ComponentOne

Binding Expressions for WPF and Silverlight

ComponentOne, a division of GrapeCity
201 South Highland Avenue, Third Floor
Pittsburgh, PA 15206 USA

Website: http://www.componentone.com
Sales: sales@componentone.com
Telephone: 1.800.858.2739 or 1.412.681.4343 (Pittsburgh, PA USA Office)

Trademarks

The ComponentOne product name is a trademark and ComponentOne is a registered trademark of GrapeCity, Inc. All other trademarks used herein are the properties of their respective owners.

Warranty

ComponentOne warrants that the media on which the software is delivered is free from defects in material and workmanship, assuming normal use, for a period of 90 days from the date of purchase. If a defect occurs during this time, you may return the defective media to ComponentOne, along with a dated proof of purchase, and ComponentOne will replace it at no charge. After 90 days, you can obtain a replacement for the defective media by sending it and a check for $25 (to cover postage and handling) to ComponentOne.

Except for the express warranty of the original media on which the software is delivered is set forth here, ComponentOne makes no other warranties, express or implied. Every attempt has been made to ensure that the information contained in this manual is correct as of the time it was written. ComponentOne is not responsible for any errors or omissions. ComponentOne’s liability is limited to the amount you paid for the product. ComponentOne is not liable for any special, consequential, or other damages for any reason.

Copying and Distribution

While you are welcome to make backup copies of the software for your own use and protection, you are not permitted to make copies for the use of anyone else. We put a lot of time and effort into creating this product, and we appreciate your support in seeing that it is used by licensed users only.

Table of Contents

Binding Expressions for WPF and Silverlight Overview

Never write another binding converter again with **Binding Expressions for WPF and Silverlight**.

Save development time and write cleaner XAML by taking advantage of inline expression syntax. Concatenate strings, compute formulas, and even apply if/else logic directly in your binding statements without the need for complicated, code-behind converters.

See Also

Help with WPF and Silverlight Edition

Key Features

Help with WPF and Silverlight Edition

Getting Started

- For information on installing ComponentOne Studio WPF Edition, licensing, technical support, namespaces and creating a project with the control, please visit Getting Started with WPF Edition.
- For information on installing ComponentOne Studio Silverlight Edition, licensing, technical support, namespaces and creating a project with the control, please visit Getting Started with Silverlight Edition.

Key Features

Binding Expressions for WPF and Silverlight provides the following unique key features:

- **Support for Operators and Functions**
  Perform simple operations (+, -, *, /) and functions directly in your binding statement. Supported functions include Sum, Average, and Concatenate. For a list of operators and functions, see C1Binding Expression Syntax Elements.
- **Use Quotes without Breaking your XAML**
  Use quotes in your binding expressions without breaking your XAML. **C1Binding Expressions** support inline quotes in two forms: &**quot**; or use of the pipe (**|**) character.
- **Logical Expressions**
  Apply simple if/else logic directly in your binding statement. **C1Binding** supports an 'IF' function with the following syntax: if(condition, true value, false value).
- **Clean and Expressive XAML**
  By using **C1Binding Expressions**, your XAML will be much cleaner and more expressive. And it's 100% self-contained. You do not need any external converters referenced from elsewhere in your solution.
- **Calculation Engine**
  In addition to the C1Binding class, the **C1.Silverlight.Binding** and **C1.WPF.Binding** assemblies provide a C1CalcEngine class that is responsible for evaluating the expressions. The **C1CalcEngine** class can be used independently of the **C1Binding**, to implement calculated input boxes, spreadsheets, and many other applications. See the **C1CalcEngine** class for more information and some usage scenarios.
- **Silverlight 5 Support**
  The **C1.Silverlight.Binding** classes require Silverlight 5. This is because they rely on MarkupExtensions, which are not available in earlier versions of Silverlight.

Introduction to Binding Expressions

WPF and Silverlight use Binding objects to connect data values to UI properties. For example, to display the name of a customer in a **TextBlock** element, you could use this XAML:

XAML	Copy Code

```
<TextBlock
   Text="{BindingFirstName}" />
```

In this example, the **Text** property of the **TextBlock** will be automatically synchronized with the content of the **FirstName** property of the data object currently assigned to the **DataContext** property of the **TextBox** element or any of its ancestors.

The **Binding** class is convenient but not very flexible. For example, if you wanted to bind the **TextBlock** above to the customer's full name, or to make the text red when the customer account is negative, you would need a **Converter**.

The **Converter** parameter of the **Binding** object allows you to add arbitrary logic to the binding, converting values when they are transferred between the source and the target.

To illustrate, and still using the example above, imagine we wanted the **TextBlock** to show the customer's full name, and to display the text in red when the customer's account is negative. To accomplish this, we would need two converters, implemented as follows:

XAML	Copy Code

```
/// <summary>
 /// Converter for Foreground property: returns a red brush if the
 /// customer amount is negative, black otherwise.
 /// </summary>
 publicclass ForegroundConverter : System.Windows.Data.IValueConverter
 {
   public object Convert(object value, Type targetType, object parameter,
     CultureInfo culture)
   {
     var c = value as Customer;
     return c.Amount < 0
       ? new SolidColorBrush(Colors.Red)
       : new SolidColorBrush(Colors.Black);
   }
   public object ConvertBack(object value, Type targetType, object parameter,
     CultureInfo culture)
   {
     throw new NotImplementedException();
   }
 }
/// <summary>
 /// Converter for customer's full name: returns a string containing
 /// the customer's full name.
 /// </summary>
 publicclass FullNameConverter : System.Windows.Data.IValueConverter
 {
   public object Convert(object value, Type targetType, object parameter,
     CultureInfo culture)
   {
     var c = value as Customer;
     return string.Format("{0} {1}", c.FirstName, c.LastName);
   }
   public object ConvertBack(object value, Type targetType, object parameter,
     CultureInfo culture)
   {
     throw new NotImplementedException();
   }
 }
```

Once the converters have been defined, they can be used in XAML as follows:

XAML	Copy Code

```
<Grid>
   <Grid.Resources>
     <local:FullNameConverterx:Key="_cvtFullName" />
     <local:ForegroundConverterx:Key="_cvtForeground" />
   </Grid.Resources>
   <TextBlock
     Text="{BindingConverter={StaticResource _cvtFullName}}"
     Foreground="{BindingConverter={StaticResource _cvtForeground}}" />
 </Grid>
```

This solution works fine, but there are a few shortcomings:

- You can’t tell what the output will look like by looking at the XAML. The color converter could, for example, return any color, using any logic. The only way to figure this out would be to look at the converter implementation.
- If you wanted to set the **Opacity**, **Visibility**, or **FontWeight** properties of the **TextBlock** element, you would need to create more converter classes, and they would be tightly coupled with your XAML.
- The semantics of the converter class only stipulate that it should convert an object into another object. This provides a lot of flexibility, but it also makes converters relatively fragile.

See Also

The C1Binding Class

The C1CalcEngine Class

The C1Binding Class

The C1Binding class addresses these issues by allowing you to use rich expressions in your bindings. Using **C1Binding**, the example above could be written as:

XAML	Copy Code

```
<TextBlock
   Text="{c1:C1BindingExpression='concatenate(FirstName, | |, LastName)' }"
   Foreground="{c1:C1BindingExpression='if(Amount &lt; 0, |red|, |black|)' }"
/>
```

Notice how much simpler and more direct this version is. There are no resources and no converters. The meaning of the bindings is clear from the XAML.

If the XAML author decided that the full name should be displayed as "Last, First" instead, he could make the change directly in the XAML. If he wanted negative amounts to be shown in bold, that would be easy too:

XAML	Copy Code

```
<TextBlock
   Text="{c1:C1BindingExpression='concatenate(FirstName, | |, LastName)' }"
   Foreground="{c1:C1BindingExpression='if(Amount &lt; 0, |red|, |black|)' }"
/>
   FontWeight="{c1:C1BindingExpression='if(Amount &lt; 0, |bold|, |normal|)'
}" />
```

In the expressions above, the vertical bars represent quotes. They could also be written as ***&quot;***, but the vertical bars are easier to type and read.

The "**Expression=**" term in the expressions above is optional in WPF. It is required in Silverlight 5, currently in beta. It may not be required by the time Silverlight 5 ships.

See Also

Using C1Binding Expression Syntax

C1Binding Usage Scenarios

C1Binding Limitations

Using C1Binding Expression Syntax

C1Binding objects use the C1CalcEngine class to parse and evaluate expressions. The syntax supported is similar to the one used in Microsoft Excel®.

You can use all the usual logical operators (=, >, <,.<>, >=, <=), arithmetic (+, -, *, /, ^), and you can group expressions using parentheses. For example:

"1+2*3 < (1+2)*3" returns TRUE (7 < 9).

Any public properties in the binding source object can be used in expressions as well. For example:

"Price * 8.5%" returns the value of the **Price** property times 0.085.

Properties that return collections or dictionaries are also supported. For example, assuming the **DataContext** object has a **Children** property:

"Children(Children.Count-1).Name" returns the name of the last child.

Finally, the **C1CalcEngine** supports a subset of the functions available in Excel. See C1Binding Expression Syntax Elements for a list of these functions.

C1Binding Usage Scenarios

Simple Arithmetic:

Suppose you want to display taxes due on a certain amount. Using traditional bindings, you would need a converter to calculate the tax amount. Using C1Binding, you can calculate the amount using a simple expression:

XAML	Copy Code

```
<TextBlock
```

```
   Text="{c1:C1BindingExpression='Amount * 8.5%' }"
```

Combining Values:

Suppose you want to extend the scenario above and display the total amount in addition to the tax amount. You could do that using several **TextBlock** elements, but it would be more efficient to use a single **TextBlock** and a **C1Binding** based on the CONCATENATE function:

XAML	Copy Code

```
<TextBlock
   Text="{c1:C1BindingExpression=
     'CONCATENATE(Amount, | Tax: |, Amount * 8.5%' }"
```

Formatting Values:

C1Binding and regular **Binding** objects have a **StringFormat** property that you can use to control the format of bound values. If your binding expression requires formatting several parts of the output, you can use the TEXT function as follows:

XAML	Copy Code

```
<TextBlock
   Text="{c1:C1BindingExpression=
     'CONCATENATE(TEXT(Amount, |c|), | Tax: |, TEXT(Amount * 8.5%, |c|))' }"
```

Conditional Formatting:

Suppose you want to create a user interface where amounts greater than a certain value are displayed in bold and red. You can accomplish this using the IF function, which performs conditional logic:

XAML	Copy Code

```
<TextBlock
   Text="{c1:C1BindingExpression='Amount', StringFormat='c'}"
   Foreground="{c1:C1BindingExpression='if(Amount &gt; 1000, |red|, |black|)'
}" />
```

```
   FontWeight="{c1:C1BindingExpression='if(Amount &gt; 1000, |bold|,
|normal|)' }" />
```

C1Binding Limitations

C1Binding objects can only be used in one-way binding scenarios.

The reason for this is simple: regular bindings are based on simple properties, which can be evaluated or assigned values. **C1Binding**, however, is based on expressions, which can be evaluated but not assigned to. For example, you can assign a new value to the "Amount" property, but not to the "Credit – Debit" expression.

In addition to this, **C1Binding** objects have the same limitations as regular **Binding** objects:

- They can only be assigned to dependency properties, which only exist in dependency objects (such as **UIElement** or **FrameworkElement** objects).
- If the binding is to be automatically updated when the property changes, then the binding source object must implement the **INotifyPropertyChanged** interface.
- If the types of the source and target properties are not compatible, then the target property must have the appropriate converters (e.g., most types can be converted from string values, including **Color**, **FontWeight**, and so on).

The C1CalcEngine Class

The C1CalcEngine class is responsible for parsing and evaluating the expressions used in C1Binding objects, but it is a public class and can also be used independently of **C1Binding**.

The **C1CalcEngine** class has two main methods: **Parse** and **Evaluate**:

1. **Parse** parses strings into **Expression** objects.
2. **Evaluate** parses strings and evaluates the resulting expression.

See Also

C1CalcEngine Usage Scenarios

C1CalcEngine Usage Scenarios

Calculated TextBox:

Suppose you want to add a calculated **TextBox** to your application. Users should be able to type expressions into the **TextBox**. The expressions should be evaluated when the control loses focus or when the user presses Enter. You could do this with the following code:

XAML	Copy Code

```
publicMainWindow()
 {
   InitializeComponent();
   var tb = new TextBox();
   tb.LostFocus += (s,e) => { Evaluate(s); };
   tb.KeyDown += (s, e) => { Evaluate(s); };
 }
 void Evaluate(object sender)
 {
   var tb = sender as TextBox;
   var ce = new C1.WPF.Binding.C1CalcEngine();
   try
   {
     var value = ce.Evaluate(tb.Text);
     tb.Text = value.ToString();
   }
   catch (Exception x)
   {
     MessageBox.Show("Error in Expression: " + x.Message);
   }
 }
```

Turning Grids into Spreadsheets:
Another typical use for the C1CalcEngine class is the implementation of calculated cells in grid controls and data objects. We have used it to implement several samples that add spreadsheet functionality to grid controls. The code is too long to include here, but you can see one of the samples in action here: http://demo.componentone.com/Silverlight/ExcelBook/.

C1Binding Expression Syntax Elements

The C1Binding class uses the C1CalcEngine class to parse and evaluate expressions. The syntax elements described below apply to both classes.

Case

All expressions are case-insensitive.

Operators

Expressions may contain the following operators:

Operator Type	**Operator Symbols**
Comparison	< > = <= >=
Addition/Subtraction	+ -
Multiplication/Division	* /
Power	^
Grouping	( ) , .

The operators follow the usual precedence rules, so "1 + 2 * 3" equals 7. Expressions may be grouped using parentheses, so "(1 + 2) * 3" equals 9.

Functions

Expressions may include calls to the following functions:

Logical Functions

Name	**Description**	**Syntax**
AND	Returns TRUE if all of its arguments are TRUE	=AND(logical1[, logical2,...])
FALSE	Returns the logical value FALSE	=FALSE
IF	Specifies a logical test to perform	=IF(logical_test, value_if_true, value_if_false)
NOT	Reverses the logic of its argument	=NOT(logical)
OR	Returns TRUE if any argument is TRUE	=OR(logical1[, logical2,...])
TRUE	Returns the logical value TRUE	=TRUE

Mathematical Functions

Name	**Description**	**Syntax**
ABS	Returns the absolute value of a number	=ABS(number)
ACOS	Returns the arccosine of a number	=ACOS(number)
ASIN	Returns the arcsine of a number	=ASIN(number)
ATAN	Returns the arctangent of a number	=ATAN(number)
ATAN2	Returns the arctangent from x- and y-coordinates	=ATAN2(x_num, y_num)
CEILING	Rounds a number to the nearest integer	=CEILING(number)

	or to the nearest multiple of significance	
COS	Returns the cosine of a number	=COS(number)
COSH	Returns the hyperbolic cosine of a number	=COSH(number)
EXP	Returns e raised to the power of a given number	=EXP(number)
FLOOR	Rounds a number down, toward zero	=FLOOR(number)
INT	Rounds a number down to the nearest integer	=INT(number)
LN	Returns the natural logarithm of a number	=LN(number)
LOG	Returns the logarithm of a number to a specified base	=LOG(number[, base])
LOG10	Returns the base-10 logarithm of a number	=LOG10(number)
PI	Returns the value of the PI constant	=PI()
POWER	Returns the result of a number raised to a power	=POWER(number, power)
RAND	Returns a random number between 0 and 1	=RAND()
RANDBETWEEN	Returns a random number between the numbers you specify	=RANDBETWEEN(bottom, top)
SIGN	Returns the sign of a number	=SIGN(number)
SIN	Returns the sine of the given angle	=SIN(number)

SINH	Returns the hyperbolic sine of a number	=SINH(number)
SQRT	Returns a positive square root	=SQRT(number)
SUM	Adds its arguments	=SUM(number1[, number2, ...])
TAN	Returns the tangent of a number	=TAN(number)
TANH	Returns the hyperbolic tangent of a number	=TANH(number)
TRUNC	Truncates a number to an integer	=TRUNC(number)";

Statistical Functions

Name	**Description**	**Syntax**
AVERAGE	Returns the average of its arguments	
AVERAGEA	Returns the average of its arguments, including numbers, text, and logical values	=AVERAGE(number1 [, number2, ...])
COUNT	Counts how many numbers are in the list of arguments	=AVERAGEA(number1 [, number2, ...])
COUNTA	Counts how many values are in the list of arguments	=COUNT(number1 [, number2, ...])
COUNTBLANK	Counts the number of blank cells within a range	=COUNTA(number1 [, number2, ...])
COUNTIF	Counts the number of cells within a range that meet the given criteria	=COUNTIF(range, criteria)
MAX	Returns the maximum value in a list of	=MAX(number1 [, number2, ...])

	arguments	
MAXA	Returns the maximum value in a list of arguments, including numbers, text, and logical values	=MAXA(number1 [, number2, ...])
MIN	Returns the minimum value in a list of arguments	=MIN(number1 [, number2, ...])
MINA	Returns the smallest value in a list of arguments, including numbers, text, and logical values	=MINA(number1 [, number2, ...])
STDEV	Estimates standard deviation based on a sample	=STDEV(number1 [, number2, ...])
STDEVA	Estimates standard deviation based on a sample, including numbers, text, and logical values	=STDEVA(number1 [, number2, ...])
STDEVP	Calculates standard deviation based on the entire population	=STDEVP(number1 [, number2, ...])
STDEVPA	Calculates standard deviation based on the entire population, including numbers, text, and logical values	=STDEVPA(number1 [, number2, ...])
VAR	Estimates variance based on a sample	=VAR(number1 [, number2, ...])
VARA	Estimates variance based on a sample, including numbers, text, and logical values	=VARA(number1 [, number2, ...])
VARP	Calculates variance based on the entire population	=VARP(number1 [, number2, ...])
VARPA	Calculates variance based on the entire population, including numbers, text,	=VARPA(number1 [, number2, ...])

	and logical values	

Text Functions

Name	**Description**	**Syntax**
CODE	Returns a numeric code for the first character in a text string	=CODE(text)
CONCATENATE	Joins several text items into one text item	=CONCATENATE(text1 [, text2, …])
FIND	Finds one text value within another (case-sensitive)	=FIND(find_text, within_text [, start_num])
LEFT	Returns the leftmost characters from a text value	=LEFT(text[, num_chars])
LEN	Returns the number of characters in a text string	=LEN(text)
LOWER	Converts text to lowercase	=LOWER(text)
MID	Returns a specific number of characters from a text string starting at the position you specify	=MID(text, start_num, num_chars)
PROPER	Capitalizes the first letter in each word of a text value	=PROPER(text)
REPLACE	Replaces characters within text	=REPLACE(old_text, stat_num, num_chars, new_text)
REPT	Repeats text a given number of times	=REPT(trext, number_times)
RIGHT	Returns the rightmost characters	=RIGHT(text[, num_chars])

	from a text value	
SEARCH	Finds one text value within another (not case-sensitive)	=SEARCH(find_text, within_text[, start_num])
SUBSTITUTE	Substitutes new text for old text in a text string	=SUBSTITUTE(text, old_text, new_text[, instance_num])
T	Converts its arguments to text	=T(value)
TEXT	Formats a number and converts it to text	=TEXT(value, format_text)
TRIM	Removes spaces from text	=TRIM(text)
UPPER	Converts text to uppercase	=UPPER(text)
VALUE	Converts a text argument to a number	=VALUE(text)

C1Binding Sample

The **C1BindingDemo** sample installed with the product shows the differences between regular binding and C1Binding as well as template binding for each.

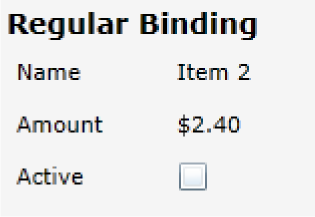
For example, here is the XAML used to perform regular binding:

XAML	Copy Code

```
<!-- regular binding -->
        <BorderGrid.Row="1"Background="WhiteSmoke"Padding="8" >
            <StackPanelOrientation="Vertical" >
```

```
                <TextBlockText="Regular
Binding"Style="{StaticResource_styTitle}" />
                <StackPanelOrientation="Horizontal"Margin="5" >
                    <TextBlockText="Name"Width="80" />
                    <TextBlockText="{BindingName}" />
                </StackPanel>
                <StackPanelOrientation="Horizontal"Margin="5" >
                    <TextBlockText="Amount"Width="80" />
                    <TextBlockText="{BindingAmount,StringFormat=c}" />
                </StackPanel>
                <StackPanelOrientation="Horizontal"Margin="5" >
                    <TextBlockText="Active"Width="80" />
                    <CheckBoxIsChecked="{BindingActive}" />
                </StackPanel>
            </StackPanel>
        </Border>
```

When the project runs, the regular binding looks like this:



Regular Binding

Name Item 2

Amount $2.40

Active ☐

We can compare this to the XAML used to perform **C1Binding**:

XAML Copy Code

```
<!-- C1Binding -->
        <BorderGrid.Row="2"Background="Yellow"Padding="8" >
            <StackPanelOrientation="Vertical" >
                <TextBlockText="C1Binding"Style="{StaticResource_styTitle}"
/>
                <StackPanelOrientation="Horizontal"Margin="5" >
                    <TextBlockText="Name"Width="80" />
                    <TextBlock
                        Text="{c1:C1BindingExpression=Name}"
                        FontWeight="{c1:C1BindingExpression='if(Active,
|Bold|, |Normal|)'}"
                        Foreground="{c1:C1BindingExpression='if(Active,
|Blue|, |Red|)'}"
                        />
                </StackPanel>
                <StackPanelOrientation="Horizontal"Margin="5" >
                    <TextBlockText="Amount"Width="80" />
                    <TextBlock

Text="{c1:C1BindingExpression='concatenate(text(Amount, |c|), | (tax: |,
text(Amount * 8.5%, |c|), |)|)',StringFormat=c}"
                        FontWeight="{c1:C1BindingExpression='if(Active,
|Bold|, |Normal|)'}"
                        />
                </StackPanel>
                <StackPanelOrientation="Horizontal"Margin="5" >
                    <TextBlockText="Active"Width="80" />
                    <CheckBoxIsChecked="{c1:C1BindingExpression=Active}" />
                </StackPanel>
            </StackPanel>
        </Border>
```

In this example, **C1Binding** expressions are used to make the binding a little more interesting. In this XAML, conditional formatting is used to display the **FontWeight** as **Bold** if the item is active or as **Normal** if not. The **ForeGround** is **Blue** if the item is active or **Red** if not.

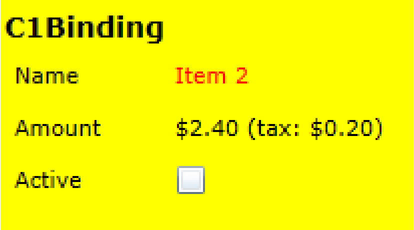
XAML	Copy Code

```

<TextBlock
                        Text="{c1:C1BindingExpression=Name}"
                        FontWeight="{c1:C1BindingExpression='if(Active,
|Bold|, |Normal|)'}"
                        Foreground="{c1:C1BindingExpression='if(Active,
|Blue|, |Red|)'}"
                        />
```

Then the amount is shown with the tax, which has been calculated using a multiplication operator, using the CONCATENATE function.

XAML	Copy Code

```
<TextBlockText="Amount"Width="80" />
                    <TextBlock

Text="{c1:C1BindingExpression='concatenate(text(Amount, |c|), | (tax: |,
text(Amount * 8.5%, |c|), |)|)',StringFormat=c}"
                        FontWeight="{c1:C1BindingExpression='if(Active,
|Bold|, |Normal|)'}"
                        />
```


C1Binding

Name Item 2

Amount $2.40 (tax: $0.20)

Active □

The template binding is very similar, however, items are listed in a **<ListBox.ItemTemplate>**. The same **C1Binding** expressions are used as in the **C1Binding** XAML.

To get the regular binding to look and perform like the **C1Binding**, many **Converter** parameters would need to be used. **C1Binding** keeps the XAML clean and simple.

Reference

C1.Silverlight.Binding.5 Assembly

Namespaces

C1.Silverlight.Binding Namespace

Overview

Classes

	Class	Description
	C1Binding	A C1.Silverlight.Binding that supports formulas.
	C1CalcEngine	Class that parses strings and returns Expression objects that can be evaluated.
	C1StaticResource	A **StaticResource** that supports a path specification.
	Expression	Base class that represents parsed expressions.
	FunctionDefinition	Function definition class (keeps function name, parameter counts, and delegate).
	PropertyChangedEventManager	Provides an implementation so that you can use the "weak event listener" pattern to attach listeners for the **PropertyChanged** event.
	Tally	A class that aggregates values and provides summary statistics (including count, sum, average, standard deviation, and variance).
	XObjectExpression	Expression that represents an external object.

Interfaces

	Interface	Description
	IValueObject	Interface supported by external objects that have to return a value other than themselves (e.g. a cell range object should return the cell content instead of the range itself).
	IWeakEventListener	Provides event listening support for classes that expect to receive events through the WeakEvent pattern and a WeakEventManager.

Delegates

	Delegate	Description
	CalcEngineFunction	Delegate that represents CalcEngine functions.

See Also

Reference

C1.Silverlight.Binding.5 Assembly

Classes

C1Binding

Example

A C1.Silverlight.Binding that supports formulas.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1Binding
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1Binding : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1Binding** class is similar to a regular C1.Silverlight.Binding, but supports expressions instead of simple paths.

The expression syntax is similar to the one used in Excel formulas, including support for the IF statement which makes it easy to create bindings that make **Converters** unnecessary.

Example

The XAML below uses the **C1Binding** class to bind several properties of a **TextBlock** element to a data value. Notice that the bindings are used to provide values for properties of several types, and no converters are required. Notice how the XAML snippet above specifies the 'Expression' property. This is required only in Silverlight, and could be omitted in WPF. Notice also how vertical bars ('|') are used to specify quotes inside expressions. You could also use &quot; instead, but the bars made the code more readable.

- C#

```
<TextBlock
    Text="{c1:C1Binding Expression=CustomerName}"
    Visibility="{c1:C1Binding Expression='if(Active, |Visible|,
|Collapsed|)'}"
    FontWeight="{c1:C1Binding Expression='if(Sales > 200, |Bold|,
|Normal|)'}"
    Foreground="{c1:C1Binding Expression='if(Sales > 200, |Blue|, |Red|)'}"
/>
```

Inheritance Hierarchy

System.Object
  System.Windows.Markup.MarkupExtension
    C1.Silverlight.Binding.C1Binding

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Members
C1.Silverlight.Binding Namespace

Overview

Example

A C1.Silverlight.Binding that supports formulas.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1Binding
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1Binding : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1Binding** class is similar to a regular C1.Silverlight.Binding, but supports expressions instead of simple paths.

The expression syntax is similar to the one used in Excel formulas, including support for the IF statement which makes it easy to create bindings that make **Converters** unnecessary.

Example

The XAML below uses the **C1Binding** class to bind several properties of a **TextBlock** element to a data value. Notice that the bindings are used to provide values for properties of several types, and no converters are required. Notice how the XAML snippet above specifies the 'Expression' property. This is required only in Silverlight, and could be omitted in WPF. Notice also how vertical bars ('|') are used to specify quotes inside expressions. You could also use &quot; instead, but the bars made the code more readable.

- C#

```
<TextBlock
    Text="{c1:C1Binding Expression=CustomerName}"
    Visibility="{c1:C1Binding Expression='if(Active, |Visible|,
|Collapsed|)'}"
    FontWeight="{c1:C1Binding Expression='if(Sales > 200, |Bold|,
|Normal|)'}"
    Foreground="{c1:C1Binding Expression='if(Sales > 200, |Blue|, |Red|)'}"
/>
```

Inheritance Hierarchy

System.Object
System.Windows.Markup.MarkupExtension
C1.Silverlight.Binding.C1Binding

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Members
C1.Silverlight.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by C1Binding.

Public Constructors

	Name	Description
≡◆	C1Binding Constructor	Overloaded.

Top

Public Properties

	Name	Description
▤	Expression	Gets or sets the expression used to calculate the binding value.
▤	FallbackValue	Gets or sets the value to use when the binding is unable to return a value.
▤	StringFormat	Gets or sets a string that specifies how to format the binding value.
▤	TargetNullValue	Gets or sets the value that is used in the target when the value of the source is null.

Top

Public Methods

	Name	Description
	ProvideValue	Overridden. Returns an object that is provided as the value of the target property for this markup extension.

Top

See Also

Reference

C1Binding Class
C1.Silverlight.Binding Namespace

C1Binding Constructor

Overload List

Overload	Description
C1Binding Constructor(String)	Initializes a new instance of a C1Binding.
C1Binding Constructor()	Initializes a new instance of a C1Binding.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

C1Binding Constructor(String)

Binding expression.

Initializes a new instance of a C1Binding.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal expression As System.String _
)
```

C#

```
public C1Binding(
   System.string expression
)
```

Parameters

expression

Binding expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members
Overload List

C1Binding Constructor()

Initializes a new instance of a C1Binding.

Syntax

Visual Basic (Declaration)

```
Public Function New()
```

C#

```
public C1Binding()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members
Overload List

Methods

%%scrap%%

" -->

For a list of all members of this type, see C1Binding members.

Public Methods

	Name	Description
≡◆	ProvideValue	Overridden. Returns an object that is provided as the value of the target property for this markup extension.

Top

See Also

Reference

C1Binding Class
C1.Silverlight.Binding Namespace

ProvideValue Method

A service provider helper that can provide services for the markup extension.

Returns an object that is provided as the value of the target property for this markup extension.

Syntax

Visual Basic (Declaration)	

```
Public Overrides Function ProvideValue( _
   ByVal serviceProvider As System.IServiceProvider _
) As System.Object
```

C#

```
public override System.object ProvideValue(
   System.IServiceProvider serviceProvider
)
```

Parameters

serviceProvider

A service provider helper that can provide services for the markup extension.

Return Value

The object value to set on the property where the extension is applied.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

Properties

%%scrap%%

" -->

For a list of all members of this type, see C1Binding members.

Public Properties

	Name	Description
	Expression	Gets or sets the expression used to calculate the binding value.
	FallbackValue	Gets or sets the value to use when the binding is unable to return a value.

▤	StringFormat	Gets or sets a string that specifies how to format the binding value.
▤	TargetNullValue	Gets or sets the value that is used in the target when the value of the source is null.

Top

See Also

Reference

C1Binding Class
C1.Silverlight.Binding Namespace

Expression Property

Gets or sets the expression used to calculate the binding value.

Syntax

Visual Basic (Declaration)

```
Public Property Expression As System.String
```

C#

```
public System.string Expression {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

FallbackValue Property

Gets or sets the value to use when the binding is unable to return a value.

Syntax

Visual Basic (Declaration)

```
Public Property FallbackValue As System.Object
```

C#

```
public System.object FallbackValue {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

StringFormat Property

Gets or sets a string that specifies how to format the binding value.

Syntax

Visual Basic (Declaration)

```
Public Property StringFormat As System.String
```

C#

```
public System.string StringFormat {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

TargetNullValue Property

Gets or sets the value that is used in the target when the value of the source is null.

Syntax

Visual Basic (Declaration)

```
Public Property TargetNullValue As System.Object
```

C#

```
public System.object TargetNullValue {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

C1CalcEngine

Class that parses strings and returns Expression objects that can be evaluated.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1CalcEngine
```

C#

```
public class C1CalcEngine
```

Remarks

This class has three extensibility points:

Use the **DataContext** property to add an object's properties to the engine scope.

Use the **RegisterFunction** method to define custom functions.

Override the **GetExternalObject** method to add arbitrary variables to the engine scope.

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.C1CalcEngine

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Members
C1.Silverlight.Binding Namespace

Overview

Class that parses strings and returns Expression objects that can be evaluated.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1CalcEngine
```

C#

```
public class C1CalcEngine
```

Remarks

This class has three extensibility points:

Use the **DataContext** property to add an object's properties to the engine scope.

Use the **RegisterFunction** method to define custom functions.

Override the **GetExternalObject** method to add arbitrary variables to the engine scope.

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.C1CalcEngine

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Members
C1.Silverlight.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by C1CalcEngine.

Public Constructors

	Name	Description
	C1CalcEngine Constructor	Initializes a new instance of a C1CalcEngine.

Top

Public Properties

	Name	Description
	CacheExpressions	Gets or sets whether the calc engine should keep a cache with parsed expressions.
	CultureInfo	Gets or sets the CultureInfo to use when parsing numbers and dates.
	DataContext	Gets or sets the DataContext for this C1CalcEngine.

	Functions	Gets the dictionary that contains function definitions.
	IdentifierChars	Gets or sets a string that specifies special characters that are valid for identifiers.
	OptimizeExpressions	Gets or sets whether the calc engine should optimize expressions when they are parsed.
	Variables	Gets the dictionary that contains simple variables (not in the DataContext).

Top

Public Methods

	Name	Description
	Evaluate	Evaluates a string.
	GetExternalObject	Gets an external object based on an identifier.
	Parse	Parses a string into an Expression.
	RegisterFunction	Overloaded. Registers a function that can be evaluated by this C1CalcEngine.

Top

See Also

Reference

C1CalcEngine Class
C1.Silverlight.Binding Namespace

C1CalcEngine Constructor

Initializes a new instance of a C1CalcEngine.

Syntax

Visual Basic (Declaration)	

```
Public Function New()
```

C#

```
public C1CalcEngine()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Methods

%%scrap%%

" -->

For a list of all members of this type, see C1CalcEngine members.

Public Methods

	Name	Description
≡◆	Evaluate	Evaluates a string.
≡◆	GetExternalObject	Gets an external object based on an identifier.
≡◆	Parse	Parses a string into an Expression.
≡◆	RegisterFunction	Overloaded. Registers a function that can be evaluated by this C1CalcEngine.

Top

See Also

Reference

C1CalcEngine Class
C1.Silverlight.Binding Namespace

Evaluate Method

Expression to evaluate.

Evaluates a string.

Syntax

Visual Basic (Declaration)

```
Public Function Evaluate( _
   ByVal expression As System.String _
) As System.Object
```

C#

```
public System.object Evaluate(
   System.string expression
)
```

Parameters

expression

Expression to evaluate.

Return Value

The value of the expression.

Remarks

If you are going to evaluate the same expression several times, it is more efficient to parse it only once using the Parse method and then using the Expression.Evaluate method to evaluate the parsed expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

GetExternalObject Method

Gets an external object based on an identifier.

Syntax

Visual Basic (Declaration)

```
Public Overridable Function GetExternalObject( _
   ByVal identifier As System.String _
) As System.Object
```

C#

```
public virtual System.object GetExternalObject(
   System.string identifier
)
```

Parameters

identifier

Remarks

This method is useful when the engine needs to create objects dynamically. For example, a spreadsheet calc engine would use this method to dynamically create cell range objects based on identifiers that cannot be enumerated at design time (such as "AB12", "A1:AB12", etc.)

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Parse Method

String to parse.

Parses a string into an Expression.

Syntax

Visual Basic (Declaration)

```
Public Function Parse( _
   ByVal expression As System.String _
) As Expression
```

C#

```
public Expression Parse(
   System.string expression
)
```

Parameters

expression

String to parse.

Return Value

An Expression object that can be evaluated.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

RegisterFunction Method

Registers a function that can be evaluated by this C1CalcEngine.

Overload List

Overload	Description
RegisterFunction(String,Int32,Int32,CalcEngineFunction)	Registers a function that can be evaluated by this C1CalcEngine.

RegisterFunction(String,Int32,CalcEngineFunction)	Registers a function that can be evaluated by this C1CalcEngine.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

RegisterFunction(String,Int32,Int32,CalcEngineFunction) Method
Function name.

Minimum parameter count.

Maximum parameter count.

Delegate that evaluates the function.

Registers a function that can be evaluated by this C1CalcEngine.

Syntax

Visual Basic (Declaration)

```
Public Overloads Sub RegisterFunction( _
   ByVal functionName As System.String, _
   ByVal parmMin As System.Integer, _
   ByVal parmMax As System.Integer, _
   ByVal fn As CalcEngineFunction _
)
```

C#

```
public void RegisterFunction(
   System.string functionName,
   System.int parmMin,
   System.int parmMax,
   CalcEngineFunction fn
)
```

Parameters

functionName

Function name.

parmMin

Minimum parameter count.

parmMax

Maximum parameter count.

fn

Delegate that evaluates the function.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members
Overload List

RegisterFunction(String,Int32,CalcEngineFunction) Method
Function name.

Parameter count.

Delegate that evaluates the function.

Registers a function that can be evaluated by this C1CalcEngine.

Syntax

Visual Basic (Declaration)

```
Public Overloads Sub RegisterFunction( _
   ByVal functionName As System.String, _
   ByVal parmCount As System.Integer, _
   ByVal fn As CalcEngineFunction _
)
```

C#

```
public void RegisterFunction(
   System.string functionName,
   System.int parmCount,
   CalcEngineFunction fn
)
```

Parameters

functionName

Function name.

parmCount

Parameter count.

fn

Delegate that evaluates the function.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members
Overload List

Properties

%%scrap%%

" -->

For a list of all members of this type, see C1CalcEngine members.

Public Properties

	Name	Description
▤	CacheExpressions	Gets or sets whether the calc engine should keep a cache with parsed expressions.
▤	CultureInfo	Gets or sets the CultureInfo to use when parsing numbers and dates.
▤	DataContext	Gets or sets the DataContext for this C1CalcEngine.
▤	Functions	Gets the dictionary that contains function definitions.
▤	IdentifierChars	Gets or sets a string that specifies special characters that are valid for identifiers.
▤	OptimizeExpressions	Gets or sets whether the calc engine should optimize expressions when they are parsed.
▤	Variables	Gets the dictionary that contains simple variables (not in the DataContext).

Top

See Also

Reference

C1CalcEngine Class
C1.Silverlight.Binding Namespace

CacheExpressions Property

Gets or sets whether the calc engine should keep a cache with parsed expressions.

Syntax

Visual Basic (Declaration)

```
Public Property CacheExpressions As System.Boolean
```

C#

```
public System.bool CacheExpressions {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

CultureInfo Property

Gets or sets the CultureInfo to use when parsing numbers and dates.

Syntax

Visual Basic (Declaration)

```
Public Property CultureInfo As System.Globalization.CultureInfo
```

C#

```
public System.Globalization.CultureInfo CultureInfo {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

DataContext Property

Gets or sets the DataContext for this C1CalcEngine.

Syntax

Visual Basic (Declaration)

```
Public Overridable Property DataContext As System.Object
```

C#

```
public virtual System.object DataContext {get; set;}
```

Remarks

Once a DataContext is set, all public properties of the object become available to the CalcEngine, including sub-properties such as "Address.Street". These may be used with expressions just like any other constant.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Functions Property

Gets the dictionary that contains function definitions.

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property Functions As
System.Collections.Generic.Dictionary(Of String,FunctionDefinition)
```

C#

```
public System.Collections.Generic.Dictionary<string,FunctionDefinition>
Functions {get;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

IdentifierChars Property

Gets or sets a string that specifies special characters that are valid for identifiers.

Syntax

Visual Basic (Declaration)

```
Public Property IdentifierChars As System.String
```

C#

```
public System.string IdentifierChars {get; set;}
```

Remarks

Identifiers must start with a letter or an underscore, which may be followed by additional letters, underscores, or digits. This string allows you to specify additional valid characters such as ':' or '!' (used in Excel range references for example).

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

OptimizeExpressions Property

Gets or sets whether the calc engine should optimize expressions when they are parsed.

Syntax

Visual Basic (Declaration)

```
Public Property OptimizeExpressions As System.Boolean
```

C#

```
public System.bool OptimizeExpressions {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Variables Property

Gets the dictionary that contains simple variables (not in the DataContext).

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property Variables As
System.Collections.Generic.Dictionary(Of String,Object)
```

C#

```
public System.Collections.Generic.Dictionary<string,object> Variables {get;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

C1StaticResource

Example

A **StaticResource** that supports a path specification.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1StaticResource
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1StaticResource : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1StaticResource** class is similar to a regular **StaticResource** markup, but supports a path specification instead of simple objects.

Paths are specified using the same expression syntax used in the C1Binding class. This allows you to use object members as resources.

Example

The XAML below uses the **C1StaticResource** class to bind a grid column using a value converter that is implemented in a **ViewModel** object. The XAML snippet above assigns the **_model.JobsConverter** object to the **Converter** property of the binding. This cannot be done using a nested binding because the **Binding** class is not a dependency object.

- C#

```
<c1:C1FlexGrid
  ItemsSource="{Binding Source={StaticResource _model}, Path=People}"
  AutoGenerateColumns="False" >
  <c1:C1FlexGrid.Columns>
    <c1:Column
      Binding="{Binding JobID,
        Mode=TwoWay,
        Converter={c1:C1StaticResource Key=_model,Path=JobsConverter}}" />
  </c1:C1FlexGrid.Columns>
</c1:C1FlexGrid>
```

Inheritance Hierarchy

System.Object
  System.Windows.Markup.MarkupExtension
    C1.Silverlight.Binding.C1StaticResource

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Members
C1.Silverlight.Binding Namespace

Overview

Example

A **StaticResource** that supports a path specification.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1StaticResource
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1StaticResource : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1StaticResource** class is similar to a regular **StaticResource** markup, but supports a path specification instead of simple objects.

Paths are specified using the same expression syntax used in the C1Binding class. This allows you to use object members as resources.

Example

The XAML below uses the **C1StaticResource** class to bind a grid column using a value converter that is implemented in a **ViewModel** object. The XAML snippet above assigns the **_model.JobsConverter** object to the **Converter** property of the binding. This cannot be done using a nested binding because the **Binding** class is not a dependency object.

- C#

```
<c1:C1FlexGrid
  ItemsSource="{Binding Source={StaticResource _model}, Path=People}"
  AutoGenerateColumns="False" >
  <c1:C1FlexGrid.Columns>
    <c1:Column
      Binding="{Binding JobID,
        Mode=TwoWay,
        Converter={c1:C1StaticResource Key=_model,Path=JobsConverter}}" />
  </c1:C1FlexGrid.Columns>
</c1:C1FlexGrid>
```

Inheritance Hierarchy

System.Object
  System.Windows.Markup.MarkupExtension
    C1.Silverlight.Binding.C1StaticResource

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Members
C1.Silverlight.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by C1StaticResource.

Public Constructors

	Name	Description
≡◆	C1StaticResource Constructor	Overloaded.

Top

Public Properties

	Name	Description

	Key	Key that identifies the resource.
	Path	Expression that defines the data to extract from the resource.

Top

Public Methods

	Name	Description
	ProvideValue	Overridden. Returns an object that is provided as the value of this markup extension.

Top

See Also

Reference

C1StaticResource Class
C1.Silverlight.Binding Namespace

C1StaticResource Constructor

Overload List

Overload	Description
C1StaticResource Constructor()	Initializes a new instance of a C1StaticResource.
C1StaticResource Constructor(String)	Initializes a new instance of a C1StaticResource.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

C1StaticResource Constructor()

Initializes a new instance of a C1StaticResource.

Syntax

Visual Basic (Declaration)

```
Public Function New()
```

C#

```
public C1StaticResource()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members
Overload List

C1StaticResource Constructor(String)

Key that identifies the resource.

Initializes a new instance of a C1StaticResource.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal resourceKey As System.String _
)
```

C#

```
public C1StaticResource(
   System.string resourceKey
)
```

Parameters

resourceKey

Key that identifies the resource.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members
Overload List

Methods

%%scrap%%

" -->

For a list of all members of this type, see C1StaticResource members.

Public Methods

	Name	Description
≡◆	ProvideValue	Overridden. Returns an object that is provided as the value of this markup extension.

Top

See Also

Reference

C1StaticResource Class
C1.Silverlight.Binding Namespace

ProvideValue Method

A service provider helper that can provide services for the markup extension.

Returns an object that is provided as the value of this markup extension.

Syntax

Visual Basic (Declaration)

```
Public Overrides Function ProvideValue( _
   ByVal serviceProvider As System.IServiceProvider _
) As System.Object
```

C#

```
public override System.object ProvideValue(
   System.IServiceProvider serviceProvider
)
```

Parameters

serviceProvider

A service provider helper that can provide services for the markup extension.

Return Value

The object value to set on the property where the extension is applied.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

Properties

%%scrap%%

" -->

For a list of all members of this type, see C1StaticResource members.

Public Properties

	Name	Description
▤	Key	Key that identifies the resource.

	Path	Expression that defines the data to extract from the resource.

Top

See Also

Reference

C1StaticResource Class
C1.Silverlight.Binding Namespace

Key Property

Key that identifies the resource.

Syntax

Visual Basic (Declaration)

```
Public Property Key As System.String
```

C#

```
public System.string Key {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

Path Property

Expression that defines the data to extract from the resource.

Syntax

Visual Basic (Declaration)

```
Public Property Path As System.String
```

C#

```
public System.string Path {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

Expression

Base class that represents parsed expressions.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Expression
```

C#

```
public class Expression
```

Remarks

For example: `Expression expr = scriptEngine.Parse(strExpression); object val = expr.Evaluate();`

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.Expression
    C1.Silverlight.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Members
C1.Silverlight.Binding Namespace

Overview

Base class that represents parsed expressions.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Expression
```

C#

```
public class Expression
```

Remarks

For example: `Expression expr = scriptEngine.Parse(strExpression); object val = expr.Evaluate();`

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.Expression
    C1.Silverlight.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Members
C1.Silverlight.Binding Namespace

Members

Methods

The following tables list the members exposed by Expression.

Public Methods

	Name	Description
◆	CompareTo	Compares the value of this Expression to the value of another Expression.
◆	Evaluate	Returns the value of the Expression.
◆	Optimize	Returns an equal or simpler (but equivalent) version of the Expression.

Top

Public Operators

Implicit Type Conversion	Overloaded. Evaluates the expression and returns the result converted to a string.

Top

See Also

Reference

Expression Class
C1.Silverlight.Binding Namespace

Methods

%%scrap%%

" -->

For a list of all members of this type, see Expression members.

Public Methods

	Name	Description
◆	CompareTo	Compares the value of this Expression to the value of another Expression.
◆	Evaluate	Returns the value of the Expression.
◆	Optimize	Returns an equal or simpler (but equivalent) version of the Expression.

Top

See Also

Reference

Expression Class
C1.Silverlight.Binding Namespace

CompareTo Method

Expression to compare to.

Compares the value of this Expression to the value of another Expression.

Syntax

Visual Basic (Declaration)

```
Public Function CompareTo( _
   ByVal other As Expression _
) As System.Integer
```

C#

```
public System.int CompareTo(
   Expression other
)
```

Parameters

other

Expression to compare to.

Return Value

An integer that indicates whether the value of this expression is smaller than, equal, or greater than the value of the other expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

Evaluate Method

Returns the value of the Expression.

Syntax

Visual Basic (Declaration)

```
Public Overridable Function Evaluate() As System.Object
```

C#

```
public virtual System.object Evaluate()
```

Return Value

The value of the Expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

Optimize Method

Returns an equal or simpler (but equivalent) version of the Expression.

Syntax

Visual Basic (Declaration)

```
Public Overridable Function Optimize() As Expression
```

C#

```
public virtual Expression Optimize()
```

Return Value

An equal or simpler but equivalent version of the expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

Operators

Implicit Type Conversion Operator

Evaluates the expression and returns the result converted to a string.

Overload List

Overload	Description
Implicit Type Conversion(String,Expression)	Evaluates the expression and returns the result converted to a string.
Implicit Type Conversion(Double,Expression)	Evaluates the expression and returns the result converted to a double.
Implicit Type Conversion(Boolean,Expression)	Evaluates the expression and returns the result converted to a Boolean value.
Implicit Type Conversion(DateTime,Expression)	Evaluates the expression and returns the result

	converted to a **System.DateTime** value.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

FunctionDefinition

Function definition class (keeps function name, parameter counts, and delegate).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class FunctionDefinition
```

C#

```
public class FunctionDefinition
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.FunctionDefinition

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Members
C1.Silverlight.Binding Namespace

Overview

Function definition class (keeps function name, parameter counts, and delegate).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class FunctionDefinition
```

C#

```
public class FunctionDefinition
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.FunctionDefinition

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Members
C1.Silverlight.Binding Namespace

Members

Fields

The following tables list the members exposed by FunctionDefinition.

Public Constructors

	Name	Description

≡◆	FunctionDefinition Constructor	Initializes a new instance of a FunctionDefinition.

Top

Public Fields

	Name	Description
◆	Function	Gets or sets the CalcEngineFunction that evaluates this function.
◆	ParmMax	Gets or sets the maximum number of parameters allowed by this FunctionDefinition.
◆	ParmMin	Gets or sets the minimum number of parameters required by this FunctionDefinition.

Top

See Also

Reference

FunctionDefinition Class
C1.Silverlight.Binding Namespace

FunctionDefinition Constructor

The minimum number of parameters required by this function.

The maximum number of parameters allowed by this function.

The CalcEngineFunction delegate that evaluates this function

Initializes a new instance of a FunctionDefinition.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal parmMin As System.Integer, _
   ByVal parmMax As System.Integer, _
   ByVal function As CalcEngineFunction _
)
```

C#

```
public FunctionDefinition(
   System.int parmMin,
   System.int parmMax,
   CalcEngineFunction function
)
```

Parameters

parmMin

The minimum number of parameters required by this function.

parmMax

The maximum number of parameters allowed by this function.

function

The CalcEngineFunction delegate that evaluates this function

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

Fields

>

Name	Description
◆ Function	Gets or sets the CalcEngineFunction that evaluates this function.
◆ ParmMax	Gets or sets the maximum number of parameters allowed by this FunctionDefinition.
◆ ParmMin	Gets or sets the minimum number of parameters required by this FunctionDefinition.

Top

See Also

Reference

FunctionDefinition Class
C1.Silverlight.Binding Namespace

Function Field

Gets or sets the CalcEngineFunction that evaluates this function.

Syntax

Visual Basic (Declaration)

```
Public Function As CalcEngineFunction
```

C#

```
public CalcEngineFunction Function
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

ParmMax Field

Gets or sets the maximum number of parameters allowed by this FunctionDefinition.

Syntax

Visual Basic (Declaration)

```
Public ParmMax As System.Integer
```

C#

```
public System.int ParmMax
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

ParmMin Field

Gets or sets the minimum number of parameters required by this FunctionDefinition.

Syntax

Visual Basic (Declaration)

```
Public ParmMin As System.Integer
```

C#

```
public System.int ParmMin
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

PropertyChangedEventManager

Provides an implementation so that you can use the "weak event listener" pattern to attach listeners for the **PropertyChanged** event.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class PropertyChangedEventManager
```

C#

```
public class PropertyChangedEventManager
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.PropertyChangedEventManager

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PropertyChangedEventManager Members
C1.Silverlight.Binding Namespace

Overview

Provides an implementation so that you can use the "weak event listener" pattern to attach listeners for the **PropertyChanged** event.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class PropertyChangedEventManager
```

C#

```
public class PropertyChangedEventManager
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.PropertyChangedEventManager

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PropertyChangedEventManager Members
C1.Silverlight.Binding Namespace

Members

Methods

The following tables list the members exposed by PropertyChangedEventManager.

Public Constructors

	Name	Description
	PropertyChangedEventManager Constructor	

Top

Public Methods

	Name	Description
S	AddListener	Adds the specified listener to the list of listeners on the specified source.
S	RemoveListener	Removes the specified listener from the list of listeners on the specified source.

Top

See Also

Reference

PropertyChangedEventManager Class
C1.Silverlight.Binding Namespace

PropertyChangedEventManager Constructor

Syntax

Visual Basic (Declaration)

```
Public Function New()
```

C#

```
public PropertyChangedEventManager()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PropertyChangedEventManager Class
PropertyChangedEventManager Members

Methods

>

Name	Description
≡◆ S AddListener	Adds the specified listener to the list of listeners on the specified source.
≡◆ S RemoveListener	Removes the specified listener from the list of listeners on the specified source.

Top

See Also

Reference

PropertyChangedEventManager Class
C1.Silverlight.Binding Namespace

AddListener Method

The object with the event.

The object to add as a listener.

The name of the property that exists on source upon which to listen for changes.

Adds the specified listener to the list of listeners on the specified source.

Syntax

Visual Basic (Declaration)

```
Public Shared Sub AddListener( _
   ByVal source As System.ComponentModel.INotifyPropertyChanged, _
   ByVal listener As IWeakEventListener, _
   ByVal propertyName As System.String _
)
```

C#

```
public static void AddListener(
   System.ComponentModel.INotifyPropertyChanged source,
   IWeakEventListener listener,
   System.string propertyName
)
```

Parameters

source

The object with the event.

listener

The object to add as a listener.

propertyName

The name of the property that exists on source upon which to listen for changes.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PropertyChangedEventManager Class
PropertyChangedEventManager Members

RemoveListener Method

The object with the event.

The object to remove as a listener.

The name of the property that exists on source upon which to listen for changes.

Removes the specified listener from the list of listeners on the specified source.

Syntax

Visual Basic (Declaration)

```
Public Shared Sub RemoveListener( _
   ByVal source As System.ComponentModel.INotifyPropertyChanged, _
   ByVal listener As IWeakEventListener, _
   ByVal propertyName As System.String _
)
```

C#

```
public static void RemoveListener(
   System.ComponentModel.INotifyPropertyChanged source,
   IWeakEventListener listener,
   System.string propertyName
)
```

Parameters

source

The object with the event.

listener

The object to remove as a listener.

propertyName

The name of the property that exists on source upon which to listen for changes.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PropertyChangedEventManager Class
PropertyChangedEventManager Members

Tally

A class that aggregates values and provides summary statistics (including count, sum, average, standard deviation, and variance).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Tally
```

C#

```
public class Tally
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.Tally

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Members
C1.Silverlight.Binding Namespace

Overview

A class that aggregates values and provides summary statistics (including count, sum, average, standard deviation, and variance).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Tally
```

C#

```
public class Tally
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.Tally

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Members
C1.Silverlight.Binding Namespace

Members

Methods

The following tables list the members exposed by Tally.

Public Constructors

	Name	Description

≡◆	Tally Constructor	Overloaded.

Top

Public Methods

	Name	Description
≡◆	Add	Adds an Expression to the Tally, updating the associated summary statistics.
≡◆	AddValue	Adds a value to the Tally, updating the associated summary statistics.
≡◆	Average	Gets the average of the values in the Tally.
≡◆	Count	Gets the total count of values in the Tally.
≡◆	Max	Gets the maximum value in the Tally.
≡◆	Min	Gets the minimum value in the Tally.
≡◆	Range	Gets the range of values in the Tally (maximum minus minimum).
≡◆	Std	Gets the standard deviation of the values in the Tally.
≡◆	StdP	Gets the population standard deviation of the values in the Tally.
≡◆	Sum	Gets the sum of the values in the Tally.
≡◆	Var	Gets the variance of the values in the Tally.
≡◆	VarP	Gets the population variance of the values in the Tally.

Top

See Also

Reference

Tally Class
C1.Silverlight.Binding Namespace

Tally Constructor

Overload List

Overload	Description
Tally Constructor(Boolean)	Initializes a new instance of a Tally.
Tally Constructor()	Initializes a new instance of a Tally.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Tally Constructor(Boolean)

Whether this Tally can be used only with numeric values.

Initializes a new instance of a Tally.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
    ByVal numbersOnly As System.Boolean _
)
```

C#

```
public Tally(
    System.bool numbersOnly
)
```

Parameters

numbersOnly

Whether this Tally can be used only with numeric values.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members
Overload List

Tally Constructor()

Initializes a new instance of a Tally.

Syntax

Visual Basic (Declaration)

```
Public Function New()
```

C#

```
public Tally()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members
Overload List

Methods

%%scrap%%

" -->

For a list of all members of this type, see Tally members.

Public Methods

	Name	Description
≡◆	Add	Adds an Expression to the Tally, updating the associated summary statistics.
≡◆	AddValue	Adds a value to the Tally, updating the associated summary statistics.
≡◆	Average	Gets the average of the values in the Tally.
≡◆	Count	Gets the total count of values in the Tally.
≡◆	Max	Gets the maximum value in the Tally.
≡◆	Min	Gets the minimum value in the Tally.
≡◆	Range	Gets the range of values in the Tally (maximum minus minimum).
≡◆	Std	Gets the standard deviation of the values in the Tally.
≡◆	StdP	Gets the population standard deviation of the values in the Tally.
≡◆	Sum	Gets the sum of the values in the Tally.
≡◆	Var	Gets the variance of the values in the Tally.
≡◆	VarP	Gets the population variance of the values in the Tally.

Top

See Also

Reference

Tally Class
C1.Silverlight.Binding Namespace

Add Method

Expression to add to the Tally.

Adds an Expression to the Tally, updating the associated summary statistics.

Syntax

Visual Basic (Declaration)

```
Public Sub Add( _
   ByVal e As Expression _
)
```

C#

```
public void Add(
   Expression e
)
```

Parameters

e

Expression to add to the Tally.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

AddValue Method

Value to add to the Tally

Adds a value to the Tally, updating the associated summary statistics.

Syntax

Visual Basic (Declaration)

```
Public Sub AddValue( _
   ByVal value As System.Object _
)
```

C#

```
public void AddValue(
   System.object value
)
```

Parameters

value

Value to add to the Tally

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Average Method

Gets the average of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Average() As System.Double
```

C#

```
public System.double Average()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Count Method

Gets the total count of values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Count() As System.Double
```

C#

```
public System.double Count()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Max Method

Gets the maximum value in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Max() As System.Double
```

C#

```
public System.double Max()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Min Method

Gets the minimum value in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Min() As System.Double
```

C#

```
public System.double Min()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Range Method

Gets the range of values in the Tally (maximum minus minimum).

Syntax

Visual Basic (Declaration)

```
Public Function Range() As System.Double
```

C#

```
public System.double Range()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Std Method

Gets the standard deviation of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Std() As System.Double
```

C#

```
public System.double Std()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

StdP Method

Gets the population standard deviation of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function StdP() As System.Double
```

C#

```
public System.double StdP()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Sum Method

Gets the sum of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Sum() As System.Double
```

C#

```
public System.double Sum()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Var Method

Gets the variance of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Var() As System.Double
```

C#

```
public System.double Var()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

VarP Method

Gets the population variance of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function VarP() As System.Double
```

C#

```
public System.double VarP()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

XObjectExpression

Expression that represents an external object.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class XObjectExpression
   Inherits Expression
```

C#

```
public class XObjectExpression : Expression
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.Expression
    C1.Silverlight.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Members
C1.Silverlight.Binding Namespace

Overview

Expression that represents an external object.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class XObjectExpression
   Inherits Expression
```

C#

```
public class XObjectExpression : Expression
```

Inheritance Hierarchy

System.Object
  C1.Silverlight.Binding.Expression
    C1.Silverlight.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Members
C1.Silverlight.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by XObjectExpression.

Public Properties

	Name	Description
▤	Value	Gets the inner value of this expression (may be an actual value or an IValueObject).

Top

Public Methods

	Name	Description

	CompareTo	Compares the value of this Expression to the value of another Expression. (Inherited from C1.Silverlight.Binding.Expression)
	Evaluate	Overridden. Returns the value of the Expression.
	GetEnumerator	Gets an **System.Collections.IEnumerator** that can be used to iterate over a collection.
	Optimize	Returns an equal or simpler (but equivalent) version of the Expression. (Inherited from C1.Silverlight.Binding.Expression)

Top

See Also

Reference

XObjectExpression Class
C1.Silverlight.Binding Namespace

Methods

%%scrap%%

" -->

For a list of all members of this type, see XObjectExpression members.

Public Methods

	Name	Description
	CompareTo	Compares the value of this Expression to the value of another Expression. (Inherited from C1.Silverlight.Binding.Expression)
	Evaluate	Overridden. Returns the value of the Expression.
	GetEnumerator	Gets an **System.Collections.IEnumerator** that can be used to iterate over a collection.
	Optimize	Returns an equal or simpler (but equivalent) version of the Expression. (Inherited from C1.Silverlight.Binding.Expression)

Top

See Also

Reference

XObjectExpression Class
C1.Silverlight.Binding Namespace

Evaluate Method

Returns the value of the Expression.

Syntax

Visual Basic (Declaration)

```
Public Overrides Function Evaluate() As System.Object
```

C#

```
public override System.object Evaluate()
```

Return Value

The value of the Expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Class
XObjectExpression Members

GetEnumerator Method

Gets an **System.Collections.IEnumerator** that can be used to iterate over a collection.

Syntax

Visual Basic (Declaration)

```
Public Function GetEnumerator() As System.Collections.IEnumerator
```

C#

```
public System.Collections.IEnumerator GetEnumerator()
```

Return Value

An **System.Collections.IEnumerator** that can be used to iterate over a collection.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Class
XObjectExpression Members

Properties

>

Name	Description
Value	Gets the inner value of this expression (may be an actual value or an IValueObject).

Top

See Also

Reference

XObjectExpression Class
C1.Silverlight.Binding Namespace

Value Property

Gets the inner value of this expression (may be an actual value or an IValueObject).

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property Value As System.Object
```

C#

```
public System.object Value {get;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Class
XObjectExpression Members

Interfaces

IValueObject

Interface supported by external objects that have to return a value other than themselves (e.g. a cell range object should return the cell content instead of the range itself).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Interface IValueObject
```

C#

```
public interface IValueObject
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IValueObject Members
C1.Silverlight.Binding Namespace

Overview

Interface supported by external objects that have to return a value other than themselves (e.g. a cell range object should return the cell content instead of the range itself).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Interface IValueObject
```

C#

```
public interface IValueObject
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IValueObject Members
C1.Silverlight.Binding Namespace

Members

Methods

The following tables list the members exposed by IValueObject.

Public Methods

	Name	Description
	GetValue	Gets the value of this object.

Top

See Also

Reference

IValueObject Interface
C1.Silverlight.Binding Namespace

Methods

%%scrap%%

" -->

For a list of all members of this type, see IValueObject members.

Public Methods

	Name	Description
≡◆	GetValue	Gets the value of this object.

Top

See Also

Reference

IValueObject Interface
C1.Silverlight.Binding Namespace

GetValue Method

Gets the value of this object.

Syntax

Visual Basic (Declaration)	
`Function GetValue() As System.Object`	
C#	
`System.object GetValue()`	

Return Value

The value of this object.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IValueObject Interface
IValueObject Members

IWeakEventListener

Provides event listening support for classes that expect to receive events through the WeakEvent pattern and a WeakEventManager.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Interface IWeakEventListener
```

C#

```
public interface IWeakEventListener
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IWeakEventListener Members
C1.Silverlight.Binding Namespace

Overview

Provides event listening support for classes that expect to receive events through the WeakEvent pattern and a WeakEventManager.

Object Model

Syntax

Visual Basic (Declaration)	
`Public Interface IWeakEventListener`	
C#	
`public interface IWeakEventListener`	

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IWeakEventListener Members
C1.Silverlight.Binding Namespace

Members

Methods

The following tables list the members exposed by IWeakEventListener.

Public Methods

	Name	Description
≡◆	ReceiveWeakEvent	Receives events from the centralized event manager.

Top

See Also

Reference

IWeakEventListener Interface
C1.Silverlight.Binding Namespace

Methods

%%scrap%%

" -->

For a list of all members of this type, see IWeakEventListener members.

Public Methods

	Name	Description
≡◆	ReceiveWeakEvent	Receives events from the centralized event manager.

Top

See Also

Reference

IWeakEventListener Interface
C1.Silverlight.Binding Namespace

ReceiveWeakEvent Method

The type of the WeakEventManager calling this method.

Object that originated the event.

Event data.

Receives events from the centralized event manager.

Syntax

Visual Basic (Declaration)

```
Function ReceiveWeakEvent( _
   ByVal managerType As System.Type, _
   ByVal sender As System.Object, _
   ByVal e As System.EventArgs _
) As System.Boolean
```

C#

```
System.bool ReceiveWeakEvent(
   System.Type managerType,
   System.object sender,
   System.EventArgs e
)
```

Parameters

managerType

The type of the WeakEventManager calling this method.

sender

Object that originated the event.

e

Event data.

Return Value

true if the listener handled the event. It is considered an error by the WeakEventManager handling in WPF to register a listener for an event that the listener does not handle. Regardless, the method should return false if it receives an event that it does not recognize or handle.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IWeakEventListener Interface
IWeakEventListener Members

Delegates

CalcEngineFunction

List of Expression objects that represent the parameters to be used in the function call.

Delegate that represents CalcEngine functions.

Syntax

Visual Basic (Declaration)	

```
Public Delegate Function CalcEngineFunction( _
   ByVal parms As System.Collections.Generic.List(Of Expression) _
) As System.Object
```

C#

```
public delegate System.object CalcEngineFunction(
   System.Collections.Generic.List<Expression> parms
)
```

Parameters

parms

List of Expression objects that represent the parameters to be used in the function call.

Return Value

The function result.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

CalcEngineFunction Members
C1.Silverlight.Binding Namespace

C1.WPF.Binding.4 Assembly

Namespaces

C1.WPF.Binding Namespace

Overview

Classes

	Class	Description

	C1Binding	A C1.WPF.Binding that supports formulas.
	C1CalcEngine	Class that parses strings and returns Expression objects that can be evaluated.
	C1StaticResource	A **StaticResource** that supports a path specification.
	Expression	Base class that represents parsed expressions.
	FunctionDefinition	Function definition class (keeps function name, parameter counts, and delegate).
	Tally	A class that aggregates values and provides summary statistics (including count, sum, average, standard deviation, and variance).
	XObjectExpression	Expression that represents an external object.

Interfaces

	Interface	Description
	IValueObject	Interface supported by external objects that have to return a value other than themselves (e.g. a cell range object should return the cell content instead of the range itself).

Delegates

	Delegate	Description
	CalcEngineFunction	Delegate that represents CalcEngine functions.

See Also

Reference

C1.WPF.Binding.4 Assembly

Classes

C1Binding

Example

A C1.WPF.Binding that supports formulas.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1Binding
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1Binding : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1Binding** class is similar to a regular C1.WPF.Binding, but supports expressions instead of simple paths.

The expression syntax is similar to the one used in Excel formulas, including support for the IF statement which makes it easy to create bindings that make **Converters** unnecessary.

Example

The XAML below uses the **C1Binding** class to bind several properties of a **TextBlock** element to a data value. Notice that the bindings are used to provide values for properties of several types, and no converters are required. Notice how the XAML snippet above specifies the 'Expression' property. This is required only in Silverlight, and could be omitted in WPF. Notice also how vertical bars ('|') are used to specify quotes inside expressions. You could also use &quot; instead, but the bars made the code more readable.

- C#

```
<TextBlock
    Text="{c1:C1Binding Expression=CustomerName}"
    Visibility="{c1:C1Binding Expression='if(Active, |Visible|,
|Collapsed|)'}"
    FontWeight="{c1:C1Binding Expression='if(Sales > 200, |Bold|,
|Normal|)'}"
```

```
    Foreground="{c1:C1Binding Expression='if(Sales > 200, |Blue|, |Red|)'}"
/>
```

Inheritance Hierarchy

System.Object
  System.Windows.Markup.MarkupExtension
    C1.WPF.Binding.C1Binding

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Members
C1.WPF.Binding Namespace

Overview

Example

A C1.WPF.Binding that supports formulas.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1Binding
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1Binding : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1Binding** class is similar to a regular C1.WPF.Binding, but supports expressions instead of simple paths.

The expression syntax is similar to the one used in Excel formulas, including support for the IF statement which makes it easy to create bindings that make **Converters** unnecessary.

Example

The XAML below uses the **C1Binding** class to bind several properties of a **TextBlock** element to a data value. Notice that the bindings are used to provide values for properties of several types, and no converters are required. Notice how the XAML snippet above specifies the 'Expression' property. This is required only in Silverlight, and could be omitted in WPF. Notice also how vertical bars ('|') are used to specify quotes inside expressions. You could also use &quot; instead, but the bars made the code more readable.

- C#

```
<TextBlock
    Text="{c1:C1Binding Expression=CustomerName}"
    Visibility="{c1:C1Binding Expression='if(Active, |Visible|,
|Collapsed|)'}"
    FontWeight="{c1:C1Binding Expression='if(Sales > 200, |Bold|,
|Normal|)'}"
    Foreground="{c1:C1Binding Expression='if(Sales > 200, |Blue|, |Red|)'}"
/>
```

Inheritance Hierarchy

System.Object
  System.Windows.Markup.MarkupExtension
    C1.WPF.Binding.C1Binding

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Members
C1.WPF.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by C1Binding.

Public Constructors

	Name	Description
◆	C1Binding Constructor	Overloaded.

Top

Public Properties

	Name	Description
▤	Expression	Gets or sets the expression used to calculate the binding value.
▤	FallbackValue	Gets or sets the value to use when the binding is unable to return a value.
▤	StringFormat	Gets or sets a string that specifies how to format the binding value.
▤	TargetNullValue	Gets or sets the value that is used in the target when the value of the source is null.

Top

Public Methods

	Name	Description
◆	ProvideValue	Overridden. Returns an object that is provided as the value of the target property for this markup extension.

Top

See Also

Reference

C1Binding Class
C1.WPF.Binding Namespace

C1Binding Constructor

Overload List

Overload	Description

C1Binding Constructor(String)	Initializes a new instance of a C1Binding.
C1Binding Constructor()	Initializes a new instance of a C1Binding.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

C1Binding Constructor(String)

Binding expression.

Initializes a new instance of a C1Binding.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal expression As System.String _
)
```

C#

```
public C1Binding(
   System.string expression
)
```

Parameters

expression

Binding expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members
Overload List

C1Binding Constructor()

Initializes a new instance of a C1Binding.

Syntax

Visual Basic (Declaration)	
`Public Function New()`	
C#	
`public C1Binding()`	

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members
Overload List

Methods

%%scrap%%

" -->

For a list of all members of this type, see C1Binding members.

Public Methods

	Name	Description

	ProvideValue	Overridden. Returns an object that is provided as the value of the target property for this markup extension.

Top

See Also

Reference

C1Binding Class
C1.WPF.Binding Namespace

ProvideValue Method

A service provider helper that can provide services for the markup extension.

Returns an object that is provided as the value of the target property for this markup extension.

Syntax

Visual Basic (Declaration)

```
Public Overrides Function ProvideValue( _
   ByVal serviceProvider As System.IServiceProvider _
) As System.Object
```

C#

```
public override System.object ProvideValue(
   System.IServiceProvider serviceProvider
)
```

Parameters

serviceProvider

A service provider helper that can provide services for the markup extension.

Return Value

The object value to set on the property where the extension is applied.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

Properties

%%scrap%%

" -->

For a list of all members of this type, see C1Binding members.

Public Properties

	Name	Description
	Expression	Gets or sets the expression used to calculate the binding value.
	FallbackValue	Gets or sets the value to use when the binding is unable to return a value.
	StringFormat	Gets or sets a string that specifies how to format the binding value.
	TargetNullValue	Gets or sets the value that is used in the target when the value of the source is null.

Top

See Also

Reference

C1Binding Class
C1.WPF.Binding Namespace

Expression Property

Gets or sets the expression used to calculate the binding value.

Syntax

Visual Basic (Declaration)	

```
Public Property Expression As System.String
```

C#

```
public System.string Expression {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

FallbackValue Property

Gets or sets the value to use when the binding is unable to return a value.

Syntax

Visual Basic (Declaration)

```
Public Property FallbackValue As System.Object
```

C#

```
public System.object FallbackValue {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

StringFormat Property

Gets or sets a string that specifies how to format the binding value.

Syntax

Visual Basic (Declaration)

```
Public Property StringFormat As System.String
```

C#

```
public System.string StringFormat {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

TargetNullValue Property

Gets or sets the value that is used in the target when the value of the source is null.

Syntax

Visual Basic (Declaration)

```
Public Property TargetNullValue As System.Object
```

C#

```
public System.object TargetNullValue {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1Binding Class
C1Binding Members

C1CalcEngine

Class that parses strings and returns Expression objects that can be evaluated.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1CalcEngine
```

C#

```
public class C1CalcEngine
```

Remarks

This class has three extensibility points:

Use the **DataContext** property to add an object's properties to the engine scope.

Use the **RegisterFunction** method to define custom functions.

Override the **GetExternalObject** method to add arbitrary variables to the engine scope.

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.C1CalcEngine

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Members
C1.WPF.Binding Namespace

Overview

Class that parses strings and returns Expression objects that can be evaluated.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1CalcEngine
```

C#

```
public class C1CalcEngine
```

Remarks

This class has three extensibility points:

Use the **DataContext** property to add an object's properties to the engine scope.

Use the **RegisterFunction** method to define custom functions.

Override the **GetExternalObject** method to add arbitrary variables to the engine scope.

Inheritance Hierarchy

System.Object
   C1.WPF.Binding.C1CalcEngine

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Members
C1.WPF.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by C1CalcEngine.

Public Constructors

	Name	Description
	C1CalcEngine Constructor	Initializes a new instance of a C1CalcEngine.

Top

Public Properties

	Name	Description
	CacheExpressions	Gets or sets whether the calc engine should keep a cache with parsed expressions.
	CultureInfo	Gets or sets the CultureInfo to use when parsing numbers and dates.
	DataContext	Gets or sets the DataContext for this C1CalcEngine.
	Functions	Gets the dictionary that contains function definitions.
	IdentifierChars	Gets or sets a string that specifies special characters that are valid for identifiers.
	OptimizeExpressions	Gets or sets whether the calc engine should optimize expressions when they are parsed.
	Variables	Gets the dictionary that contains simple variables (not in the DataContext).

Top

Public Methods

	Name	Description
	Evaluate	Evaluates a string.

◆	GetExternalObject	Gets an external object based on an identifier.
◆	Parse	Parses a string into an Expression.
◆	RegisterFunction	Overloaded. Registers a function that can be evaluated by this C1CalcEngine.

Top

See Also

Reference

C1CalcEngine Class
C1.WPF.Binding Namespace

C1CalcEngine Constructor

Initializes a new instance of a C1CalcEngine.

Syntax

Visual Basic (Declaration)

```
Public Function New()
```

C#

```
public C1CalcEngine()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Methods

%%scrap%%

" -->

For a list of all members of this type, see C1CalcEngine members.

Public Methods

	Name	Description
	Evaluate	Evaluates a string.
	GetExternalObject	Gets an external object based on an identifier.
	Parse	Parses a string into an Expression.
	RegisterFunction	Overloaded. Registers a function that can be evaluated by this C1CalcEngine.

Top

See Also

Reference

C1CalcEngine Class
C1.WPF.Binding Namespace

Evaluate Method

Expression to evaluate.

Evaluates a string.

Syntax

Visual Basic (Declaration)

```
Public Function Evaluate( _
   ByVal expression As System.String _
) As System.Object
```

C#

```
public System.object Evaluate(
   System.string expression
)
```

Parameters

expression

Expression to evaluate.

Return Value

The value of the expression.

Remarks

If you are going to evaluate the same expression several times, it is more efficient to parse it only once using the Parse method and then using the Expression.Evaluate method to evaluate the parsed expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

GetExternalObject Method

Gets an external object based on an identifier.

Syntax

Visual Basic (Declaration)

```
Public Overridable Function GetExternalObject( _
   ByVal identifier As System.String _
) As System.Object
```

C#

```
public virtual System.object GetExternalObject(
   System.string identifier
)
```

Parameters

identifier

Remarks

This method is useful when the engine needs to create objects dynamically. For example, a spreadsheet calc engine would use this method to dynamically create cell range objects based on identifiers that cannot be enumerated at design time (such as "AB12", "A1:AB12", etc.)

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Parse Method

String to parse.

Parses a string into an Expression.

Syntax

Visual Basic (Declaration)

```
Public Function Parse( _
   ByVal expression As System.String _
) As Expression
```

C#

```
public Expression Parse(
   System.string expression
)
```

Parameters

expression

String to parse.

Return Value

An Expression object that can be evaluated.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

RegisterFunction Method

Registers a function that can be evaluated by this C1CalcEngine.

Overload List

Overload	Description
RegisterFunction(String,Int32,Int32,CalcEngineFunction)	Registers a function that can be evaluated by this C1CalcEngine.
RegisterFunction(String,Int32,CalcEngineFunction)	Registers a function that can be evaluated by this C1CalcEngine.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

RegisterFunction(String,Int32,Int32,CalcEngineFunction) Method

Function name.

Minimum parameter count.

Maximum parameter count.

Delegate that evaluates the function.

Registers a function that can be evaluated by this C1CalcEngine.

Syntax

Visual Basic (Declaration)

```
Public Overloads Sub RegisterFunction( _
   ByVal functionName As System.String, _
   ByVal parmMin As System.Integer, _
   ByVal parmMax As System.Integer, _
   ByVal fn As CalcEngineFunction _
)
```

C#

```
public void RegisterFunction(
   System.string functionName,
   System.int parmMin,
   System.int parmMax,
   CalcEngineFunction fn
)
```

Parameters

functionName

Function name.

parmMin

Minimum parameter count.

parmMax

Maximum parameter count.

fn

Delegate that evaluates the function.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members
Overload List

RegisterFunction(String,Int32,CalcEngineFunction) Method
Function name.

Parameter count.

Delegate that evaluates the function.

Registers a function that can be evaluated by this C1CalcEngine.

Syntax

Visual Basic (Declaration)

```
Public Overloads Sub RegisterFunction( _
   ByVal functionName As System.String, _
   ByVal parmCount As System.Integer, _
   ByVal fn As CalcEngineFunction _
)
```

C#

```
public void RegisterFunction(
   System.string functionName,
   System.int parmCount,
   CalcEngineFunction fn
)
```

Parameters

functionName

Function name.

parmCount

Parameter count.

fn

Delegate that evaluates the function.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members
Overload List

Properties

%%scrap%%

" -->

For a list of all members of this type, see C1CalcEngine members.

Public Properties

	Name	Description
	CacheExpressions	Gets or sets whether the calc engine should keep a cache with parsed expressions.
	CultureInfo	Gets or sets the CultureInfo to use when parsing numbers and dates.
	DataContext	Gets or sets the DataContext for this C1CalcEngine.
	Functions	Gets the dictionary that contains function definitions.
	IdentifierChars	Gets or sets a string that specifies special characters that are valid for identifiers.
	OptimizeExpressions	Gets or sets whether the calc engine should optimize expressions when they are parsed.
	Variables	Gets the dictionary that contains simple variables (not in the DataContext).

Top

See Also

Reference

C1CalcEngine Class
C1.WPF.Binding Namespace

CacheExpressions Property

Gets or sets whether the calc engine should keep a cache with parsed expressions.

Syntax

Visual Basic (Declaration)

```
Public Property CacheExpressions As System.Boolean
```

C#

```
public System.bool CacheExpressions {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

CultureInfo Property

Gets or sets the CultureInfo to use when parsing numbers and dates.

Syntax

Visual Basic (Declaration)

```
Public Property CultureInfo As System.Globalization.CultureInfo
```

C#

```
public System.Globalization.CultureInfo CultureInfo {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

DataContext Property

Gets or sets the DataContext for this C1CalcEngine.

Syntax

Visual Basic (Declaration)

```
Public Overridable Property DataContext As System.Object
```

C#

```
public virtual System.object DataContext {get; set;}
```

Remarks

Once a DataContext is set, all public properties of the object become available to the CalcEngine, including sub-properties such as "Address.Street". These may be used with expressions just like any other constant.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Functions Property

Gets the dictionary that contains function definitions.

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property Functions As
System.Collections.Generic.Dictionary(Of String,FunctionDefinition)
```

C#

```
public System.Collections.Generic.Dictionary<string,FunctionDefinition>
Functions {get;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

IdentifierChars Property

Gets or sets a string that specifies special characters that are valid for identifiers.

Syntax

Visual Basic (Declaration)

```
Public Property IdentifierChars As System.String
```

C#

```
public System.string IdentifierChars {get; set;}
```

Remarks

Identifiers must start with a letter or an underscore, which may be followed by additional letters, underscores, or digits. This string allows you to specify additional valid characters such as ':' or '!' (used in Excel range references for example).

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

OptimizeExpressions Property

Gets or sets whether the calc engine should optimize expressions when they are parsed.

Syntax

Visual Basic (Declaration)

```
Public Property OptimizeExpressions As System.Boolean
```

C#

```
public System.bool OptimizeExpressions {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

Variables Property

Gets the dictionary that contains simple variables (not in the DataContext).

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property Variables As
System.Collections.Generic.Dictionary(Of String,Object)
```

C#

```
public System.Collections.Generic.Dictionary<string,object> Variables {get;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1CalcEngine Class
C1CalcEngine Members

C1StaticResource

Example

A **StaticResource** that supports a path specification.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1StaticResource
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1StaticResource : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1StaticResource** class is similar to a regular **StaticResource** markup, but supports a path specification instead of simple objects.

Paths are specified using the same expression syntax used in the C1Binding class. This allows you to use object members as resources.

Example

The XAML below uses the **C1StaticResource** class to bind a grid column using a value converter that is implemented in a **ViewModel** object. The XAML snippet above assigns the **_model.JobsConverter** object to the **Converter** property of the binding. This cannot be done using a nested binding because the **Binding** class is not a dependency object.

- C#

```
<c1:C1FlexGrid
  ItemsSource="{Binding Source={StaticResource _model}, Path=People}"
  AutoGenerateColumns="False" >
  <c1:C1FlexGrid.Columns>
    <c1:Column
      Binding="{Binding JobID,
        Mode=TwoWay,
        Converter={c1:C1StaticResource Key=_model,Path=JobsConverter}}" />
  </c1:C1FlexGrid.Columns>
</c1:C1FlexGrid>
```

Inheritance Hierarchy

System.Object
  System.Windows.Markup.MarkupExtension
    C1.WPF.Binding.C1StaticResource

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Members
C1.WPF.Binding Namespace

Overview

Example

A **StaticResource** that supports a path specification.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class C1StaticResource
   Inherits System.Windows.Markup.MarkupExtension
```

C#

```
public class C1StaticResource : System.Windows.Markup.MarkupExtension
```

Remarks

The **C1StaticResource** class is similar to a regular **StaticResource** markup, but supports a path specification instead of simple objects.

Paths are specified using the same expression syntax used in the C1Binding class. This allows you to use object members as resources.

Example

The XAML below uses the **C1StaticResource** class to bind a grid column using a value converter that is implemented in a **ViewModel** object. The XAML snippet above assigns the **_model.JobsConverter** object to the **Converter** property of the binding. This cannot be done using a nested binding because the **Binding** class is not a dependency object.

- C#

```
<c1:C1FlexGrid
  ItemsSource="{Binding Source={StaticResource _model}, Path=People}"
  AutoGenerateColumns="False" >
  <c1:C1FlexGrid.Columns>
    <c1:Column
      Binding="{Binding JobID,
        Mode=TwoWay,
        Converter={c1:C1StaticResource Key=_model,Path=JobsConverter}}" />
  </c1:C1FlexGrid.Columns>
</c1:C1FlexGrid>
```

Inheritance Hierarchy

System.Object
  System.Windows.Markup.MarkupExtension
    C1.WPF.Binding.C1StaticResource

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Members
C1.WPF.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by C1StaticResource.

Public Constructors

	Name	Description
	C1StaticResource Constructor	Overloaded.

Top

Public Properties

	Name	Description
	Key	Key that identifies the resource.
	Path	Expression that defines the data to extract from the resource.

Top

Public Methods

	Name	Description
	ProvideValue	Overridden. Returns an object that is provided as the value of this markup extension.

Top

See Also

Reference

C1StaticResource Class
C1.WPF.Binding Namespace

C1StaticResource Constructor

Overload List

Overload	Description

C1StaticResource Constructor()	Initializes a new instance of a C1StaticResource.
C1StaticResource Constructor(String)	Initializes a new instance of a C1StaticResource.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

C1StaticResource Constructor()

Initializes a new instance of a C1StaticResource.

Syntax

Visual Basic (Declaration)	
`Public Function New()`	
C#	
`public C1StaticResource()`	

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members
Overload List

C1StaticResource Constructor(String)

Key that identifies the resource.

Initializes a new instance of a C1StaticResource.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal resourceKey As System.String _
)
```

C#

```
public C1StaticResource(
   System.string resourceKey
)
```

Parameters

resourceKey

Key that identifies the resource.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members
Overload List

Methods

%%scrap%%

" -->

For a list of all members of this type, see C1StaticResource members.

Public Methods

	Name	Description

	ProvideValue	Overridden. Returns an object that is provided as the value of this markup extension.

Top

See Also

Reference

C1StaticResource Class
C1.WPF.Binding Namespace

ProvideValue Method

A service provider helper that can provide services for the markup extension.

Returns an object that is provided as the value of this markup extension.

Syntax

Visual Basic (Declaration)

```
Public Overrides Function ProvideValue( _
   ByVal serviceProvider As System.IServiceProvider _
) As System.Object
```

C#

```
public override System.object ProvideValue(
   System.IServiceProvider serviceProvider
)
```

Parameters

serviceProvider

A service provider helper that can provide services for the markup extension.

Return Value

The object value to set on the property where the extension is applied.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

Properties

%%scrap%%

" -->

For a list of all members of this type, see C1StaticResource members.

Public Properties

	Name	Description
▤	Key	Key that identifies the resource.
▤	Path	Expression that defines the data to extract from the resource.

Top

See Also

Reference

C1StaticResource Class
C1.WPF.Binding Namespace

Key Property

Key that identifies the resource.

Syntax

Visual Basic (Declaration)

```
Public Property Key As System.String
```

C#

```
public System.string Key {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

Path Property

Expression that defines the data to extract from the resource.

Syntax

Visual Basic (Declaration)

```
Public Property Path As System.String
```

C#

```
public System.string Path {get; set;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

C1StaticResource Class
C1StaticResource Members

Expression

Base class that represents parsed expressions.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Expression
```

C#

```
public class Expression
```

Remarks

For example: `Expression expr = scriptEngine.Parse(strExpression); object val = expr.Evaluate();`

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.Expression
    C1.WPF.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Members
C1.WPF.Binding Namespace

Overview

Base class that represents parsed expressions.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Expression
```

C#

```
public class Expression
```

Remarks

For example: `Expression expr = scriptEngine.Parse(strExpression); object val = expr.Evaluate();`

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.Expression
    C1.WPF.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Members
C1.WPF.Binding Namespace

Members

Methods

The following tables list the members exposed by Expression.

Public Methods

	Name	Description
≡◆	CompareTo	Compares the value of this Expression to the value of another Expression.
≡◆	Evaluate	Returns the value of the Expression.
≡◆	Optimize	Returns an equal or simpler (but equivalent) version of the Expression.

Top

Public Operators

Implicit Type Conversion	Overloaded. Evaluates the expression and returns the result converted to a string.

Top

See Also

Reference

Expression Class
C1.WPF.Binding Namespace

Methods

%%scrap%%

" -->

For a list of all members of this type, see Expression members.

Public Methods

	Name	Description
	CompareTo	Compares the value of this Expression to the value of another Expression.
	Evaluate	Returns the value of the Expression.
	Optimize	Returns an equal or simpler (but equivalent) version of the Expression.

Top

See Also

Reference

Expression Class
C1.WPF.Binding Namespace

CompareTo Method

Expression to compare to.

Compares the value of this Expression to the value of another Expression.

Syntax

Visual Basic (Declaration)

```
Public Function CompareTo( _
   ByVal other As Expression _
) As System.Integer
```

C#

```
public System.int CompareTo(
   Expression other
)
```

Parameters

other

Expression to compare to.

Return Value

An integer that indicates whether the value of this expression is smaller than, equal, or greater than the value of the other expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

Evaluate Method

Returns the value of the Expression.

Syntax

Visual Basic (Declaration)

```
Public Overridable Function Evaluate() As System.Object
```

C#

```
public virtual System.object Evaluate()
```

Return Value

The value of the Expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

Optimize Method

Returns an equal or simpler (but equivalent) version of the Expression.

Syntax

Visual Basic (Declaration)

```
Public Overridable Function Optimize() As Expression
```

C#

```
public virtual Expression Optimize()
```

Return Value

An equal or simpler but equivalent version of the expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

Operators

Implicit Type Conversion Operator

Evaluates the expression and returns the result converted to a string.

Overload List

Overload	Description
Implicit Type Conversion(String,Expression)	Evaluates the expression and returns the result converted to a string.
Implicit Type Conversion(Double,Expression)	Evaluates the expression and returns the result converted to a double.
Implicit Type Conversion(Boolean,Expression)	Evaluates the expression and returns the result converted to a Boolean value.
Implicit Type Conversion(DateTime,Expression)	Evaluates the expression and returns the result converted to a **System.DateTime** value.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Expression Class
Expression Members

FunctionDefinition

Function definition class (keeps function name, parameter counts, and delegate).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class FunctionDefinition
```

C#

```
public class FunctionDefinition
```

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.FunctionDefinition

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Members
C1.WPF.Binding Namespace

Overview

Function definition class (keeps function name, parameter counts, and delegate).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class FunctionDefinition
```

C#

```
public class FunctionDefinition
```

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.FunctionDefinition

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Members
C1.WPF.Binding Namespace

Members

Fields

The following tables list the members exposed by FunctionDefinition.

Public Constructors

	Name	Description
≡◆	FunctionDefinition Constructor	Initializes a new instance of a FunctionDefinition.

Top

Public Fields

	Name	Description
◆	Function	Gets or sets the CalcEngineFunction that evaluates this function.
◆	ParmMax	Gets or sets the maximum number of parameters allowed by this FunctionDefinition.
◆	ParmMin	Gets or sets the minimum number of parameters required by this FunctionDefinition.

Top

See Also

Reference

FunctionDefinition Class
C1.WPF.Binding Namespace

FunctionDefinition Constructor

The minimum number of parameters required by this function.

The maximum number of parameters allowed by this function.

The CalcEngineFunction delegate that evaluates this function

Initializes a new instance of a FunctionDefinition.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal parmMin As System.Integer, _
   ByVal parmMax As System.Integer, _
   ByVal function As CalcEngineFunction _
)
```

C#

```
public FunctionDefinition(
   System.int parmMin,
   System.int parmMax,
   CalcEngineFunction function
)
```

Parameters

parmMin

The minimum number of parameters required by this function.

parmMax

The maximum number of parameters allowed by this function.

function

The CalcEngineFunction delegate that evaluates this function

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

Fields

>

Name	Description
◆ Function	Gets or sets the CalcEngineFunction that evaluates this function.
◆ ParmMax	Gets or sets the maximum number of parameters allowed by this FunctionDefinition.
◆ ParmMin	Gets or sets the minimum number of parameters required by this FunctionDefinition.

Top

See Also

Reference

FunctionDefinition Class
C1.WPF.Binding Namespace

Function Field

Gets or sets the CalcEngineFunction that evaluates this function.

Syntax

Visual Basic (Declaration)

```
Public Function As CalcEngineFunction
```

C#

```
public CalcEngineFunction Function
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

ParmMax Field

Gets or sets the maximum number of parameters allowed by this FunctionDefinition.

Syntax

Visual Basic (Declaration)

```
Public ParmMax As System.Integer
```

C#

```
public System.int ParmMax
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

ParmMin Field

Gets or sets the minimum number of parameters required by this FunctionDefinition.

Syntax

Visual Basic (Declaration)

```
Public ParmMin As System.Integer
```

C#

```
public System.int ParmMin
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

FunctionDefinition Class
FunctionDefinition Members

Tally

A class that aggregates values and provides summary statistics (including count, sum, average, standard deviation, and variance).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Tally
```

C#

```
public class Tally
```

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.Tally

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Members
C1.WPF.Binding Namespace

Overview

A class that aggregates values and provides summary statistics (including count, sum, average, standard deviation, and variance).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class Tally
```

C#

```
public class Tally
```

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.Tally

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Members
C1.WPF.Binding Namespace

Members

Methods

The following tables list the members exposed by Tally.

Public Constructors

	Name	Description

	Tally Constructor	Overloaded.

Top

Public Methods

	Name	Description
	Add	Adds an Expression to the Tally, updating the associated summary statistics.
	AddValue	Adds a value to the Tally, updating the associated summary statistics.
	Average	Gets the average of the values in the Tally.
	Count	Gets the total count of values in the Tally.
	Max	Gets the maximum value in the Tally.
	Min	Gets the minimum value in the Tally.
	Range	Gets the range of values in the Tally (maximum minus minimum).
	Std	Gets the standard deviation of the values in the Tally.
	StdP	Gets the population standard deviation of the values in the Tally.
	Sum	Gets the sum of the values in the Tally.
	Var	Gets the variance of the values in the Tally.
	VarP	Gets the population variance of the values in the Tally.

Top

See Also

Reference

Tally Class
C1.WPF.Binding Namespace

Tally Constructor

Overload List

Overload	Description
Tally Constructor(Boolean)	Initializes a new instance of a Tally.
Tally Constructor()	Initializes a new instance of a Tally.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Tally Constructor(Boolean)

Whether this Tally can be used only with numeric values.

Initializes a new instance of a Tally.

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal numbersOnly As System.Boolean _
)
```

C#

```
public Tally(
   System.bool numbersOnly
)
```

Parameters

numbersOnly

Whether this Tally can be used only with numeric values.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members
Overload List

Tally Constructor()

Initializes a new instance of a Tally.

Syntax

Visual Basic (Declaration)

```
Public Function New()
```

C#

```
public Tally()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members
Overload List

Methods

%%scrap%%

" -->

For a list of all members of this type, see Tally members.

Public Methods

	Name	Description
◆	Add	Adds an Expression to the Tally, updating the associated summary statistics.
◆	AddValue	Adds a value to the Tally, updating the associated summary statistics.
◆	Average	Gets the average of the values in the Tally.
◆	Count	Gets the total count of values in the Tally.
◆	Max	Gets the maximum value in the Tally.
◆	Min	Gets the minimum value in the Tally.
◆	Range	Gets the range of values in the Tally (maximum minus minimum).
◆	Std	Gets the standard deviation of the values in the Tally.
◆	StdP	Gets the population standard deviation of the values in the Tally.
◆	Sum	Gets the sum of the values in the Tally.
◆	Var	Gets the variance of the values in the Tally.
◆	VarP	Gets the population variance of the values in the Tally.

Top

See Also

Reference

Tally Class
C1.WPF.Binding Namespace

Add Method

Expression to add to the Tally.

Adds an Expression to the Tally, updating the associated summary statistics.

Syntax

Visual Basic (Declaration)

```
Public Sub Add( _
   ByVal e As Expression _
)
```

C#

```
public void Add(
   Expression e
)
```

Parameters

e

Expression to add to the Tally.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

AddValue Method

Value to add to the Tally

Adds a value to the Tally, updating the associated summary statistics.

Syntax

Visual Basic (Declaration)

```
Public Sub AddValue( _
   ByVal value As System.Object _
)
```

C#

```
public void AddValue(
   System.object value
)
```

Parameters

value

Value to add to the Tally

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Average Method

Gets the average of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Average() As System.Double
```

C#

```
public System.double Average()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Count Method

Gets the total count of values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Count() As System.Double
```

C#

```
public System.double Count()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Max Method

Gets the maximum value in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Max() As System.Double
```

C#

```
public System.double Max()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Min Method

Gets the minimum value in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Min() As System.Double
```

C#

```
public System.double Min()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Range Method

Gets the range of values in the Tally (maximum minus minimum).

Syntax

Visual Basic (Declaration)

```
Public Function Range() As System.Double
```

C#

```
public System.double Range()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Std Method

Gets the standard deviation of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Std() As System.Double
```

C#

```
public System.double Std()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

StdP Method

Gets the population standard deviation of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function StdP() As System.Double
```

C#

```
public System.double StdP()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Sum Method

Gets the sum of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Sum() As System.Double
```

C#

```
public System.double Sum()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

Var Method

Gets the variance of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function Var() As System.Double
```

C#

```
public System.double Var()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

VarP Method

Gets the population variance of the values in the Tally.

Syntax

Visual Basic (Declaration)

```
Public Function VarP() As System.Double
```

C#

```
public System.double VarP()
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

Tally Class
Tally Members

XObjectExpression

Expression that represents an external object.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class XObjectExpression
   Inherits Expression
```

C#

```
public class XObjectExpression : Expression
```

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.Expression
    C1.WPF.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Members
C1.WPF.Binding Namespace

Overview

Expression that represents an external object.

Object Model

Syntax

Visual Basic (Declaration)

```
Public Class XObjectExpression
   Inherits Expression
```

C#

```
public class XObjectExpression : Expression
```

Inheritance Hierarchy

System.Object
  C1.WPF.Binding.Expression
    C1.WPF.Binding.XObjectExpression

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Members
C1.WPF.Binding Namespace

Members

Properties Methods

The following tables list the members exposed by XObjectExpression.

Public Properties

	Name	Description
▤	Value	Gets the inner value of this expression (may be an actual value or an IValueObject).

Top

Public Methods

	Name	Description

≡◆	CompareTo	Compares the value of this Expression to the value of another Expression. (Inherited from C1.WPF.Binding.Expression)
≡◆	Evaluate	Overridden. Returns the value of the Expression.
≡◆	GetEnumerator	Gets an **System.Collections.IEnumerator** that can be used to iterate over a collection.
≡◆	Optimize	Returns an equal or simpler (but equivalent) version of the Expression. (Inherited from C1.WPF.Binding.Expression)

Top

See Also

Reference

XObjectExpression Class
C1.WPF.Binding Namespace

Methods

%%scrap%%

" -->

For a list of all members of this type, see XObjectExpression members.

Public Methods

	Name	Description
≡◆	CompareTo	Compares the value of this Expression to the value of another Expression. (Inherited from C1.WPF.Binding.Expression)
≡◆	Evaluate	Overridden. Returns the value of the Expression.
≡◆	GetEnumerator	Gets an **System.Collections.IEnumerator** that can be used to iterate over a collection.
≡◆	Optimize	Returns an equal or simpler (but equivalent) version of the Expression. (Inherited from C1.WPF.Binding.Expression)

Top

See Also

Reference

XObjectExpression Class
C1.WPF.Binding Namespace

Evaluate Method

Returns the value of the Expression.

Syntax

Visual Basic (Declaration)

```
Public Overrides Function Evaluate() As System.Object
```

C#

```
public override System.object Evaluate()
```

Return Value

The value of the Expression.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Class
XObjectExpression Members

GetEnumerator Method

Gets an **System.Collections.IEnumerator** that can be used to iterate over a collection.

Syntax

Visual Basic (Declaration)

```
Public Function GetEnumerator() As System.Collections.IEnumerator
```

C#

```
public System.Collections.IEnumerator GetEnumerator()
```

Return Value

An **System.Collections.IEnumerator** that can be used to iterate over a collection.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Class
XObjectExpression Members

Properties

>

Name	Description
Value	Gets the inner value of this expression (may be an actual value or an IValueObject).

Top

See Also

Reference

XObjectExpression Class
C1.WPF.Binding Namespace

Value Property

Gets the inner value of this expression (may be an actual value or an IValueObject).

Syntax

Visual Basic (Declaration)

```
Public ReadOnly Property Value As System.Object
```

C#

```
public System.object Value {get;}
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

XObjectExpression Class
XObjectExpression Members

Interfaces

IValueObject

Interface supported by external objects that have to return a value other than themselves (e.g. a cell range object should return the cell content instead of the range itself).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Interface IValueObject
```

C#

```
public interface IValueObject
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IValueObject Members
C1.WPF.Binding Namespace

Overview

Interface supported by external objects that have to return a value other than themselves (e.g. a cell range object should return the cell content instead of the range itself).

Object Model

Syntax

Visual Basic (Declaration)

```
Public Interface IValueObject
```

C#

```
public interface IValueObject
```

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IValueObject Members
C1.WPF.Binding Namespace

Members

Methods

The following tables list the members exposed by IValueObject.

Public Methods

	Name	Description
	GetValue	Gets the value of this object.

Top

See Also

Reference

IValueObject Interface
C1.WPF.Binding Namespace

Methods

%%scrap%%

" -->

For a list of all members of this type, see IValueObject members.

Public Methods

	Name	Description
◆	GetValue	Gets the value of this object.

Top

See Also

Reference

IValueObject Interface
C1.WPF.Binding Namespace

GetValue Method

Gets the value of this object.

Syntax

Visual Basic (Declaration)	
`Function GetValue() As System.Object`	
C#	
`System.object GetValue()`	

Return Value

The value of this object.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

IValueObject Interface
IValueObject Members

Delegates

CalcEngineFunction

List of Expression objects that represent the parameters to be used in the function call.

Delegate that represents CalcEngine functions.

Syntax

Visual Basic (Declaration)

```
Public Delegate Function CalcEngineFunction( _
   ByVal parms As System.Collections.Generic.List(Of Expression) _
) As System.Object
```

C#

```
public delegate System.object CalcEngineFunction(
   System.Collections.Generic.List<Expression> parms
)
```

Parameters

parms

List of Expression objects that represent the parameters to be used in the function call.

Return Value

The function result.

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

CalcEngineFunction Members
C1.WPF.Binding Namespace

SmartAssembly.Attributes Namespace

Overview

Inheritance Hierarchy

Classes

	Class	Description
	PoweredByAttribute	

See Also

Reference

C1.WPF.Binding.4 Assembly

Classes

PoweredByAttribute

Object Model

Syntax

Visual Basic (Declaration)

```
Public NotInheritable Class PoweredByAttribute
   Inherits System.Attribute
```

C#

```
public sealed class PoweredByAttribute : System.Attribute
```

Inheritance Hierarchy

System.Object
  System.Attribute
    SmartAssembly.Attributes.PoweredByAttribute

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PoweredByAttribute Members
SmartAssembly.Attributes Namespace

Overview

Object Model

Syntax

Visual Basic (Declaration)

```
Public NotInheritable Class PoweredByAttribute
   Inherits System.Attribute
```

C#

```
public sealed class PoweredByAttribute : System.Attribute
```

Inheritance Hierarchy

System.Object
  System.Attribute
    SmartAssembly.Attributes.PoweredByAttribute

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PoweredByAttribute Members
SmartAssembly.Attributes Namespace

Members

Properties Methods

The following tables list the members exposed by PoweredByAttribute.

Public Constructors

	Name	Description
◆	PoweredByAttribute Constructor	

Top

Public Properties

	Name	Description
▤	TypeId	(Inherited from System.Attribute)

Top

Public Methods

	Name	Description
◆	Equals	(Inherited from System.Attribute)
◆	GetHashCode	(Inherited from System.Attribute)
◆	IsDefaultAttribute	(Inherited from System.Attribute)
◆	Match	(Inherited from System.Attribute)

Top

See Also

Reference

PoweredByAttribute Class
SmartAssembly.Attributes Namespace

PoweredByAttribute Constructor

Syntax

Visual Basic (Declaration)

```
Public Function New( _
   ByVal param1 As System.String _
)
```

C#

```
public PoweredByAttribute(
   System.string param1
)
```

Parameters

param1

Requirements

Target Platforms: Windows 7, Windows Vista SP1 or later, Windows XP SP3, Windows Server 2008 (Server Core not supported), Windows Server 2008 R2 (Server Core supported with SP1 or later), Windows Server 2003 SP2

See Also

Reference

PoweredByAttribute Class
PoweredByAttribute Members