province not found	Destination address province not found	Ensure destination province is entered correctly or remove the destination province element.
208	The Destination postcode is invalid in the request.	Destination address postcode invalid	Ensure the destination postcode element is entered correctly.
209	Destination address postcode has an invalid format. For example CV9 XTT is incorrect for a UK postcode.	Destination address postcode invalid format	Ensure the destination postcode element is entered correctly and is no longer than 9 characters.
210	Destination address postcode matches more than one town. This may occur if a partial postcode is entered.	Destination address postcode not unique	Ensure the destination postcode is entered correctly.
211	Destination address has an invalid town / postcode combination. For example if you entered a London postcode for Birmingham.	Destination address town/postcode combination invalid	Ensure destination postcode and/or town are entered correctly.
212	Destination address postcode is not within the valid range of postcodes for the town.	Destination address postcode not in valid range	Ensure destination postcode and/or town are entered correctly.
213	Destination postcode is missing from the request XML.	Destination address postcode not found	Ensure destination postcode is entered correctly
214	Destination country not found in the XML	Destination address country not supplied	Country is mandatory in the request.
215	Destination address country is not valid ISO 3166-1 ALPHA-2. (2 character alpha).	Destination address country not valid iso.	Ensure destination country is entered correctly.
216	Destination address town results in more than one match.	Destination address not unique	Ensure destination town and/or postcode are entered correctly. Do not use partial town or postcodes.
217	Destination address postcode is below the minimum length for selected country	Destination address postcode is below minimum length for selected country	Ensure destination town and/or postcode are entered correctly. Do not use partial town or postcodes.
301	The depot ID for the origin depot is not valid.	Delivery in depot id is not valid	Depot ID should be one of the valid TNT towngroup codes contained within the towngroup dataset.
302	The depot ID for the self collect delivery address is not valid.	Self collect depot id is not valid	Depot ID should be one of the valid TNT towngroup codes contained within the towngroup dataset.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

Error Code	Error Description	Default Message	Resolution
401	The TNT backend system was unable to determine the route between the source and destination addresses.	Unable to determine traffic lane for route	Contact TNT helpdesk.
999	The user name and password specified in the request have not been recognised.	login details not recognised	Enter the correct username and password combination. Contact the TNT helpdesk if the problem persists.

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

8. Connecting to ExpressConnect Pricing

ExpressConnect Pricing is a HTTP web service, which defines an XML interface for data transfer. The XSD for the service can be found in the pricing section of the website. In the section below we have an example using Java.

8.1.1 XML

XML is very suited to environments where Hessian does not offer support. Performance is reduced as the messages are larger, however, it can be simpler to develop as the message are human readable and can be constructed using simple String manipulation.

Even for a simple XML connection a number of jar files are recommended,

- a. commons-httpclient-3.1.jar (<http://hc.apache.org/httpclient-3.x/>)
- b. commons-logging-1.1.1.jar (<http://commons.apache.org/>)
- c. commons-codec-1.3.jar (<http://commons.apache.org/>)

See below an example request set up using Apache HttpClient 4.2.1. Please note the usage of BASIC authorization header for authentication.

```

public class XMLTestApp {
    public static void main(String args[] throws ClientProtocolException,
        IOException {

        // Create the xml request
        StringBuilder xmlRequest = new StringBuilder();
        xmlRequest.append("<?xml version='1.0' encoding='UTF-8' ?>");
        xmlRequest.append("<priceRequest>");
        // Add price request contents here
        xmlRequest.append("</priceRequest>");

        // Declare ExpressConnect Pricing URL and initialise Http POST request
        String url = "https://express.tnt.com/expressconnect/pricing/getprice";
        HttpPost postRequest = new HttpPost(url);

        // Create the request entity
        HttpEntity requestEntity = new StringEntity(xmlRequest.toString(),
            ContentType.create("text/xml", "UTF-8"));
        postRequest.setEntity(requestEntity);

        // Create the authentication header
        String userPassword = "USER:USER";
        byte[] basicAuthEncoding = Base64.encodeBase64(userPassword.getBytes());
        postRequest.addHeader("Authorization", "Basic "
            + new String(basicAuthEncoding));

        // Initialise HttpClient and execute POST request
        HttpClient httpClient = new DefaultHttpClient();
        HttpResponse result = httpClient.execute(postRequest);

        // Check HTTP status code of the response
        System.out.println("Status Code: "
            + result.getStatusLine().getStatusCode());

        // Read content of result stream
        HttpEntity entity = result.getEntity();
        if (entity != null) {
            System.out.println(IOutils.toString(entity.getContent()));
        }
    }
}

```

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

9. Appendix A: XML elements definition (input)

These tables contain all of the possible Input XML nodes and some information about their use. All elements are mandatory and must contain just one instance unless stated in the table below.

XML elements which do not contain a value but are merely a container for other elements are noted as **Container element**, the elements contained therein are described in the rows immediately below. Please remember that field values should be escaped using the `![CDATA[]]` notation. At a minimum, all address fields should be escaped. Failure to escape these areas could result in unexpected problems if the value in an element contains an Ampersand (&).

For all data types described with a type prefix of 'xsd:' see [XSD Data Types](#).

For all data types in the form 'stringMinLengthXMaxLengthY', 'stringMaxLengthY', 'integerMin0Max9', 'doubleMaxExclusiveX' or 'longMaxLengthX' see Section [Custom Data Types](#).

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

XML ELEMENT	DESCRIPTION	COMMENTS
priceRequest	Top level element for a message to ExpressConnect Pricing. Container element	The root XML element for a Price Request message. This element must contain one or more price requests, each one represented by a priceCheck element.
priceCheck	The priceCheck element represents a single request for a price request. Container element	Multiple price requests, each one in a priceCheck element may be submitted. The information within this element will be used to validate the addresses, determine the services, and produce the estimate costs for the shipment. Each priceCheck element must contain a rateId attribute that uniquely identifies the price request. The value of the key may be any alphanumeric string - typically this value will be an integer value i.e. the first request will have key="1", second value of key="2" and so on. Any validation errors will be tagged with this key so that you can identify the request in error.
appId	Type = xsd:string	This is the application ID and should be set to 'PC'. This is a mandatory element.
appVersion	Type = xsd:decimal	This denotes the version of ExpressConnect Pricing that will be used. If a value of 3.0 is supplied then the new interface functionality will be evoked. If no appVersion is supplied it will be assumed that the existing interface is used. Only 3.0 will be considered valid, if any other value is supplied then an error will be returned.
rateId	Type = xsd:string	It is recommended that the value is unique within the priceRequest to allow easy identification of results to requests. This is a mandatory element.
sender	Element to hold the origin address details. Container element	This element provides the origin address details and must be supplied for each Price request message.

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

delivery	Element to hold the destination address details. Container element	This element provides the destination address details and must be supplied for each Price request message.
country	Type = xsd:string	The ISO 3166-1 Alpha-2 country code for the country of the given address. This is a mandatory element.
town	Type = xsd:string	The town name as recognised by TNT.
postcode	Type = xsd:string	Postcode or zip code
collectionDateTime	Type=xsd:dateTime	The format of a date time is yyyy-mmddThh:MM:ss
Product	Element to hold details of the pricing service and options requested. Container element	Information relating to the TNT product chosen for this priceCheck. If this is supplied, the price will be for the service, otherwise a list of services will be returned.
id	Type = xsd:string	The identifier for the requested product/service e.g. 15N
division	Type = xsd:string	The identifier for the requested product/service division. This is an optional element and if not specified will default to an appropriate division based on the origin and destination country selected
type	Type = xsd:string	Type of service for product chosen. Values are "D" for a document or "N" for non-documents. This is a mandatory element.

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

options	Element to hold the options specified for the pricing request. Container element	The options element can contain up to 4 options
option	Type = xsd:string	The option must be a valid code from the TNT dataset.
account	Element to hold the account details specified for the pricing request. Container element	This is an optional element but must be supplied to obtain accurate price estimates.
accountNumber	Type = xsd:string	TNT account number, which is the 9 or 10 digit number assigned by the TNT sales person.
accountCountry	Type = xsd:string	ISO 3166-1 Alpha-2 country code for the country in which the TNT account is registered.
insurance	Element to hold the insurance details specified for the pricing request. Container element	This element holds the insurance details needed to retrieve an insurance quote as part of the price response. If an insurance option such as IN is specified than these elements become mandatory.
insuranceValue	Type = xsd:decimal	The value of the insurance required to cover the goodsValue specified in the request. This is limited to a maximum of 110% of the goodsValue
goodsValue	Type = xsd:decimal	The value of the goods in the currency specified in the request
termsOfPayment	Type = xsd:string	Whether the sender or receiver to paying for the shipment. Values are "S" for a sender or "R"

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

		for receiver. Optional element, which default to S if not supplied.
currency	Type = xsd:string	The ISO 4217 Alpha-3 currency codes for the currency of the price request.
priceBreakDown	Type = xsd:boolean	This is an optional element that determines if a full price breakdown is returned or just the high level rate. The default value is false.
consignmentDetails	Element to hold the summary consignment details for the pricing request. Container element	This element is optional as long as the <code>pieceLine</code> are specified instead
totalWeight	Type = xsd:decimal	The total weight of the shipment, Mandatory element, must be supplied in Kilograms.
totalVolume	Type = xsd:decimal	The total volume of the shipment, Mandatory element if non document, must be supplied in cubic metres.
totalNumberOfPieces	Type = xsd:integer	The total number of pieces this <code>priceCheck</code> contains. In cases where only some of the pieces are being submitted, this value should contain the total number of pieces in the consignment, not the total number of pieces in the request.
pieceLine	Element to hold the individual piece line details for the pricing request. Container element	The <code>pieceLine</code> element is used to describe the individual piece lines which make up the consignment in greater detail. It is an optional element and can occur multiple times up to a maximum of 99 piece lines. The details specified for the <code>pieceLine</code> must be compatible with those specified in the <code>consignmentDetails</code> . If no <code>pieceLine</code> are specified then a default one will be created.

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

numberOfPieces	Type = xs:integer	Number of pieces within this piece line.
pieceMeasurements	Element to hold the individual pieceLine measurements. Container element	
weight	Type = xs:decimal	The weight of an individual piece in KG.
length	Type = xs:decimal	The length in metres
width	Type = xs:decimal	The width in metres
height	Type = xs:decimal	The height in metres.
pallet	Type = xs:boolean	Whether the packages have been placed onto a pallet. The default value is false

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

10. Appendix B: XSD Data Types

XSD DATA TYPE	DESCRIPTION
dateTime	Data expected in the format CCYY-MM-DD'T'hh:mm:ss where CC indicates century, YY year, MM month in 2 digit format, DD day of the month, hh hour of the day, mm minutes and ss seconds. For example 5:30 p.m. on 30th December 2008 would be 2008-12-30T17:30:00. This data type describes instances identified by the combination of a date and a time. It is described in Chapter 5.4 of ISO 8601 and the W3C XML Schema Recommendation. Its lexical space is the extended format: [-]CCYY-MM-DDThh:mm:ss[Z](+ -)hh:mm]
double	Numeric data with decimal places in the range -9007199254740991..9007199254740991.
int	Numeric data without decimal places in the range -2147483648..2147483647
long	Numeric data without decimal places in the range -9223372036854775808..9223372036854775807
string	XML compatible alphanumeric data.
boolean	1 – true, 0 – false
decimal	The maximum number of decimal digits you can specify is 18.
Integer	An integer value

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

11. Appendix C: ISO 3166-1 Alpha-2 Country Codes

Country names	ISO 3166-1-alpha-2 code
A	
AFGHANISTAN	AF
ÅLAND ISLANDS	AX
ALBANIA	AL
ALGERIA	DZ
AMERICAN SAMOA	AS
ANDORRA	AD
ANGOLA	AO
ANGUILLA	AI
ANTARCTICA	AQ
ANTIGUA AND BARBUDA	AG
ARGENTINA	AR
ARMENIA	AM
ARUBA	AW
AUSTRALIA	AU
AUSTRIA	AT
AZERBAIJAN	AZ
B	
BAHAMAS	BS
BAHRAIN	BH
BANGLADESH	BD
BARBADOS	BB
BELARUS	BY
BELGIUM	BE
BELIZE	BZ
BENIN	BJ
BERMUDA	BM
BHUTAN	BT
BOLIVIA	BO
BOSNIA AND HERZEGOVINA	BA
BOTSWANA	BW
BOUVET ISLAND	BV
BRAZIL	BR

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

BRITISH INDIAN OCEAN TERRITORY	IO
BRUNEI DARUSSALAM	BN
BULGARIA	BG
BURKINA FASO	BF
BURUNDI	BI
C	
CAMBODIA	KH
CAMEROON	CM
CANADA	CA
CAPE VERDE	CV
CAYMAN ISLANDS	KY
CENTRAL AFRICAN REPUBLIC	CF
CHAD	TD
CHILE	CL
CHINA	CN
CHRISTMAS ISLAND	CX
COCOS (KEELING) ISLANDS	CC
COLOMBIA	CO
COMOROS	KM
CONGO	CG
CONGO, THE DEMOCRATIC REPUBLIC OF THE	CD
COOK ISLANDS	CK
COSTA RICA	CR
CÔTE D'IVOIRE	CI
CROATIA	HR
CUBA	CU
CYPRUS	CY
CZECH REPUBLIC	CZ
D	
DENMARK	DK
DJIBOUTI	DJ
DOMINICA	DM
DOMINICAN REPUBLIC	DO
E	
ECUADOR	EC
EGYPT	EG

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

EL SALVADOR	SV
EQUATORIAL GUINEA	GQ
ERITREA	ER
ESTONIA	EE
ETHIOPIA	ET
F	
FALKLAND ISLANDS (MALVINAS)	FK
FAROE ISLANDS	FO
FIJI	FJ
FINLAND	FI
FRANCE	FR
FRENCH GUIANA	GF
FRENCH POLYNESIA	PF
FRENCH SOUTHERN TERRITORIES	TF
G	
GABON	GA
GAMBIA	GM
GEORGIA	GE
GERMANY	DE
GHANA	GH
GIBRALTAR	GI
GREECE	GR
GREENLAND	GL
GRENADA	GD
GUADELOUPE	GP
GUAM	GU
GUATEMALA	GT
GUERNSEY	GG
GUINEA	GN
GUINEA-BISSAU	GW
GUYANA	GY
H	
HAITI	HT
HEARD ISLAND AND MCDONALD ISLANDS	HM
HOLY SEE (VATICAN CITY STATE)	VA
HONDURAS	HN
HONG KONG	HK

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

HUNGARY	HU
I	
ICELAND	IS
INDIA	IN
INDONESIA	ID
IRAN, ISLAMIC REPUBLIC OF	IR
IRAQ	IQ
IRELAND	IE
ISLE OF MAN	IM
ISRAEL	IL
ITALY	IT
J	
JAMAICA	JM
JAPAN	JP
JERSEY	JE
JORDAN	JO
K	
KAZAKHSTAN	KZ
KENYA	KE
KIRIBATI	KI
KOREA, DEMOCRATIC PEOPLE'S REPUBLIC OF	KP
KOREA, REPUBLIC OF	KR
KUWAIT	KW
KYRGYZSTAN	KG
L	
LAO PEOPLE'S DEMOCRATIC REPUBLIC	LA
LATVIA	LV
LEBANON	LB
LESOTHO	LS
LIBERIA	LR
LIBYAN ARAB JAMAHIRIYA	LY
LIECHTENSTEIN	LI
LITHUANIA	LT
LUXEMBOURG	LU
M	
MACAO	MO

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

MACEDONIA, THE FORMER YUGOSLAV REPUBLIC OF	MK
MADAGASCAR	MG
MALAWI	MW
MALAYSIA	MY
MALDIVES	MV
MALI	ML
MALTA	MT
MARSHALL ISLANDS	MH
MARTINIQUE	MQ
MAURITANIA	MR
MAURITIUS	MU
MAYOTTE	YT
MEXICO	MX
MICRONESIA, FEDERATED STATES OF	FM
MOLDOVA	MD
MONACO	MC
MONGOLIA	MN
MONTENEGRO	ME
MONTSERRAT	MS
MOROCCO	MA
MOZAMBIQUE	MZ
MYANMAR	MM
N	
NAMIBIA	NA
NAURU	NR
NEPAL	NP
NETHERLANDS	NL
NETHERLANDS ANTILLES	AN
NEW CALEDONIA	NC
NEW ZEALAND	NZ
NICARAGUA	NI
NIGER	NE
NIGERIA	NG
NIUE	NU
NORFOLK ISLAND	NF
NORTHERN MARIANA ISLANDS	MP

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

NORWAY	NO
O	
OMAN	OM
P	
PAKISTAN	PK
PALAU	PW
PALESTINIAN TERRITORY, OCCUPIED	PS
PANAMA	PA
PAPUA NEW GUINEA	PG
PARAGUAY	PY
PERU	PE
PHILIPPINES	PH
PITCAIRN	PN
POLAND	PL
PORTUGAL	PT
PUERTO RICO	PR
Q	
QATAR	QA
R	
RÉUNION	RE
ROMANIA	RO
RUSSIAN FEDERATION	RU
RWANDA	RW
S	
SAINT BARTHÉLEMY	BL
SAINT HELENA	SH
SAINT KITTS AND NEVIS	KN
SAINT LUCIA	LC
SAINT MARTIN	MF
SAINT PIERRE AND MIQUELON	PM
SAINT VINCENT AND THE GRENADINES	VC
SAMOA	WS
SAN MARINO	SM
SAO TOME AND PRINCIPE	ST
SAUDI ARABIA	SA
SENEGAL	SN

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

SERBIA	RS
SEYCHELLES	SC
SIERRA LEONE	SL
SINGAPORE	SG
SLOVAKIA	SK
SLOVENIA	SI
SOLOMON ISLANDS	SB
SOMALIA	SO
SOUTH AFRICA	ZA
SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS	GS
SPAIN	ES
SRI LANKA	LK
SUDAN	SD
SURINAME	SR
SVALBARD AND JAN MAYEN	SJ
SWAZILAND	SZ
SWEDEN	SE
SWITZERLAND	CH
SYRIAN ARAB REPUBLIC	SY
T	
TAIWAN, PROVINCE OF CHINA	TW
TAJKISTAN	TJ
TANZANIA, UNITED REPUBLIC OF	TZ
THAILAND	TH
TIMOR-LESTE	TL
TOGO	TG
TOKELAU	TK
TONGA	TO
TRINIDAD AND TOBAGO	TT
TUNISIA	TN
TURKEY	TR
TURKMENISTAN	TM
TURKS AND CAICOS ISLANDS	TC
TUVALU	TV
U	
UGANDA	UG
UKRAINE	UA

Developer Notes

ExpressConnect Pricing Integration Guide	Version: 2.0
	Date: 12/03/2013

UNITED ARAB EMIRATES	AE
UNITED KINGDOM	GB
UNITED STATES	US
UNITED STATES MINOR OUTLYING ISLANDS	UM
URUGUAY	UY
UZBEKISTAN	UZ
V	
VANUATU	VU
VATICAN CITY STATE	see HOLY SEE
VENEZUELA	VE
VIET NAM	VN
VIRGIN ISLANDS, BRITISH	VG
VIRGIN ISLANDS, U.S.	VI
W	
WALLIS AND FUTUNA	WF
WESTERN SAHARA	EH
Y	
YEMEN	YE
Z	
ZAMBIA	ZM
ZIMBABWE	ZW

Developer Notes