



# eTrans

## Developer Documentation v1.1.0

©1998-2002 Deep Sky Technologies, Inc. All Rights Reserved.  
Published and Distributed Worldwide by Deep Sky Technologies, Inc.

Deep Sky Technologies, Inc.  
P.O. Box 6897  
Vero Beach, FL 32961-6897  
772.794.9494



# Software License and Limited Warranty

Please read this license carefully before using the software. By using the software, you agree to become bound by the terms of this agreement, which includes the software license and warranty disclaimer (collectively referred to herein as the "agreement"). This agreement constitutes the complete agreement between you and Deep Sky Technologies, Inc. If you do not agree to the terms of this agreement, do not use the software and promptly destroy all copies in your possession, physical and electronically.

**1. Ownership of Software:** The enclosed manual and computer programs ("Software") were developed and are copyrighted by Deep Sky Technologies, Inc. ("DSTi") and are licensed, not sold, to you by DSTi for use under the following terms, and DSTi reserves any rights not expressly granted to you. DSTi retains ownership of all copies of the Software itself. Neither the manual nor the Software may be copied in whole or in part except as explicitly stated below.

**2. License:** DSTi, as Licensor, grants to you, the Licensee, a non-exclusive, non-transferable right to use this Software subject to the terms of the license as described below:

- a. You may make backup copies of the Software for your use provided they bear the DSTi copyright notice.
- b. You may use this Software in an unlimited number of custom or commercial databases or applications created by the original licensee. No additional product license or royalty is required.

**3. Restrictions:** You may not distribute copies of the Software to others (except as an integral part of a database or application within the terms of this License) or electronically transfer the Software from one computer to another over a network. You may distribute copies of the Software as an integral part of a development shell or non-compiled commercial database as long as the DSTi copyright notices and documentation remain intact with the distribution. The Software contains trade secrets and to protect them you may not decompile, reverse engineer, disassemble, or otherwise reduce the Software to a human perceivable form. You may not modify, adapt, translate, rent, lease, loan or resell for profit the software or any part thereof.

**4. Termination:** This license is effective until terminated. This license will terminate immediately without notice from DSTi if you fail to comply with any of its provisions. Upon termination you must

destroy the Software and all copies thereof, and you may terminate this license at any time by doing so.

**5. Update Policy:** DSTi may create, from time to time, updated versions of the Software. At its option, DSTi will make such updates available to the Licensee.

**6. Warranty Disclaimer:** The software is provided "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. DSTi does not warrant, guarantee, or make any representations regarding the use, or the results of the use, of the software or written materials in the terms of correctness, accuracy, reliability, currentness or otherwise. The entire risk as to the results and performance of the software is assumed by the Licensee. If the software or written materials are defective you, and not DSTi or its dealers, distributors, agents, or employees, assume the entire cost of all necessary servicing, repair or correction. No oral or written information or advice given by DSTi, its dealers, distributors, agents, or employees shall create a warranty or in any way increase the scope of this warranty, and you may not rely on such information or advice. This warranty gives you specific legal rights. You may have other rights, which vary from state to state.

**7. Governing Law:** This agreement shall be governed by the laws of the State of Florida.

## Copyrights and Trademarks

All trade names referenced in this document are the trademark or registered trademark of their respective holder.

BASh, BASh Pro, TCP Deux, TCP Deux Pro, SMTP Deux, SMTP Deux Pro, POP3 Client Deux, POP3 Client Deux Pro, FTP Client Deux, FTP Client Deux Pro, HTTP Client Deux, HTTP Client Deux Pro, eTrans, TCP Server Deux, TCP Server Deux Pro, HTTP Server Deux, HTTP Server Deux Pro, and HTTP Log Deux are copyright Deep Sky Technologies, Inc.

4th Dimension, ACI, ACI US, 4D Compiler, 4D, 4D Server, 4D Client, and 4D Insider are trademarks of 4D, Inc.

Internet ToolKit plugin provided courtesy, and with permission, of Christian Quest.

Macintosh and MacOS are trademarks of Apple Computer, Inc.

Windows is a trademark of Microsoft Corporation.

## Preface

The eTrans component is designed to work in conjunction with components from Deep Sky Technologies, Inc. Specifically, the eTrans component requires that the HTTP Client Deux and TCP Deux components already be installed in your database structure file. In addition, all of the above components require that the BASH component, available for free from DSTi, also be installed into your database structure file.

Make certain that you view the compatibility matrix for components available to make certain you are using compatible versions of the different components required. There is a compatibility matrix available in this manual; the most recent compatibility matrix is available on the DSTi web site.

## Acknowledgements

The creation of the eTrans component is not directly attributable to any single person. Particular pieces of functionality within the eTrans component may be from the direct knowledge and experience of certain developers, but the overall concept and construction of the eTrans component has come from all of the developers at Deep Sky Technologies, Inc.

In particular, the tireless efforts of James A. Crate have contributed the most to the eTrans component. His ability and patience, finding and wading through the veritable plethora of documentation, have proven invaluable in the development of the eTrans component.

In addition, this component would have been much more difficult without the efforts Robert T. McGoye has put into providing the incredibly useful HTTP Client Deux component, as well as the tremendous amount of work he has put into TCP Deux and BASH core routines.

Finally, I, Steven G. Willis, might have had something to do with the creation of the eTrans component...

## Features

eTrans is a 4th Dimension component that provides a single, consistent interface to a variety of online credit card authorization services. This makes it easy to utilize multiple authorization services, or switch between them simply and easily. Even if you only need to use one of the services, the eTrans component offers the advantage of having a simple interface to build and process the transactions.

eTrans requires that HTTP Client Deux be installed. In addition, most online credit card authorization services require SSL connections, so ITK 2.5.x will be necessary if you need to use SSL.

HTTP Client Deux works "on top" of TCP Deux, another 4D component available from DSTi. This provides completely transparent operation with 4D Internet Commands v6.7.x, Internet ToolKit v2.0.x, and Internet ToolKit v2.5.x; any of these plugins can be used without any coding changes. The only exception to this is for HTTPS requests to function properly you must be using a valid copy of ITK Pro v2.5.x.

All of the components require that the BASH 4D component be installed for it to operate correctly. BASH is a 4D component available for free from DSTi and it handles all of the core routines needed to properly run TCP Deux, HTTP Client Deux, and eTrans.

All of the code within the eTrans component is ready to use immediately. There are no compiler issues to be concerned about; all of the variables are typed in compiler methods for even the most stringent 4D developers. Just install the component in your structure and start using it. It is really that simple!

## System Requirements

The eTrans component is compatible with both Macintosh and Windows installations of 4th Dimension.

Since it is a component, it does require at least version 6.7 of 4th Dimension or above, including 4D Insider v6.7 or above for installation.

Other than the normal hardware and software requirements for your version of 4th Dimension, there are no other minimum requirements for proper use of this software.

## Support

Support is provided for the eTrans component free of charge for all currently licensed users. Included support services provided for all currently licensed users encompasses all of the online support services available through the DSTi web site (email, FAQ, messaging, etc.). Check the DSTi web site for current direct support options available; we are always working to offer more resources for your needs.

Contact information, including email address(es), phone number(s), and a Contact Us request form, for Deep Sky Technologies, Inc., can be found on the DSTi web site located at:

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

If there are terms or conventions which you find difficult to understand in relation to the eTrans component or TCP protocols and servers in general, feel free to contact Deep Sky Technologies, Inc., support. We will be more than happy to help you in any way we reasonably can. And, only through your questions do we know what subjects to include in future versions of this manual.

## Components

A component groups various 4D objects (tables, project methods, forms, menu bars, variables, etc.) representing one or more additional functions. Developing a 4D component providing electronic mail functionality is one such example. A component is autonomous and must be able to be installed in any 4D structure.

Components are defined, generated, and installed with the help of 4D Insider. The component definition is based on the cross referencing analysis performed by 4D Insider (target objects and source objects).

Unlike libraries and groups, components embed the idea of security of objects that they compose. During the development phase of the component, each object is attributed an access type, "Public", "Protected" or "Private". This attribute determines whether each object will be visible or modifiable in 4th Dimension and in 4D Insider once the component is installed within a 4D database.

# Installing and Updating eTrans

Installing eTrans or updating an existing version of eTrans within a 4D database is performed using 4D Insider. The activity primarily consists of installing the eTrans component in a database structure opened with 4D Insider (installing the eTrans component in a library is not supported at this time).

4D Insider will manage possible conflict issues within the installation and will inform you as they are detected. Though, with the naming conventions used within the eTrans component and the limited number of object names, conflicts should be very rare.

To install or update the eTrans component, follow these very simple steps:

**Open the uncompiled structure that you wish to install eTrans into using 4D Insider.**

**Choose the "Install/Update..." command in the "Components" menu.**

A standard open file dialog box will appear.

**Select the eTrans component file and click on the "Open" button.**

4D Insider parses the eTrans component and prepares to integrate it with your open database. 4D Insider will detect if the operation is an installation or an update of the eTrans component.

In the event of a new installation, all eTrans objects are installed.

In the event of an update, 4D Insider compares the version numbers of both the currently installing eTrans component and the already installed eTrans component. If the date of the "new" component is older than the already installed component, a dialog box will alert you, allowing you to then "Continue" or "Cancel" the update.

4D Insider replaces old objects with newer objects within the eTrans component and adds new objects from the new eTrans component. 4D Insider takes into account "public" objects having been modified by you (e.g. "\_ERROR" methods)

and will prompt you to either save or replace them. If any other conflicts arise from the installation or update of the eTrans component, 4D Insider will prompt you with an appropriate dialog box.

### **Save the database in 4D Insider.**

### **Place a copy of the Affix eTrans document in the 4DX folder.**

The Affix eTrans document contains essential data and resources for many of the methods within the eTrans component. For many of the methods within the eTrans component to function properly, the Affix eTrans document must be in the 4DX folder for the current structure.

On Macintosh, the Affix eTrans document is entitled **Affix\_eTrans.4DX** (**Affix\_eTrans.4CX** under MacOS X and 4D v6.8.x) and is located in the Mac4DX directory in the eTrans component's archive. The document should be copied into the Mac4DX folder of your current structure. If the eTrans component is going to be used in all of your 4D projects, the Affix eTrans document can instead be placed within the Mac4DX folder within the 4D folder of your system.

On Windows, the Affix eTrans document is actually two documents: **Affix\_eTrans.4DX** and **Affix\_eTrans.RSR**. These two documents correspond to the data fork and the resource fork of the Affix eTrans document used on Macintosh. These documents are located in the Win4DX directory in the Logger Deux component's archive. These documents should be copied into the Win4DX folder of your current structure. If the Logger Deux component is going to be used in all of your 4D projects, the Affix eTrans document can instead be placed within the Win4DX folder within the 4D folder of your system.

For client/server installations in cross-platform environments, both the Macintosh and Windows versions of the Affix eTrans document should be installed.

### **Call the method *INIT\_ETRANS* early in the On Startup database method.**

To initialize the eTrans component in your code, place a call to the method ***INIT\_ETRANS*** early in your **On Startup** data-

base method. However, this method must be after the calls to *INIT\_BASH*, *INIT\_TCPd*, and *INIT\_TCPsd*.

Details about the *INIT\_ETRANS* method can be found in the module and method documentation, below.

The eTrans component is now installed/updated in your database and is listed on the "Components" page of the 4D Explorer.

### **Managing Installation Conflicts**

On very rare occasions, when the eTrans component is installed or updated in your 4D database, several questions and conflicts may arise. In the event of an update, 4D Insider will detect that you have modified one or more "Public" objects in eTrans after the initial installation. Or, one or more objects of the same type and of the same name may already exist in your database and in the eTrans component.

4D Insider detects and solves these conflicts during installation:

#### **Modified public objects (updates only)**

In this case, 4D Insider alerts you by a dialog box, allowing you to choose an update mode:

- Replace the object
- Replace all objects
- Do not replace the object
- Stop installation

#### **Name conflicts**

In this case, 4D Insider stops the eTrans installation process, alerts you through a dialog box and saves the list of objects in conflict. This list is stored as a text file in the 4D database folder.

Naming conflicts between logical objects, such as variables, are managed by 4D Insider, in a manner that allows database

compilation and avoids conflicts between eTrans and other 4D components.

It may be necessary to rename certain objects in your database or in other components in order to be able to install the eTrans component.

If any naming conflicts do occur between eTrans and other 4D components, please notify Deep Sky Technologies, Inc., immediately.

### **Affix eTrans Document**

The Affix eTrans document (entitled **Affix\_eTrans.4DX** on MacOS 8/9; entitled **Affix\_eTrans.4CX** on MacOS X and 4D v6.8.x; entitled **Affix\_eTrans.4DX** and **Affix\_eTrans.RSR** on Windows) contains essential data and resources for many of the methods within the eTrans component. For many of the methods within the eTrans component to function properly, the Affix eTrans document must be in the 4DX folder for the current structure. For distributed and installed versions of your 4D projects, the Affix eTrans document must be available in the 4DX folder for the eTrans methods to continue to function properly.

**Note:** it is best to consider the Affix eTrans document another plug-in within your 4D project. Though there no actual plug-in calls available within the Affix eTrans document, it does contain data and resources essential to the operation of the eTrans methods. The eTrans component has been designed to find the Affix eTrans document correctly in all environments (any platform, any 4DX folder, single user or clients/server, etc.). Since the Affix eTrans document is configured similarly to plug-ins, 4D and the eTrans component will automatically manage the document for you in all of the possible installations of a 4D project.

# Uninstalling eTrans

4D Insider allows you to uninstall the eTrans component from your 4D database.

To uninstall eTrans from your 4D database:

**Using 4D Insider, open your database containing the copy of eTrans to be uninstalled.**

**In the "Main" listing window, select the eTrans component.**

**Consider again how great the eTrans component is and make certain that you will really no longer need it in your 4D database.**

**Select the "Uninstall..." command in the "Components" menu.**

This command is only active when a component is installed in the database. A dialog box appears allowing you to confirm or cancel the operation. If you are uncertain about the previous step then the cancel option is probably your best choice at this time.

**Click "OK" to validate the operation.**

**Remove the Affix eTrans document from your 4DX folder.**

**Remove the call to the method *INIT\_ETRANS* from your On Startup database method.**

All objects from the eTrans component are deleted from your 4D database. Obviously, you are now very sad to no longer have the eTrans component in your 4D database. Crying is allowed...

## Basics of eTrans

eTrans provides a simple interface for working with different credit card authorization services. At the heart of this system is the eTrans transaction.

Methods are provided to create transactions for various services. After creating the transaction successfully, extra optional information can be added to the eTrans transaction. When all desired information has been added to the eTrans transaction, the transaction can be processed with the appropriate authorization service provider. The transaction result is returned in a BLOB for easy access.

eTrans does not do any processing on the results returned from processing a transaction. However, BASH, which is required to use eTrans, has many parsing routines to make working with the transaction results much easier.

## Direct vs. Relay Transactions

eTrans can perform two different types of online credit card authorization transactions. These two different types are **direct** transactions and **relay** transactions.

**Direct** transactions are those in which eTrans communicates with the credit card authorization service, and returns the result **directly** in the HTTP response. There is no other communication outside of this direct request and response.

**Relay** transactions are intended to allow the customer to enter their credit card information in a web page served by the credit card authorization service provider. The information is then **relayed** to a URL which the developer has specified. This two-step process is more difficult to coordinate, and more susceptible to errors. It also requires that you have a server capable of receiving these relayed transactions. Most people only use these relay services when they do not have the facilities to securely accept the credit card information themselves.

While eTrans supports relay transactions, it does so by sending the request with all the information, just as if the user had entered it into the provider's site directly. So, you can not use eTrans to perform relay transactions in the traditional manner, where the customer enters their credit card information directly on the provider's website.

We have found during development of eTrans that all of the services we support with relay transactions will return the transaction results in the HTTP response. In some cases, they return the fields directly, so you only need to set the relay URL to some URL that can accept a POST and return a result. In other cases, the HTTP response will contain the HTML returned by the relay URL, in which case you need to set the relay URL to a "bounce" CGI which will simply echo the posted response back to the service provider.

## Supported Authorization Services

eTrans supports several online credit card authorization services with both direct and relay transactions. Following is a list of the credit card authorization services that eTrans supports:

**AuthorizeNet** (available online at <http://www.authorizenet.com/>): eTrans supports AuthorizeNet for both direct and relay transactions. In addition, eTrans supports the eCheck services offered by AuthorizeNet, in both direct and relay transactions as well. eCheck transactions allow you to draw directly from a bank account, just as if a check had been written.

**PlanetPayment** (available online at <http://www.planetpayment.com/>): PlanetPayment is currently the same as AuthorizeNet, but oriented toward international usage. eTrans has the same support for PlanetPayment as for AuthorizeNet. In addition, eTrans supports PlanetPayment's exchange rate API to allow developers to easily check exchange rates.

**E-xact** (available online at <http://www.e-xact.com/>): eTrans only supports relay transactions for E-xact. The authorization info is returned in the HTTP response for the transaction, but a relay URL will need to be provided for E-xact to function correctly.

**LinkPoint** (available online at <http://www.linkpoint.com/>): eTrans only supports direct transactions with LinkPoint. However, even though eTrans handles it as a direct transaction, it is really a relay transaction in disguise. For this reason, you'll need to specify a URL, which doesn't even need to exist, on your LinkPoint account setup. The same URL will be required to create a LinkPoint transaction.

**WorldPay** (available online at <http://www.worldpay.com/>): WorldPay only supports relay transactions, so eTrans can only provide capability to run relay transactions. WorldPay does not require a relay URL to be configured, but does not return the authorization info in the response. The only way to retrieve the authorization info in the response is to configure a relay URL which echoes that information back to WorldPay, which can include that information in the response to the transaction. In addition, WorldPay requires an email address in the transaction in order to send a receipt; if you are handling communications with your customer yourself, you'll need to provide your own email address to receive these emails.

**IonGate** (available online at <http://www.iongate.com/>): eTrans supports both direct and relay transactions with IonGate.

**FirePay** (available online at <http://www.firepay.com/>): FirePay only supports direct response transactions. eTrans supports the purchase transaction, but neither of the failure recovery transactions outlined in the FirePay technical reference. FirePay also supports their own internal user account system. These accounts are specified with a credit card type of "FP" (FirePay).

**SurePay** (available online at <http://www.surepay.com/>): SurePay only supports direct response transactions. eTrans uses the SurePay SDK, not their new Verisign PayFlow Link and PayFlow Pro products. Because of this, you can communicate exclusively with SurePay instead of having to work through the systems of multiple separate companies.

# Multi-Currency Support

Since PlanetPayment is oriented toward international usage, it supports multiple currencies. PlanetPayment accounts can be configured to accept transactions in multiple currencies. A field is provided to specify the currency of the current transaction. If the merchant does not transact in that currency, the amount will be converted according to the current exchange rate.

PlanetPayment also offers an exchange rate API. This API is supported as just another transaction type, to maintain the simple, consistent interface within eTrans. This API allows the developer to determine the current exchange rates quickly and easily.

Currency codes follow the ISO 4217 standard. While these codes can be easily found on the internet, we have included a list of currency codes here for your convenience:

<b>Code</b>	<b>Location Used</b>	<b>Monetary Unit</b>
AED	United Arab Emirates	Dirham
AFA	Afghanistan	Afghani
ALL	Albania	Lek
AMD	Armenia	Dram
ANG	Netherlands Antilles	Guilder
AON	Angola	New Kwanza
ARP	Argentina	Peso
ATS	Austria	Schilling
AUD	Australia	Dollar
AWF	Aruba	Florin
AZM	Azerbaijan	Manat
BAK	Bosnia	Convertible Mark
BBD	Barbados	Dollar
BDT	Bangladesh	Taka
BEF	Belgium	Franc
BGL	Bulgaria	Lev
BHD	Bahrain	Dinar

<b>Code</b>	<b>Location Used</b>	<b>Monetary Unit</b>
BIF	Burundi	Burundi Franc
BMD	Bermuda	Dollar
BND	Brunei Darussalam	Dollar
BOB	Bolivia	Boliviano
BRL	Brazil	Brazilian Real
BSD	Bahamas	Dollar
BTR	Bhutan	Rupee
BWP	Botswana	Pula
BYR	Belarus	Ruble
BZD	Belize	Dollar
CAD	Canada	Dollar
CDF	Congo-Kinshasa	Congolese Franc
CHF	Switzerland	Franc
CLP	Chile	Peso
CNY	China	Yuan Renminbi
COP	Colombia	Peso
CRC	Costa Rica	Colon
CZK	Czech Republic	Koruna
CUP	Cuba	Peso
CVE	Cape Verde	Escudo
CYP	Cyprus	Pound
DEM	Germany	Deutsche Mark
DJF	Djibouti	Franc
DKK	Denmark	Krone
DOP	Dominican Republic	Peso
DZD	Algeria	Algerian Dinar
ECS	Ecuador	Sucre
EEK	Estonia	Kroon
EGP	Egypt	Pound
ERN	Eritrea	Nakfa
ESP	Spain	Peseta

<b>Code</b>	<b>Location Used</b>	<b>Monetary Unit</b>
ETB	Ethiopia	Birr
EUR	European Union (EMU)	Euro
FIM	Finland	Markka
FJD	Fiji	Dollar
FKP	Falkland Islands	Pound
FRF	France	Franc
GBP	Great Britain	Pound Sterling
GEL	Georgia	Lari
GHC	Ghana	Cedi
GIP	Gibraltar	Pound
GMD	Gambia	Dalasi
GNF	Guinea	Franc
GRD	Greece	Drachma
GTQ	Guatemala	Quetzal
GYD	Guyana	Dollar
HKD	Hong Kong	Dollar
HNL	Honduras	Lempira
HRK	Croatia	Kuna
HTG	Haiti	Gourde
HUF	Hungary	Forint
IDR	Indonesia	Rupiah
IEP	Eire (Ireland)	Punt
IEP	Ireland (Eire)	Punt
ILS	Israel	Shekel
INR	India	Rupee
IQD	Iraq	Dinar
IRR	Iran	Rial
ISK	Iceland	Krona
ITL	Italy	Lira
JMD	Jamaica	Dollar
JOD	Jordan	Dinar

<b>Code</b>	<b>Location Used</b>	<b>Monetary Unit</b>
JPY	Japan	Yen
KES	Kenya	Shilling
KGS	Kyrgyzstan	Som
KHR	Cambodia	Riel
KMF	Comoros	Franc
KPW	Korea (North)	Won
KRW	Korea (South)	Won
KWD	Kuwait	Dinar
KYD	Cayman Islands	Dollar
KZT	Kazakstan	Tenge
LAK	Laos	Kip
LBP	Lebanon	Pound
LKR	Sri Lanka	Rupee
LRD	Liberia	Dollar
LSL	Lesotho	Loti
LTL	Lithuania	Lita
LUF	Luxembourg	Franc
LVL	Latvia	Lat
LYD	Libya	Dinar
MAD	Morocco	Dirham
MDL	Moldova	Leu
MGF	Madagascar	Malagasy Franc
MKD	Macedonia	Denar
MMK	Myanmar (Burma)	Kyat
MNT	Mongolia	Tugrik
MOP	Macau	Pataca
MRO	Mauritania	Ouguiya
MTL	Malta	Lira
MUR	Mauritius	Rupee
MVR	Maldives	Rufiyaa
MWK	Malawi	Kwacha

<b>Code</b>	<b>Location Used</b>	<b>Monetary Unit</b>
MXP	Mexico	Peso
MYR	Malaysia	Ringgit
MZM	Mozambique	Metical
NAD	Namibia	Dollar
NGN	Nigeria	Naira
NIO	Nicaragua	Cordoba Oro
NLG	Holland	Guilder
NOK	Norway	Krone
NPR	Nepal	Nepalese Rupee
NZD	New Zealand	Dollar
OMR	Oman	Sul Rial
PAB	Panama	Balboa
PEN	Peru	Nuevo Sol
PGK	Papua New Guinea	Kina
PHP	Philippines	Peso
PKR	Pakistan	Rupee
PLZ	Poland	Zloty
PTE	Portugal	Escudo
PYG	Paraguay	Guarani
QAR	Qatar	Rial
ROL	Romania	Leu
RUR	Russia	Ruble
RWF	Rwanda	Rwanda Franc
SAR	Saudi Arabia	Riyal
SBD	Solomon Islands	Dollar
SBL	Seborga	Luigino
SCR	Seychelles	Rupee
SDD	Sudan	Dinar
SEK	Sweden	Krona
SGD	Singapore	Dollar
SHP	Saint Helena	Pound

<b>Code</b>	<b>Location Used</b>	<b>Monetary Unit</b>
SIT	Slovenia	Tolar
SKK	Slovakia	Koruna
SLL	Sierra Leone	Leone
SOS	Somalia	Shilling
SRG	Suriname	Guilder
STD	São Tome and Principe	Dobra
SVC	El Salvador	Colon
SYP	Syria	Pound
SZL	Swaziland	Lilangeni
THB	Thailand	Baht
TJR	Tajikistan	Ruble
TMM	Turkmenistan	Manat
TND	Tunisia	Dinar
TOP	Tonga	Pa'anga
TRL	Turkey	Lira
TTD	Trinidad and Tobago	Dollar
TWD	Taiwan	Dollar
TZS	Tanzania	Shilling
UAH	Ukraine	Hryvnia
UGX	Uganda	Shilling
USD	United States of America	US Dollar
UYU	Uruguay	Peso
UZS	Uzbekistan	Som
VEB	Venezuela	Bolivar
VND	Viet Nam	Dong
VUV	Vanuatu	Vatu
WST	West Samoa	Tala
YER	Yemen	Rial
YUN	Yugoslavia	New Dinar
ZAR	South Africa	Rand
ZMK	Zambia	Kwacha

<b>Code</b>	<b>Location Used</b>	<b>Monetary Unit</b>
ZWD	Zimbabwe	Zimbabwe Dollar

## eTrans Constants

There are a minimal number of custom constants included with the eTrans component package. These constants are grouped into two convenient constant groups for easier referencing and organization.

Where appropriate, it is highly recommended that the custom constants included with the eTrans component be utilized within your code; this will considerably simplify future feature enhancements to the core code within eTrans.

## eTrans Transaction Types

The eTrans Transaction Types constants group is a group of constants provided with the eTrans component. This group of constants provides a compact identification mechanism for different transaction types that can be handled by the eTrans component. A different constant defines each of the different transaction types supported.

The following table lists all of the eTrans Transaction Type constants, their values, and the common name associated with each value:

Constant	Value	Transaction Type
ETRANS_TT_AN_Direct	1	Authorize.Net Credit Card (Direct)
ETRANS_TT_AN_Relay	5	Authorize.Net Credit Card (Relay)
ETRANS_TT_ANEC_Direct	2	Authorize.Net E-Check (Direct)
ETRANS_TT_ANEC_Relay	6	Authorize.Net E-Check (Relay)
ETRANS_TT_EXact_Relay	9	E-Xact Credit Card (Relay)
ETRANS_TT_LP_Direct	10	LinkPoint Credit Card (Direct)
ETRANS_TT_PP_Direct	3	PlanetPayment Credit Card (Direct)
ETRANS_TT_PP_Relay	7	PlanetPayment Credit Card (Relay)
ETRANS_TT_PPEC_Direct	4	PlanetPayment E-Check (Direct)
ETRANS_TT_PPEC_Relay	8	PlanetPayment E-Check (Relay)
ETRANS_TT_PP_Exchange	11	PlanetPayment Exchange Rate API
ETRANS_TT_WP_Relay	12	WorldPay Credit Card (Relay)
ETRANS_TT_WP_Exchange	13	WorldPay Exchange Rate API
ETRANS_TT_IG_Direct	14	IonGate Credit Card (Direct)
ETRANS_TT_IG_Relay	15	IonGate Credit Card (Relay)
ETRANS_TT_FP_Direct	16	FirePay Credit Card (Direct)
ETRANS_TT_SP_Direct_Test	17	SurePay Credit Card (Direct) Test Transaction
ETRANS_TT_SP_Direct	18	SurePay Credit Card (Direct) Live Transaction

## eTrans Field Types

The eTrans Field Types constants group is a group of constants provided with the eTrans component. This group of constants provides a compact identification mechanism for the fields supported by the various transaction types. Each transaction type supports a subset of these fields. These field constants are mapped to the actual field names required for each transaction type. This abstraction of the field names allows you to switch between transaction types much more easily. A different constant defines each of the different transaction fields supported.

The following table lists all of the eTrans Field Type constants and their values:

Constant	Value
ETRANS_AccountID	69
ETRANS_Amount	7
ETRANS_AuthCode	8
ETRANS_BankABACode	10
ETRANS_BankAcctNum	11
ETRANS_BankAcctType	12
ETRANS_BankName	13
ETRANS_CardNum	14
ETRANS_CardSecurityCode	68
ETRANS_CardType	62
ETRANS_City	15
ETRANS_Company	16
ETRANS_Country	17
ETRANS_CustomerID	18
ETRANS_CustomerIP	19
ETRANS_CustomerTaxID	20
ETRANS_DateTime	9
ETRANS_DelimitData	2
ETRANS_DelimiterCharacter	1
ETRANS_Description	21
ETRANS_DriversLicenseDOB	24

Constant	Value
ETRANS_DriversLicenseNum	22
ETRANS_DriversLicenseState	23
ETRANS_EmailAddressCustomer	25
ETRANS_EmailAddressMerchant	37
ETRANS_EmailCustomer	26
ETRANS_EmailMerchant	27
ETRANS_EncapsulateCharacter	3
ETRANS_ExpirationDate	28
ETRANS_ExpirationMonth	58
ETRANS_ExpirationYear	59
ETRANS_Fax	29
ETRANS_FirstName	30
ETRANS_FooterEmailReceipt	31
ETRANS_Freight	32
ETRANS_FromCurrency	66
ETRANS_FullName	60
ETRANS_HeaderEmailReceipt	33
ETRANS_InvoiceNum	34
ETRANS_LastName	35
ETRANS_Login	36
ETRANS_Method	38
ETRANS_Password	39
ETRANS_Phone	40
ETRANS_PONum	41
ETRANS_Province	64
ETRANS_RelayResponse	4
ETRANS_ShipToCity	43
ETRANS_ShipToCompany	44
ETRANS_ShipToCountry	45
ETRANS_ShipToFirstName	46
ETRANS_ShipToFullName	61

<b>Constant</b>	<b>Value</b>
ETRANS_ShipToLastName	47
ETRANS_ShipToProvince	65
ETRANS_ShipToState	48
ETRANS_ShipToStreetAddress	42
ETRANS_ShipToZip	49
ETRANS_State	50
ETRANS_StreetAddress	6
ETRANS_Subtotal	63
ETRANS_Tax	51
ETRANS_TaxExempt	52
ETRANS_TestRequest	53
ETRANS_ToCurrency	67
ETRANS_TransactionID	54
ETRANS_TransactionType	55
ETRANS_URL	5
ETRANS_Version	56
ETRANS_Zip	57

## Field Type Mappings

The following is a detailed listing of the eTrans Field Types supported for each transaction type.

## eTrans Transaction Types

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETRANS\_AN\_Create\_Direct***.

Constant	Field Name	Description
ETRANS_DelimitData	x_ADC_Delim_Data	Set when transaction is created Cannot be modified
ETRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETRANS_URL	x_ADC_URL	Set when transaction is created Cannot be modified
ETRANS_Login	x_Login	AuthorizeNet login name
ETRANS_CardNum	x_Card_Num	Credit card number Must be only numeric characters
ETRANS_ExpirationDate	x_Exp_Date	Credit card expiration date Must be in one of the following formats: mmyy mm/yy mm/yyyy
ETRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers
ETRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers

Constant	Field Name	Description
ETRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETRANS_BankName	x_Bank_Name	Bank name
ETRANS_CardSecurityCode	x_Card_Code	CVV2 code
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character

Constant	Field Name	Description
ETRANS_Method	x_Method	AuthorizeNet transaction method Must be "CC" if used
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country

Constant	Field Name	Description
ETRANS_Phone	x_Phone	Customer phone number
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Addresses	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

# AuthorizeNet Credit Card (Relay)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_AN\_Create\_Relay***.

Constant	Field Name	Description
ETTRANS_RelayResponse	x_ADC_Relay_Response	Set when transaction is created Cannot be modified
ETTRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETTRANS_URL	x_ADC_URL	URL for relay response
ETTRANS_Login	x_Login	AuthorizeNet login name
ETTRANS_CardNum	x_Card_Num	Credit card number Must be only numeric characters
ETTRANS_ExpirationDate	x_Exp_Date	Credit card expiration date Must be in one of the following formats: mmyy mm/yy mm/yyyy
ETTRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETTRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETTRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETTRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers
ETTRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers

Constant	Field Name	Description
ETRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETRANS_BankName	x_Bank_Name	Bank name
ETRANS_CardSecurityCode	x_Card_Code	CVV2 code
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character

Constant	Field Name	Description
ETRANS_Method	x_Method	AuthorizeNet transaction method Must be "CC" if used
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country

<b>Constant</b>	<b>Field Name</b>	<b>Description</b>
ETRANS_Phone	x_Phone	Customer phone number
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Addresses	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

## AuthorizeNet eCheck (Direct)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETRANS\_ANEC\_Create\_Direct***.

Constant	Field Name	Description
ETRANS_DelimitData	x_ADC_Delim_Data	Set when transaction is created Cannot be modified
ETRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETRANS_URL	x_ADC_URL	Set when transaction is created Cannot be modified
ETRANS_Method	x_Method	Set when transaction is created Cannot be modified
ETRANS_CardNum	x_Card_Num	Set when transaction is created Cannot be modified
ETRANS_ExpirationDate	x_Exp_Date	Set when transaction is created Cannot be modified
ETRANS_Login	x_Login	AuthorizeNet login name
ETRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers
ETRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers
ETRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETRANS_BankName	x_Bank_Name	Bank name
ETRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"

Constant	Field Name	Description
ETRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE

Constant	Field Name	Description
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country
ETRANS_Phone	x_Phone	Customer phone number
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company

Constant	Field Name	Description
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Addresses	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

## AuthorizeNet eCheck (Relay)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETRANS\_ANEC\_Create\_Relay***.

Constant	Field Name	Description
ETRANS_RelayResponse	x_ADC_Relay_Response	Set when transaction is created Cannot be modified
ETRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETRANS_Method	x_Method	Set when transaction is created Cannot be modified
ETRANS_CardNum	x_Card_Num	Set when transaction is created Cannot be modified
ETRANS_ExpirationDate	x_Exp_Date	Set when transaction is created Cannot be modified
ETRANS_URL	x_ADC_URL	URL for relay response
ETRANS_Login	x_Login	AuthorizeNet login name
ETRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers
ETRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers
ETRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETRANS_BankName	x_Bank_Name	Bank name
ETRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"

Constant	Field Name	Description
ETRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE

Constant	Field Name	Description
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country
ETRANS_Phone	x_Phone	Customer phone number
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company

Constant	Field Name	Description
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Addresses	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

## PlanetPayment Credit Card (Direct)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETRANS\_PP\_Create\_Direct***.

Constant	Field Name	Description
ETRANS_DelimitData	x_ADC_Delim_Data	Set when transaction is created Cannot be modified
ETRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETRANS_URL	x_ADC_URL	Set when transaction is created Cannot be modified
ETRANS_Login	x_Login	AuthorizeNet login name
ETRANS_CardNum	x_Card_Num	Credit card number Must be only numeric characters
ETRANS_ExpirationDate	x_Exp_Date	Credit card expiration date Must be in one of the following formats: mmyy mm/yy mm/yyyy
ETRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_FromCurrency	x_Currency_Code	ISO 4217 currency code of this transaction
ETRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers
ETRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers

Constant	Field Name	Description
ETRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETRANS_BankName	x_Bank_Name	Bank name
ETRANS_CardSecurityCode	x_Card_Code	CVV2 code
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character

Constant	Field Name	Description
ETRANS_Method	x_Method	AuthorizeNet transaction method Must be "CC" if used
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country

Constant	Field Name	Description
ETRANS_Phone	x_Phone	Customer phone number
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Addresses	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

## PlanetPayment Credit Card (Relay)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETRANS\_PP\_Create\_Relay***.

Constant	Field Name	Description
ETRANS_RelayResponse	x_ADC_Relay_Response	Set when transaction is created Cannot be modified
ETRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETRANS_URL	x_ADC_URL	URL for relay response
ETRANS_Login	x_Login	AuthorizeNet login name
ETRANS_CardNum	x_Card_Num	Credit card number Must be only numeric characters
ETRANS_ExpirationDate	x_Exp_Date	Credit card expiration date Must be in one of the following formats: mmyy mm/yy mm/yyyy
ETRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_FromCurrency	x_Currency_Code	ISO 4217 currency code of this transaction
ETRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers

Constant	Field Name	Description
ETRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers
ETRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETRANS_BankName	x_Bank_Name	Bank name
ETRANS_CardSecurityCode	x_Card_Code	CVV2 code
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyyyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character

Constant	Field Name	Description
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character
ETRANS_Method	x_Method	AuthorizeNet transaction method Must be "CC" if used
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state

<b>Constant</b>	<b>Field Name</b>	<b>Description</b>
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country
ETRANS_Phone	x_Phone	Customer phone number
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Address	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

## PlanetPayment eCheck (Direct)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETRANS\_PPEC\_Create\_Direct***.

Constant	Field Name	Description
ETRANS_DelimitData	x_ADC_Delim_Data	Set when transaction is created Cannot be modified
ETRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETRANS_URL	x_ADC_URL	Set when transaction is created Cannot be modified
ETRANS_Method	x_Method	Set when transaction is created Cannot be modified
ETRANS_CardNum	x_Card_Num	Set when transaction is created Cannot be modified
ETRANS_ExpirationDate	x_Exp_Date	Set when transaction is created Cannot be modified
ETRANS_Login	x_Login	AuthorizeNet login name
ETRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers
ETRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers
ETRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETRANS_BankName	x_Bank_Name	Bank name
ETRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"

Constant	Field Name	Description
ETRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_FromCurrency	x_Currency_Code	ISO 4217 currency code of this transaction
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character

Constant	Field Name	Description
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country
ETRANS_Phone	x_Phone	Customer phone number

<b>Constant</b>	<b>Field Name</b>	<b>Description</b>
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Addresses	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

## PlanetPayment eCheck (Relay)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_PPEC\_Create\_Relay***.

Constant	Field Name	Description
ETTRANS_RelayResponse	x_ADC_Relay_Response	Set when transaction is created Cannot be modified
ETTRANS_Version	x_Version	Set when transaction is created Cannot be modified
ETTRANS_Method	x_Method	Set when transaction is created Cannot be modified
ETTRANS_CardNum	x_Card_Num	Set when transaction is created Cannot be modified
ETTRANS_ExpirationDate	x_Exp_Date	Set when transaction is created Cannot be modified
ETTRANS_URL	x_ADC_URL	URL for relay response
ETTRANS_Login	x_Login	AuthorizeNet login name
ETTRANS_Amount	x_Amount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETTRANS_BankAcctNum	x_Bank_Acct_Num	Bank account number Can only contain numbers
ETTRANS_BankABACode	x_Bank_ABA_Code	Bank ABA code Can only contain numbers
ETTRANS_BankAcctType	x_Bank_Acct_Type	Bank Account Type Must be one of the following: CHECKING SAVINGS
ETTRANS_BankName	x_Bank_Name	Bank name
ETTRANS_Tax	x_Tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"

Constant	Field Name	Description
ETRANS_Freight	x_Freight	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_FromCurrency	x_Currency_Code	ISO 4217 currency code of this transaction
ETRANS_CustomerIP	x_Customer_IP	Customer IP address Must be dotted IP address format
ETRANS_CustomerTaxID	x_Customer_Tax_ID	Customer Tax ID 20 digits, numbers and spaces only
ETRANS_DriversLicenseNum	x_Drivers_License_Num	Drivers license number
ETRANS_DriversLicenseState	x_Drivers_License_State	Drivers license state Two-character state abbreviation
ETRANS_DriversLicenseDOB	x_Drivers_License_DOB	Date of birth on drivers license Use format yyyymmdd
ETRANS_EmailAddressCustomer	x_Email	Valid email address
ETRANS_EmailAddressMerchant	x_Merchant_Email	Valid email address
ETRANS_EmailCustomer	x_Email_Customer	Specify whether to send email to customer Must be one of the following: TRUE FALSE
ETRANS_EmailMerchant	x_Email_Merchant	Specify whether to send email to merchant Must be one of the following: TRUE FALSE
ETRANS_HeaderEmailReceipt	x_Header_Email_Receipt	Header for email receipt sent to customer
ETRANS_FooterEmailReceipt	x_Footer_Email_Receipt	Footer for email receipt sent to customer
ETRANS_DelimiterCharacter	x_ADC_Delim_Character	Delimiter character for transaction results Can be any character
ETRANS_EncapsulateCharacter	x_ADC_Encapsulate_Character	Character that encapsulates each field in results Can be any character

Constant	Field Name	Description
ETRANS_TaxExempt	x_Tax_Exempt	Specify whether customer is tax-exempt Must be one of the following: TRUE FALSE
ETRANS_TestRequest	x_Test_Request	Specify whether this is a test request Must be one of the following: TRUE FALSE
ETRANS_TransactionType	x_Type	AuthorizeNet transaction type Must be one of the following: AUTH_CAPTURE AUTH_ONLY CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_Password	x_Password	Only required when x_Type is one of: CAPTURE_ONLY CREDIT VOID PRIOR_AUTH_CAPTURE
ETRANS_AuthCode	x_Auth_Code	Authorization code from previous transaction This may be used for credits or voids
ETRANS_TransactionID	x_Trans_ID	Transaction ID
ETRANS_InvoiceNum	x_Invoice_Num	Invoice number
ETRANS_PONum	x_PO_Num	Purchase order number
ETRANS_Description	x_Description	Transaction description
ETRANS_CustomerID	x_Cust_ID	Customer ID
ETRANS_Company	x_Company	Customer company
ETRANS_FirstName	x_First_Name	Customer first name
ETRANS_LastName	x_Last_Name	Customer last name
ETRANS_StreetAddress	x_Address	Customer billing address
ETRANS_City	x_City	Customer billing city
ETRANS_State	x_State	Customer billing state
ETRANS_Zip	x_Zip	Customer billing zip code
ETRANS_Country	x_Country	Customer billing country
ETRANS_Phone	x_Phone	Customer phone number

<b>Constant</b>	<b>Field Name</b>	<b>Description</b>
ETRANS_Fax	x_Fax	Customer FAX number
ETRANS_ShipToCompany	x_Ship_To_Company	Shipping address company
ETRANS_ShipToFirstName	x_Ship_To_First_Name	Shipping first name
ETRANS_ShipToLastName	x_Ship_To_Last_Name	Shipping last name
ETRANS_ShipToStreetAddress	x_Ship_To_Addresses	Shipping street address
ETRANS_ShipToCity	x_Ship_To_City	Shipping address city
ETRANS_ShipToState	x_Ship_To_State	Shipping address state
ETRANS_ShipToZip	x_Ship_To_Zip	Shipping address zip code
ETRANS_ShipToCountry	x_Ship_To_Country	Shipping address country

# PlanetPayment Exchange Rate API

All of these fields are required when the transaction is created.

This transaction type is created by using the method ***ETRANS\_PP\_Create\_Exchange***.

Constant	Field Name	Description
ETRANS_Login	x_Login	AuthorizeNet login name
ETRANS_Password	x_Password	AuthorizeNet password
ETRANS_Method	x_AMC_Response_Method	Desired format of the response Must be one of the following: HTML XML DELIM
ETRANS_FromCurrency	x_AMC_From_Currency_Code	ISO 4217 currency code
ETRANS_ToCurrency	x_AMC_To_Currency_Code	ISO 4217 currency code

## LinkPoint Credit Card (Direct)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_LP\_Create\_Direct***.

Constant	Field Name	Description
ETTRANS_TransactionType	2000	Set when transaction is created Cannot be modified
ETTRANS_Method	mode	Set when transaction is created Cannot be modified
ETTRANS_Login	storename	LinkPoint store name
ETTRANS_URL	referurl	Order form URL as specified in LinkPoint account setup
ETTRANS_CardNum	cardnumber	Credit card number Must be 16 or less numeric characters
ETTRANS_CardType	cctype	Credit card type Single character, must be one of the following: A (American Express) D (Discover) M (MasterCard) V (VISA)
ETTRANS_ExpirationMonth	expmonth	Credit card expiration month
ETTRANS_ExpirationYear	expyear	4-digit credit card expiration year
ETTRANS_Amount	chargetotal	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "####.00" Max amount of 30,000
ETTRANS_Subtotal	subtotal	Subtotal, before tax and shipping Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "####.00"
ETTRANS_Tax	tax	Sales tax amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "####.00"

Constant	Field Name	Description
ETRANS_Freight	shipping	Shipping charge amount Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETRANS_InvoiceNum	oid	Invoice number Cannot exceed 30 characters
ETRANS_FullName	bname	Customer billing full name Cannot exceed 30 characters
ETRANS_StreetAddress	baddr1	Customer billing address Cannot exceed 30 characters
ETRANS_City	bcity	Customer billing city Cannot exceed 30 characters
ETRANS_State	bstate	Customer billing state 2-character state abbreviation
ETRANS_Province	bstate2	Customer billing province Used if country is not US
ETRANS_Zip	bzip	Customer billing zip code Cannot exceed 10 characters
ETRANS_Country	bcountry	Customer billing country 2-character country abbreviation Required if country is not US
ETRANS_Phone	phone	Customer phone number Cannot exceed 20 characters
ETRANS_Fax	fax	Customer FAX number Cannot exceed 20 characters
ETRANS_ShipToFullName	sname	Shipping full name Cannot exceed 30 characters
ETRANS_ShipToStreetAddress	saddr1	Shipping street address Cannot exceed 30 characters
ETRANS_ShipToCity	scity	Shipping address city Cannot exceed 30 characters
ETRANS_ShipToState	sstate	Shipping address state 2-character state abbreviation
ETRANS_ShipToProvince	sstate2	Shipping address province Used if country is not US
ETRANS_ShipToZip	szip	Shipping address zip code Cannot exceed 10 characters
ETRANS_ShipToCountry	scountry	Shipping address country 2-character country abbreviation Required if country is not US

## E-xact Credit Card (Relay)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETRANS\_EX\_Create\_Relay***.

Constant	Field Name	Description
ETRANS_Password	post_Password	Set when transaction is created Cannot be modified This does not contain the actual password, but is a secure (MD5) hash of the password and other required fields
ETRANS_DateTime	post_DateTime	Set when transaction is created Cannot be modified Current timestamp in ISO 8601 format
ETRANS_Method	post_Method	Set when transaction is created Cannot be modified
ETRANS_Login	post_ExactID	E-xact login name
ETRANS_URL	post_Return_URL	URL for relay response
ETRANS_TransactionType	post_TransactionType	E-xact transaction type Must be one of the following: Purchase Refund PreAuthorizedPurchase PreAuthorizedPurchaseCompletion
ETRANS_FullName	post_CardHoldersName	Cardholder's full name
ETRANS_CardNum	x_Card_Num	Credit card number Must be 16 or less numeric characters
ETRANS_ExpirationMonth	post_Expiry_Month	Credit card expiration month (MM)
ETRANS_ExpirationYear	post_Expiry_Year	4-digit credit card expiration year
ETRANS_Amount	post_DollarAmount	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "####.#0" Max value of 99999.99
ETRANS_DelimiterCharacter	post_Delimiter	Delimiter character for transaction results Can be any character except comma (,)

<b>Constant</b>	<b>Field Name</b>	<b>Description</b>
ETTRANS_AuthCode	post_Authorization_Num	Authorization code from previous transaction This may be used for credits or voids Cannot exceed 8 characters
ETTRANS_StreetAddress	post_VerificationStreet1	Customer street address Used for address verification
ETTRANS_TransactionID	post_SequenceNo	Transaction ID
ETTRANS_InvoiceNum	post_Reference_No	Invoice number
ETTRANS_PONum	post_UDD4	Purchase order number
ETTRANS_Description	post_UDD3	Transaction description
ETTRANS_CustomerID	post_UDD1	Customer ID
ETTRANS_EmailAddressCustomer	post_UDD2	Customer email address
ETTRANS_Company	post_UDD5	Customer company

## WorldPay Credit Card (Relay)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_WP\_Create\_Relay***.

Constant	Field Name	Description
ETTRANS_TransactionType	op-parsePayment-0	Set when transaction is created Cannot be modified
ETTRANS_Login	instId	WorldPay Installation ID
ETTRANS_TransactionID	cartId	Cart ID (Basket ID or Invoice ID)
ETTRANS_FromCurrency	currency	Transaction Currency
ETTRANS_Amount	amount	Amount to be charged of specified currency Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETTRANS_CardNum	cardNo	Credit card number Must be 16 or less numeric characters
ETTRANS_ExpirationMonth	cardExpMonth	Credit card expiration month (MM)
ETTRANS_ExpirationYear	cardExpYear	4-digit credit card expiration year
ETTRANS_FullName	cardName	Cardholder's full name Cannot exceed 40 characters
ETTRANS_ShipToFullName	name	Customer name Set when transaction is created to same value as ETTRANS_FullName; it can be changed later if necessary. Cannot exceed 40 characters
ETTRANS_StreetAddress	address	Cardholder's street address Used for address verification
ETTRANS_Country	country	Cardholder's country Used for address verification 2-character ISO country code
ETTRANS_EmailAddressCustomer	email	Email address to send transaction receipt Cannot exceed 80 characters
ETTRANS_Zip	postCode	Cardholder's Zip Code Cannot exceed 12 characters

Constant	Field Name	Description
ETTRANS_CardSecurityCode	cardCVV	Credit card security code Can be 3 or 4 numeric digits
ETTRANS_AccountID	acctId	WorldPay account id Used when more than one account can handle transaction
ETTRANS_Method	authMode	Authorization mode Allowed values: A E
ETTRANS_Phone	tel	Customer phone number Cannot exceed 30 characters
ETTRANS_Fax	fax	Customer fax number Cannot exceed 30 characters
ETTRANS_Description	desc	Transaction description
ETTRANS_TestRequest	testMode	Specifies type of testing to be performed

# WorldPay Exchange Rate API

All of these fields are required when the transaction is created.

This transaction type is created by using the method ***ETRANS\_WP\_Create\_Exchange***.

Constant	Field Name	Description
ETRANS_Login	instId	WorldPay installation ID
ETRANS_Password	infoPW	WorldPay info password
ETRANS_Method	op	Operation to be performed Must be one of: rates rates-latest rates-today rates-tomorrow

## IonGate Credit Card (Direct)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_IG\_Create\_Direct***.

Constant	Field Name	Description
ETTRANS_URL	RECEIPTURL	Set to "DISPLAY" when transaction is created Cannot be modified
ETTRANS_Login	LOGIN	IonGate login name
ETTRANS_Amount	AMOUNT	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETTRANS_CardNum	CARDNUM	Credit card number
ETTRANS_CardType	CARDTYPE	Credit card type Allowed Values: VISA MASTERCARD AMEX DISCOVER DINERS JCB
ETTRANS_ExpirationDate	EXPIRES	Credit card expiration date Must be in mmyy format
ETTRANS_FullName	CARDNAME	Cardholder's full name Cannot exceed 40 characters
ETTRANS_StreetAddress	ADDRESS	Cardholder's billing street address Cannot exceed 90 characters
ETTRANS_City	CITY	Cardholder's billing city Cannot exceed 40 characters
ETTRANS_State	STATE	Cardholder's billing State Two-character state abbreviation
ETTRANS_Zip	ZIP	Cardholder's billing zip code Cannot exceed 10 characters
ETTRANS_Country	COUNTRY	Cardholder's billing country
ETTRANS_EmailAddressCustomer	EMAIL	Customer email address

Constant	Field Name	Description
ETRANS_EmailMerchant	MERCHEMAIL	Flag to suppress email to merchant Allowed Values: NO
ETRANS_Phone	PHONE	Customer phone number Cannot exceed 25 characters
ETRANS_CustomerID	CUSTNO	Customer ID Cannot exceed 20 characters
ETRANS_InvoiceNum	INVOICENO	Invoice Number Cannot exceed 20 characters
ETRANS_Description	DESCRIPTION	Transaction Description Cannot exceed 100 characters
ETRANS_AccountID	USER05	Customer Account ID
ETRANS_CardSecurityCode	CVV2NUM	Credit Card Security Code
ETRANS_Company	USER01	Customer company
ETRANS_PONum	USER02	Purchase order number
ETRANS_TransactionID	USER03	Transaction ID
ETRANS_TransactionType	USER04	Transaction Type

## IonGate Credit Card (Relay)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_IG\_Create\_Relay***.

Constant	Field Name	Description
ETTRANS_RelayResponse	RETURNALLER- RORS	Set to "YES" when transaction is created Cannot be changed
ETTRANS_URL	RECEIPTURL	URL to receive relay response
ETTRANS_Login	LOGIN	IonGate login name
ETTRANS_Amount	AMOUNT	Amount to be charged Can only contain numbers and decimal (.) If it contains a decimal, must be formatted "#####.00"
ETTRANS_CardNum	CARDNUM	Credit card number
ETTRANS_CardType	CARDTYPE	Credit card type Allowed Values: VISA MASTERCARD AMEX DISCOVER DINERS JCB
ETTRANS_ExpirationDate	EXPIRES	Credit card expiration date Must be in mmyy format
ETTRANS_FullName	CARDNAME	Cardholder's full name Cannot exceed 40 characters
ETTRANS_StreetAddress	ADDRESS	Cardholder's billing street address Cannot exceed 90 characters
ETTRANS_City	CITY	Cardholder's billing city Cannot exceed 40 characters
ETTRANS_State	STATE	Cardholder's billing State Two-character state abbreviation
ETTRANS_Zip	ZIP	Cardholder's billing zip code Cannot exceed 10 characters
ETTRANS_Country	COUNTRY	Cardholder's billing country

Constant	Field Name	Description
ETTRANS_EmailAddressCustomer	EMAIL	Customer email address
ETTRANS_EmailMerchant	MERCHEMAIL	Flag to suppress email to merchant Allowed Values: NO
ETTRANS_Phone	PHONE	Customer phone number Cannot exceed 25 characters
ETTRANS_CustomerID	CUSTNO	Customer ID Cannot exceed 20 characters
ETTRANS_InvoiceNum	INVOICENO	Invoice Number Cannot exceed 20 characters
ETTRANS_Description	DESCRIPTION	Transaction Description Cannot exceed 100 characters
ETTRANS_AccountID	USER05	Customer Account ID
ETTRANS_CardSecurityCode	CVV2NUM	Credit Card Security Code
ETTRANS_Company	USER01	Customer company
ETTRANS_PONum	USER02	Purchase order number
ETTRANS_TransactionID	USER03	Transaction ID
ETTRANS_TransactionType	USER04	Transaction Type

## FirePay Credit Card (Direct)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_FP\_Create\_Direct***.

Constant	Field Name	Description
ETTRANS_Method	operation	Set to "P" when transaction is created Cannot be modified
ETTRANS_Version	clientVersion	Set to "1.03" when transaction is created Cannot be modified
ETTRANS_AccountID	account	FirePay Account ID
ETTRANS_Login	merchantID	FirePay transaction processing id
ETTRANS_Password	merchantPwd	FirePay transaction processing password
ETTRANS_TransactionID	merchantTxn	Unique transaction ID for this transaction
ETTRANS_Amount	amount	Amount to be charged Can only contain numbers Integer number of cents for this transaction
ETTRANS_CardNum	cardNumber	Credit card number
ETTRANS_CardType	cardType	Credit card type Allowed Values: VI MC AM DI FP
ETTRANS_ExpirationDate	cardExp	Credit card expiration date Must be in mm/yy format
ETTRANS_FullName	custName1	Cardholder's full name Cannot exceed 255 characters
ETTRANS_StreetAddress	streetAddr	Cardholder's billing street address Cannot exceed 255 characters
ETTRANS_City	city	Cardholder's billing city Cannot exceed 255 characters
ETTRANS_State	province	Cardholder's billing state/province Two-character state/province abbreviation

Constant	Field Name	Description
ETRANS_Zip	zip	Cardholder's billing zip code Cannot exceed 10 characters
ETRANS_Country	country	Cardholder's billing country Two-character country abbreviation
ETRANS_EmailAddressCustomer	email	Customer email address
ETRANS_Phone	phone	Customer phone number Cannot exceed 255 characters
ETRANS_Description	merchantData	Transaction Description Cannot exceed 255 characters

## SurePay Credit Card (Direct)

All required fields are set when the transaction is created. Some values may be hard-coded, and other values will be required as parameters for the method that creates the transaction. All other fields are optional.

This transaction type is created by using the method ***ETTRANS\_SP\_Create\_Direct***.

Constant	Field Name	Description
ETTRANS_Login	merchant	SurePay merchant ID
ETTRANS_Password	password	SurePay transaction processing password
ETTRANS_InvoiceNum	ordernumber	Order number for this transaction Cannot exceed 10 numeric digits
ETTRANS_Amount	unitprice	Order subtotal, before shipping or taxes Amount to be charged Can only contain numbers ETTRANS_Freight and ETTRANS_Tax will be added to this amount to calculate the full order total
ETTRANS_CardNum	number	Credit card number
ETTRANS_ExpirationDate	expiration	Credit card expiration date Must be in mm/yy format
ETTRANS_FullName	fullname	Cardholder's full name Cannot exceed 32 characters
ETTRANS_StreetAddress	address1	Cardholder's billing street address Cannot exceed 100 characters
ETTRANS_City	city	Cardholder's billing city Cannot exceed 30 characters
ETTRANS_State	state	Cardholder's billing state/province Two-character state/province abbreviation required if country is US
ETTRANS_Zip	zip	Cardholder's billing zip code Cannot exceed 20 characters
ETTRANS_Country	country	Cardholder's billing country Two-character country abbreviation
ETTRANS_ShipToFullName	fullname	Recipient's full name Cannot exceed 32 characters
ETTRANS_ShipToStreetAddress	address1	Recipient's street address Cannot exceed 100 characters

Constant	Field Name	Description
ETRANS_ShipToCity	city	Recipient's city Cannot exceed 30 characters
ETRANS_ShipToState	state	Recipient's state/province Two-character state/province abbreviation required if country is US
ETRANS_ShipToZip	zip	Recipient's zip code Cannot exceed 20 characters
ETRANS_ShipToCountry	country	Recipient's country Two-character country abbreviation
ETRANS_TransactionType	sku	SKU for line item, used to identify orders in SurePay Manager Reports Defaults to "sku_01" Cannot exceed 20 characters
ETRANS_Description	description	Description for order's single line item Defaults to "desc_01" Cannot exceed 200 characters
ETRANS_PONum	ponumber	PO number for order Cannot exceed 20 characters
ETRANS_EmailAddressCustomer	email	Customer's email address Cannot exceed 255 characters
ETRANS_CustomerIP	ipaddress	Customer's IP address
ETRANS_Phone	phone	Customer's phone number
ETRANS_Fax	fax	Customer's Fax number
ETRANS_Freight	shippingcost	Total shipping charge for order
ETRANS_Tax	taxamount	Total sales tax charge for order
ETRANS_CardSecurityCode	cvv2	Credit card security code (CVV code)

## Modules

All of the code within the eTrans component is organized into modules. Each module is designated by a three (3) to five (5) character module prefix. All of the module prefixes are used within the name of every object within the module (methods names, variable names, semaphore names, etc.). This allows for the easy identification of any object within the eTrans component.

Each module contains a set of methods which can be used to easily implement a web server once the eTrans component is installed. Method names all begin with the module prefix followed by an underscore ("\_") characters. The remainder of the method name then describes the function of the method.

## eTrans Module

The eTrans Module handles the server aspects of eTrans. Use the methods in the HTTPsd module to define or clear eTrans Services, as well as pause and resume the eTrans server.

---

### ENV\_Get\_ETRANS\_HardName\_Long

*ENV\_Get\_ETRANS\_HardName\_Long* => *Long Hard Name*

*ENV\_Get\_ETRANS\_HardName\_Long*  
=> *Long Hard Name* : Text

	Parameter	Type	Description
	<i>Long Hard Name</i>	Text	Full, hard coded name of the eTrans component including versioning information

The method **ENV\_Get\_ETRANS\_HardName\_Long** returns the full, hard coded name of the eTrans component, including versioning information.

*Long Hard Name* is the full, hard coded name of the eTrans component. As of this release, this will always return the value "Affix\_eTrans\_v1.1.0".

---

### ENV\_Get\_ETRANS\_HardName\_Short

*ENV\_Get\_ETRANS\_HardName\_Short* => *Short Hard Name*

*ENV\_Get\_ETRANS\_HardName\_Short*  
=> *Short Hard Name* : Text

	Parameter	Type	Description
	<i>Short Hard Name</i>	Text	Short hard coded name of the eTrans component.

The method **ENV\_Get\_ETRANS\_HardName\_Short** returns a shortend, hard coded name of the eTrans component.

*Short Hard Name* is the shortened, hard coded name of the eTrans component. As of this release, this will always return the value "Affix\_eTrans".

---

## **ETRANS\_ANEC\_Create\_Direct**

*ETRANS\_ANEC\_Create\_Direct* ( *AuthorizeNet Login* ; *Transaction Amount* ; *Bank Account Number* ; *Bank ABA Code* ) => *Referenced Transaction Record*

*ETRANS\_ANEC\_Create\_Direct*

```
(
    -> AuthorizeNet Login : Text
    -> Transaction Amount : Real
    -> Bank Account Number : String[80]
    -> Bank ABA Code : String[80]
)
=> Referenced Transaction Record : Pointer
```

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>AuthorizeNet Login</i>	Text	AuthorizeNet login name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Bank Account Number</i>	String[80]	Bank Account Number that eCheck is to be drawn from
	<i>Bank ABA Code</i>	String[80]	Bank ABA Code that eCheck is to be drawn from
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_ANEC\_Create\_Direct*** will create a transaction record for an AuthorizeNet eCheck direct response transaction. AuthorizeNet eCheck transactions aren't charged against a credit card, they are drawn directly from a bank account instead.

*AuthorizeNet Login* is the login for your AuthorizeNet store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Bank Account Number* is the Account Number of the bank account this eCheck transaction is to be drawn from. *Bank Account Number* can only contain numbers.

*Bank ABA Code* is the ABA code identifying the bank which this eCheck transaction is to be drawn from. *Bank ABA Code* can only contain numbers.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETTRANS\_Transact** method to actually process the transaction.

## **ETTRANS\_ANEC\_Create\_Relay**

*ETTRANS\_ANEC\_Create\_Relay* ( *AuthorizeNet Login* ; *Transaction Amount* ; *Bank Account Number* ; *Bank ABA Code* ; *Response Full URL* ) => *Referenced Transaction Record*

```

ETTRANS_ANEC_Create_Relay
(
    -> AuthorizeNet Login : Text
    -> Transaction Amount : Real
    -> Bank Account Number : String[80]
    -> Bank ABA Code : String[80]
    -> Response Full URL : Text
)
=> Referenced Transaction Record : Pointer
    
```

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>AuthorizeNet Login</i>	Text	AuthorizeNet login name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Bank Account Number</i>	String[80]	Bank Account Number that eCheck is to be drawn from
	<i>Bank ABA Code</i>	String[80]	Bank ABA Code that eCheck is to be drawn from
	<i>Response Full URL</i>	Text	Full URL for AuthorizeNet to send relay response to

	Parameter	Type	Description
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_ANEC\_Create\_Relay*** will create a transaction record for an AuthorizeNet eCheck relay response transaction. AuthorizeNet eCheck transactions aren't charged against a credit card, they are drawn directly from a bank account instead.

*AuthorizeNet Login* is the login for your AuthorizeNet store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Bank Account Number* is the Account Number of the bank account this eCheck transaction is to be drawn from. *Bank Account Number* can only contain numbers.

*Bank ABA Code* is the ABA code identifying the bank which this eCheck transaction is to be drawn from. *Bank ABA Code* can only contain numbers.

*Response Full URL* is the full URL for AuthorizeNet to send the response to. When the transaction is processed, AuthorizeNet will send the transaction result to the *Response Full URL*. *Response Full URL* needs to be capable of accepting an HTTP POST, and return a response to the AuthorizeNet server.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the ***ETRANS\_Transact*** method to actually process the transaction.

## **ETRANS\_AN\_Create\_Direct**

***ETRANS\_AN\_Create\_Direct*** ( *AuthorizeNet Login ; Transaction Amount ; Card Number ; Expiration Date* ) => *Referenced Transaction Record*

### *ETTRANS\_AN\_Create\_Direct*

```
(  
    -> AuthorizeNet Login : Text  
    -> Transaction Amount : Real  
    -> Card Number : String[80]  
    -> Expiration Date : String[10]  
)  
=> Referenced Transaction Record : Pointer
```

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>AuthorizeNet Login</i>	Text	AuthorizeNet login name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	String[80]	Credit card number to be charged
	<i>Expiration Date</i>	String[10]	Expiration Date of the credit card
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETTRANS\_AN\_Create\_Direct*** will create a transaction record for an AuthorizeNet credit card direct response transaction. This transaction will be authorized against a credit card.

*AuthorizeNet Login* is the login for your AuthorizeNet store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against. *Card Number* cannot contain any non-numeric characters.

*Expiration Date* is the expiration date of the credit card to be charged. *Expiration Date* needs to be in the format "mmyy", "mm/yy", or "mm/yyyy".

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the ***ETTRANS\_Transact*** method to actually process the transaction.

---

## **ETTRANS\_AN\_Create\_Relay**

*ETTRANS\_AN\_Create\_Relay* ( *AuthorizeNet Login* ; *Transaction Amount* ; *Card Number* ; *Expiration Date* ; *Response Full URL* ) => *Referenced Transaction Record*

*ETTRANS\_AN\_Create\_Relay*

(  
    -> *AuthorizeNet Login* : Text  
    -> *Transaction Amount* : Real  
    -> *Card Number* : String[80]  
    -> *Expiration Date* : String[10]  
    -> *Response Full URL* : Text  
)  
=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>AuthorizeNet Login</i>	Text	AuthorizeNet login name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	String[80]	Credit card number to be charged
	<i>Expiration Date</i>	String[10]	Expiration Date of the credit card
	<i>Response Full URL</i>	Text	Full URL for AuthorizeNet to send relay response to
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETTRANS\_AN\_Create\_Relay*** will create a transaction record for an AuthorizeNet credit card relay response transaction. This transaction will be authorized against a credit card.

*AuthorizeNet Login* is the login for your AuthorizeNet store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against. *Card Number* cannot contain any non-numeric characters.

*Expiration Date* is the expiration date of the credit card to be charged. *Expiration Date* needs to be in the format "mmyy", "mm/yy", or "mm/yyyy".

*Response Full URL* is the full URL for AuthorizeNet to send the response to. When the transaction is processed, AuthorizeNet will send the transaction result to *Response Full URL*. *Response Full URL* needs to be capable of accepting an HTTP POST, and return a response to the AuthorizeNet server.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETRANS\_Transact** method to actually process the transaction.

---

## ETRANS\_ERROR

*ETRANS\_ERROR* ( *eTrans Error Number*; *Special Error Text*; *Calling Method Name* )

*ETRANS\_ERROR*

```
(  
    -> eTrans Error Number : Longint  
    -> Special Error Text : Text  
    -> Calling Method Name : Text  
)
```

	Parameter	Type	Description
	<i>eTrans Error Number</i>	Longint	Internal eTrans error number
	<i>Special Error Text</i>	Text	Special text to describe the exact error instance
	<i>Calling Method Name</i>	Text	Name of the method that the error condition occurred in

The method **ETRANS\_ERROR** acts as a callback method from within the eTrans component for errors that may occur. Any time an error condition is detected within the eTrans module, a call to the method **ETRANS\_ERROR** is made.

The internal *eTrans Error Number* is passed to this method as the first parameter. The *Special Error Text* parameter will contain any relevant error text which is specific to the error which occurred. It is not uncommon for the *Special Error Text* value to be empty. The *Calling Method Name* will always contain the name of the eTrans method which call the **ETRANS\_ERROR** method.

The **ETRANS\_ERROR** method has been implemented as a source for a consistent interface and/or error tracking mechanism to be available while using the eTrans component. This method can be modified to suit the needs of the database in which the eTrans component has been installed.

### ETRANS\_EX\_Create\_Relay

*ETRANS\_EX\_Create\_Relay* ( *E-xact Login ; E-xact Password ; Transaction Type ; Transaction Amount ; Card Number ; Expiration Month ; Expiration Year ; Cardholder Name ; Response Full URL* ) => *Referenced Transaction Record*

#### *ETRANS\_EX\_Create\_Relay*

```
(
    -> E-xact Login : Text
    -> E-xact Password : Text
    -> Transaction Type : Text
    -> Transaction Amount : Real
    -> Card Number : Text
    -> Expiration Month : Longint
    -> Expiration Year : Longint
    -> Cardholder Name : Text
    -> Response Full URL : Text
)
=> Referenced Transaction Record : Pointer
```

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>E-xact Login</i>	Text	E-xact login name
	<i>E-xact Password</i>	Text	E-xact password
	<i>Transaction Type</i>	Text	Transaction Type
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	Text	Credit card number to be charged

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>Expiration Month</i>	Longint	Month of credit card's expiration date
	<i>Expiration Year</i>	Longint	Year of credit card's expiration date
	<i>Cardholder Name</i>	Text	Credit card holder's full name
	<i>Response Full URL</i>	Text	Full URL for AuthorizeNet to send relay response to
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_EX\_Create\_Relay*** will create a transaction record for an E-xact credit card relay response transaction. This transaction will be authorized against a credit card.

*E-xact Login* is the login for your E-xact account.

*E-xact Password* is the password for your E-xact account. The password is not built into the transaction, but is only used to create a security hash to be verified on the server.

*Transaction Type* is the E-xact transaction type. This must be one of the following:

Purchase  
 Refund  
 PreAuthorizedPurchase  
 PreAuthorizedPurchaseCompletion

*Transaction Amount* is the amount to charge for this transaction. This must be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against. *Card Number* cannot contain any non-numeric characters.

*Expiration Month* is the month of the expiration date of the credit card to be charged.

*Expiration Year* is the year of the expiration date of the credit card to be charged.

*Response Full URL* is the full URL for AuthorizeNet to send the response to. When the transaction is processed, Authori-

zeNet will send the transaction result to *Response Full URL*. *Response Full URL* needs to be capable of accepting an HTTP POST, and return a response to the E-xact server.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETRANS\_Transact** method to actually process the transaction.

### ETRANS\_FP\_Create\_Direct

**ETRANS\_FP\_Create\_Direct** ( *FirePay Account Number ; FirePay Transaction Processing ID ; FirePay Transaction Processing Password ; Transaction ID ; Transaction Amount ; Card Number ; Card Type ; Expiration Date* ) => *Referenced Transaction Record*

**ETRANS\_FP\_Create\_Direct**

```
(
    -> FirePay Account Number : Text
    -> FirePay Transaction Processing ID : Text
    -> FirePay Transaction Processing Password : Text
    -> Transaction ID : Text
    -> Transaction Amount : Longint
    -> Card Number : Text
    -> Card Type : Text
    -> Expiration Date : Text
)
=> Referenced Transaction Record : Pointer
```

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>FirePay Account Number</i>	Text	FirePay account number
	<i>FirePay Transaction Processing ID</i>	Text	FirePay account transaction processing id
	<i>FirePay Transaction Processing Password</i>	Text	FirePay account transaction processing Password
	<i>Transaction ID</i>	Text	Unique identifier for this transaction
	<i>Transaction Amount</i>	Longint	Amount of transaction in cents
	<i>Card Number</i>	Text	Credit card number to be charged

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>Card Type</i>	Text	Type of credit card
	<i>Expiration Date</i>	Text	Credit card's expiration date (mm/yy)
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_FP\_Create\_Direct*** will create a transaction record for a FirePay credit card direct response transaction. This transaction will be authorized against a credit card. FirePay also supports their own internal user account system. These accounts are specified with a credit card type of "FP" (FirePay).

*FirePay Account Number* is your FirePay account number.

*FirePay Transaction Processing ID* is your FirePay account transaction processing id.

*FirePay Transaction Processing Password* is your FirePay account transaction processing password.

*Transaction ID* is a unique identifier for this transaction.

*Transaction Amount* is the number of cents to charge for this transaction. This needs to be a longint number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against.

*Card Type* is the type of the credit card to be charged. *Card Type* must be one of the following values: "VI", "MC", "AM", "DI", "FP".

*Expiration Date* is the expiration date of the credit card to be charged. *Expiration Date* needs to be in mm/yy format.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the ***ETRANS\_Transact*** method to actually process the transaction.

---

## **ETRANS\_IG\_Create\_Direct**

*ETRANS\_IG\_Create\_Direct* ( *IonGate Login ID* ; *Transaction Amount* ; *Card Number* ; *Card Type* ; *Expiration Date*; *Cardholder Name* ; *Billing Street Address* ; *Billing City* ; *Billing State* ; *Billing Zip Code* ) => *Referenced Transaction Record*

### *ETRANS\_IG\_Create\_Direct*

(  
    -> *IonGate Login ID* : Text  
    -> *Transaction Amount* : Real  
    -> *Card Number* : Text  
    -> *Card Type* : Text  
    -> *Expiration Date* : Text  
    -> *Cardholder Name* : Text  
    -> *Billing Street Address* : Text  
    -> *Billing City* : Text  
    -> *Billing State* : Text  
    -> *Billing Zip Code* : Text  
)  
=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>IonGate Login ID</i>	Text	IonGate Login ID
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	Text	Credit card number to be charged
	<i>Card Type</i>	Text	Type of credit card
	<i>Expiration Date</i>	Text	Credit card's expiration date (mmyy)
	<i>Cardholder Name</i>	Text	Cardholder's full name
	<i>Billing Street Address</i>	Text	Cardholder's billing street address
	<i>Billing City</i>	Text	Cardholder's billing city
	<i>Billing State</i>	Text	Cardholder's billing state abbreviation
	<i>Billing Zip Code</i>	Text	Cardholder's billing zip code
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_IG\_Create\_Direct*** will create a transaction record for an IonGate credit card direct response transaction. This transaction will be authorized against a credit card.

*IonGate Login ID* is your IonGate account login id.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against.

*Card Type* is the type of the credit card to be charged. This must be one of the following values: "VISA", "MASTERCARD", "AMEX", "DISCOVER", "DINERS", "JCB".

*Expiration Date* is the expiration date of the credit card to be charged. Expiration Date needs to be in mmyy format.

*Cardholder Name* is the credit card holder's full name.

*Billing Street Address* is the cardholder's billing street address. This is used for address verification.

*Billing City* is the cardholder's billing zip code.

*Billing State* is the cardholder's billing zip code.

*Billing Zip Code* is the cardholder's billing zip code. This is used for address verification.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETTRANS\_Transact** method to actually process the transaction.

---

## **ETTRANS\_IG\_Create\_Relay**

*ETTRANS\_IG\_Create\_Relay* ( *IonGate Login ID* ; *Transaction Amount* ; *Card Number* ; *Card Type* ; *Expiration Date*; *Cardholder Name* ; *Billing Street Address* ; *Billing City* ; *Billing State* ; *Billing Zip Code* ; *Relay Receipt URL* ) => *Referenced Transaction Record*

*ETTRANS\_IG\_Create\_Relay*

(

- > *IonGate Login ID* : Text
- > *Transaction Amount* : Real
- > *Card Number* : Text
- > *Card Type* : Text
- > *Expiration Date* : Text
- > *Cardholder Name* : Text
- > *Billing Street Address* : Text
- > *Billing City* : Text
- > *Billing State* : Text
- > *Billing Zip Code* : Text
- > *Relay Receipt URL* : Text

)

=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>IonGate Login ID</i>	Text	IonGate Login ID
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	Text	Credit card number to be charged
	<i>Card Type</i>	Text	Type of credit card
	<i>Expiration Date</i>	Text	Credit card's expiration date (mmyy)
	<i>Cardholder Name</i>	Text	Cardholder's full name
	<i>Billing Street Address</i>	Text	Cardholder's billing street address
	<i>Billing City</i>	Text	Cardholder's billing city
	<i>Billing State</i>	Text	Cardholder's billing state abbreviation
	<i>Billing Zip Code</i>	Text	Cardholder's billing zip code
	<i>Relay Receipt URL</i>	Text	Receipt URL for relay response
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_IG\_Create\_Direct*** will create a transaction record for an IonGate credit card relay response transaction. This transaction will be authorized against a credit card.

*IonGate Login ID* is your IonGate account login id.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against.

*Card Type* is the type of the credit card to be charged. This must be one of the following values: "VISA", "MASTERCARD", "AMEX", "DISCOVER", "DINERS", "JCB".

*Expiration Date* is the expiration date of the credit card to be charged. Expiration Date needs to be in mmyy format.

*Cardholder Name* is the credit card holder's full name.

*Billing Street Address* is the cardholder's billing street address. This is used for address verification.

*Billing City* is the cardholder's billing zip code.

*Billing State* is the cardholder's billing zip code.

*Billing Zip Code* is the cardholder's billing zip code. This is used for address verification.

*Relay Receipt URL* is the receipt URL for the relay response. This should be a secure URL, and be capable of processing the result of this transaction.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETRANS\_Transact** method to actually process the transaction.

---

## ETRANS\_LP\_Create\_Direct

*ETRANS\_LP\_Create\_Direct* ( *LinkPoint Store Name* ; *Transaction Amount* ; *Card Number* ; *Expiration Month* ; *Expiration Year* ; *Card Type* ; *Order Form URL* ; *Billing Street Address* ; *Billing Zip Code* ) => *Referenced Transaction Record*

*ETRANS\_LP\_Create\_Direct*

(

- > *LinkPoint Store Name* : Text
- > *Transaction Amount* : Real
- > *Card Number* : Text
- > *Expiration Month* : Longint
- > *Expiration Year* : Longint

- > *Card Type* : Text
- > *Order Form URL* : Text
- > *Billing Street Address* : Text
- > *Billing Zip Code* : Text

)

=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>LinkPoint Store Name</i>	Text	LinkPoint Store Name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	Text	Credit card number to be charged
	<i>Expiration Month</i>	Longint	Month of credit card's expiration date
	<i>Expiration Year</i>	Text	Year of credit card's expiration date
	<i>Card Type</i>	Text	Type of credit card
	<i>Order Form URL</i>	Text	Order form URL specified in the LinkPoint account setup
	<i>Billing Street Address</i>	Text	Cardholder's billing street address
	<i>Billing Zip Code</i>	Text	Cardholder's billing zip code
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_LP\_Create\_Direct*** will create a transaction record for an LinkPoint credit card direct response transaction. This transaction will be authorized against a credit card.

*LinkPoint Store Name* is your LinkPoint store name.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against. Card Number cannot contain any non-numeric characters.

*Expiration Month* is the month of the expiration date of the credit card to be charged.

*Expiration Year* is the year of the expiration date of the credit card to be charged.

*Credit Card Type* is the type of the credit card to be charged. This must be one of the following values: "A", "D", "M", "V".

*Order Form URL* is the "Order Submission Form URL" as set in your LinkPoint account setup. This order form URL does not even need to exist, it just needs to match what has been entered in the LinkPoint account setup.

*Billing Street Address* is the cardholder's billing street address. This is used for address verification.

*Billing Zip Code* is the cardholder's billing zip code. This is used for address verification.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the ***ETTRANS\_Transact*** method to actually process the transaction.

---

## **ETTRANS\_PPEC\_Create\_Direct**

*ETTRANS\_PPEC\_Create\_Direct* ( *PlanetPayment Login* ; *Transaction Amount* ; *Bank Account Number* ; *Bank ABA Code* ) => *Referenced Transaction Record*

*ETTRANS\_PPEC\_Create\_Direct*

```
(  
    -> PlanetPayment Login : Text  
    -> Transaction Amount : Real  
    -> Bank Account Number : String[80]  
    -> Bank ABA Code : String[80]  
)  
=> Referenced Transaction Record : Pointer
```

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>PlanetPayment Login</i>	Text	PlanetPayment login name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Bank Account Number</i>	String[80]	Bank Account Number that eCheck is to be drawn from

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>Bank ABA Code</i>	String[80]	Bank ABA Code that eCheck is to be drawn from
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method **ETTRANS\_PPEC\_Create\_Direct** will create a transaction record for a PlanetPayment eCheck direct response transaction. PlanetPayment eCheck transactions aren't charged against a credit card, they are drawn directly from a bank account instead.

*PlanetPayment Login* is the login for your PlanetPayment store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Bank Account Number* is the Account Number of the bank account this eCheck transaction is to be drawn from. *Bank Account Number* can only contain numbers.

*Bank ABA Code* is the ABA code identifying the bank which this eCheck transaction is to be drawn from. *Bank ABA Code* can only contain numbers.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETTRANS\_Transact** method to actually process the transaction.

## **ETTRANS\_PPEC\_Create\_Relay**

**ETTRANS\_PPEC\_Create\_Relay** ( *PlanetPayment Login ; Transaction Amount ; Bank Account Number ; Bank ABA Code ; Response Full URL* ) => *Referenced Transaction Record*

**ETTRANS\_PPEC\_Create\_Relay**

(

-> *PlanetPayment Login* : Text

- > *Transaction Amount* : Real
- > *Bank Account Number* : String[80]
- > *Bank ABA Code* : String[80]
- > *Response Full URL* : Text

)

=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>PlanetPayment Login</i>	Text	PlanetPayment login name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Bank Account Number</i>	String[80]	Bank Account Number that eCheck is to be drawn from
	<i>Bank ABA Code</i>	String[80]	Bank ABA Code that eCheck is to be drawn from
	<i>Response Full URL</i>	Text	Full URL for AuthorizeNet to send relay response to
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_PPEC\_Create\_Relay*** will create a transaction record for a PlanetPayment eCheck relay response transaction. PlanetPayment eCheck transactions aren't charged against a credit card, they are drawn directly from a bank account instead.

*PlanetPayment Login* is the login for your PlanetPayment store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Bank Account Number* is the Account Number of the bank account this eCheck transaction is to be drawn from. *Bank Account Number* can only contain numbers.

*Bank ABA Code* is the ABA code identifying the bank which this eCheck transaction is to be drawn from. *Bank ABA Code* can only contain numbers.

*Response Full URL* is the full URL for AuthorizeNet to send the response to. When the transaction is processed, AuthorizeNet will send the transaction result to the *Response Full URL*. *Response Full URL* needs to be capable of accepting an HTTP POST, and return a response to the AuthorizeNet server.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETTRANS\_Transact** method to actually process the transaction.

## ETRANS\_PP\_Check\_Exchange\_Rates

*ETTRANS\_PP\_Check\_Exchange\_Rates* ( *PlanetPayment Login* ; *PlanetPayment Password* ; *From Currency* ; *To Currency* ; *From Currency Descriptions* ; *From Currency Codes* ; *From Currency Numeric Codes* ; *To Currency Descriptions* ; *To Currency Codes* ; *To Currency Numeric Codes* ; *Exchange Rates* ) => *Result Code*

*ETTRANS\_PP\_Check\_Exchange\_Rates*

```
(
    -> PlanetPayment Login : Text
    -> PlanetPayment Password : Text
    -> From Currency : Text
    -> To Currency : Text
    -> From Currency Descriptions : Pointer
    -> From Currency Codes : Pointer
    -> From Currency Numeric Codes : Pointer
    -> To Currency Descriptions : Pointer
    -> To Currency Codes : Pointer
    -> To Currency Numeric Codes : Pointer
    -> Exchange Rates : Pointer
)
```

=> *Result Code* : Longint

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>PlanetPayment Login</i>	Text	PlanetPayment login name
	<i>PlanetPayment Password</i>	Text	PlanetPayment password
	<i>From Currency</i>	Text	ISO 4217 "from" currency code
	<i>To Currency</i>	Text	ISO 4217 "to" currency code
	<i>From Currency Descriptions</i>	Pointer	Reference to text array to receive from currency descriptions NOTE: pass a NULL pointer to skip

Parameter	Type	Description
<i>From Currency Codes</i>	Pointer	Reference to text array to receive from currency ISO 4217 codes NOTE: pass a NULL pointer to skip
<i>From Currency Numeric Codes</i>	Pointer	Reference to longint array to receive from currency numeric codes NOTE: pass a NULL pointer to skip
<i>To Currency Descriptions</i>	Pointer	Reference to text array to receive to currency descriptions NOTE: pass a NULL pointer to skip
<i>To Currency Codes</i>	Pointer	Reference to text array to receive to currency ISO 4217 codes NOTE: pass a NULL pointer to skip
<i>To Currency Numeric Codes</i>	Pointer	Reference to longint array to receive to currency numeric codes NOTE: pass a NULL pointer to skip
<i>Exchange Rates</i>	Pointer	Reference to real array to receive exchange rates
<i>Result Code</i>	Longint	Transaction Result Code = 0 for successful request = 1 for invalid response method = 2 for invalid login = 3 for merchant account inactive = 4 for invalid from currency code = 5 for invalid to currency code = 6 for exchange rate unavailable < 0 for error in routine

The method ***ETRANS\_PP\_Check\_Exchange\_Rates*** is a convenience method which creates a transaction for PlanetPayment's exchange rate API and parses the response into arrays for the developer.

*PlanetPayment Login* is the login for your PlanetPayment store account.

*PlanetPayment Password* is the password for your PlanetPayment store account.

*From Currency* is an ISO 4217 currency code to convert from. Pass "\*\*\*\*" as this parameter to find the conversion from all supported currencies to a specific currency. **NOTE:** both *From Currency* and *To Currency* cannot be "\*\*\*\*".

*To Currency* is an ISO 4217 currency code to convert to. Pass "\*\*\*\*" as this parameter to find the conversion to all supported currencies from a specific currency. **NOTE:** both *From Currency* and *To Currency* cannot be "\*\*\*\*".

*From Currency Descriptions* is a reference to a text array to receive the full currency descriptions of the currencies to calculate from. Pass a NULL pointer to skip this parameter.

*From Currency Codes* is a reference to a text array to receive the ISO 4217 currency codes of the currencies to calculate from. Pass a NULL pointer to skip this parameter.

*From Currency Numeric Codes* is a reference to a longint array to receive the numeric country code of the currencies to calculate from. Pass a NULL pointer to skip this parameter.

*To Currency Descriptions* is a reference to a text array to receive the full currency descriptions of the currencies to calculate to. Pass a NULL pointer to skip this parameter.

*To Currency Codes* is a reference to a text array to receive the ISO 4217 currency codes of the currencies to calculate to. Pass a NULL pointer to skip this parameter.

*To Currency Numeric Codes* is a reference to a longint array to receive the numeric country code of the currencies to calculate to. Pass a NULL pointer to skip this parameter.

*Exchange Rates* is a reference to a real array to receive the currency conversion exchange rates.

*Result Code* is the result code of the exchange rate API transaction. *Result Code* is set to zero (0) if there was no error, greater than zero (>0) if there was an error in the request, and less than zero (<0) if there was an error in the routine.

---

## ETRANS\_PP\_Create\_Direct

*ETTRANS\_PP\_Create\_Direct* ( *PlanetPayment Login ; Transaction Amount ; Card Number ; Expiration Date* ) => *Referenced Transaction Record*

## ***ETTRANS\_PP\_Create\_Direct***

```
(  
    -> PlanetPayment Login : Text  
    -> Transaction Amount : Real  
    -> Card Number : String[80]  
    -> Expiration Date : String[10]  
)  
  
=> Referenced Transaction Record : Pointer
```

<b>Parameter</b>	<b>Type</b>	<b>Description</b>
<i>PlanetPayment Login</i>	Text	PlanetPayment login name
<i>Transaction Amount</i>	Real	Amount of transaction
<i>Card Number</i>	String[80]	Credit card number to be charged
<i>Expiration Date</i>	String[10]	Expiration Date of the credit card
<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETTRANS\_PP\_Create\_Direct*** will create a transaction record for a PlanetPayment credit card direct response transaction. This transaction will be authorized against a credit card.

*PlanetPayment Login* is the login for your PlanetPayment store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against. *Card Number* cannot contain any non-numeric characters.

*Expiration Date* is the expiration date of the credit card to be charged. *Expiration Date* needs to be in the format "mmyy", "mm/yy", or "mm/yyyy".

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the ***ETTRANS\_Transact*** method to actually process the transaction.

---

## ETRANS\_PP\_Create\_Exchange

*ETTRANS\_PP\_Create\_Exchange* ( *PlanetPayment Login* ; *PlanetPayment Password* ;  
*From Currency* ; *To Currency* ; *Response Type* ) => *Referenced Transaction  
Record*

*ETTRANS\_PP\_Create\_Exchange*  
(  
    -> *PlanetPayment Login* : Text  
    -> *PlanetPayment Password* : Text  
    -> *From Currency* : Text  
    -> *To Currency* : Text  
    -> *Response Type* : Text  
)  
=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>PlanetPayment Login</i>	Text	PlanetPayment login name
	<i>PlanetPayment Password</i>	Text	PlanetPayment password
	<i>From Currency</i>	Text	ISO 4217 "from" currency code
	<i>To Currency</i>	Text	ISO 4217 "to" currency code
	<i>Response Type</i>	Text	Response type of transaction
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETTRANS\_PP\_Create\_Exchange*** will create a transaction record for a PlanetPayment exchange rate API transaction. This transaction will be interface with PlanetPayment's exchange rate API, and return the results in the format specified.

*PlanetPayment Login* is the login for your PlanetPayment store account.

*PlanetPayment Password* is the password for your PlanetPayment store account.

*From Currency* is an ISO 4217 currency code to convert from. Pass "\*\*\*\*" as this parameter to find the conversion from all supported currencies to a specific currency. **NOTE:** both *From Currency* and *To Currency* cannot be "\*\*\*\*".

*To Currency* is an ISO 4217 currency code to convert to. Pass "\*\*\*\*" as this parameter to find the conversion to all supported currencies from a specific currency. **NOTE:** both *From Currency* and *To Currency* cannot be "\*\*\*\*".

*Response Type* is the desired format of the response. Allowed values are "DELIM", "XML", or "HTML". Setting a value of "DELIM" will return a response with fields delimited by the pipe symbol ("|") and rows delimited with <carriage return>+<line feed> (CRLF).

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* with the **ETRANS\_Transact** method to actually process the transaction.

## ETRANS\_PP\_Create\_Relay

*ETRANS\_PP\_Create\_Relay* ( *PlanetPayment Login* ; *Transaction Amount* ; *Card Number* ; *Expiration Date* ; *Response Full URL* ) => *Referenced Transaction Record*

*ETRANS\_PP\_Create\_Relay*

```
(
  -> PlanetPayment Login : Text
  -> Transaction Amount : Real
  -> Card Number : String[80]
  -> Expiration Date : String[10]
  -> Response Full URL : Text
)
```

=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>PlanetPayment Login</i>	Text	PlanetPayment login name
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	String[80]	Credit card number to be charged
	<i>Expiration Date</i>	String[10]	Expiration Date of the credit card
	<i>Response Full URL</i>	Text	Full URL for AuthorizeNet to send relay response to

	Parameter	Type	Description
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method **ETRANS\_PP\_Create\_Relay** will create a transaction record for a PlanetPayment credit card relay response transaction. This transaction will be authorized against a credit card.

*PlanetPayment Login* is the login for your PlanetPayment store account.

*Transaction Amount* is the amount to charge for this transaction. This needs to be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against. *Card Number* cannot contain any non-numeric characters.

*Expiration Date* is the expiration date of the credit card to be charged. *Expiration Date* needs to be in the format "mmyy", "mm/yy", or "mm/yyyy".

*Response Full URL* is the full URL for AuthorizeNet to send the response to. When the transaction is processed, AuthorizeNet will send the transaction result to *Response Full URL*. *Response Full URL* needs to be capable of accepting an HTTP POST, and return a response to the AuthorizeNet server.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETRANS\_Transact** method to actually process the transaction.

---

## ETRANS\_Return\_TR

*ETRANS\_Return\_TR* ( *Referenced Transaction Record* )

*ETRANS\_Return\_TR*  
(

-> *Referenced Transaction Record* : Pointer  
 )

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>Referenced Transaction Record</i>	Pointer	Reference to eTrans transaction record

The method ***ETRANS\_Return\_TR*** will return the referenced transaction record. After calling this method, the referenced transaction record can no longer be used.

*Referenced Transaction Record* is a pointer to an eTrans transaction record built using one of the transaction creation methods.

### **ETRANS\_Set\_Fields\_TR**

*ETRANS\_Set\_Fields\_TR* ( *Referenced Transaction Record* ; *Referenced Field Types* ; *Referenced Field Values* ) => *qi Modification Successful*

*ETRANS\_Set\_Fields\_TR*

(  
 -> *Referenced Transaction Record* : Pointer  
 -> *Referenced Field Types* : Pointer  
 -> *Referenced Field Values* : Pointer  
 )  
 => *qi Modification Successful* : Longint

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>Referenced Transaction Record</i>	Pointer	Reference to eTrans transaction record
	<i>Referenced Field Types</i>	Pointer	Reference to longint array of eTrans field types to add or modify
	<i>Referenced Field Values</i>	Pointer	Reference to text array of field values
	<i>qi Modification Successful</i>	Longint	qi for whether modifications to eTrans transaction record is successful

The method ***ETRANS\_Set\_Fields\_TR*** will add or modify the specified fields in the eTrans transaction record. A qi is returned indicating if the modification was successful. This

is an "all or nothing" modification; if there are any errors, then the eTrans transaction record will not be modified at all.

Each field type is checked to see if it is valid for the transaction type of the eTrans transaction record. Each field value is also checked for validity according to its field type.

*Referenced Transaction Record* is a pointer to an eTrans transaction record built using one of the transaction creation methods.

*Referenced Field Types* is a pointer to a Longint array containing the eTrans field types to add or modify (see eTrans Field Types constant group). If the specified field type does not exist in *Referenced Transaction Record*, it will be added.

*Referenced Field Values* is a pointer to a Text array containing the field values for each of the field types in *Referenced Field Types*. These values are checked for validity according to the specified field type.

*qi Modification Successful* is a qi indicator for whether the modification to *Referenced Transaction Record* was successful. If the modification is successful, it will be set to one (1). If the modification is not successful, it will be set to zero (0).

---

## ETRANS\_Set\_Field\_TR

*ETTRANS\_Set\_Field\_TR* ( *Referenced Transaction Record* ; *Field Type* ; *Field Value* ) =>  
*qi Modification Successful*

*ETTRANS\_Set\_Field\_TR*

(  
    -> *Referenced Transaction Record* : Pointer  
    -> *Field Type* : Longint  
    -> *Field Value* : Text  
)  
=> *qi Modification Successful* : Longint

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>Referenced Transaction Record</i>	Pointer	Reference to eTrans transaction record
	<i>Field Type</i>	Longint	eTrans field type to add or modify
	<i>Field Value</i>	Text	Field value for the specified field type
	<i>qi Modification Successful</i>	Longint	qi for whether modifications to eTrans transaction record is successful

The method ***ETRANS\_Set\_Field\_TR*** will add or modify the specified field in the eTrans transaction record. A qi is returned indicating if the modification was successful. This is an "all or nothing" modification; if there are any errors, then the eTrans transaction record will not be modified at all.

The field type is checked to see if it is valid for the transaction type of the eTrans transaction record. The field value is also checked for validity according to its field type.

*Referenced Transaction Record* is a pointer to an eTrans transaction record built using one of the transaction creation methods.

*Field Type* is the eTrans field type to add or modify (see eTrans Field Types constant group). If the specified field type does not exist in *Referenced Transaction Record*, it will be added.

*Field Value* is the field value for the field type specified in *Field Type*. This value is checked for validity according to the specified field type.

*qi Modification Successful* is a qi indicator for whether the modification to *Referenced Transaction Record* was successful. If the modification is successful, it will be set to one (1). If the modification is not successful, it will be set to zero (0).

## **ETRANS\_SP\_Create\_Direct**

**ETTRANS\_SP\_Create\_Direct** ( *SurePay Merchant ID ; SurePay Password ; qi Live Transaction ; Order Number ; Order Subtotal ; Card Number ; Expiration Date ; Billing Name ; Billing Address ; Billing City ; Billing State ; Billing Country ; Billing Zip Code* ) => Referenced Transaction Record

**ETTRANS\_SP\_Create\_Direct**

(  
 -> *SurePay Merchant ID* : Text  
 -> *SurePay Password* : Text  
 -> *qi Live Transaction* : Longint  
 -> *Order Number* : Longint  
 -> *Order Subtotal* : Real  
 -> *Card Number* : Text  
 -> *Expiration Date* : Text  
 -> *Billing Name* : Text  
 -> *Billing Address* : Text  
 -> *Billing City* : Text  
 -> *Billing State* : Text  
 -> *Billing Country* : Text  
 -> *Billing Zip Code* : Text  
 )  
 => *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>SurePay Merchant ID</i>	Text	SurePay merchant ID
	<i>SurePay Password</i>	Text	SurePay password to process transactions
	<i>qi Live Transaction</i>	Longint	qi for whether this is a live transaction
	<i>Order Number</i>	Longint	Order number for this transaction
	<i>Order Subtotal</i>	Real	Order subtotal (before shipping and freight)
	<i>Card Number</i>	Text	Credit card number to be charged
	<i>Expiration Date</i>	Text	Credit card's expiration date (mm/yy)
	<i>Billing Name</i>	Text	Full billing name
	<i>Billing Address</i>	Text	Billing street address
	<i>Billing City</i>	Text	Billing city
	<i>Billing State</i>	Text	State or province of billing address
	<i>Billing Country</i>	Text	2-character country abbreviation of billing address
	<i>Billing Zip Code</i>	Text	Billing zip or postal code

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETTRANS\_SP\_Create\_Direct*** will create a transaction record for a SurePay credit card direct response transaction. This transaction will be authorized against a credit card. The billing information collected will also be used as the shipping address. Shipping address fields can also be set individually with ***ETTRANS\_Set\_Field\_TR***.

*SurePay Merchant ID* is your SurePay merchant ID.

*SurePay Password* is your SurePay password to be used for processing transactions.

*qi Live Transaction* is a qi for whether this is a live transaction. Set *qi Live Transaction* to one (1) to indicate a live transaction, or to zero (0) to indicate a test transaction.

*Order Number* is the order number or other unique identifier for this transaction.

*Order Subtotal* is the subtotal of the order, before shipping and sales tax charges. This needs to be a longint number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against.

*Expiration Date* is the expiration date of the credit card to be charged. *Expiration Date* needs to be in mm/yy format.

*Billing Name* is the full name of the cardholder.

*Billing Address* is the cardholder's billing street address.

*Billing City* is the city from the cardholder's billing address.

*Billing State* is the state from the cardholder's billing address. For states in the US, *Billing State* needs to be the two-character state abbreviation.

*Billing Country* is the two-character ISO country code from the cardholder's billing address.

*Billing Zip Code* is the postal code from the cardholder's billing address.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the ***ETRANS\_Transact*** method to actually process the transaction.

## **ETRANS\_Transact**

*ETRANS\_Transact* ( *Referenced Transaction Record* ; *Referenced Response* ) => HTTP Response Code

```
ETRANS_Transact
(
    -> Referenced Transaction Record : Pointer
    -> Referenced Response : Pointer
)
=> HTTP Response Code : Longint
```

	Parameter	Type	Description
	<i>Referenced Transaction Record</i>	Pointer	Reference to eTrans transaction record
	<i>Referenced Response</i>	Pointer	Pointer to a BLOB to contain the transaction response
	<i>HTTP Response Code</i>	Longint	HTTP response code returned from processing the eTrans transaction

The method ***ETRANS\_Transact*** will process the transaction with the service provider and return the response from the remote server.

*Referenced Transaction Record* is a pointer to an eTrans transaction record built using one of the transaction creation methods.

*Referenced Response* is a pointer to a BLOB which will be set to the response from the remote server. *Referenced Response* does not contain the HTTP header. The result from

processing the eTrans transaction is contained within *Referenced Response*.

*HTTP Response Code* is the HTTP response code from processing the eTrans transaction. If there is an error in this method before processing the eTrans transaction, then *HTTP Response Code* will be set to negative one (-1).

## ETRANS\_WP\_Check\_Exchange\_Rates

*ETTRANS\_WP\_Check\_Exchange\_Rates* ( *WorldPay ID* ; *WorldPay Password* ; *From Currency Codes* ; *To Currency Codes* ; *Exchange Rates* ; *Rate Date* ) => *Result Code*

```
ETTRANS_WP_Check_Exchange_Rates
(
  -> WorldPay ID : Text
  -> WorldPay Password : Text
  -> From Currency Codes : Pointer
  -> To Currency Codes : Pointer
  -> Exchange Rates : Pointer
  -> Rate Date : Pointer
)
=> Result Code : Longint
```

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>WorldPay ID</i>	Text	WorldPay Installation ID
	<i>WorldPay Password</i>	Text	WorldPay info password
	<i>From Currency Codes</i>	Pointer	Reference to text array to receive from currency ISO 4217 codes
	<i>To Currency Codes</i>	Pointer	Reference to text array to receive to currency ISO 4217 codes
	<i>Exchange Rates</i>	Pointer	Reference to real array to receive exchange rates
	<i>Rate Date</i>	Pointer	Reference to text variable to receive date rates are current
	<i>Result Code</i>	Longint	Transaction Result Code = 0 for success, rates are current = 1 for success, rates not current < 0 for error in routine

The method ***ETRANS\_WP\_Check\_Exchange\_Rates*** is a convenience method which creates a transaction for WorldPay's exchange rate API and parses the response into arrays for the developer.

*WorldPay ID* is the WorldPay installation ID.

*WorldPay Password* is the WorldPay info password.

*From Currency Codes* is a reference to a text array to receive the ISO 4217 currency codes of the currencies to convert from.

*To Currency Codes* is a reference to a text array to receive the ISO 4217 currency codes of the currencies to convert to.

*Exchange Rates* is a reference to a real array to receive the currency conversion exchange rates.

*Exchange Rates* is a reference to a text variable to receive the date string for when the rates were generated. This rate string is formatted YYYY-MM-DD.

*Result Code* is the result code of the exchange rate API transaction. *Result Code* is set to zero (0) if there was no error, one (1) if the request was successful but the rates are not current, and less than zero (<0) if there was an error in the routine.

---

## **ETRANS\_WP\_Create\_Exchange**

***ETRANS\_WP\_Create\_Exchange*** ( *WorldPay ID* ; *WorldPay Password* ; *Method* ) =>  
*Referenced Transaction Record*

***ETRANS\_WP\_Create\_Exchange***  
(  
    -> *WorldPay ID* : Text  
    -> *WorldPay Password* : Text  
    -> *Method* : Text  
)  
=> *Referenced Transaction Record* : Pointer

Parameter	Type	Description
<i>WorldPay ID</i>	Text	WorldPay Installation ID
<i>WorldPay Password</i>	Text	WorldPay info password
<i>Method</i>	Text	Rates to get Must be one of: rates rates-latest rates-today rates-tomorrow
<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_WP\_Create\_Exchange*** will create a transaction record for a WorldPay exchange rate API transaction. This transaction will interface with WorldPay's exchange rate API, and return the results as text.

*WorldPay ID* is the WorldPay installation ID..

*WorldPay Password* is the info password specified on your WorldPay account. Note that this is different from your account password.

*Method* is the rates to retrieve.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* with the ***ETRANS\_Transact*** method to actually process the transaction.

## **ETRANS\_WP\_Create\_Relay**

***ETRANS\_WP\_Create\_Relay*** ( *WorldPay ID* ; *Cart ID* ; *Currency Code* ; *Transaction Amount* ; *Card Number* ; *Expiration Month* ; *Expiration Year* ; *Cardholder Name* ; *Cardholder Address* ; *Cardholder Country* ; *Response Email Address* )  
=> *Referenced Transaction Record*

***ETRANS\_WP\_Create\_Relay***  
(

- > *WorldPay ID* : Text
- > *Cart ID* : Text
- > *Currency Code* : Text
- > *Transaction Amount* : Real
- > *Card Number* : Text
- > *Expiration Month* : Longint
- > *Expiration Year* : Longint
- > *Cardholder Name* : Text
- > *Cardholder Address* : Text
- > *Cardholder Country* : Text
- > *Response Email Address* : Text

)

=> *Referenced Transaction Record* : Pointer

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>WorldPay ID</i>	Text	WorldPay Installation ID
	<i>Cart ID</i>	Text	Cart ID (e.g. Basket ID or Invoice ID)
	<i>Currency Code</i>	Text	ISO 4217 transaction currency code
	<i>Transaction Amount</i>	Real	Amount of transaction
	<i>Card Number</i>	Text	Credit card number to be charged
	<i>Expiration Month</i>	Longint	Month of credit card's expiration date
	<i>Expiration Year</i>	Longint	Year of credit card's expiration date
	<i>Cardholder Name</i>	Text	Cardholder's full name
	<i>Cardholder Address</i>	Text	Cardholder's street address
	<i>Cardholder Country</i>	Text	Cardholder's country (2-character ISO country code)
	<i>Response Email Address</i>	Text	Email address to send response to
	<i>Referenced Transaction Record</i>	Pointer	Reference to transaction record built by this method

The method ***ETRANS\_WP\_Create\_Relay*** will create a transaction record for WorldPay credit card relay response transaction. This transaction will be authorized against a credit card. Be sure to see the notes in the **Supported Authorization Services** section for issues with WorldPay relay transactions.

*WorldPay ID* is the WorldPay Installation ID for the WorldPay account being used.

*Card ID* is an identifier for this transaction. It is normally the card ID, or shopping basket ID, or invoice ID, or something else necessary to identify this transaction.

*Currency Code* is the ISO 4217 currency code identifying the currency to be used for this transaction.

*Transaction Amount* is the amount to charge for this transaction. This must be a real number greater than zero (0).

*Card Number* is the credit card number for this transaction to be authorized against. *Card Number* cannot contain any non-numeric characters.

*Expiration Month* is the month of the expiration date of the credit card to be charged.

*Expiration Year* is the 4-digit year of the expiration date of the credit card to be charged.

*Cardholder Name* is the cardholder's full name. This value is used for both ETRANS\_FullName and ETRANS\_ShipToFullName.

*Cardholder Address* is the cardholder's billing street address. This is used for address verification purposes.

*Cardholder Country* is the cardholder's billing country. This is used for address verification purposes. *Cardholder Country* must be specified as the 2-character ISO country code.

*Referenced Transaction Record* is a pointer to the transaction record created by this method. If an error occurs while creating the transaction record, *Referenced Transaction Record* will be a NULL pointer. Use *Referenced Transaction Record* to add more data to the transaction, and with the **ETRANS\_Transact** method to actually process the transaction.

---

## INIT\_ETRANS

*INIT\_ETRANS* ( *eTrans Serial Number* )

*INIT\_ETRANS*

(  
    -> *eTrans Serial Number* : Text  
)

	<b>Parameter</b>	<b>Type</b>	<b>Description</b>
	<i>eTrans Serial Number</i>	Text	Serial number for the eTrans component

The method ***INIT\_ETRANS*** initializes the eTrans component. A single call to this method should be made early in the On Startup database method in your 4D application. Make certain the call to this method follows the initialization calls to the BASH, TCP Deux, and HTTP Client Deux components, but before any other calls to the eTrans component.

*eTrans Serial Number* is the serial number given to you by Deep Sky Technologies, Inc., when you purchased the eTrans component. If you are using a demo or beta copy of the eTrans component, look on the Deep Sky Technologies, Inc. web site or in the archive downloaded for the appropriate serial number.

## Version History

The following is a brief version history of the eTrans component. It details release notes, bug fixes, and changes for each version publicly available.

# eTrans v1.1.0

*released Wednesday, May 15, 2002 (20020515)*

## **Changes:**

Corrected bug in serialization checking routines in which the compiled state of the application was not being properly checked against allowable serialization.

Fixed a bug in the declaration of the “amount” parameter in the method ETRANS\_FP\_Create\_Direct.

Fixed rare bug in which some 4D serials and name could cause component to not serialize properly.

Added support for SurePay transactions; added new error code (in error codes document) for a new errors with SurePay.

Changed Authorize.Net and PlanetPayment transactions from v3.0 to v3.1 format to add support for CVV2 code. This also changes the format of the response string, adding new fields for v3.1. Response parsing routines should be checked to make sure there won't be any problems.

Updated routine for finding correct Affix document for this component; now uses generic routines in BASH for handling under 4D v6.8.x and MacOS X.

Added support for new 4D plugin architecture under 4D v6.8.x in which plugins can be located next to the current application in any environment.

Created a carbon affix document entitled “Affix\_eTrans.4CX”.

Changed name of Mac affix document from “Affix\_eTrans” to “Affix\_eTrans.4DX”.

Increased minimum version of BASH required to v1.7.0.

Increased minimum version of TCP Deux required to v1.1.0.

Increased minimum version of HTTP Client Deux required to v1.1.0.

# eTrans v1.0.0

*released 20020110*

## **Changes:**

Added support for IonGate direct and relay transactions.

Added support for FirePay direct transactions.

Fixed a bug in the checking of serial numbers when initializing the component.

Changed the name of the affix document on Windows from "Afx\_eTrans" to "Affix\_eTrans".

## Errors

The listing of error codes and conditions is obviously a continuously updated process. With practically every new version of eTrans, the error codes and conditions can and do change. Though it has been a long time coming, we are now documenting much of the error conditions that can occur in eTrans.

Different methods in the eTrans component can generate errors when used incorrectly or when used under the wrong circumstances. When an error condition is encountered, the methods within the eTrans component will call the applicable “\_ERROR” method (the “ETRANS\_ERROR” method, to be exact). The “\_ERROR” methods are documented above. The first parameter sent to an “\_ERROR” method is the error code identifying the unique error condition that occurred.

With future versions of the eTrans component (and other components to come), the management and handling of error conditions will become much better documented, much clearer to understand, and easier to handle in your code. For now, though, reading this manual thoroughly is the best single source of understanding for the error conditions that can arise in using the eTrans component.

# Error Codes

When an “\_ERROR” method is called in the eTrans component, the first parameter is always an error code. This error code is always a 7 digit integer indicating the unique error condition which was detected in the code.

These 7 digit numbers actually consist of two pieces for uniquely identifying the error code and condition. The first five digits is an internal code for the module which an error occurred within. The last two digits identifies the unique error condition from within a module that has been detected.

The following is a quick listing of all of the error codes, grouped by the module which the error codes are from. Each error code includes the textual description of the error condition.

## **ETRANS: 29038**

- 01 Failed to load transaction record type URLs
- 02 Transaction record type not within range of URLs loaded
- 03 HTTP connection failed
- 04 (reserved)
- 05 Failed to create transaction record
- 06 Failed to set initial values in transaction record
- 07 Not a valid transaction record type
- 08 Field type value out of range
- 09 Failed to load textual field names
- 10 Invalid transaction record field type or value for transaction record type
- 11 Stack of fields and field values not well formed
- 12 Stack of fields and field values empty
- 13 Referenced parameter is not of correct type
- 14 (reserved)
- 15 (reserved)
- 16 (reserved)
- 17 (reserved)
- 18 (reserved)
- 19 Serialization for eTrans component invalid
- 20 Failed to load SurePay XML Template

## Method Listing

The following is a listing of all of the methods within the eTrans component (COMPILER methods are not included in this listing), including all private methods which are not directly available in the API when the eTrans component is installed. Following each method is a list of the error codes which each method can generate.

ENV\_Get\_ETRANS\_HardName\_Long  
ENV\_Get\_ETRANS\_HardName\_Short  
ENV\_Get\_ETRANS\_RF\_FullPath  
ETRANS\_ANECR\_Confirm\_FieldData  
ETRANS\_ANEC\_Confirm\_FieldData  
ETRANS\_ANEC\_Create\_Direct  
    2903805  
    2903806  
ETRANS\_ANEC\_Create\_Relay  
    2903805  
    2903806  
ETRANS\_ANR\_Confirm\_FieldData  
ETRANS\_AN\_Confirm\_FieldData  
ETRANS\_AN\_Create\_Direct  
    2903805  
    2903806  
ETRANS\_AN\_Create\_Relay  
    2903805  
    2903806  
ETRANS\_Check\_Serial\_Full  
ETRANS\_Check\_Serial\_Quick  
ETRANS\_Confirm\_FieldData\_TR  
ETRANS\_Convert\_TRTypes\_x  
    2903807  
    2903808  
    2903809  
ETRANS\_Create\_TR  
    2903807  
ETRANS\_ERROR  
ETRANS\_EXR\_Confirm\_FieldData  
ETRANS\_EX\_Create\_Relay  
    2903805  
    2903806  
ETRANS\_FP\_Confirm\_FieldData  
ETRANS\_FP\_Create\_Direct  
    2903805  
    2903806  
ETRANS\_IGR\_Confirm\_FieldData  
ETRANS\_IG\_Confirm\_FieldData  
ETRANS\_IG\_Create\_Direct  
    2903805  
    2903806  
ETRANS\_IG\_Create\_Relay  
    2903805  
    2903806  
ETRANS\_INIT  
    2903819  
ETRANS\_Load\_SP\_PTEXTs

ETTRANS\_LP\_Confirm\_FieldData  
ETTRANS\_LP\_Create\_Direct  
2903805  
2903806  
ETTRANS\_PPECR\_Confirm\_FieldData  
ETTRANS\_PPEC\_Confirm\_FieldData  
ETTRANS\_PPEC\_Create\_Direct  
2903805  
2903806  
ETTRANS\_PPEC\_Create\_Relay  
2903805  
2903806  
ETTRANS\_PPEX\_Confirm\_FieldData  
ETTRANS\_PPR\_Confirm\_FieldData  
ETTRANS\_PP\_Check\_Exchange\_Rates  
ETTRANS\_PP\_Confirm\_FieldData  
ETTRANS\_PP\_Create\_Direct  
2903805  
2903806  
ETTRANS\_PP\_Create\_Exchange  
2903805  
2903806  
ETTRANS\_PP\_Create\_Relay  
2903805  
2903806  
ETTRANS\_Return\_TR  
2903813  
ETTRANS\_Set\_Fields\_TR  
2903810  
2903811  
2903812  
2903813  
ETTRANS\_Set\_Field\_TR  
2903810  
2903813  
ETTRANS\_SP\_Confirm\_FieldData  
ETTRANS\_SP\_Create\_Direct  
2903805  
2903806  
ETTRANS\_Transact  
2903807  
2903813  
2903819  
ETTRANS\_Transact\_POST\_Simple  
2903801  
2903802  
2903803  
2903813  
ETTRANS\_Transact\_SP\_XML  
2903801  
2903803  
2903813  
2903820  
ETTRANS\_Transact\_WP\_Relay  
2903801  
2903803  
2903813  
ETTRANS\_WPEX\_Confirm\_FieldData  
ETTRANS\_WPR\_Confirm\_FieldData

ETTRANS\_WP\_Check\_Exchange\_Rates  
ETTRANS\_WP\_Create\_Exchange  
    2903805  
    2903806  
ETTRANS\_WP\_Create\_Relay  
    2903805  
    2903806  
INIT\_ETTRANS  
RES\_Open\_ETTRANS