

Version

2

ENCORE SYSTEMS LLC

Web Development and eCommerce Integration

PayPal NVP API Class

Library User Guide

WEB DEVELOPMENT AND ECOMMERCE INTEGRATION

PayPal Class Library User Guide

The Encore Systems PayPal Class Libraries are wrappers for PayPal's APIs that are designed to simplify PayPal integration with ASP.NET websites and .NET applications. The Encore Class Libraries have a short learning curve and are well documented so that developers can spend their time designing their application and know that the PayPal integration will just work.

Two versions of the Class Library are available: one for the SOAP API and another for the name-value-pair (NVP) API.

This guide is for the NVP API class library.

© Encore Systems LLC
448 Ignacio Blvd • #196
Novato, CA 94949
510.524.4624

Table of Contents

Overview	2
Features	2
FAQ	3
Quick Start.....	5
Templates.....	5
Videos	6
Encore Class Library Reference	7
API Classes.....	7
Special Classes, Constants, and Binding Sources	9
PayPal Enumeration Classes.....	10
PayPal API Analysis Application	11

Overview

PayPal's NVP API was created to simplify Website Payments Pro development.

PayPal's original API is a SOAP web service which communicates with applications using XML messages. On some platforms, that requires the developer to build the XML messages, pass them to the SOAP web service, and then parse the XML responses. In order to simplify that process, PayPal created the name-value-pair API in which the communication takes the form of simple text strings that can be passed and retrieved in the URL much the same as query strings in standard HTML programming.

One of the difficulties of working directly with the NVP API is that the method and parameter names are entered as text strings. There is no type checking or ability to select the parameter from a list of properties. This leads to errors from typos or just not knowing what the correct parameter name is. The Encore Class Library simplifies this by encapsulating each API call into a single object that provides a list of parameter names to select from. No typos or searching the documentation for the needed parameter. The parameter list has popup help explaining each parameter and its use. To make an API call, the developer creates a single object, adds the necessary parameters from a list, and calls the Post() method.

Features

- The class library is designed to run under medium trust. This allows use in hosted environments such as GoDaddy, HostMySite, 1and1, RackSpace, etc.
- Each API call is encapsulated in its own .NET class. Each API class has its own set of parameter name constants to prevent typos and misspellings. Each parameter name has popup help showing its usage, limitations, possible values, etc. Just add the required name-value pair parameters and call the Post() member to complete any API call.
- The class library defines data objects for all PayPal enumeration types allowing simple population of DropDownLists. Includes States, Countries, Currencies, PaymentActions, CreditCardTypes, etc.

- Work with the API call parameters the way you prefer. Parameters can be manipulated as separate Name and Value strings, as Parameter objects, as NameValueCollection objects, or as Generic Lists of Parameter objects.
- The class library includes pre-defined objects for Payment Line Items, Billing Agreement Items, and Mass Pay Items. No need to number these parameters; just build up a list of item objects and add it to the parameter collection. Parameter numbering is automatic.
- All authentication credential properties such as Username, Password, Signature, etc. can be globally defaulted in the application's configuration file. For increased security, pass your own credentials function to the credentials delegate and the class library will call your function as necessary.
- Includes a complete diagnostic tool with documented source code showing every possible API call. Try any possible combination of property settings on API calls to see what PayPal's response is. Cut and paste the examples to get a quick start on your application.
- Supports Recurring Payments, Reference Transactions, 3rd-party payments, and all PayPal API calls as of Version 53.0. Choose between Live, Sandbox, or Beta Sandbox environments.

FAQ

Q. Will the class library still work if PayPal updates their API?

A. Yes. The class library was developed for a specific version of the API. As long as that version is used, the class library will continue to work as designed. Newer versions of the API do not break the older versions.

Q. What is the GetCredentials function in the example website?

A. This is an example of using the credentials delegate. The GetCredentials function returns a SoapCredentials class. Any function returning this class may be used as a delegate for setting the API credentials in each API call.

Q. Should I use the supplied GetCredentials function?

A. No. The GetCredentials function is designed for the online demo site. It stores the API credentials in Session so that multiple developers can use the site to test the class library. This would not be a good solution for a production site. Either store the credentials in the web.config or create your own delegate function that gets the credentials from a database, for example.

Q. Can I just set the API credentials in the code that sets the rest of the Request properties?

A. Yes. But, that would require setting them on every API call. Using the configuration file or a delegate is more efficient.

Q. Why am I getting Error 10501 Invalid Merchant Configuration?

A. This means that Website Payments Pro is not enabled for your account. You need to accept the billing agreement, or contact PayPal customer service to have your account configured correctly.

Quick Start

Visual Studio templates are provided for getting started quickly on various PayPal integration projects. The class library installation package is a Visual Studio Content Installer file. Just select the appropriate Project template and you will have with a completely configured, running application to give you a quick start. Or, select the appropriate Item template to add the class library to your existing project. The following templates are included:

Templates

- Example Site (C#) An ASP.NET web application demonstrating every possible API call. Use this C# application to explore the Encore Class Library and the PayPal SOAP API. Every API call can be built interactively from a form providing every possible parameter for that call. The code behind files completely document how each API object is built, configured, and called.
- Example Site (VB) Same as above for Visual Basic.
- Web Site (C#) A single page ASP.NET web site with the class library installed and configured. Use this C# application to start building a PayPal integrated site from scratch, or to see the minimum settings necessary to start one.
- Web Site (VB) Same as above for Visual Basic.
- Web Site Item (C#) Use this C# item template to integrate PayPal into an existing web application.
- Web Site Item (VB) Same as above for Visual Basic.
- Winform Application (C#) A Windows Forms application with the class library installed and configured. Use this C# application to start building a PayPal integrated desktop application from scratch, or to see the minimum settings necessary to start one.
- Winform Application (VB) Same as above for Visual Basic.
- Winform Application Item (C#) Use this C# item template to integrate PayPal into an existing desktop application.
- Winform Application Item (VB) Same as above for Visual Basic.

Videos

Check out these short videos for a quick start on various tasks.

[Class Library Installation](#)

[Obtain and Install License Keys](#)

Encore Class Library Reference

The latest documentation for the NVP API can be found on the PayPal site here: https://www.paypal.com/en_US/pdf/PP_NVPAPI_DeveloperGuide.pdf. The following class library API classes correspond directly to the API calls listed in the PayPal documentation. All classes are prefixed with “Nvp” to differentiate them from the Name SOAP class library. This allows both class libraries to be used in the same application without fully qualifying each class.

API Classes

- NvpAddressVerify returns the status of a PayPal account address, including street and zip code match results. (undocumented, untested, requires special permission)
- NvpBillOutstandingAmount bills any balance due on a recurring payments profile.
- NvpCreateBillingAgreement returns a billing agreement ID that may be used to make reference transactions against a PayPal account. (undocumented, needs special permission)
- NvpCreateRecurringPaymentsProfile creates a recurring payments profile allowing automatic periodic billing.
- NvpDoAuthorization is your request to authorize a customer order that can be fulfilled within 29 days.
- NvpDoCapture is your request to completely or partially settle an order, an authorization, or reauthorization.
- NvpDoDirectPayment charges or authorizes a credit card.
- NvpDoExpressCheckoutPayment obtains payment through Express Checkout for a final sale or requests authorization for later capture of payment.
- NvpDoNonReferencedCredit makes a non-referenced credit to a customer’s credit card.
- NvpDoReauthorization reauthorizes a previously authorized a transaction and that has passed its settlement period.

- NvpDoReferenceTransaction makes a payment based on the information provided in a previous Direct Payment, avoiding the need to provide any buyer information.
- NvpDoVoid voids an order or an authorization.
- NvpGetBalance returns the PayPal account balance of the account whose API credentials are used for the call. (undocumented)
- NvpGetBillingAgreementCustomerDetailsRequest returns information about the customer, including name and address on file with PayPal.
- NvpGetExpressCheckoutDetails returns information about the customer, including name and address on file with PayPal.
- NvpGetRecurringPaymentsProfileDetails returns the details of a recurring payments profile.
- NvpGetTransactionDetails gets details about a transaction.
- NvpManagePendingTransactionStatus accepts or denies a pending transaction. (undocumented, untested, needs special permission)
- NvpManageRecurringPaymentsProfileStatus allows a recurring payment profile to be cancelled, suspended, or reactivated.
- NvpMassPay processes payments en masse to up to 250 different recipients.
- NvpRefundTransaction refunds a payment.
- NvpSetCustomerBillingAgreementRequest indicates to PayPal that you are using Express Checkout to create billing agreements for your customer.
- NvpSetExpressCheckout indicates to PayPal that you are using Express Checkout to obtain payment from your customer.
- NvpTransactionSearch searches transaction history.
- NvpUpdateRecurringPaymentsProfile allows modification of the terms of a recurring payments profile.

Special Classes, Constants, and Binding Sources

- `NvpConfig` is a static class that returns all properties defined in the `encore.PayPal.Nvp` configuration section of the application's configuration file. Default values are returned for any property not set in the configuration file.
- `NvpCountryCodeType` encapsulates a list of countries, including name and two-character code. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `NvpCurrencyCodeType` encapsulates a list of currencies accepted by PayPal, including name and three-character code. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `NvpCredentialsDelegate` The developer can define a function that takes no parameters and has return type `NvpCredentials`, and then pass that function to any API call instead of setting credentials individually. This can increase security by passing the API credentials from an encrypted store of the developer's choice. The delegate function may also be declared in the application's configuration file to make this a global setting.
- `NvpLocaleCodeType` encapsulates a list of countries for which PayPal Express Checkout pages include culture settings. Includes name and two-character code, a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `NvpMonthCodeType` encapsulates a list of calendar month constants, including integer value, short name, and long name. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `NvpStateCodeType` encapsulates a list of states and provinces, including name and two-character code. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.
- `NvpYearCodeType` encapsulates a list of calendar year constants from the current year through current year plus eleven. Includes a `BindingSource` property to allow binding directly to `DropDownList` objects, and a `Get()` member to allow binding to an `ObjectDataSource`.

PayPal Enumeration Classes

Unlike the SOAP API, the NVP API doesn't automatically provide enumerations for all of the special PayPal types such as `BillingCodeType`, `ChannelType`, etc. The Encore Class Library includes a class of constants for each of these special PayPal types. These Encore classes also have a `Get()` method and a `BindingSource` property that allows the developer to bind these lists to `ObjectDataSources` and `DropDownLists`. Following are enumeration classes included in the class library.

- `NvpAutoBillType`
- `NvpBillingCodeType`
- `NvpBillingPeriodType`
- `NvpChannelType`
- `NvpCompleteCodeType`
- `NvpCreditCardTypeType`
- `NvpFailedPaymentActionType`
- `NvpLandingPageType`
- `NvpMerchantPullPaymentCodeType`
- `NvpPaymentActionCodeType`
- `NvpPaymentTransactionClassCodeType`
- `NvpPaymentTransactionStatusCodeType`
- `NvpReceiverInfoCodeType`
- `NvpRefundType`
- `NvpSolutionTypeType`
- `NvpStatusChangeActionType`
- `NvpTransactionEntityType`

PayPal API Analysis Application

Included with the Encore Class Library is a web application that serves many important purposes. This application demonstrates every PayPal API call. The code-behind files for each API call are well documented and show all required and optional parameters. Modular user controls for collecting and presenting PayPal API call parameters are readily available for possible adaption in your own application.

In addition, the example application serves as an analysis tool for the PayPal API. Every parameter can be set interactively when making API calls. This allows the developer to test every possible combination of parameter settings to see what PayPal's response will be.

Enter your live API credentials and the example application becomes an interactive PayPal terminal. Use it to research transactions, capture authorized payments, make refunds, modify recurring payments profiles, check your balance, enter payments for phone orders, etc.

Display the request and response parameters in a parsed, formatted list. After making any API call, click a button the open a pair of text boxes displaying exactly what was sent to and received from the API.

Available only with the NVP class library: Includes an interactive generic API call builder. Pick any API call from a drop down list and the builder automatically displays the correct parameter list. Or, build the API call from a request string – perfect for testing request strings provided by PayPal Forum users.