

SCICHART v2.0 (BETA 2)

Release Note

A letter from the SciChart Team

A **huge thank you** to those who have downloaded the rough-around-the-edges SciChart v2.0 Beta 1! We have a second beta now with some urgent fixes. Please take the time to download and evaluate the second Beta. If you are happy, we can go to full production release.

As always our release schedule is frequent. **We will be responding to support requests** to fix any issues with v2.x over the coming weeks and months. Most importantly we are looking for compatibility issues (such as this used to work, no longer works). **Regarding v1.7 we will be phasing out support for this** – i.e. we will continue to fix major bugs but new feature development has to be on v2.0, since the codebases have diverged.

We hope you enjoy this release and we thank you once again for the confidence you have expressed in us by choosing SciChart!

Whats new in SciChart 2

For a full breakdown of new features, please see the [SciChart v2.0 Preview Video](#) on our website. Also the examples suite which has many new examples showcasing the new features of this powerful WPF Chart library.

New Features at a glance include:

- Box Plot, Error Bars, Bubble Chart, Fan Chart
- Heatmap / Spectrogram
- Stacked Column Chart, Stacked Bar Chart, Stacked Mountain Chart, Stacked Columns (side by side)
- No more DataSeriesSet, MVVM API improvements
- Unlimited, Multiple X-Axes
- Swap X-Y axis (rotate chart)
- New PointMarker API
- Support for more DataTypes in DataSeries (ushort, short, uint, byte, sbyte, TimeSpan)
- New TimeSpanAxis
- New Axis Interactivity API
- Alternative, Sub-pixel (High Quality) renderer
- High Performance Axis Bands, High Performance Dashed Line support (StrokeDashArray)
- Gradient Brush support for Column, Candlestick, Mountain, Box Plot and Stacked Chart types
- PaletteProvider now works on Line Series (Colors lines point by point)
- Support for custom themes via IThemeProvider
- Faster rendering in multiple axis scenarios
- Plugin renderer architecture
- Exceptions and renderer error codes - 'Why didn't SciChart render' reason is output to Console Window
- New! online API documentation (work in progress)

Update Instructions for users of v1.5-v1.7

Several API changes have been made. Please follow these upgrade instructions if you experience one of the following compilation or runtime errors:

There is no DataSet

✖ 1	The property 'DataSet' does not exist in XML namespace 'http://schemas.abtsoftware.co.uk/scichart'. Line 13 Position 28.	TradeOverlayExampleView.xaml	13 28	Abt.Controls.SciChart.Core.Wpf.Example
✖ 2	The property 'DataSeriesIndex' does not exist in XML namespace 'http://schemas.abtsoftware.co.uk/scichart'. Line 21 Position 54.	GapsInSeries.xaml	21 54	Abt.Controls.SciChart.Core.Wpf.Example

In SciChart v1.x we had the concept of DataSet, which was bound by setting `SciChartSurface.DataSet = new DataSeriesSet<double, double>()`. DataSeries were added to the DataSet. This brought the complication that RenderableSeries and DataSeries were matched by index (RenderableSeries.DataSeriesIndex property), and changing a renderable series type, or re-ordering series required synchronizing of these indices.

In SciChart v2.0 we have removed the concept of DataSet, and now RenderableSeries bind directly to DataSeries. You may set the relationship as follows. Once set, it exists for the lifetime of the application. If RenderableSeries are removed, then the DataSeries is also removed. If a new RenderableSeries added (to swap existing series), you will need to ensure that DataSeries is set on the new RenderableSeries.

Before Syntax

```
<!-- Define the chart surface host -->
<s:SciChartSurface DataSet="{Binding PriceData}"
  Annotations="{Binding TradeAnnotations}"
  s:ThemeManager.Theme="Chrome">

  <s:SciChartSurface.RenderableSeries>
    <s:FastLineRenderableSeries SeriesColor="#279B27" />
  </s:SciChartSurface.RenderableSeries>
```

After Syntax

```
<!-- Define the chart surface host -->
<s:SciChartSurface Annotations="{Binding TradeAnnotations}" s:ThemeManager.Theme="Chrome">

  <s:SciChartSurface.RenderableSeries>
    <s:FastLineRenderableSeries DataSeries="{Binding ChartDataSeries}"
    SeriesColor="#279B27" />
  </s:SciChartSurface.RenderableSeries>
```

Unable to SuspendUpdates / ResumeUpdates since there is no DataSet

The following code compiles in v1.x but in v2.0 will not, since there is no more DataSeriesSet.

```
// Lock the dataset and clear / re-add new data
using (this.DataSeriesSet.SuspendUpdates())
{
    _series0.Clear();
    _series0.Append(dataSource.XData, dataSource.YData);
}
```

The `SuspendUpdates()` method has been moved to the `DataSeries` instead. You may use any `DataSeries` to suspend the operation of the parent `SciChartSurface`.

Binding to `DataSeriesSet` should be replaced with `DataSeries` bindings.

Because there is no `DataSeriesSet`, instead of binding `SciChartSurface.DataSet` to a `DataSet` property in your viewmodel, you will need to bind each `RenderableSeries` to a `DataSeries` property in your view model.

One way to do this easily is to use the `SciChartSurface.SeriesSource` API (which expects an observable collection of `ISeriesViewModel`). This API has not changed since v1.x

Creating `DataSeries`, `FifoSeries` when there is no `DataSet`

Because there is no `DataSeriesSet`, we must create `DataSeries` in a slightly different way. The following code is no longer valid:

```
private void CreateDataSetAndSeries()
{
    // Create a new dataset of type X=double, Y=double
    dataset = new DataSet<double, double>();

    if (isFifoCheckBox.IsChecked == true)
    {
        // Add three FIFO series to fill with data.
        // calling dataset.AddFifoSeries(int size) will create a FIFO series. New data is appended
        // until the size is met, at which point
        // old data is discarded. Internally the FIFO series is implemented as a circular buffer so
        // that old data is pushed out of the buffer
        // once the capacity has been reached
        // Note: Once a FIFO series has been added to a dataset, all subsequent series must be FIFO
        // series. In addition, the FifoSize must be the
        // same for all FIFO series in a dataset.
        dataset.AddFifoSeries(FifoSize);
        dataset.AddFifoSeries(FifoSize);
        dataset.AddFifoSeries(FifoSize);
    }
    else
    {
        // Add three standard series to fill with data.
        // calling dataset.AddSeries() will create a normal dataseries. New data is appended and no
        // old data is discarded
        dataset.AddSeries();
        dataset.AddSeries();
        dataset.AddSeries();
    }

    // Set the dataset on the chart
    sciChartSurface.DataSet = dataset;
}
```

Instead, we must use something like this:

```
private void CreateDataSetAndSeries()
{
    // Create new Dataseries of type X=double, Y=double
    series0 = new XyDataSeries<double, double>();
    series1 = new XyDataSeries<double, double>();
    series2 = new XyDataSeries<double, double>();

    if (isFifoCheckBox.IsChecked == true)
    {
        // Add three FIFO series to fill with data.
        // setting the FIFO capacity will denote this series as a FIFO series. New data is appended
        // until the size is met, at which point
        // old data is discarded. Internally the FIFO series is implemented as a circular buffer so
        // that old data is pushed out of the buffer
        // once the capacity has been reached
        // Note: Once a FIFO series has been added to a dataset, all subsequent series must be FIFO
        // series. In addition, the FifoSize must be the
        // same for all FIFO series in a dataset.
        series0.FifoCapacity = FifoSize;
    }
}
```

```

        series1.FifoCapacity = FifoSize;
        series2.FifoCapacity = FifoSize;
    }

    // Set the dataseries on the chart's RenderableSeries
    renderableSeries0.DataSeries = series0;
    renderableSeries1.DataSeries = series1;
    renderableSeries2.DataSeries = series2;
}

```

DataSeriesIndex should be removed

Now that `RenderableSeries.DataSeries` can be bound (or set) directly to a `RenderableSeries`, there is no more need for `DataSeriesIndex`.

- If you wish to re-order the `RenderableSeries`, simply set the order in the `SciChartSurface.RenderableSeries` collection
- If you wish to bind several `RenderableSeries` to a single `DataSeries` (e.g. to switch visibility), then just do that!

✖ 2 'Abt.Controls.SciChart.Visuals.RenderableSeries.IRenderableSeries' does not contain a definition for 'DataSeriesIndex' and no extension method 'DataSeriesIndex' accepting a first argument of type 'Abt.Controls.SciChart.Visuals.RenderableSeries.IRenderableSeries' could be found (are you missing a using directive or an assembly reference?) 45:53 Abt.Controls.SciChart.Core.Wpf.Example

RolloverLabelTemplate renamed to TooltipLabelTemplate

As part of refactoring, we have renamed `RolloverLabelTemplate` to `TooltipLabelTemplate`. Please update your XAML accordingly.

✖ 3 The property 'RolloverLabelTemplate' does not exist in XML namespace 'http://schemas.abtsoftware.co.uk/scichart'. Line 194 Position 48. 19:48 Abt.Controls.SciChart.Core.Wpf.Example

Before

```

<!-- Declare ChartModifiers -->
<SciChart:SciChartSurface.ChartModifier>
  <SciChart:ModifierGroup>
    <SciChart:RolloverModifier x:Name="rolloverModifier"
                             DrawVerticalLine="True"
                             RolloverLabelTemplate="{StaticResource
RolloverLabelTemplate}"
                             SourceMode="AllSeries" />
    <SciChart:SeriesSelectionModifier />
  </SciChart:ModifierGroup>
</SciChart:SciChartSurface.ChartModifier>

```

After

```

<!-- Declare ChartModifiers -->
<SciChart:SciChartSurface.ChartModifier>
  <SciChart:ModifierGroup>
    <SciChart:RolloverModifier x:Name="rolloverModifier"
                             DrawVerticalLine="True"
                             SourceMode="AllSeries"
                             TooltipLabelTemplate="{StaticResource
RolloverLabelTemplate}" />
    <SciChart:SeriesSelectionModifier />
  </SciChart:ModifierGroup>
</SciChart:SciChartSurface.ChartModifier>

```

CursorLabelTemplate renamed to TooltipLabelTemplate

Similarly, CursorLabelTemplate has been renamed to TooltipLabelTemplate. Please update your XAML accordingly.

✖ 29 The member "CursorLabelTemplate" is not recognized or is not accessible. OscilloscopeView.xaml 24: 46 Abt.Controls.SciChart.Core.Wpf.Example

Before

```
<!-- Create ChartModifiers, which are enabled or disabled from the ViewModel via the
ChartModifier property and IsModifierTypeConverter -->
<SciChart:SciChartSurface.ChartModifier>
  <SciChart:ModifierGroup>
    <SciChart:CursorModifier CursorLabelTemplate="{StaticResource
CursorLabelTemplate}" IsEnabled="{Binding ChartModifier, Mode=TwoWay, Converter={StaticResource
IsModifierTypeConverter}, ConverterParameter=CrosshairsCursor}" />
  </SciChart:ModifierGroup>
</SciChart:SciChartSurface.ChartModifier>
```

After

```
<!-- Create ChartModifiers, which are enabled or disabled from the ViewModel via the
ChartModifier property and IsModifierTypeConverter -->
<s:SciChartSurface.ChartModifier>
  <s:ModifierGroup>
    <s:CursorModifier TooltipLabelTemplate="{StaticResource CursorLabelTemplate}"
IsEnabled="{Binding ChartModifier, Mode=TwoWay, Converter={StaticResource IsModifierTypeConverter},
ConverterParameter=CrosshairsCursor}" />
  </s:ModifierGroup>
</s:SciChartSurface.ChartModifier>
```

FastMountainRenderableSeries.AreaColor is renamed to AreaBrush

FastMountainRenderableSeries.AreaColor has been renamed to AreaBrush, and now requires type Brush, not Color. This RenderableSeries now supports gradient brushes (e.g. LinearGradientBrush) as well as SolidColorBrush. It also supports inline declaration of colors as hex codes just like any other WPF brush. Please note that Gradient Brushes, while they look amazing, do come at the expense of performance!

✖ 9 The property "AreaColor" is not a DependencyProperty. To be used in markup, non-attached properties must be exposed on the target type with an accessible instance property "AreaColor". For attached properties, the declaring type must provide static "GetAreaColor" and "SetAreaColor" methods. SciTraderView.xaml 95: 21 Abt.Controls.SciChart.Core.Wpf.Example

Before

```
<!-- Style for the Price mountain chart -->
<Style x:Key="PriceMountainSeriesStyle" TargetType="sciChart:FastMountainRenderableSeries">
  <Setter Property="SeriesColor" Value="#FFC6E6FF" />
  <Setter Property="AreaColor" Value="#774083B7" />
</Style>
```

After

```
<!-- Style for the Price mountain chart -->
<Style x:Key="PriceMountainSeriesStyle" TargetType="s:FastMountainRenderableSeries">
  <Setter Property="SeriesColor" Value="#FFC6E6FF" />
  <Setter Property="AreaBrush" Value="#774083B7" />
</Style>
```

Similarly, FastColumnRenderableSeries.AreaColor has been renamed to AreaBrush, FastCandlestickRenderableSeries.UpBodyColor/DownBodyColor have been renamed to UpBodyBrush/DownBodyBrush respectively. These now support gradients or solid colors.

RenderSurface renamed to RenderSurfaceBase

As part of a refactor, the type RenderSurface has been renamed to RenderSurfaceBase. If you are styling the RenderSurface please update your XAML accordingly.

Before

```
<!-- Gives the rendersurface a glow effect, similar to a VDU -->
<Style TargetType="s:RenderSurface">
  <Setter Property="Effect">
    <Setter.Value>
      <DropShadowEffect BlurRadius="5" ShadowDepth="0" Color="#FFB3E8F6"/>
    </Setter.Value>
  </Setter>
</Style>
```

After

```
<!-- Gives the rendersurface a glow effect, similar to a VDU -->
<Style TargetType="s:RenderSurfaceBase">
  <Setter Property="Effect">
    <Setter.Value>
      <DropShadowEffect BlurRadius="5"
        ShadowDepth="0"
        Color="#FFB3E8F6" />
    </Setter.Value>
  </Setter>
</Style>
```

Namespace Changes

As part of a Namespace refactor (necessary to output documentation by namespace), several of the core classes have been moved out of namespace Abt.Controls.SciChart and into child namespaces. The following is a list of namespaces which have been updated. Please adjust your source code accordingly.

Namespace	Description
Abt.Controls.SciChart.Visuals	Core SciChart controls such as the SciChartSurface, SciChartOverview, SciChartLegend.
Abt.Controls.SciChart.Visuals.Annotations	All annotation types, including CustomAnnotation
Abt.Controls.SciChart.Visuals.RenderableSeries	All renderable series, including BaseRenderableSeries, concrete RenderableSeries types and PaletteProviderBase IPaletteProvider
Abt.Controls.SciChart.Visuals.Axes	All Axis types, including IAxis, AxisBase and concrete axis implementations
Abt.Controls.SciChart.Model.DataSeries	All DataSeries types, from IDataSeries, to interfaces such as IXyDataSeries<TX,TY> to concrete types such as XyDataSeries<TX,TY>
Abt.Controls.SciChart.ChartModifiers	ChartModifiers have been moved to this namespace

Please note in XAML you only need one namespace and that is:

```
xmlns:s="http://schemas.abtsoftware.co.uk/scichart"
```

This namespace encompasses the entire SciChart library in XAML. In code-behind you will need to include individual namespaces where you use any of these components.

Controls Namespaces

✖ 11	The name 'SciChartSurface' does not exist in the current context	MainViewModel.cs	95 26	Abt.Controls.SciChart.Core.Wpf.Example
------	--	------------------	-------	--

SciChartSurface, SciChartOverview, SciChartLegend have all been moved to the Abt.Controls.SciChart.Visuals namespace.

AxisAlignment Namespace

This enumeration has been moved to the Abt.Controls.SciChart.Visuals.Axes namespace.

✖ 7	The type or namespace name 'AxisAlignment' could not be found (are you missing a using directive or an assembly reference?)	SeriesSelectionExampleView.xaml.cs	52 64	Abt.Controls.SciChart.Core.Wpf.Example
-----	---	------------------------------------	-------	--

LineDrawMode Namespace

This enumeration has been moved to the Abt.Controls.SciChart.Visuals.RenderableSeries namespace.

✖ 28	The name 'LineDrawMode' does not exist in the current context	GapsInSeries.xaml.cs	19 38	Abt.Controls.SciChart.Core.Wpf.Example
------	---	----------------------	-------	--

SourceMode Enumeration

This enumeration has been moved to the Abt.Controls.SciChart.ChartModifiers namespace.

✖ 3	The type or namespace name 'SourceMode' could not be found (are you missing a using directive or an assembly reference?)	RolloverFeedback.xaml.cs	12 62	Abt.Controls.SciChart.Core.Wpf.Example
-----	--	--------------------------	-------	--

LineAnnotation, LineArrowAnnotation etc...

All annotation types have been moved to the Abt.Controls.SciChart.Visuals.Annotations namespace

✖ 10	The type or namespace name 'LineAnnotation' could not be found (are you missing a using directive or an assembly reference?)	CreateAnnotationsDynamically.xaml.cs	14 40	Abt.Controls.SciChart.Core.Wpf.Example
✖ 11	The type or namespace name 'LineArrowAnnotation' could not be found (are you missing a using directive or an assembly reference?)	CreateAnnotationsDynamically.xaml.cs	15 45	Abt.Controls.SciChart.Core.Wpf.Example

GetLegendDataFor Enumeration Namespace

This enumeration has been moved to the Abt.Controls.SciChart.ChartModifiers namespace.

✖ 2	The name 'GetLegendDataFor' does not exist in the current context	MultipleLinesView.xaml.cs	14 39	Abt.Controls.SciChart.Core.Wpf.Example
✖ 3	The name 'GetLegendDataFor' does not exist in the current context	MultipleLinesView.xaml.cs	16 44	Abt.Controls.SciChart.Core.Wpf.Example

ExecuteOn Enumeration Namespace

✖ 4	The name 'ExecuteOn' does not exist in the current context	MouseDragToPan.xaml.cs	51 70	Abt.Controls.SciChart.Core.Wpf.Example
-----	--	------------------------	-------	--

The ExecuteOn Enumeration has been moved to the Abt.Controls.SciChart.ChartModifiers namespace

YAxisDragModifier/XAxisDragModifier.AxisModes

The AxisModes property (and enumeration) has been updated. No longer is there AxisModes.AbsoluteScale, AbsolutePan, RelativeScale, now there is just Scale, and Pan.

The YAxisDragModifier and XAxisDragModifier instead attempt to figure out whether to perform an absolute range (i.e. dragging affects the Axis.VisibleRange) or relative range (i.e. dragging affects the Axis.GrowBy) depending on the Axis.AutoRange property value.

Logarithmic Axis refactor

AxisMode.Logarithmic is Obsolete

The AxisMode enumeration still exists but the AxisBase.AxisMode property is now marked as obsolete. You will receive these errors if you are using Logarithmic Axis in SciChart:

✗ 3	The name 'AxisMode' does not exist in the current context	LogarithmicAxisView.xaml.cs	43 49	Abt.Controls.SciChart.Core.Wpf.Example
✗ 5	The name 'AxisMode' does not exist in the current context	LogarithmicAxisView.xaml.cs	44 49	Abt.Controls.SciChart.Core.Wpf.Example

```

private void YAxisLogChecked(object sender, RoutedEventArgs e)
{
    if (yAxis != null) yAxis.AxisMode = AxisMode.Logarithmic;
    if (xAxis != null) xAxis.AxisMode = AxisMode.Linear;

    if (sciChart != null) { s
}

private void XAxisLogChecked(object sender, RoutedEventArgs e)
{
    if (yAxis != null) yAxis.AxisMode = AxisMode.Linear;
    if (xAxis != null) xAxis.AxisMode = AxisMode.Logarithmic;

    if (sciChart != null) { sciChart.AnimateZoomExtents(ts); }
  
```

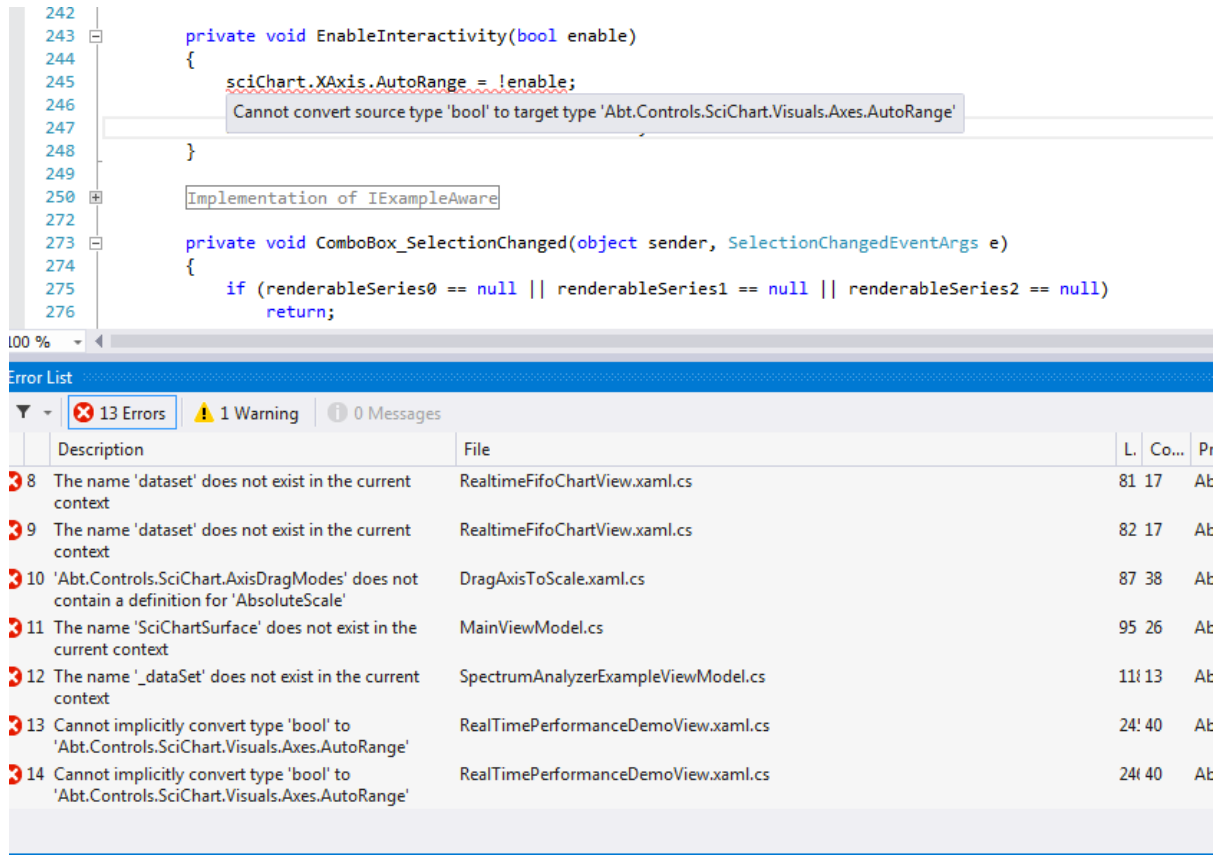
"We're sorry! AxisBase.AxisMode is obsolete. To create a chart with Logarithmic Axis, you must use LogarithmicNumericAxis instead."

Abt.Controls.SciChart.Visuals.Axes.AxisMode? (Alt+Enter)

Instead of setting NumericAxis.AxisMode = AxisMode.Logarithmic, you must now swap the NumericAxis for LogarithmicNumericAxis.

The reason why we have done this is so that we can add extra features specific to LogarithmicNumericAxis in the future.

AxisBase.AutoRange is no longer Boolean



The screenshot shows a code editor with the following C# code snippet:

```

242
243     private void EnableInteractivity(bool enable)
244     {
245         sciChart.XAxis.AutoRange = !enable;
246         // Cannot convert source type 'bool' to target type 'Abt.Controls.SciChart.Visuals.Axes.AutoRange'
247     }
248
249     // Implementation of IExampleAware
250
251     private void ComboBox_SelectionChanged(object sender, SelectionChangedEventArgs e)
252     {
253         if (renderableSeries0 == null || renderableSeries1 == null || renderableSeries2 == null)
254             return;
255     }
  
```

The Error List window shows the following errors:

	Description	File	L	Co...	Pr
8	The name 'dataset' does not exist in the current context	RealtimeFifoChartView.xaml.cs	81	17	At
9	The name 'dataset' does not exist in the current context	RealtimeFifoChartView.xaml.cs	82	17	At
10	'Abt.Controls.SciChart.AxisDragModes' does not contain a definition for 'AbsoluteScale'	DragAxisToScale.xaml.cs	87	38	At
11	The name 'SciChartSurface' does not exist in the current context	MainViewModel.cs	95	26	At
12	The name '_dataSet' does not exist in the current context	SpectrumAnalyzerExampleViewModel.cs	111	13	At
13	Cannot implicitly convert type 'bool' to 'Abt.Controls.SciChart.Visuals.Axes.AutoRange'	RealTimePerformanceDemoView.xaml.cs	24	40	At
14	Cannot implicitly convert type 'bool' to 'Abt.Controls.SciChart.Visuals.Axes.AutoRange'	RealTimePerformanceDemoView.xaml.cs	24	40	At

If you receive this error you need to adjust how you are setting Axis.AutoRange, both in XAML and in Code.

Before Code

```

private void EnableInteractivity(bool enable)
{
    sciChart.XAxis.AutoRange = !enable;
    sciChart.YAxis.AutoRange = !enable;
    sciChart.ChartModifier.IsEnabled = enable;
}
  
```

After Code

```

private void EnableInteractivity(bool enable)
{
    if (!enable)
    {
        sciChart.XAxis.AutoRange = AutoRange.Always;
        sciChart.YAxis.AutoRange = AutoRange.Always;
    }
    else
    {
        sciChart.XAxis.AutoRange = AutoRange.Auto;
        sciChart.YAxis.AutoRange = AutoRange.Auto;
    }

    sciChart.ChartModifier.IsEnabled = enable;
}
  
```

That's It!

That's all folks! We hope you enjoy using SciChart v2.0 and can see the vision of where we are taking this product. As always if you have questions please contact us, or [post on our forums](#). We are more than happy to help!

Do you have a Testimonial you would like to share?

We don't like to offer special prizes or monetary incentives to share testimonials, however if you have had good experiences with SciChart and your application has been well received by users, we'd love to hear from you. In return we can offer *even more* special treatment of support requests and you'll¹ get a warm feeling inside that you helped out our business!

If you want to share a case study then that would be great – we can host videos on our site or articles talking about your product and what went well with SciChart – a bit of bi-directional marketing. Contact us if you are interested!

¹ Warm feeling not guaranteed