

# Developing NetVIOS-Based Applications

## 1.0 INTRODUCTION

NetVIOS based applications are functional web software that are developed to run on a web site with NetVIOS OS running on it much in the same way as a PC software runs on a PC operating system. A NetVIOS-based application utilizes NetVIOS operating system features embedded in NetVIOS. Such features may include the user-based system, the resource and security and auditing system, presentation functions as well as form building features.

### **User-Based Functions**

The application does not need to create user data, access or authentication features. IT simply uses the features available in NetVIOS. For instance, consider a member based greeting card application. This application may need an unauthenticated user with a first name, last name and email address. The developer of the application will first verify if these user variables are native to NetVIOS (in NetVIOS' User or UserProfile table). If this is the case, the user type that will make use of the application is set up to include these user variables. The developer then simply accesses these variables in his codes or scripts because they will be made open by the operating (NetVIOS) system

If the application includes user variables that are not in any current NetVIOS user table, a user table is created with the required variables. The user table name is registered in the database table **UserTables**. The fields or user variables that need to be opened by the OS is also entered in the **UserFields** table. The application manuals should instruct the site owner of what user variables must be included with the site user type that is intended to access the application. Then, the implementer or site owner/administrator installing the application will ensure that the user type that will access the software includes these variables and that the required input are requested from the user.

Technically, any application can run on NetVIOS even if it creates all its own functions and uses no NetVIOS functions. This is because NetVIOS is an open non-restrictive system that freely layers on the IIS environment. However, the synergy is lost if every application creates its own member features, it's own login and its own file system and GUI. This method is expensive and annoying. For instance, a user may be asked for his name and to provide a password while using a site's chat room and again when accessing the online store. This appears to be the state of web-based applications prior to NetVIOS.

Programmatically accessing NetVIOS user variables is described in Section 2.

## **Look-Based Functions**

It would be ideal to install an application on your website and have it automatically look custom to your site. Making use of NetVIOS's feature functions, the developer may programmatically access the site's look-based variables. The script file `_sitepreferences.asp` contains the declared site features and should be server-side included on pages in which you wish to use these features.

The developer may also take advantage of NetVIOS's feature commands to define additional features. In this context, features are simply html variables such as color, dimensions, image URLs, or they may be strings or other quantities that you wish to programmatically change to affect the behavior of the software after installation. For instance, a feature may determine if you wish to allow non-registered site members perform an application's function. The site owner or installer may set this feature after installing your software. Features may also be declared as cookies in which case it is the individual site user that causes its modification or modifies its value as needed.

All you need to do as a developer is to declare your feature on the page or script in which it is used together with a default (or factory) value and NetVIOS will handle the rest. NetVIOS initializes your feature so that it is made available to the site owner/administrator to set its value. NetVIOS also handles inserting the appropriate value as well as caching your feature for quick rendering of the page. The user chapter on Features describes how to use features in greater detail.

## **GUI-Based Functions**

NetVIOS provides GUI functions for generating forms and committing the form's data to a database. NetVIOS includes the GUI controls listed in publication **NetVIOS GUI Controls**.

The GUI controls may be displayed using the **drawFormitem** command. The syntax of this command is referenced in NetVIOS API reference under the `formitems` class.

## 2.0 USING USER VARIABLES

There is a session variable for every user variable. All NetVIOS user variables are registered in the table **UserFields** in the NetVIOS database.

When a user logs into NetVIOS, the session variables are created. Only session variables that have been included in the definition of the user type are created. It is the job of the programmer to check that a variable required for an application has been declared for a user accessing that application. All applications should document what variables should be included in a user type defined to use the application.

The session variable declared for each user variable has the form “VariableWOS” i.e. the suffix “WOS” is placed after the variable.

UserFieldID	FieldName	Table	FieldDataType	FieldSize	GUIType	FieldValue	FieldList
1	UserID	Users	Integer		SystemKey		
2	LoginName	Users	String	50	Input		
3	Password	Users	String	50	Password		
4	FirstName	Users	String	50	Input		
5	LastName	Users	String	50	Input		
6	MiddleInitial	Users	String	2	Input		
7	Alias	Users	String	50	Input		
8	Email	Users	String	250	Email		
9	CreatedDate	Users	Date	0	System		
10	LastLogin	Users	Date	0	System		
11	Logins	Users	Integer	10	System		
12	UserType	Users	Integer	0	DropDownQ	Select UserTypeID, UserType From UserTypes	
13	Active	Users	Boolean	0	Button	True,False	Yes,No
14	Blocked	Users	Boolean	0	Button	True,False	Yes,No
15	HomeDirectory	Users	String	250	Input		
16	Deleted	Users	Boolean	0	Button	True,False	Yes,No
18	BirthDate	UserProfile	Date	0	Date		
19	Age	UserProfile	Integer	4	Input		
20	Gender	UserProfile	String	0	Button	F,M	Female,Male

Consider an extract of the **UserFields** table shown above. If Age and Gender have been declared for a user with a given login name, once the user logs in, the developer's application will have access to the variables Session("AgeWOS") and Session("GenderWOS") both containing the user's age and gender respectively.

### **Adding User Variables**

A developer has the option of adding new user variables to NetVIOS. To do this, the developer has two options:

1. Create new fields in an existing user table such as UserProfile.
2. Creating a new user table with the application's user variables.

In both cases, the application (developer) must register the new user variables in the UserFields table.

**In the case of (2), the user must also declare the user table in the list of user tables in the **UserTables** table of the NetVIOS database.**

Once the new user variables have been registered with NetVIOS, they become available in the definition of user types and in use at runtime just like the native NetVIOS user variables.

#### **CAUTION:**

Good programming practice would lead the developer to conclude that (2) is preferable because other applications may wish to create similar variables with an existing user table such as UserProfile and control in versioning, uninstalling, maintenance and conflict resolution becomes more difficult. **In fact, for an application to obtain NetVIOS Certification status it needs to use the second method.**

### **The UserFields table**

This table is the repository of every available NetVIOS user variable, which would be stored in diverse user tables in the NetVIOS database. The variables are registered in UserFields by providing both the exact field name and the table name of the user field. In addition, the datatype of the field and a GUI type that is required to interrogate for the variable is also indicated. The list of NetVIOS GUI types is in the document – NetVIOS GUI Controls. The data types may be Integer, Real, String, Date, Time, and Boolean.

Two additional GUI types are also allowed for user variables – System and SystemKey. A user is never interrogated for user variables with this GUI type even though the session variables for these variables are available at run time

### **User Variable Resolution**

If two variables have the same FieldName, but belong to different user tables (obviously you can not have two database fields with the same name in any case), they are treated as one. In other words, the same session variable is used for these fields and the user is interrogated only once for this field.

For example, if a table eCommerceVars contains a user field variable called Phone. If this variable is included in a user type definition together with the Phone variable in UserProfile, the user would be asked to enter a value for Phone only once and this value will be stored in both the UserProfile and eCommerceVar tables. In addition only one session variable – PhoneWOS – is available to the developer for this variable at run time when a user logs in. For this reason, it is good to publish new user variables with the NetVIOS website so that other developers may be aware of their existence. **In fact, this requirement is mandatory if you wish to have your application declared NetVIOS-compatible and listed freely on the main NetVIOS site.**

## 3.0 BUILDING NetVIOS FORMS AND PAGES

Unlike PC software, web site software have the additional requirement that they should preferably not look the same on every site. As a result, the developer should take advantage of the feature-based attributes of NetVIOS. Features can be declared and used on any page using the NetVIOS system. At run time, the system will assign values to them. The values of the system will be either a default (factory value) or a value the site owner sets them to be. NetVIOS is responsible for handling the setting of the values of the feature by the users.

### Site Features

Even though the developer may define new features for their particular application and html pages, NetVIOS already has some default site features (`_sitepreferences.asp`). Taking advantage of the default site features means that the application may be made to immediately assume a look consistent with the site upon installation. This is because some default site features such as background color and image, font color and sizes and layout features are declared in the default site preferences file.

To use the default site features on your site, simply include the file `_sitepreferences.asp` at the start of your asp page.

NOTE: If you declare duplicate features to those in the default site features, your own features will take precedence if they are declared after the section including the site feature file. The advantage of this is that your declarations always maintain their integrity even with later versions of the `_sitepreferences.asp` file.

### Forms

There are three parts to building a form:

1. Drawing up the GUI controls – textboxes, combo boxes, labels, submit buttons, buttons etc.
2. Validating the input
3. Storing or processing the input

NetVIOS provides commands to assist the developer in easily accomplishing the tasks. These commands are described below:

## Drawing the GUI Controls

Using the commands **drawformitem**, **viewformitem**, **showmonth**, **showhour**, **showminute** etc of the FORMITEMS class, the developer may present GUI controls on their page. The advantage to using NetVIOS's functions to draw the GUI controls are as follows:

- Quick development time as the commands are easily deployed. For instance, it would take the developer quite a bit of time to develop a custom control that would pop-up a window to select a color. The NetVIOS GUI control type - **color** – automatically handles this allowing the developer focus on the main functions of their application.
- Compatibility with the site. For instance, a site feature to set the size of form labels is uniformly applied to all NetVIOS controls.
- NetVIOS GUI controls are compatible with other parts of the operating system. The **User** GUI control type, for instance, automatically links with the NetVIOS database to allow the selection of the user. The **FilePath** control type links with NetVIOS WebManager to allow the selection of a file directly from the site.

The syntax for using the **drawformitem** class and a list of NetVIOS GUI types are contained in the NetVIOS API and NetVIOS GUI controls publications respectively.

## Validating the input

Form inputs may be validated either from the client end or from the server end. For most applications, both methods would be in use. However, it is preferable to shift as much of the validation to the client as much as possible so that form values do not come to the server until absolutely necessary (the server is doing enough already). Client site validation is done with JavaScript. Virtually all browsers support JavaScript without any additional program to install, which makes the use of JavaScript ideal.

NetVIOS provides commands that would dynamically insert html JavaScript commands in your pages. The command **putvalidation** – which is part of the formitems class – is used to insert the JavaScript syntax in an html page.

## Storing the Input

If you wish to save the contents of your form into a table in the NetVIOS database. The commands –**CreateInUserDbase**, **UpdateCreateInDbase**, **UpdateCreateDbaseRec**, **UpdateDbaseRec**, **UpdateWOSRec**, **CreateWOSRec** –

are used to automatically save content in a NetVIOS database. The commands - **CreateInDbase**, **UpdateInDbase** - will create or update database records in any database. Check the form of the syntax for these commands in the NetVIOS API publications.

### Combining the three activities

The developer may declare an array of form variables that may be used for accomplishing all three above tasks. The commands - **DrawGUIForm**, **DrawCatGUIForm**, **ValidateForm** - may be used with the commands above for storing input. An example of a script in which this is done is editwebtopicon.asp, editwebtop.asp, editgallery, editpicitem.asp.

## 4.0 INSTALLING NetVIOS Applications

### **The NetVIOS Application Table**

Apart from getting the application files onto the website, NetVIOS applications also need to be registered in the **AppTable** table of the NetVIOS database. A unique application code needs to be created for the application and used with the definition in the **AppTable**. This application code will also be used in the ScopeName of any feature used for the application.

### **Application Database**

As a developer, you may choose to implement your data tables in the NetVIOS database or in a different database. If you choose to implement your tables in NetVIOS, you will need to ensure that there is no conflict with existing tables in NetVIOS. You may programmatically determine this by using a SELECT FROM TABLE command and test for error. To include tables in the NetVIOS database, you may use SQL create table commands. Note that the handle for the NetVIOS data source name (DSN) is declared in the Application variables – IOSDb, IOSDbUserName and IOSDbPassword. Table names that have the Application code (which is unique) attached to it are likely to be unique.

### **Application Code**

Every application entered into the Application Database has a unique code. You may determine a code for your application after checking the NetVIOS site to ensure that no prior application uses this code. This is in fact a requirement for an application to obtain NetVIOS-Certified status. The main technical reason why applications should have a unique code is so that features defined for these applications are correctly identified with them. This is because the internal unique feature name assigned to each feature for identification and caching is a function of the application code.

### **Application Libraries or Script Files**

Many applications may be defined completely in script files such as asp, jsp or php files. However, you may also implement the processing of the applications in a library or dll. Libraries or dlls allows you to access system details and commands that are not available in scripts. In addition, with dlls you to compile your applications and secure them in the case that they are commercial.

**In general, a thorough analysis of the included NetVIOS applications will reveal how much of NetVIOS' functions are used and provide the developer with further ideas on how to utilize the powerful and intrinsic commands and features to produce powerful and functional applications very quickly and at a much lower cost.**