

TNT Post Group



a TPG company

TNT iConnections v2.0: Tracking

Technical Guide

**Issue 4.1
December 4, 2002**

Revision History

Date	Version	Description	Author
19 th August, 2002	4.0	Initial document	Michelle Dance
4 th December, 2002	4.1	Encoding added to return XML declaration	Michelle Dance



a TPG company

CONTENTS

INTRODUCTION	4
CONNECTING TO THE TNT SERVER.....	5
MICROSOFT.XMLHTTP	6
JAVASCRIPT CONNECTION CODE EXAMPLE	7
VBSCRIPT CONNECTION CODE EXAMPLE.....	7
TRACK OPTIONS	8
TRACK TYPE	8
EXTRA DETAILS	8
XML FORMAT	9
INPUT	9
TABLE OF XML ELEMENTS (INPUT)	10
OUTPUT	11
TABLE OF XML ELEMENTS (OUTPUT)	14
DTD (DOCUMENT TYPE DEFINITION)	17
ERRORS.....	18
PARSER ERRORS.....	18
ERRORS RETURNED BY THE VBGATEWAY.ASP (ENTRY FILE).....	19
APPLICATION GENERATED ERRORS	19
TABLE OF APPLICATION GENERATED ERROR CODES, MESSAGES AND RESOLUTIONS	20
SERVER ERRORS.....	20
RECOMMENDED RESOURCES	21



INTRODUCTION

TNT is providing a facility whereby you can connect to a TNT consignment tracking facility over the World Wide Web and receive tracking details for submitted consignment numbers.

With the use of the new XML (extensible mark-up language) technology, this will give consignment tracking functionality for single or multiple consignment submissions.

The application also allows you to retrieve up to four sets of extra details for each consignment tracked: Sender Address, Receiver Address, Shipment and Package details are those available. Please note, the consignment must belong to the customer submitting the request for them to have permission to receive these extra details. The customer is requested to supply a list of their account numbers that they would like to be registered against their logon details. This will be arranged by the TNT representative.

The data to track will be passed to TNT in a predetermined XML format. You must send the data to TNT by opening an http socket level connection to a given URL and port, thus connecting your own web server with that of TNT. Your XML data must be passed as a URL encoded string.

Return data from TNT will be formatted as XML, again in a predetermined structure. When this data is returned, you may parse it either for storage or presentation as required.



a TPG company

CONNECTING TO THE TNT SERVER

A new user of the TNT iConnections Tracking application will need to complete the following steps before attempting to send an XML document directly to the TNT server.

1. The IP address of the customer's **server** (not their PC) must be added to the TNT firewall rulebase, so that any calls to the TNT server from this server will be allowed through. In the case that the customer has more than one server with access required, for example, a test environment and a live environment, each server's IP address must be given. Please note that 5 working days notice is required before the requested IP addresses are added to the TNT firewall. This will be arranged by the TNT representative.
2. Each customer is setup with their own company id and password, required for each iConnections submission.
3. Before sending an XML document through a straight socket connection, you must verify that you can access TNT's iConnections applications by successfully using the test/demonstration web site provided by TNT. This is located at <http://iconnection.tnt.com:81>.

Some important points to note:

- ▶ You need to ensure that your own firewall will allow data out using port 81.
- ▶ The demonstration site does NOT support foreign (non-ASCII) characters.

The test/demonstration site simulates a simple customer interface to the back-end iConnections server. It is useful for front-end developers to use as a tool to test their XML and analyse their return XML results, prior to and during the development of their front-end application. The demonstration site links to the **production environment** so that it replicates exactly the results that the customer will achieve with their own front-end application.

4. 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 real XML submission using the code similar to that on the following pages.

Open a socket connection to the following address and port number.

Domain Name	iConnection.tnt.com
IP Address	194.200.70.13
Port	81

Send the following lines, data between quotes. N.B. 1 blank line should be sent after Content-length line.

```
"POST VBGateway.asp HTTP/1.0"
"Accept: */*"
"User-Agent: VBGateway_socket/1.0";
"Content-type: application/x-www-form-urlencoded";
"Content-length: X";      (replace X with the length of data plus 7 for "xml_in=")
""
"xml_in=XML document"    (this is the content of message)
```

XML document is the XML corresponding to the specified format (see later section).

Once the above data has been sent, read from the socket until no more data is available. The data returned will consist of some header information followed by a blank line, and then the track information.

MICROSOFT.XMLHTTP

Microsoft has added an http object to Internet Explorer 5.0 and above. If IIS is being used with IE5 installed then the socket connection is made easier. The following shows how to use the Microsoft http object. It does the same job as the connection above, but is slightly user-friendlier.

The XMLHttpRequest object provides client-side protocol support for communication with HTTP servers.

A client computer can use the XMLHttpRequest object (Microsoft.XMLHTTP) to send an arbitrary HTTP request, receive the response, and have the Microsoft XML Document Object Model (XMLDOM) parse that response.

This object is integrated with MSXML to support sending the request body directly from, and parsing the response directly into, the MSXML Document Object Model objects. When combined with the support for XSL, the XMLHttpRequest component provides an easy way to send structured queries to HTTP servers and efficiently display the results with a variety of presentations.

On the next page is some example code using the Microsoft socket object to send data into the TNT systems.



a **TPG** company

JAVASCRIPT CONNECTION CODE EXAMPLE

Var xmlin = your xml document

```
var httpOb = Server.CreateObject("Microsoft.XMLHTTP");
var URL = "iconnection.tnt.com:81/VBGateway.asp";
httpOb.Open("POST", URL, false, "", "");
httpOb.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
httpOb.Send("xml_in="+xmlin);
```

(The returned data can be obtained by viewing the ResponseText property of the http object).

```
Response.Write("httpOb.responsetext");
```

VBSCRIPT CONNECTION CODE EXAMPLE

xml_in = (your Tracking XML document as a string variable)

```
'note ----- create a socket connection object
    dim httpOb
    set httpOb = Server.CreateObject("Microsoft.XMLHTTP")

'note ----- open socket connection to given IP address and entry file
    httpOb.Open "POST", "http://iConnection.tnt.com:81/VBGateway.asp", false, "", ""

'note ----- if opening socket works set header and send XML
    httpOb.setRequestHeader "Content-Type", "application/x-www-form-urlencoded"
    httpOb.Send "xml_in"&xml_in

'note ----- store returned XML
    xml_out = httpOb.responseText

'note ----- release socket object
    set httpObj = nothing
```



a TPG company

TRACK OPTIONS

TRACK TYPE

Track Type	Description
Consignment	Tracks a standard format consignment number e.g. 256867154
CustomerRef	Tracks a customer reference e.g. DISKS
AltCon	Tracks a consignment number in the alternative format, as used in Australia e.g. CP699099T

EXTRA DETAILS

Extra Details	Description
SenderAddress	Returns the sender address and collection address for each consignment
ReceiverAddress	Returns the receiver address and delivery address for each consignment
Shipment	Returns the shipment details for each consignment
Package	Returns the package details for each consignment

Statcode	Status code
Statdesc	Status description

XML FORMAT

INPUT

This will be in the following format.

<!--Important – this address must be given for the DTD path-->

```
<?xml version="1.0" standalone="no"?><!DOCTYPE document SYSTEM
'http://164.39.41.19:81/TrackerDTD1.0/ trackerIN.dtd '>
<document>
```

 <!--Application must always be Tr.-->

```
<APPLICATION>Tr</APPLICATION>
<VERSION>1.0</VERSION>
```

<!--User name and password are mandatory and will have been supplied by your local TNT representative.-->

```
<USER_NAME>TNTTEST</USER_NAME>
<PASSWORD>password</PASSWORD>
```

<!--There are 3 track types available – see Track Options section -->

```
<TRACK_TYPE>Consignment</TRACK_TYPE>
<TRACK_DATA>
```

<!--There are 4 sets of extra details available – see Track Options section -->

```
    <ExtraDetails>SenderAddress</ExtraDetails>
    <ExtraDetails>ReceiverAddress</ExtraDetails>
    <ExtraDetails>Shipment</ExtraDetails>
    <ExtraDetails>Package</ExtraDetails>
```

<!--Plus 2 additional sets of extra details if you would like to be returned the status code as well as an English status description. The status code is useful for customers who wish to display their statuses in a language other than English-->

```
    <ExtraDetails>Statcode</ExtraDetails>
    <ExtraDetails>Statdesc</ExtraDetails>
```

<!--There may be multiple Tdata elements. They must contain data appropriate to the Track Type -->

```
<Tdata>123456782</Tdata>
<Tdata>123456794</Tdata>
```

```
</TRACK_DATA>
</document>
```

N.B. XML is case sensitive. The element tags MUST be the same case as in this example.



a TPG company

TABLE OF XML ELEMENTS (INPUT)

XML ELEMENT	MUST HOLD DATA	DESCRIPTION	CONSTRAINTS	PARENT ELEMENT
document	Y	Top level element		None
APPID	Y	Application Id	Must be 'Tr'	document
VERSION	Y	Version number	Currently '1.0'	document
USER_NAME	Y	Company Id	As allocated by TNT	document
PASSWORD	Y	Company Password	As allocated by TNT	document
TRACK_TYPE	Y	Specifies which of the 3 available tracks is to be performed	Options: 'Consignment' 'CustomerRef' 'AltCon'	document
TRACK_DATA	Y	Holds elements containing tracking data		document
ExtraDetails (optional – may be multiple)	Y	Specifies which sets of extra details are to be returned	Only available if requestor is owner or receiver of the consignment. Options: 'SenderAddress', 'ReceiverAddress', 'Shipment', 'Package' 'Statcode' * 'Statdesc' *	TRACK_DATA
Tdata (may be multiple, at least one must be present in the XML)	Y	Holds the consignment number, customer reference or alternatively formatted consignment number (depending on Track_Type specified).	One consignment per Tdata element.	TRACK_DATA

* Note that this is a new feature. As default iConnection tracking returns the status description only. To be returned the status code in the response XML, use the following format within the <TRACK_DATA> element:

```
<TRACK_DATA>
```

```
    <ExtraDetails>Statcode</ExtraDetails>
```

```
    <ExtraDetails>Statdesc</ExtraDetails>
```

```
<Tdata>123456782</Tdata>
```

```
</TRACK_DATA>
```

This will return both the status code and a description in English.



a TPG company

OUTPUT

If successfully processed, this will be in the following format.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes" ?>
<document>
```

```
<!--There may be up to 10 consignment elements -->
```

```
<Consignment>
```

```
<ConsignmentNumber>Number</ConsignmentNumber>
<OriginDepotName>Depot</OriginDepotName>
<CustomerReference>Customer reference</CustomerReference>
<CollectionDate>DD MMM YYYY</CollectionDate>
<DeliveryTown>Town name</DeliveryTown>
<DeliveryDate> DD MMM YYYY </DeliveryDate>
<DeliveryTime>HH:MM</DeliveryTime>
<Signatory>Signatory</Signatory>
<SummaryCode>Delivered</SummaryCode>
<Cou-Id-Dest>Country</Cou-Id-Dest>
<Cou-Id-Orig> Country </Cou-Id-Orig>
<SearchType>C/R</SearchType>
<SearchKey>number or reference</SearchKey>
<TOPaccountNo>account</TOPaccountNo>
<TOPaccountCouID>Country Code</TOPaccountCouID>
<SenderaccountNo>account</SenderaccountNo>
<SenderCouID>Country code</SenderCouID>
```

```
<!--A status data element exists for each point of transit -->
```

```
<StatusData>
<Status>CONSIGNMENT DELIVERED IN GOOD CONDITION</Status>
<LocalEventDate>DD MMM YYYY</LocalEventDate>
<LocalEventTime>hh:mm</LocalEventTime>
<Depot>Depot</Depot>
</StatusData>
```

```
<StatusData>
<Status>OUT FOR DELIVERY</Status>
<LocalEventDate>DD MMM YYYY</LocalEventDate>
<LocalEventTime>hh:mm</LocalEventTime>
<Depot>Depot</Depot>
</StatusData>
```

```
<StatusData>
<Status>IMPORT RECEIVED</Status>
<LocalEventDate>DD MMM YYYY</LocalEventDate>
<LocalEventTime>hh:mm</LocalEventTime>
<Depot>Depot</Depot>
</StatusData>
```

```
<StatusData>
<Status>CONSIGNMENT PASSED THROUGH TRANSIT POINT</Status>
<LocalEventDate>DD MMM YYYY</LocalEventDate>
```



a TPG company

```
<LocalEventTime>hh:mm</LocalEventTime>  
<Depot>Depot</Depot>  
</StatusData>
```

```
<StatusData>  
<Status>SHIPPED FROM ORIGINATING DEPOT</Status>  
<LocalEventDate>DD MMM YYYY</LocalEventDate>  
<LocalEventTime>hh:mm</LocalEventTime>  
<Depot>Depot</Depot>  
</StatusData>
```

<!--Sender Address details, if requested -->

```
<Sender_CollectionAddr>  
<SenderName>COMPANY NAME</SenderName>  
<SenderAddressLine1>ADDRESS</SenderAddressLine1>  
<SenderAddressLine2 />  
<SenderAddressLine3 />  
<SenderCity>City</SenderCity>  
<SenderProvince />  
<SenderPostCode>PostCode</SenderPostCode>  
<SenderCountry>Country</SenderCountry>  
<SenderTel.Number>Number</SenderTel.Number>  
<SenderContactName>Contact</SenderContactName>  
<SenderContactTel.>Tel</SenderContactTel.>  
<SenderAccountNo.>Acc No</SenderAccountNo.>  
<SenderVATNumber> VAT</SenderVATNumber>  
<CollectionName />  
<CollectionAddressLine1 />  
<CollectionAddressLine2 />  
<CollectionAddressLine3 />  
<CollectionCity />  
<CollectionProvince />  
<CollectionPostcode />  
<CollectionCountry />  
<CollectionTel.Number />  
<CollectionContactName />  
<CollectionContactTel. />  
<CollectionAccountNo />  
<CollectionVATNumber />  
</Sender_CollectionAddr>
```

<!--Receiver address details, if requested -->

```
<Receiver_DeliveryAddr>  
<ReceiverName>RECEIVER NAME</ReceiverName>  
<ReceiverAddressLine1>ADDRESS</ReceiverAddressLine1>  
<ReceiverAddressLine2></ReceiverAddressLine2>  
<ReceiverAddressLine3></ReceiverAddressLine3>  
<ReceiverCity>City name</ReceiverCity>  
<ReceiverProvince />  
<ReceiverPostCode>postcode</ReceiverPostCode>  
<ReceiverCountry>Country Code</ReceiverCountry>  
<ReceiverTel.Number />  
<ReceiverContactName>Contact</ReceiverContactName>  
<ReceiverContactTel.>Telephone number</ReceiverContactTel.>
```



a TPG company

```
<ReceiverAccountNo. />  
<ReceiverVATNumber />  
<DeliveryName />  
<DeliveryAddressLine1 />  
<DeliveryAddressLine2 />  
<DeliveryAddressLine3 />  
<DeliveryCity />  
<DeliveryProvince />  
<DeliveryPostcode />  
<DeliveryCountry />  
<DeliveryTel.Number />  
<DeliveryContactName />  
<DeliveryContactTel. />  
<DeliveryAccountNo />  
<DeliveryVATNumber />  
</Receiver_DeliveryAddr>
```

<!--Package Details, if requested -->

```
<PackagesSummary>  
<NumberOfPieces>number of pieces</NumberOfPieces>  
<Weight>weight in kg</Weight>  
<PackingDescription />  
<GoodsDescription>description</GoodsDescription>  
<InvoiceAmount>amount</InvoiceAmount>  
<InvoiceCurrency />  
</PackagesSummary>
```

<!--Shipment details, if requested -->

```
<ShipmentSummary>  
<AltConsignment />  
<Division>G/D</Division>  
<ProductName>Service name</ProductName>  
<PickupDate>DD MMM YYYY</PickupDate>  
<TermsOfPayment>Sender/Receiver</TermsOfPayment>  
</ShipmentSummary>
```

```
</Consignment>  
</document>
```

If the **Statcode** and **Statdesc** are requested each status data element is returned as follows:

```
<StatusData>  
  <StatusCode>OK</StatusCode>  
  <StatusDesc>CONSIGNMENT DELIVERED IN GOOD CONDITION</StatusDesc>  
  <LocalEventDate>DD MMM YYYY</LocalEventDate>  
  <LocalEventTime>hh:mm</LocalEventTime>  
  <Depot>Depot</Depot>  
</StatusData>
```



a TPG company

TABLE OF XML ELEMENTS (OUTPUT)

XML ELEMENT	MUST HOLD DATA	DESCRIPTION	CONSTRAINTS	MAX LENGTH	PARENT ELEMENT
document	Y	Top level element			None
Consignment (may be multiple)	Y	Holds elements containing consignment data			Document
ConsignmentNumber	N	Consignment number		15	Consignment
OriginDepot	N	Originating depot name		40	Consignment
CreationDate	N	Date consignment created		11	Consignment
CreationTime	N	Time consignment created		5	Consignment
CustomerReference	N	Customers reference		24	Consignment
CollectionDate	N	Collection date of consignment		11	Consignment
DeliveryTown	N	Delivery town of consignment		40	Consignment
DeliveryDate	N	Date consignment delivered		11	Consignment
DeliveryDateExp	N			11	Consignment
DeliveryTime	N	Time consignment delivered		5	Consignment
Signatory	N	Who signed for consignment on delivery		20	Consignment
SummaryCode	N	Description of current delivery status		10	Consignment
Cou-Id-Dest	N	Destination country of consignment		30	Consignment
Cou-Id-Orig	N	Originating country of consignment		30	Consignment
SearchType	N	Type of search used		1	Consignment
SearchKey	N	Key used to search for consignments		24	Consignment
TOPaccountNo.	N	Account Number of payer		9	Consignment
TOPaccountCouID	N	Country of payers account		30	Consignment
SenderaccountNo.	N	Account Number of sender		9	Consignment
SenderCouID	N	Country of sender		30	Consignment
StatusData (may be multiple)	N	Holds elements containing status data			Consignment
Status	N	Delivery status of consignment as of the local event date and time		65	StatusData
StatusCode	N	Status code			StatusData
StatusDesc	N	Status description in English			StatusData
LocalEventDate	N	Date when the status was updated		11	StatusData
LocalEventTime	N	Time when status was updated		5	StatusData
Depot	N	Depot Where the consignment status was updated		40	StatusData
Sender_CollectionAddr (optional - may be multiple within document)	Y		Only one per consignment element		Consignment
SenderName	Y	Name of customer sending consignment		50	Sender_CollectionAddr
SenderAddressLine1	Y	First address line of customer sending consignment		30	Sender_CollectionAddr
SenderAddressLine2	N	Second address line of customer sending consignment		30	Sender_CollectionAddr
SenderAddressLine3	N	Third line of customer sending consignment		30	Sender_CollectionAddr
SenderCity	Y	City location of customer sending consignment		30	Sender_CollectionAddr
SenderProvince	N	Province of customer sending consignment		30	Sender_CollectionAddr
SenderPostcode	Y	Post code of customer sending consignment		9	Sender_CollectionAddr
SenderCountry	Y	Country where sender is located		30	Sender_CollectionAddr
SenderTel.Number	N	Senders telephone number		17	Sender_CollectionAddr
SenderContactName	N	Contact name of sender		22	Sender_CollectionAddr



a TPG company

SenderContactTel.	N	Telephone number of senders contact person	17	Sender_CollectionAddr
SenderAccountNo.	Y	Senders account number	9	Sender_CollectionAddr
SenderVATNumber	N	Senders VAT number	20	Sender_CollectionAddr
CollectionName	N	Name of customer where consignment is collected from	50	Sender_CollectionAddr
CollectionAddressLine1	N	First address line of consignment collection address	30	Sender_CollectionAddr
CollectionAddressLine2	N	Second address line of consignment collection address	30	Sender_CollectionAddr
CollectionAddressLine3	N	Third address line of consignment collection address	30	Sender_CollectionAddr
CollectionCity	N	City where consignment can be collected from	30	Sender_CollectionAddr
CollectionProvince	N	Province where consignment can be collected from	30	Sender_CollectionAddr
CollectionPostcode	N	Post code of consignment collection location	9	Sender_CollectionAddr
CollectionCountry	N	Country of collection location	30	Sender_CollectionAddr
CollectionTel.Number	N	Telephone number of consignment collection location	17	Sender_CollectionAddr
CollectionContactName	N	Contact person name at consignment collection location	22	Sender_CollectionAddr
CollectionContactTel.	N	Telephone number of contact person at consignment collection location	17	Sender_CollectionAddr
CollectionAccountNo.	N	Account number of customer at consignment collection location	9	Sender_CollectionAddr
CollectionVATNumber	N	VAT number of collection customer	20	Sender_CollectionAddr
Receiver_DeliveryAddr (optional - may be multiple within document)	Y			Consignment
ReceiverName	Y	Name of customer receiving consignment	50	Receiver_DeliveryAddr
ReceiverAddressLine1	Y	First address line of receiver	30	Receiver_DeliveryAddr
ReceiverAddressLine2	N	Second address line of receiver	30	Receiver_DeliveryAddr
ReceiverAddressLine3	N	Third address line of receiver	30	Receiver_DeliveryAddr
ReceiverCity	Y	City of receiver	30	Receiver_DeliveryAddr
ReceiverProvince	N	Province of receiver	30	Receiver_DeliveryAddr
ReceiverPostcode	Y	Post Code of receiver	9	Receiver_DeliveryAddr
ReceiverCountry	Y	Country of receiver	30	Receiver_DeliveryAddr
ReceiverTel.Number	N	Telephone number of receiver	17	Receiver_DeliveryAddr
ReceiverContactName	Y	Contact name of receiver	22	Receiver_DeliveryAddr
ReceiverContactTel.	N	Telephone number of receivers contact person	17	Receiver_DeliveryAddr
ReceiverAccountNo.	N	Account number of receiver	9	Receiver_DeliveryAddr
ReceiverVATNumber	N	VAT of receiver	20	Receiver_DeliveryAddr
DeliveryName	N	Name of customer where consignment is to be delivered	50	Receiver_DeliveryAddr
DeliveryAddressLine1	N	Address line 1 of delivery address	30	Receiver_DeliveryAddr
DeliveryAddressLine2	N	Address line 2 of delivery address	30	Receiver_DeliveryAddr
DeliveryAddressLine3	N	Address line 3 of delivery address	30	Receiver_DeliveryAddr
DeliveryCity	N	City where consignment is to be delivered	30	Receiver_DeliveryAddr
DeliveryProvince	N	Province where consignment is to be delivered	30	Receiver_DeliveryAddr
DeliveryPostcode	N	Post code where consignment is to be delivered.	9	Receiver_DeliveryAddr
DeliveryCountry	N	Country where consignment is to be delivered	30	Receiver_DeliveryAddr
DeliveryTel.Number	N	Telephone number of customer	17	Receiver_DeliveryAddr



a TPG company

		where consignment is to be delivered			
DeliveryContactName	N	Contact person at delivery address		22	Receiver_DeliveryAddr
DeliveryContactTel.	N	Telephone number of contact person at delivery address		17	Receiver_DeliveryAddr
DeliveryAccountNo.	N	Account number of customer at delivery address		9	Receiver_DeliveryAddr
DeliveryVATNumber	N	VAT number of customer at delivery address		20	Receiver_DeliveryAddr
PackagesSummary (optional - may be multiple within document)	Y		Only one per consignment element		Consignment
NumberofPieces	N	Number of pieces in package		5	PackagesSummary
Weight	N	Weight of package		9	PackagesSummary
PackingDescription	N	Package description		20	PackagesSummary
GoodsDescription	N	Goods Description		30	PackagesSummary
InvoiceAmount	N	Invoice amount		14	PackagesSummary
InvoiceCurrency	N	Invoice currency		30	PackagesSummary
Articles	N	Articles present in package			PackagesSummary
ShipmentSummary (may be multiple within document)	Y		Only one per consignment element		Consignment
AltConsignment	N	Alternate Consignment Number		15	ShipmentSummary
Division	N	Division code		3	ShipmentSummary
ProductCode	N	Service code		30	ShipmentSummary
OptionCodes	N	Options		30	ShipmentSummary
DueDate	Y	Date due for delivery		11	ShipmentSummary
TermsOfPayment	Y	Payer (Sender or receiver)		8	ShipmentSummary
ShipmentInstruct (may be multiple)					ShipmentSummary
InstructionText	N	Shipment instruction text		50	ShipmentInstruct



a  TPG company

DTD (Document Type Definition)

This may be cut and pasted into a file (for example, in Notepad) and saved as a .dtd file for testing your XML format.

Tracking Input

```
<!-- Tracker input DTD. Version 2.0 -->
<!ELEMENT document
(APPLICATION,VERSION,USER_NAME?,PASSWORD?,TRACK_TYPE,TRACK_DATA)>
<!ELEMENT APPLICATION (#PCDATA)>
<!ELEMENT VERSION (#PCDATA)>
<!ELEMENT USER_NAME (#PCDATA)>
<!ELEMENT PASSWORD (#PCDATA)>
<!ELEMENT TRACK_TYPE (#PCDATA)>
<!ELEMENT TRACK_DATA (ExtraDetails*,Tdata+)>
<!ELEMENT Tdata (#PCDATA)>
<!ELEMENT ExtraDetails (#PCDATA)>
```

ERRORS

There are a number of different errors that may occur when using TNT iConnections. Most of these you will encounter only during the development phase when unsure of the XML format and trying to send XML through a socket connection for the first time. However, your code will need to be able to handle some of the others, such as server unavailability, once your TNT iConnections software is up and running.

PARSER ERRORS

Errors generated by the XML parser are returned as XML and look like this:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes" ?>
<parse_error>
<error_reason>Element content is invalid according to the DTD/Schema. Expecting:
TRACK_TYPE.</error_reason>
<error_line>7</error_line>
<error_linepos>13</error_linepos>
<error_srcText>
- ![CDATA[
<TRACK_DATA>
]]>
</error_srcText>
</parse_error>
```

These can be resolved by making changes to the XML format. Usually the error message will be descriptive, telling you that there was an expected element missing. Refer to the DTD and XML examples to remedy this.

If the parser error gives line number 1 and position 13 to locate the error, this suggests that the XML was invalid as XML, not just as Tracking XML. The most common cause of this is when XML is sent to the TNT server and is not URL encoded. The spaces in the XML are lost as it crosses the World Wide Web, arriving at the TNT server looking like this:

```
<?xmlversion="1.0"standalone="no"><!DOCTYPEdocumentSYSTEM'http://164.39.41.19:81/trackerDTD1.0/trackerIN.dtd'><document>...etc...</document>
```

You can force the URL encoding of your XML by replacing all spaces with a + sign before sending it.

ERRORS RETURNED BY THE VBGATEWAY.ASP (ENTRY FILE)

The entry file will check your login details and return an error if they can not be validated. Ensure that you have submitted them in the correct case, as they were given to you by your TNT contact.

If the XML/request for XML that you send in to the TNT server is an empty string, or not formatted as expected, the following message will be returned:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes" ?>
<runtime_error>
<error_reason>no data received at gateway from client</error_reason>
<error_srcText><![CDATA[]]></error_srcText>
</runtime_error>
```

Ensure that you sent your xml/request in a string starting 'xml_in=' (see the example connection code).

APPLICATION GENERATED ERRORS

Application errors are included in your returned XML as they occur and take the following format:

```
<runtime_error>
<error_reason >reason</errorreason>
<error_src_Text>
- <![CDATA[ ]]>
</error_src_Text>
</runtime_error>
```

For possible errors see Error Table below



a TPG company

TABLE OF APPLICATION GENERATED ERROR CODES, MESSAGES AND RESOLUTIONS

Error Code (For TNT use)	Error Description	Resolution
005	Number of Extra Details requested exceeds the Maximum	There should only be four extra detail elements at maximum in the XML document.
006	Request Type not recognised	Track Type MUST be 'consignment', 'customerref' or 'altcon'
010	Request contains oversized search argument	Check XML document's Tdata elements, one contains a connumber too long for the specified format
011	Search Limit has been exceeded for complete customer request	A maximum of 50 consignments may be tracked per connection
012	Request contains non-alphanumeric search argument	If the track type is consignment or altcon, then the T data element must be alphanumeric
090	Invalid Application	Application MUST be 'Tr'
098	Search has exceeded timeout limit	Resubmit, if same error occurs, reduce number of consignments to be tracked and resubmit
099	There is an error in the server - Timeout exceeded	Resubmit, if error reoccurs, contact TNT Help Desk
999	Could not validate user	The User element must contain the client's valid customer id
	Account Invalid for Extra Details	The consignment's account values did not match any of the track requesting company's accounts, therefore this company is not permitted to view account numbers or any extra details.

SERVER ERRORS

Server errors are returned as HTML messages, not XML. These will occur if the TNT server is unavailable or the service has timed out. In these cases, contact the TNT representative for more information.

RECOMMENDED RESOURCES

To download the Internet Explorer 5.5 browser:

<http://www.microsoft.com/windows/ie/previous/default.asp>

Useful XML resources:

<http://www.xml.com>

<http://www.topxml.com>

<http://www.xmlinfo.com/>

<http://msdn.microsoft.com/library/>