



## **Professional Validation And More What's New In Version 2.0**

Copyright © 2004, Peter L. Blum. All Rights Reserved

## What's New In Version 2.0

*Note: All of these features are built into the PeterBlum.VAM.dll and its script files. Follow the [Installation Guide.pdf](#) file to install these files. The "New Controls" will not appear in the Visual Studio.net or Web Matrix Toolboxes unless you follow the installation steps in that document.*

### New Controls

- New validator: EmailAddressValidator (and EmailAddressCondition). Reviews the pattern of text for an email address to make sure it confirms with standards. In addition, you can hook it up to [Advanced Intellect's aspNetMX](#) validation engine for more thorough validation. See page 35 of the User's Guide.
- New validator: CreditCardNumberValidator (and CreditCardNumberCondition). Reviews the patterns in credit card numbers to confirm they match basic rules of credit card numbers. You can define a list of credit card brands that you support so that other credit card numbers are considered illegal entries. *This feature does not run the credit card information to a service that approves or rejects the credit card.* See page 66 of the User's Guide.
- New validator: ABARoutingNumberValidator (and ABARoutingNumberCondition). Reviews check routing numbers according to the standard setup by the American Bankers Association. *This feature does not test the number against the official institution routing numbers.* See page 37 of the User's Guide.
- New validator: CharacterValidator (and CharacterCondition). Confirms that every character in a text field is in a set of characters that you specify. See page 31 of the User's Guide.
- New textbox: FilteredTextBox. Filters out characters that you don't want as the user types. So many fields in a database have a limited number of characters that they support, such as a social security number, phone number, and credit card number. The FilteredTextBox may be useful on most text entry situations. See page 155 of the User's Guide.
- New control: CombinedErrorMessages. Now you can replace individual Error Formatters on each validator with a single Error Formatter. Removes the clutter when multiple error messages are shown as now they appear as a single piece of text under one error formatter. For example, an Image with Alert Error Formatter will list all error messages associated with a particular field in the alert box. You determine how the error messages are separated: by line breaks or text separators. See page 45 of the User's Guide.

### Validation Enhancements

#### User Interface

- New error formatter: HyperLinkErrorFormatter ("Hyperlink with alert"). When an error occurs, a hyperlink appears, with some static text like "Explain". When the hyperlink is clicked, it shows an alert with the error message. It is a great way to reduce the page size while using a familiar metaphor to click. See page 43 of the User's Guide.
- The new HiliteFields feature allows other controls to change their style when a validation error is shown. For example, change the font color of the field label or the background color of an enclosing <DIV>. This is another way to catch the user's attention. See "Change the Style of Other Fields Nearby the Error" on page 93 of the User's Guide.
- CustomConditions, CustomValidators and any validator that uses the **ServerCondition** property now support the ability to change the **ErrorMessage** and **SummaryErrorMessage** within your custom condition, on both the client and server side. See the topic "Changing The Error Message On The Server Side" on page 57 and the topic "Changing The Error Message On The Client Side" on page 61 of the User's Guide.
- Use the new page-level property **Globals.Page.FocusAfterAlert** to set focus to the field with the error after the user invokes the alert box on the Image With Alert (AlertImageErrorFormatter) and HyperLink With Alert (HyperLinkErrorFormatter) error formatters. See page 100 of the User's Guide.
- Use the new global properties **DefaultRequiredFieldToolTip** and **DefaultRequiredFieldToolTipLookupID** to establish a default tooltip for the RequiredFieldMarker control. For example, have a tool tip that says, "Required". See page 136 of the User's Guide.
- Use the new ValidationSummary property **ScrollIntoView** to scroll the browser to the ValidationSummary when the user submits the page and the ValidationSummary control is shown. This is a client-side only feature. See page 130 of the User's Guide.

- Use the new page-level property **Globals.Page.SetFocusFunctionName** to hook your own script into the code that sets focus to a field with an error. See page 104 of the User's Guide.

## Evaluating Conditions

- The `RegexValidator` and `RegexCondition` now have the property **IgnoreBlankText**. It determines how to treat blank text. When `true`, blank text is not evaluated (like it does now and like Microsoft's `RegularExpressionValidator` does.) When `false`, blank text is treated as an error. See page 65 of the User's Guide.
- Validators and `FieldStateControllers` have a new property, **ExtraControlsToRunThisAction**. This allows you to introduce more controls that when clicked or changed, run the validator or `FieldStateController`. This is frequently used with `RadioButtons`, where your condition identifies one radiobutton and this collection identifies the others that will unmark the primary radiobutton.

On the `FieldStateController`, it replaces the properties **ControlIDToRunThisAction** and **ControlToRunThisAction**. Any existing code will continue to work but it's converted to the new property at runtime. In design time, you will no longer see **ControlIDToRunThisAction** in the Property Editor. See pages 52 (for validators) and 147 (for `FieldStateControllers`) of the User's Guide.

- Validators no longer require that you set **Enabled=false** when the **ControlToEvaluate** property is set to **Visible=false**. Instead, they automatically act as if **Enabled** is set to `false`.
- Easily support third party textboxes, checkboxes, radiobuttons, listboxes, and dropdownlists that were not subclassed from their Microsoft equivalents and do not support the `ValidationPropertyAttribute`. (VAM already handles the rest.) See the topic "Supporting Third Party Data Entry Controls" in the [Installation Guide](#).
- Support for client-side validation on third party client-side menu and toolbar controls. When a menu command is selected that you want to post back, you can have it run client-side validation first. (You could always setup server-side validation on anything that posts back.) See the topic "Using Menu and Toolbar Controls to Submit the Page" on page 111 of the User's Guide.
- Support for [Infragistics NetAdvantage](#) ASP.NET Grid (UltraWebGrid) v2.1 and v3. See page 121 of the User's Guide.
- Use the new page-level property **Globals.Page.SubmitOrder** to determine the order of these three actions that occur when the page is submitted: validation, confirm message (when using the **ConfirmMessage** property), custom submit function (when using the **CustomSubmitFunctionName** property.) See page 103 of the User's Guide.

## Specialized TextBox Enhancements

- The `IntegerTextBox`, `DecimalTextBox` and `CurrencyTextBox` now support an optional Spinner control. A spinner is a pair of arrows to the right of the text box that increment and decrement the value. See the topic "Adding a Spinner" on page 158 of the User's Guide.
- `DecimalTextBox` has a new property, **TrailingZeroDecimalPlaces**, that determines if additional trailing zeros are added. It can also remove the decimal part when the value equals a whole number. See page 160 of the User's Guide.
- `CurrencyTextBox` has a new property, **HideDecimalWhenZero**, that determines if the decimal part is shown when the value is a whole number. When used, the text "12.00" will be reformatted as "12" while "12.1" will reformat as "12.10". See page 161 of the User's Guide.

## Other Features

- The new Global Settings Editor application provides a Windows UI to edit the custom.VAM.config file. It supports the `<GlobalProperties>` and `<StringLookup>` sections of that file. Run it from the Start menu in the Professional Validation And More program group. See "The Global Settings Editor" on page 179 of the User's Guide.
- Convert Pages To VAM.exe now converts the datagrid's `EditCommandColumn` and `ButtonColumn` classes.
- The page level property **Globals.Page.ConfirmMessage** now supports localization and string lookups using the new **Globals.Page.ConfirmMessageLookupID** property. See page 102 of the User's Guide.
- Many improvements to the User's Guide based on user feedback.
- New Installation Guide, replacing the Installation section of the User's Guide and the entire Upgrading Assistance.pdf.