Volume

1

Software Reference Manual

Updated through Version 2.0 build 59

Table of Contents

Part 1 – Core Concepts

Introduction

Welcome to the world of Blocks. Blocks is a powerful and flexible data analysis platform that brings data and information to your fingertips unlike any other software. At its core, it is a unique platform that combines the power of Microsoft .net programming in a dynamic run-time assembly environment.

If these words sound scary, fear not. Blocks is an easy to use and intuitive software platform for viewing and analyzing data. It comes pre-configured with hundreds of technical and fundamental market strategies and tools, along with hundreds of basic and advanced calculations to give you the best overall and detailed picture of the market. While Blocks at its core is a modular environment, you can use it "right out of the box" with our pre-configured workspace, Personal Chartist. Blocks' unique QuickEdit interface allows you to configure parameters and indicators to find the best settings for your analysis, without needing to rewrite the code behind the calculation. As you get more comfortable using Blocks you can combine the pre-existing pieces to create literally thousands of unique combinations. Part 1 of this reference manual guides you through the features and tools that come pre-configured with the Blocks software platform.

As you get more comfortable with Blocks, you will want to customize your working environment to fit your own style. Blocks uses a new flexible docking layout system that gives you a unique dashboard to view all the tools, charts, lists and data. Because it's fully customizable, you can set up your desktop in a way that makes you the most productive. Part 2 explores the customization and Layout system included in Blocks.

Eventually you may wish to dive "under the hood" and explore one of the main benefits of the Blocks platform: the Block Diagram. The Block Diagram is a unique and easy-to-use patent-pending visual flowchart designer. Nearly every part of the software is controlled via a Block Diagram. If you can understand a flow chart, you can "program" in Blocks with the Block Diagram. Part 2 of this manual introduces you to the Block Diagram, while Part 3 takes you through the more advanced features of Block Diagrams.

If you are comfortable with other scripting languages or if you are familiar with C # or VB.net programming then you can harness even more power from Blocks by creating your own code blocks from source code. Blocks has a unique run-time edit and compile engine that allows you to introduce and edit code into diagrams while the application is running. This lets you visually change and update your calculations on the fly.

The Block Diagrams connect real Microsoft .net compiled code. It does not use any scripting language. Scripting Languages are notoriously slow as they need to be "interpreted" by the main application and they require to you to learn a non-standard programming syntax. Using .net (pronounced dot net) you have access to nearly the entire .net programming framework (some namespaces have been restricted for security) and can leverage existing code examples, training, websites and hundreds of published books on writing .net code.

Because .net is compiled code, Blocks gives you unparalleled processing speed

(compared to a scripting language). Combine the code blocks with the simplicity and flexibility of a run-time assembly through Block Diagrams and you have the most powerful, flexible data analysis platform on the market.

This manual is not an introduction into technical analysis, the stock market, or how to make money. This manual is a reference only and documents each of the main tools and features of the Blocks software.

Chapter 1: Introducing Blocks

The main entities of the Blocks Software are:

Workspace: A workspace is a collection of Tools that you can save and load. It's the basic working area. Think of it as your desktop where you use your tools. For more information on Workspaces, see page 12 or page 49.

Tools & Windows : Tools are similar to traditional applications that run in your Windows XP operating system. Every Tool is hosted in a Tool Window that is part of a workspace and can be docked and positioned anywhere you choose. Example tools are Stock Charts, Web Browsers, WatchLists, Pie Charts, Time & Sales tools etc. For more information on Tool Windows see page 63.

Tool Parts (plots, panes, columns etc): Each Tool can have parts. A tool part is a generalization for any sub item of a tool that can be saved to disk and loaded into the tool. For example a Chart can have plots, A WatchList can have columns. A scan can have conditions.

Blocks: Blocks are simply calculations that consume and/or provide data. You can think of them as basic functions like addition, subtraction and multiplication or as complex as physics, weather modeling, and space navigation. For more information on blocks, see page 69.

Block Diagrams: Diagrams are the assembly of blocks to form an algorithm or data flow. Blocks Diagrams always consume or supply data. For instance a diagram could represent a basic calculation like 5 + 2 or provide the Daily Open, High, Low and Close data for a stock symbol or calculate the average rainfall for each region of the world. For more information on Block diagrams, see chapter 6.

Installing Blocks

Installing from the web

If you haven't already, open your web browser and navigate to http://www.Blocks.com/Download .Once there, click the Download button.
On the File Download- Security Warning window, click the Run button to start the Installation Process.

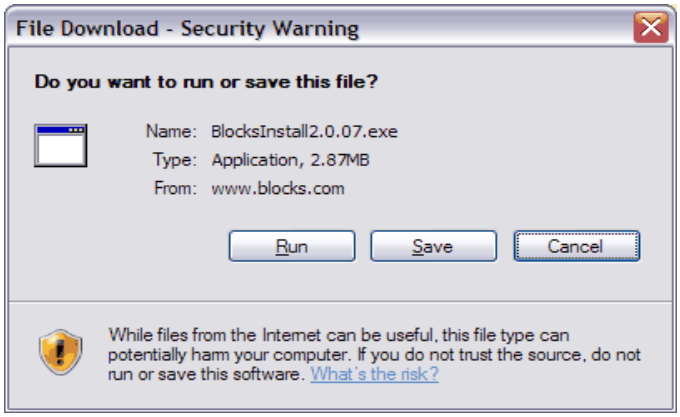


Figure 1 - Internet Explorer File Download Dialog

Installing from a CD

Insert the CD ROM into an available CD ROM or DVD drive. Navigate to the CD drive and run the Setup.exe file.

Installation Instruction

Read the license agreement and follow the installation instructions choosing the installation location on your hard disk.

Starting the Program

Start the Blocks software by clicking on Start – All Programs - The Blocks Company – Blocks. Alternatively you can click on the Blocks icon on your desktop

Signing In

When you start the program for the first time, you will be prompted to provide your Sign In information (Figure 2). This is the UserName and Password you created on the Worden.Com website. If you are currently a TeleChart customer and have TeleChart installed on the same computer as Blocks, it will read your Sign In information from TeleChart and pre-populate the fields.

If you do not know your information, simply click on the *I forgot my Sign-In information* link below the buttons.

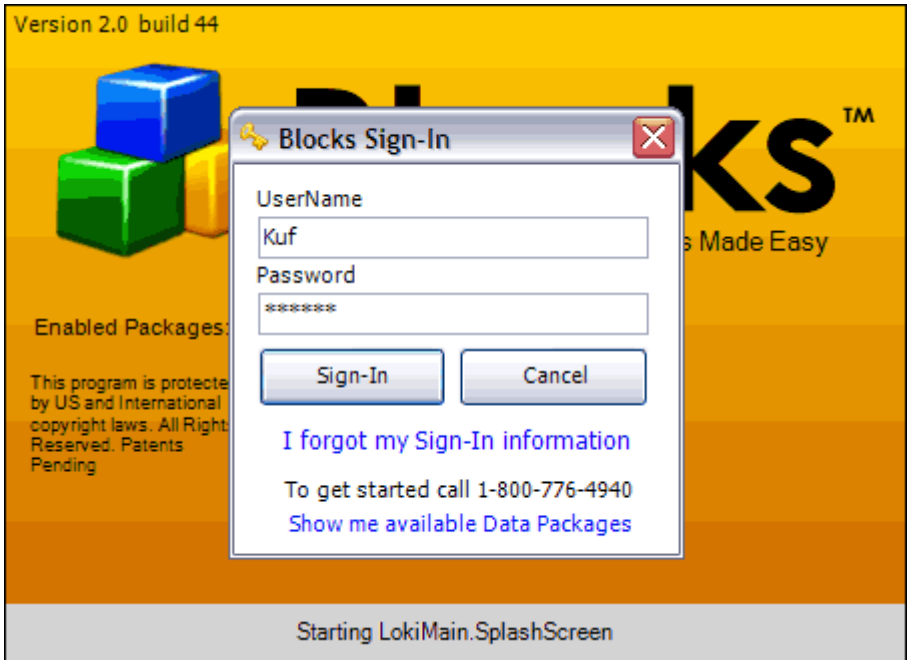


Figure 2 – Blocks Sign-In

Downloading and Installing Purchased Data Packages

After you have signed in successfully, Blocks will check the data packages you have purchased and download any additional data needed to run.
As of the time of this printing the current Data Packages are: Strategy Trader, Mega Minute and Fund Investor. You will be prompted to download and install each package you have purchased. This will be a one time process.

Open Workspace

When you first open Blocks, you will be presented with the Open Workspace dialog (Figure 3). Figure 3 is a common file dialog that is used in many places in the program. The tabs are always the same, and you can type to filter the results.

Personal Chartist and *Mega Minute OneTouch Trader* are two workspaces we have pre-configured with some basic settings. The Tabs down the side of the dialog allow you to from the System Library (default), your own computer (My Computer) or from recently used workspaces.

The *Mega Minute* workspace is the same as *Personal Chartist* with some more real-time columns added to the WatchList. The majority of this manual will use *Personal Chartist* when referencing screenshots of the program. Select the workspace of your choice to get started.

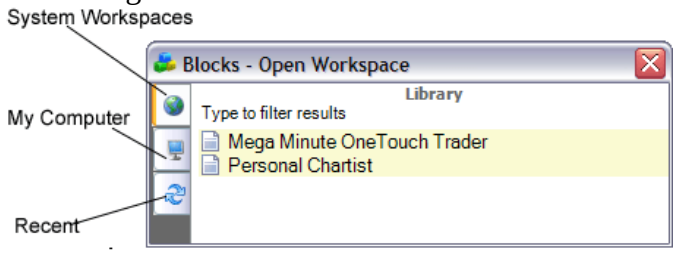


Figure 3 – Open Workspace

Workspace Introduction

Figure 4 is the *Personal Chartist* Workspace (it may not look exact as it my have been modified after the time of this printing). Let's go over a few of the basic features of the workspace.

A Workspace is a collection of Tools that are positioned in the main window frame (or additional frames). Personal Chartist comes with 3 tools by default: WatchList, Bar Chart and Google Finance Company Info. Additional tools can be added and positioned anywhere you like. For more information on positioning tools, see page 64.

You can add additional tools to the workspace by using the Start Button from the main toolbar. See chapter 4 for more details.

A workspace can also have a set of hotkeys to perform actions on the tools. These can be set under the System – Hotkey manager. For more information on Hotkeys see page 66.

Figure 4 – Personal Chartist Workspace

Tool introduction

Let's start with the Tools that are in the Personal Chartist workspace. There are 3 main tools, from left to right they are labeled as: WatchList, Bar Chart, Google Finance Company Info.

The WatchList Tool (figure 5) is everything on the left hand side of the Workspace. It contains a WatchList selection (pull down) currently on the "*Nasdaq 100 Component Stocks*" WatchList. It also has the following near the top of the tool: *WatchList* button, *Strategy* button, *Performance* button, *Main List* tab, Industry (*Ind*) tab, Sub-Industry (*Sub*) tab and *TD* AMERITRADE current positions tab.
The Strategy button allows you to perform scans and back tests on an entire list of symbols in a WatchList using a series of conditions.

For more information on Scanning and back testing see page 49

The bottom part of the WatchList tool that takes up the majority of the visible tool area are the symbols in the selected WatchList and the currently loaded WatchList columns (Symbol & Company Name)

For more information about the WatchList tool, see chapter 3.

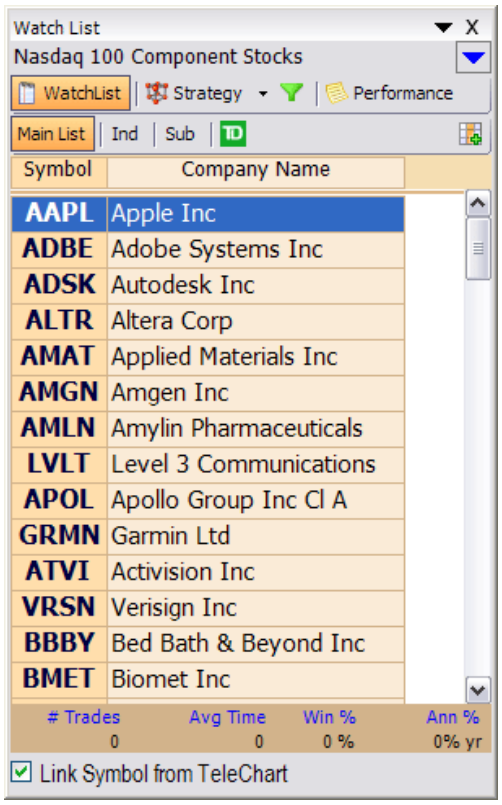


Figure 5 – WatchList Tool

The Bar Chart Tool (Figure 6) is the large black chart that takes up a majority of the viewable space in the Workspace. It has a toolbar along the top that include many features to manipulate the chart. The two that you will use most are the Add Study button and the Time Frame selection.

The Add Study button allows you to add additional studies (indicators) to your chart. The Add Study button lists the hundreds of indicators in the System Library and also allows you to choose any saved studies from your hard drive.
The Time Frame selection allows you to choose a frequency (Day, Week, option expiration Friday) and a number (where appropriate). For a Daily chart, choose 1 Day.

For a 3 day chart choose 3 Day etc. If you are on an intraday time frame choose the streaming checkbox to have the chart update on every tick.

See Chapter 2 for more information on the Chart Tool

Figure 6 – Bar Chart Tool

Tabbed behind the Bar Chart is the Google Finance Company Info (Figure 7). This is a web browser that is configured to show the Google web page for the currently selected symbol. There are many more browser tools you can add to your workspace using the Start button. See page 63 for more information on adding Tools the workspace.

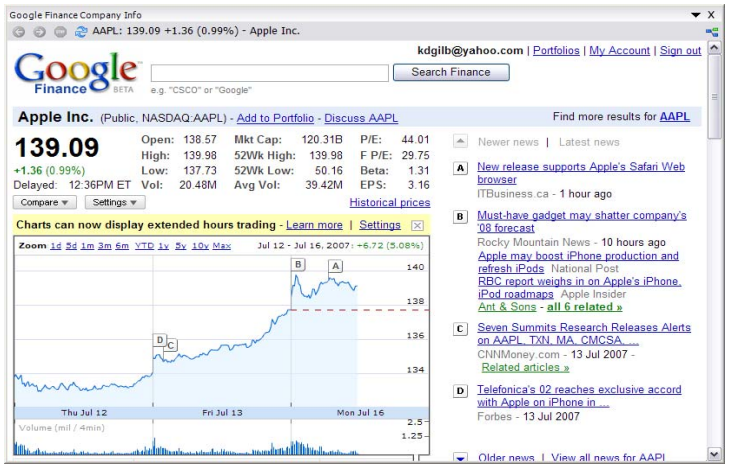


Figure 7 – Google Finance Browser Tool

Chapter 2: Chart Tool

The Bar Chart tool displays studies in a multi-pane format. This tool is simple to use out of the box, and is very customizable to suit your personal preferences.

Chart Properties Overview

The Chart tool has a number of properties that can be set to control the display and behavior of the chart. You can access the chart properties by right clicking on the chart background and selecting Properties – Chart or by Clicking on the down arrow in the window header ▼ and selecting Properties.

Table 1 lists the common and basic properties of a chart. See Figure 8 for a complete list and screenshot of the properties.

Back Color	The background color of the chart
Extra Left Margin	The number of pixels to buffer the left side of the chart
Extra Right Margin	The number of pixels to buffer the right side of the chart
Legend Font	The font to use for the legend of studies
Legend Position	Location to put the legend in a pane
Legend Width	The default width of the legend
Name	The unique name of the chart in the workspace
Zoom By	When using the zoom slider, the means of zooming. Options are PlotCount (Default) or date span.

Table 1 – Basic Chart Properties

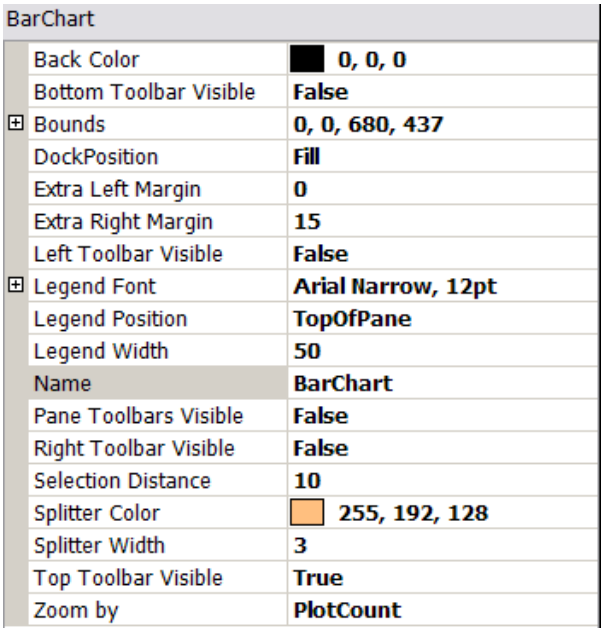


Figure 8 – Bar Chart Properties

Top Toolbar

The top toolbar contains controls and labels for manipulating the chart and displaying some data. Figure 9 displays each of the items on the top toolbar starting with the far left button. The top toolbar can be toggled on/off by right clicking on the chart and selecting *Toolbars – Top* from the menu

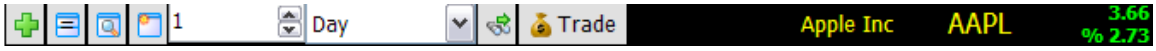


Figure 9 – Top Toolbar

Add Study Button

The *Add Study* button allows you to choose from the hundreds of indicators in the library, or any personal saved studies from your local hard drive. For more information on adding studies, see page 19.

Equalize Panes

The *Equalize Panes* button sets all the panes in the chart to the same height.

Show All Panes

The *Show All Panes* buttons will un-tab any pane that has been tabbed to the side. For more information on tabbing panes, see page 28.

Create New Plot in New Pane

Creates a new (empty) plot in a new pane. Use this for creating a new calculation from scratch.

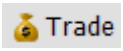
Time Frame Picker

The Time Frame picker allows you to choose the timeframe of the chart. The S checkbox will cause the chart to update on the tick (streaming) if you have real-time data enabled (Mega Minute).

Toggle Data Display

Toggles the right toolbar that displays the Open, High, Low, Close, Volume, Net and Percent change for the chart at the date the pointer is currently on.

Trade Button



Toggles the OneTouch™ Trader from the right side of the chart and opens a new Trade Ticket. For more information on OneTouch™ Trading, see chapter 11.

Company Name

Displays the name of the company the Bar Chart is displaying data for

Symbol

Displays the Symbol the Bar chart is displaying data for

Net/Percent Change

Displays the Net and Percent change for the last bar on the chart (time frame dependant. IE 1 minute bars will show the last minute net/% change)

Main Pane

The Main pane is the first and default pane on the chart. A chart can have an infinite number of panes to display studies (See figure 10). Think of a pane as a mini window on a chart that fills the width of the chart but can have a variable height. For more information on panes, see page 26. The Main pane has two visible plots: Current Symbol in the legend, the Green Price History (displayed as HLC Bars) and a 50 period moving average.

For more information on plots, see page 22.

Figure 10 – Main Chart Pane

Date Scale

The date scale is below the chart and displays all date information for all studies across all panes (see figure 11). If you have items on multiple time frames, the large time frames will display on the appropriate frequency in relation to the small ones (IE a study on a minute time frame and daily time frame can be on the same chart. The daily will plot on the appropriate days).

For more information on the Date Scale, see page 34.

Figure 11 – Date Scale

Value Scale (Scale Display)

The value scale displays the value information on a specific scale of the pane. Each pane can have 1 or more scales. By default, a pane comes with one scale and one scale display.

For more information on Scale Displays, see page 30.

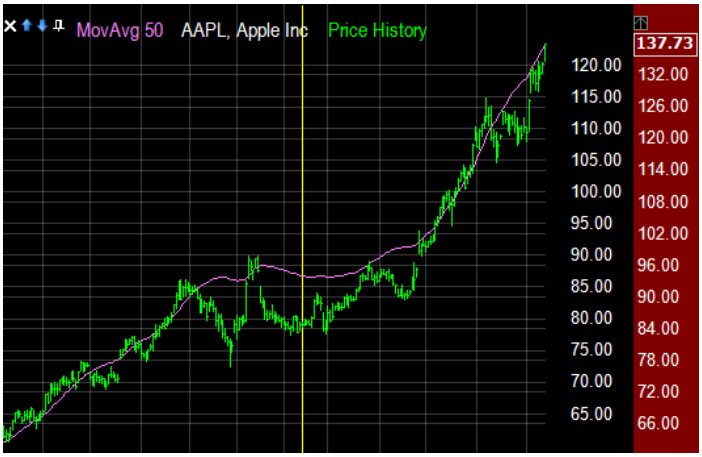


Figure 12 – A Chart with two scale displays on the right. One has a red background.

OneTouch™ Trading

OneTouch trading is an on-chart trade ticket that displays your current position(s) with the number of shares and the net gain, and any pending orders for the selected symbol you have on the table. New orders can be placed directly on the OneTouch™ Trading and existing orders can be modified.

For more information on OneTouch™ trading, see chapter 11.

Figure 13 – OneTouch™ Trading. On chart position information and order entry.

Date and Zoom Sliders

The Date and Zoom sliders are along the bottom of the chart. The Zoom controls the number of dates or bars that is visible at any one time. The Date Slider controls the last date visible on the chart. The top slider is the date slider and the bottom controls the Zoom (figure 14)

Figure 14 – Date and Zoom Sliders

Chart Specifics

Add Study Button

Adding additional studies from the library

The Blocks software comes with hundreds of studies and indicators, and thousands of combinations (through the ability to daisy chain indicators together). The list of available indicators is too much to list here, but you can always find information about each indicator at http://kb.worden.com. The Library is automatically updated each time you start Blocks and each item is available for online or offline use.
Many indicators are "child" indicators (like moving average, channels and Bollinger bands) that apply a calculation to an existing study. Studies can be chained off of each other to form calculations like the following: The MACD of Stochastics of P/E ratio.

To add a study to your chart, click on the *Add Study* button from the toolbar at the top of the chart. You will be presented with a list of studies (figure 15). Any studies you have loaded recently will be highlighted in yellow.

From the Library list, you can begin to type the name of the study that you wish to load and the list will be filtered to match. Alternatively, you can browse the folders or the list to see all the studies that are available.

To load a saved study from your hard drive, use the *My Computer* tab on the left hand side of the list.

Once you have selected the study to load, simply click on the study name to add it to your chart.

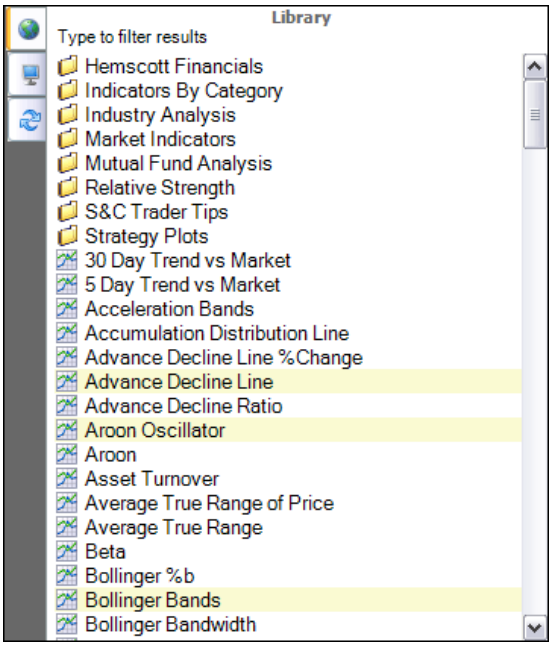


Figure 15 – Add Study Library List

Some Studies may require more data to calculate. For instance when adding Bollinger Bands to your chart, you will be presented with a list of studies already on your chart (Figure 16). Select the indicator you would like to apply the Bollinger Bands to.

Figure 16 – Adding a child indicator (Bollinger Bands) requires another plot. Select the plot from the list.

When adding the child indicator, if it does not come on its own pane, it will be added to the same pane and scale layer as the indicator you choose to apply it to.

For more information on Scale Layers and Panes see page 26.

Adding personal saved studies

Adding personal studies is no different from adding a study from the library. Click the *Add Study* button. You will be presented with the Library of indicators (Figure 15). Choose the second tab on the left (My Computer) . This will display all the panes and plots that have been saved to your hard drive (Default save location is My Documents\Blocks Files\Tool Parts\Chart)

You can type to filter the results, or navigate your folders. To choose a different disk location, use the … button in the upper right corner to pick the drive/folder to display (Figure 17).

For more information on saving studies to your computer, see page 22.

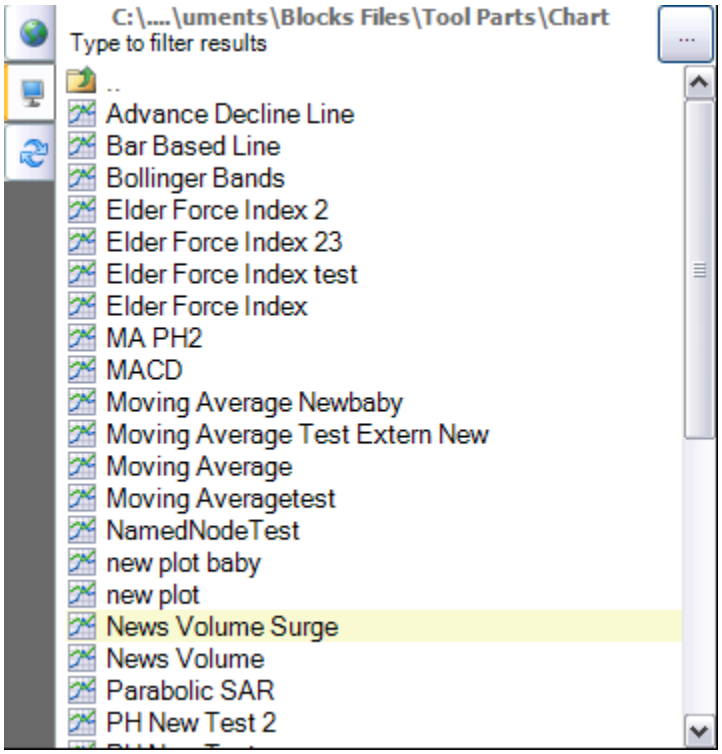


Figure 17 – Browse My Computer for Saved Studies

Studies (Plots)

A Study (also referred to as a plot or indicator) is the squiggly line that is drawn on a chart that represents some time & value based data, where the date/time is displayed horizontally on the X axis and the Value is displayed vertically on the Y Axis. Every study in Blocks is derived from DateLineSingle. Every plot has a diagram to determine the data to display on the screen (see chapter 6 for more information on block diagrams).

Loading/Saving

You can load a study by using the Add Study button (see page 16) or right clicking on the chart and selecting *Add Study – Open*.

To save any changes you have made to a plot, simple right click on the legend name, or right click on the plot itself and choose Save As.

There are hundreds of studies available in the Blocks library (see page 16 for more information on loading existing plots) but if you have a new calculation you wish to create from scratch, you can use the New Plot option (see page 26 for more information on creating new plots)

Scale With…

Plots can be on the same scale or on their own scale. Plots that share a scale will both contribute to the High and low value and will be plotted relative to one another.
To scale a plot on its own, right click on the plot and choose *Scale With - on own scale*.
To scale a plot with another plot, right click on the plot and choose *Scale With*, and then pick the plot(s) that you wish to scale with.

QuickEdit

Quick Edit allows you to change properties of a plot, including (but not limited to) the plot color, Line style and any additional parameters of a calculation (additional parameters might be the period of a moving average, or the width of Bollinger bands)
You can bring up quick edit for any plot by left clicking on the line or the label (legend) of the plot.
Figure 18 shows the quick edit for a Price History plot. From here you can change the plot color, Style, Refresh Rate and show/hide the plot.

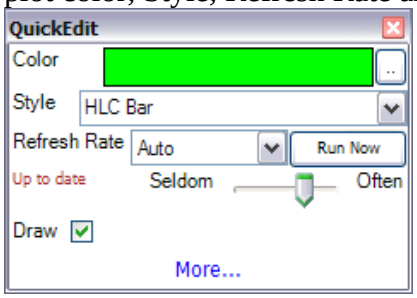


Figure 18 – Plot Quick Edit

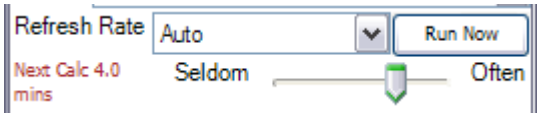
Plot Refresh Rate and Plot Calculation Status

The Plot refresh rate determines how often to perform the calculation for a plot. The values are Auto, Manual and Always. Normally calculations occur very quickly and take only milliseconds to perform. Sometimes, depending on the complexity of the calculation or on the number of symbols that is being calculated (like a list average for a large symbol list), it can take anywhere from a few seconds to a few minutes to calculate. If you are performing the calculation using real time data, the data could "tick" before it has finished calculating. If the refresh rate was set to Always, this would cause the calculation to immediately recalculate and end up bringing your computer to a halt as it constantly recalculates.

Setting this to Auto (default & recommended) will cause it to recalculate based on the amount of time it needed to calculate on the previous change. The update frequency (seldom to often) is a multiplier applied to how long it took to perform the initial calculation. Using the Auto setting will increase performance on both slow and fast machines. Slow machines will take longer to calculate and will therefore wait longer between calculations while faster machines will finish sooner and update the values sooner.

Manual lets you choose a manual interval, ignoring how long it takes to calculate a study.

Plots that are currently calculating will have a calculating icon. Plots that have new data but are waiting to calculate will have a timer icon next to their legend. If you see this icon and wish to perform the calculation immediately, or wish to see how much longer it will wait before calculating, click on the plot to view QuickEdit. It will list the time until next calculation and will allow you to choose the Run Now option to force it to recalculate



Show/Hide Study

You can quickly show or hide a plot by choosing the Draw checkbox on a plot's QuickEdit. If you have hidden a plot you can re-display the QuickEdit dialog by clicking on the plot name in the legend (top of pane). You can also map a hot key to toggle displaying a plot. See chapter 4 for more information on mapping hot keys.

Copy/Paste Study

You can copy a plot from one pane, and paste it into another pane using the Copy / Paste option. Alternatively you can copy the block diagram by using the Copy Logic option (for more information on Block Diagrams, see chapter 5).
To copy a plot, right click on the plot or on the legend of the plot and choose copy.
To paste a plot in a different pane, right click on the pane and click paste (after performing the copy steps above).

Plot Type

There are 4 different types of plots you can add to your chart:

Numeric Plots

Numeric plots are a series of dates and numbers. Most stock market and technical analysis indicators are Numeric plots. The seven line styles you can apply to a numeric chart are listed in Table 2.

Action Plots

Action plots are similar to line plots but instead of having a numeric value at a given date, they simply mark the start and end date of some event. An example of an action plot would be stock trades (Buy and Sell of a stock would be one action). Action plots are displayed as an up and down arrow with a shaded region between the start and end of the action.

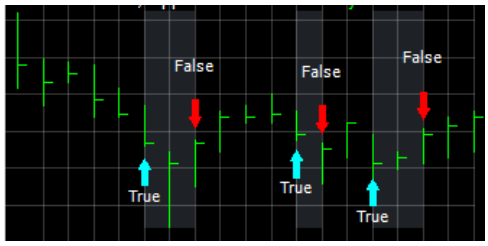


Figure 19 – Action Plots

Channel Plots

Channel Plots are the equivalent of two Line plots (one for the channel top, the other for the channel bottom) and are shaded between the two lines. An example of a Channel plot would be Bollinger Bands

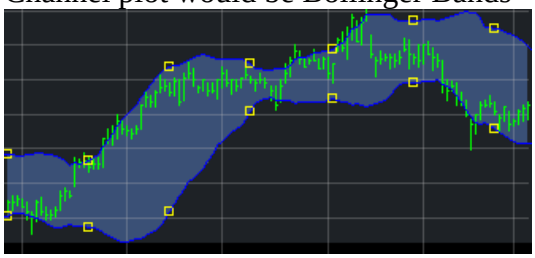


Figure 20 – Channel Plot

True Markers

True markers display when something is true on a given date. An example would be Splits, Dividends, Note Markers, and News Markers.

Figure 21 - True Markers

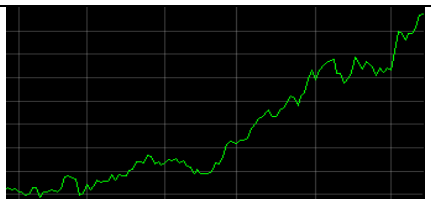
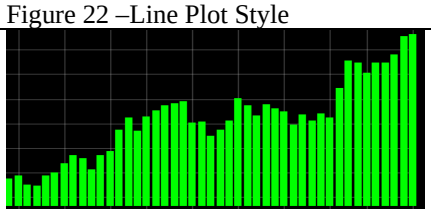
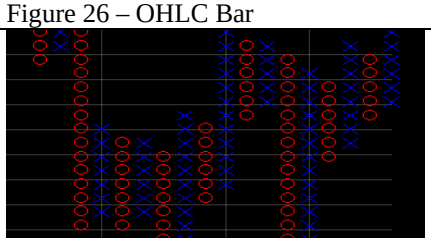
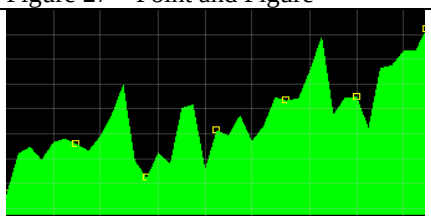
Line A basic line chart that connects each date to each Value	 Figure 22 –Line Plot Style
Bar A traditional bar chart that extends from the bottom of the chart to the value for the date.	 Figure 23 – Bar Style
Candle A market Candlestick plot that displays the open, high, low and last value for a date a Candlestick format	 Figure 24 – Candlestick Style
HLC Bar Displays the High and low for a date in a vertical line, with a horizontal line extending to the right for the last value. If only close data is available, it will render as a line	 Figure 25 – HLC Bar
OHLC Bar A HLC Bar with a horizontal line extending to the left for the open value of a given date If only close data is available, it will render as a line	 Figure 26 – OHLC Bar
Point and Figure A point and Figure plot draws X and O values to represent an up trend and a down trend. Use in combination with the point and figure bar builder block	 Figure 27 – Point and Figure
Shaded Region A line plot that will shade above or below the line to a specified value. Can be used like an area plot if the shade value is set to 0	 Figure 28 – Shaded Region

Table 2 –Line Styles

Creating New

To create a new plot from scratch, right click on the chart and select *Create New – Plot* or *Create New – Plot in New Pane*. You will be presented with the New Plot dialog (Figure 29); from here you can name your plot, choose the plot type and choose the scale group to scale the plot with. Click *Ok* to accept your settings. You will be presented with the Block diagram for your new plot. For more information on block diagrams, see chapter 6.

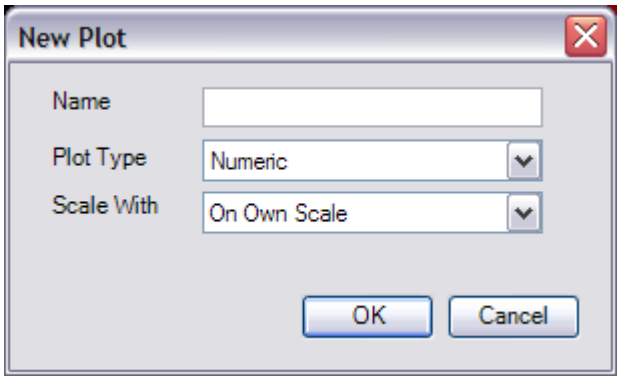


Figure 29 – New Plot Dialog

Plot Properties

Plots also have a number of properties that you can set (right click – Properties). Some of these are also configured from the QuickEdit.

Dash Style	The line dash style for the plot
Enabled	Same as Draw in quick edit
Legend	The text to display in the legend for this plot
Line Thickness	The thickness of the plot
Name	The Unique name of the plot
Refresh Rate	The Plot refresh rate, use the QuickEdit to set this value
Show Name in Legend	Set this to true to use the Name property in the legend
Show Old Results while calculating	Set this to True to display the old results while calculating new values. If set to false, the plot will disappear as it calculates. Setting this to True can reduce flicker on constantly updating plots
Style	The line style to use from Table 2.

Panes

A chart contains one or more panes. Panes are the canvas that contains 1 or more plots. They take the entire horizontal width of the chart but have a variable vertical height. The plots on a pane are grouped with each other for scaling, or each plot can be scaled independently. As such, a pane can have one or more scale displays to display the values of each scale layer. When multiple panes are on a chart, you can adjust the height of a pane by dragging the separator between panes, or using the mouse wheel to change the size of the pane under your mouse.

Figure 30 – A chart with two panes.

Each pane has a pane control box in the upper left corner (Figure 31).

Figure 31 – Pane Control box

The X will remove the pane from the chart and will no longer perform any calculations or share any results of its calculation (see chapter 9 for daisy chaining (linking) data calculations).

The blue up and down arrows will adjust the order of the pane (top to bottom) relative to the other panes on a chart.

The thumb-tack icon will pin the pane (also called tabbing a pane) to the left side of the chart. This will hide the pane temporarily on the side while still allowing the calculation to run. (Figure 32)

TIP: Panes that are tabbed to the side will still perform their calculations. This can affect system performance in a negative way if too many panes are pinned to the side. If you are finished with a pane, delete it from your chart.

Figure 32 – A tabbed pane. Note the Volume bars tabbed to the left of the chart

Saving a Pane as a study

If you have made modifications to any of the system studies (see page 22 for more information on loading studies) and would like to save those changes to load at a later date, you can right click on the pane, and choose Save Pane.

NOTE: Any external data connections you have made in your pane will be saved as a reference and may not be available when you load the pane back (see chapter 9 for more details on linking data in diagrams)

Adding Additional Scale displays

If you have multiple plots on a pane that are not scaled with each other, you may wish to load a second or third scale display to view the values of the other scales. You can add additional scale displays by right clicking on a Pane and choosing *Create New – Scale Display*. For more information on configuring a scale display, see page 30.

Figure 33 – A pane with two scale displays (yellow and red)

Copy / Paste Panes

You can copy a pane and paste it into any chart by using the copy/paste options. Simply right click on a pane and choose *Copy – Pane*. Choose the chart you wish to paste the pane into (it can be the same chart to make two copies of the pane), right click on the chart and select *Paste*. It will make a duplicate copy of the pane with all the plots and all their settings.

New Pane

To create a new pane, right click on a chart and *choose Create New – Plot in new Pane.*

Removing a Pane

To delete a pane from the chart, you can use the X in the upper left corner of the pane, or right click on the pane and choose *Remove Pane*. For more information on controlling the pane display see page 26.

Value Scale (Scale Display and Grid Lines)

A Scale display shows the values for a given set of plots on a pane. Each plot on a pane can have its own scaling or can be grouped together to display relative to each other on the same scale (For more information on Scaling plots see page 22). By default, a new pane comes with a Scale display for displaying the Y Value scale of a plot. The Scale Display also draws the gridlines on a pane for a give scale.

You can customize a scale display by right clicking on it and choosing, Properties. Table 3 lists the scale display properties (Figure 34 shows a screen shot of the properties).

Align Text Left	True/False. Aligns the Left if true, Right if false
Background Color	The background color of the scale display
Border Color	The border color of the scale display
Draw Border	If set to true, draws a border around the scale display
Fill Background	IF set to true, draws the background color, if false, draws the chart background color
Grid Color	The color of the gridlines on the chart
Grid Lines	If set to true, draws gridlines for the scale on the pane
Grid Mode	Sets the scale line spacing method to use when drawing the scale values and grid lines
Grid Opacity Percent	A number from 0 to 100 to determine the opacity of the grid lines. 0 is transparent, 100 is opaque
Name	The name of the scale display, unique to the current workspace
Number of Grid Lines	The number of grid lines to use when Grid Mode is set to LineCount
Orientation	Right, Left and None. Right displays the scale on the right side of the pane (default) Left shows it on the left side of the pane
Percent between grid line	Value to use when Grid Mode is set to FixedPercent
Percent display color	Percent color to use to render the % label at the bottom of the scale when Grid Mode is set to AutoPercentScale
Text Color	The color of the values on the scale
Text Format Method	The String format code to use when formatting the decimal numbers on the scale
Value between grid lines	The value to use when Grid Mode is set to Divisor

Table 3 – Scale Display Properties

Value Scale Display	
Align Text Left	**True**
Background Color	**255, 255, 128**
Border Color	**Silver**
Draw Border	**False**
Fill Background	**True**
⊞ Font	**Arial, 10pt**
Grid Color	**Silver**
Grid Line Style	**Solid**
Grid lines	**True**
Grid Mode	**AutoDivisor**
Grid Opacity Percent	**30**
Name	**Value Scale Display**
Number of grid lines	**5**
Orientation	**Right**
Percent between grid line	**1**
Percent display color	**255, 255, 128**
Text Color	**Black**
Text Format Method	**0.00**
Value between grid lines	**1**

Figure 34 – Value Scale Display Properties

Scale Method

The scale method of a Scale Display sets the method to use when Auto scaling. Setting this value decides what mathematical calculation to use determining the space between values. Valid methods are Arithmetic – Exponential and Logarithmic calculations. To change the Scale Method, right click on a scale display and choose *Scaling Method* - and then select the appropriate method. Some scaling methods have additional properties you can set; you can choose the *More Properties* option under the Scaling Method sub menu.

Scaling Examples

Manual Scaling 1 to 100

To set a manual scale from 1 to 100, right click on a scale display and choose *Scale Method- More properties*. Set the scale mode to *Manual*. Set Min value to 1 and the Max value to 100.

Logarithmic scaling

On a Logarithmic scale the distance between each value on the scale is a fixed percentage. The percentage will vary depending on the volatility of the values on the scale. To use logarithmic scaling, right click on a scale display and choose *Scaling Method – Logarithmic*

Arithmetic Scaling

Each value is a fixed distance from the next value on the scale. For example, each line may represent a 2.00 increase in price.
To use Arithmetic scaling, right click on a scale display and choose *Scaling Method – Arithmetic.*

Exponential Scaling

To use Exponential scaling, right click on a scale display and choose *Scaling Method – Exponential.*

Value Pointer

A value pointer is a horizontal line across a pane, marking the value at a particular position. A Scale display can have 0 or more value pointers. Value pointers can be used for display only (to show a specific or calculated value) or can be used for choosing a value to use in a calculation. Figure 35 shows a red value pointer at $111.86.
When used in combination with a date pointer you can achieve a crosshair to view the value at a particular date. For more information on date pointers, see page 35.
If a value pointer has dragging enabled (default) you can drag the pointer value and get the date and value for the point under the mouse (Figure 36).

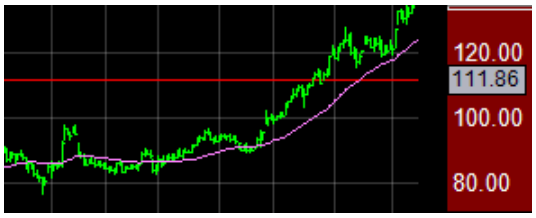


Figure 35 – A Value pointer on a chart

Figure 36 – Dragging a value pointer will show the date and value at the mouse position

Examples of value pointer uses are:

- To display the latest price on the price scale
- To display the target price based on a calculation
- To choose the top or bottom of a range for shading (On a Stochastics plot)
- To choose the value to use in a calculation (example above the price on a value pointer)
- To display the value at a given point on the chart

To view the properties of a value pointer, right click on the value display and choose properties. You will be presented the properties as seen in figure 37. Table 4 describes each of the properties and their use.

If you wish to programmatically set the value of a pointer, you can use the block diagram (also accessible by right clicking on the display and choosing properties). For more information on block diagrams, see chapter 6.

If a value pointer is off the scale, a small box with an arrow will indicate that a pointer is off the scale. Simply click and drag the box to bring the pointer back into view

Adding additional Value pointers

To add a value pointer to your scale display, right click on the scale display and choose Add Tool – Value Pointer.

Allow Drag	When set to true, you can drag the pointer to any date on the scale. When false, you cannot change the value with your mouse
Back Color	The backcolor of the value pointer on the Scale Display
Dash Style	The dash style of the line to use
Draw Value Box	If set to true, draws the label on the Scale Display, if set to false, only draws a vertical line across the pane
Label Only	If set to True, does not display the line across the pane, only displays the value on the Scale display.
Line Color	The color of the horizontal line
Line Opacity	The opacity of the line from 0 to 100. 0 = transparent, 100 = opaque.
Line Size	Number of pixels high to draw the horizontal line
Name	The name of the item, unique to the workspace
Outline Color	The border color of the label box on the scale display
Step Size	The amount to move up or down when dragging the pointer
Text Color	The color of the value on the scale display
Text Font	The font to display the value on the scale display
Text Format	The String Format code to use when displaying the value
Value to draw line at	The manual value to draw the line at (can be changed via diagram or by dragging if AllowDrag = true)

Table 4

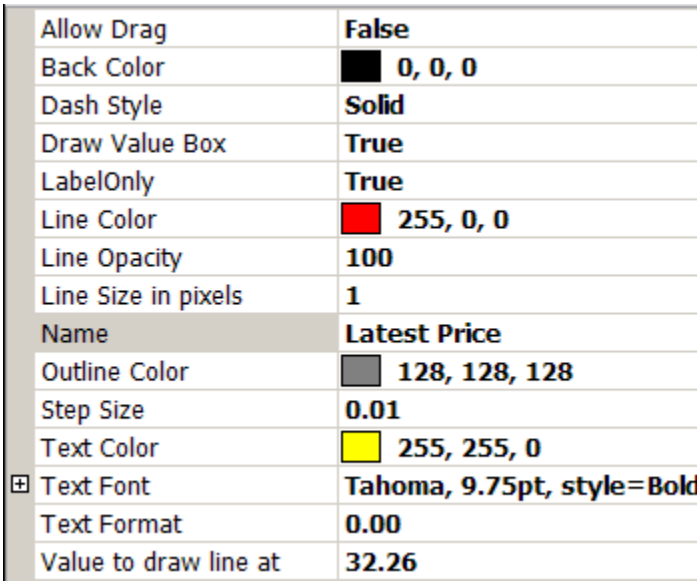


Figure 37 – Value Pointer Properties

Date Scale & Vertical Date Grid

The date scale displays the dates for all the plots on all the panes. Unlike a scale display of a pane that only displays the values of a specific scale on a specific pane (see page 18) the date scale displays all the dates that are plotted on the chart across all panes. A chart by default has a date scale display.

To customize the date scale, right click on the date display area and choose properties from the menu.

Table 5 describes the properties of the Date Scale.

Back Color	Background color of the date scale
Background highlight color	Background color to use when alternating background color
Border Color	The border color to use for the date scale
Font	The font to use to display the dates
Grid Color	The grid color to use when drawing the vertical grid for dates
Grid Line Style	The line style to use when drawing the vertical grid lines
Grid Mode	The frequency to draw the dates on the grid Auto will try to set the most granular as possible that can fit given the set of dates to render.
Grid Opacity Percent	The opacity of the vertical grid lines from 0 to 100. 0 = transparent, 100 = opaque.
Line Count	The number of dates to display when grid mode is set to Line Count
Name	The name of the date display unique to the workspace

Orientation	Bottom/Top. The location to display the dates on the chart. Bottom is default
Paint Background	True/False. Draw the background color
Paint Border	True/False Draw the border
Text Color	The foreground color of the text

Table 5 – Date Scale Properties

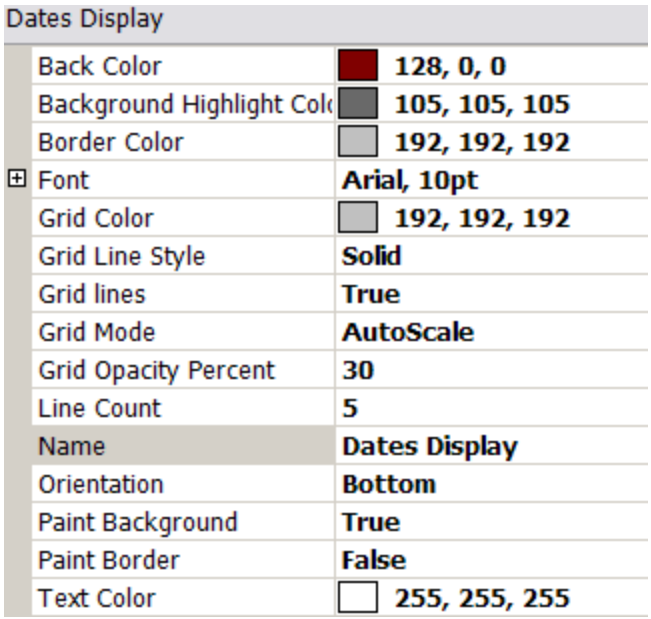


Figure 38 – Date Scale Properties

Setting the Display Mode

By default, the date scale attempts to display as much date information as possible that can fit legibly in the display area. You can also choose to set the date scale to a fixed frequency. Right click on the Date scale and choose *Display Mode*. From the sub menu you can choose: Autoscale, daily, weekly, monthly, quarterly, yearly, 5 year and Line count.

Date Pointer

The date pointer tool is akin to the value pointer tool (see page 32). The date pointer tool draws a vertical line on the chart at the specified date on the date scale. You can have 0 to many date pointers on a chart. The most common use of the date pointer is to highlight the value at a specific date, or to determine the date of a particular point on a chart.

Unlike the value pointer, the date pointer cannot be set by a block diagram. If you wish to programmatically highlight dates on a chart, use the True Marker Plot (see page 24)

When used in combination with a value pointer, you can achieve a crosshair to pinpoint a date and a value on a chart.

To add additional date pointers to a chart, right click on the Date Scale and select *Add Tool – Date Pointer*.

To customize the appearance of a date pointer, right Click on the date pointer and choose properties. Figure 39 shows all the properties you can edit for the date pointer. If a date pointer is off the scale, a small box with an arrow will indicate that a pointer is off the scale. Simply click and drag the box to bring the pointer back into view. Figure 39 shows a date pointer is off the scale to the right.

Figure 39 - Date Pointer off the scale

Date Pointer	
Allow Drag	**True**
Background Color	**178, 180, 191**
Date to draw line at	**2/16/2007**
Draw Value Box	**True**
Line Color	**255, 255, 0**
Line Opacity	**100**
Line Size in pixels	**Solid**
Line Size in pixels	**1**
Name	**Date Pointer**
Outline Color	**0, 0, 0**
Text Color	**0, 0, 0**
⊞ Text Font	**Arial, 9pt**
Text Format	**M/d/yy**

Figure 40 – Date Pointer Properties

Allow Drag	True lets you move the date pointer, False leaves it in a fixed position
Background Color	The Back color of the date pointer
Date to draw line at	The date on the scale to draw the pointer at
Draw Value box	If true, draws the date value on the scale using the Text format property
Line Color	The color of the vertical line
Line Opacity	The opacity of the line from 100 to 0
Line Style	The Line style to use
Line Size in pixels	The Width of the vertical line
Name	The unique name for this date pointer
Outline Color	The color to use for the box outline
Text Color	The color to draw the text
Text Font	The font to use when drawing the text
Text Format	The date format code to use when drawing the date value. Format code examples are: M/d/yy 5/6/07 MM/dd/yy 05/06/07 M/d/yy hh:mm:ss 5/6/07 03:05:56

Chapter 3: WatchList Tool

The WatchList Tool is very robust and fills many unique roles. Its main purpose is to display the symbols in a WatchList and select an active symbol. It also serves as an area to add column data about symbols and display them in a table format. It allows you to sort the symbols based on any calculation. It also includes a Strategy tool that you can use to scan and back test a list of symbols and display a performance report based on the back test (Scanning and Back testing will be covered in detail in chapter 4). Needless to say this is a powerful tool with many features.

Figure 41 – The WatchList tool

The main parts of the WatchList tool are:

WatchList Picker

The WatchList picker allows you to choose a list of symbols. Different data packages come with different lists of symbols. You can also create your own personal WatchLists that will be displayed in the WatchList picker

Symbol list

The symbol list is the list of symbols in a WatchList, as well as any columns that can display data about the symbol of the same row

Strategy Button, Run Strategy and Performance Tab

The Strategy button opens up the condition editor and the other sub items that make up the scanning and back test tool. See chapter 4 for more information on Scanning and Testing

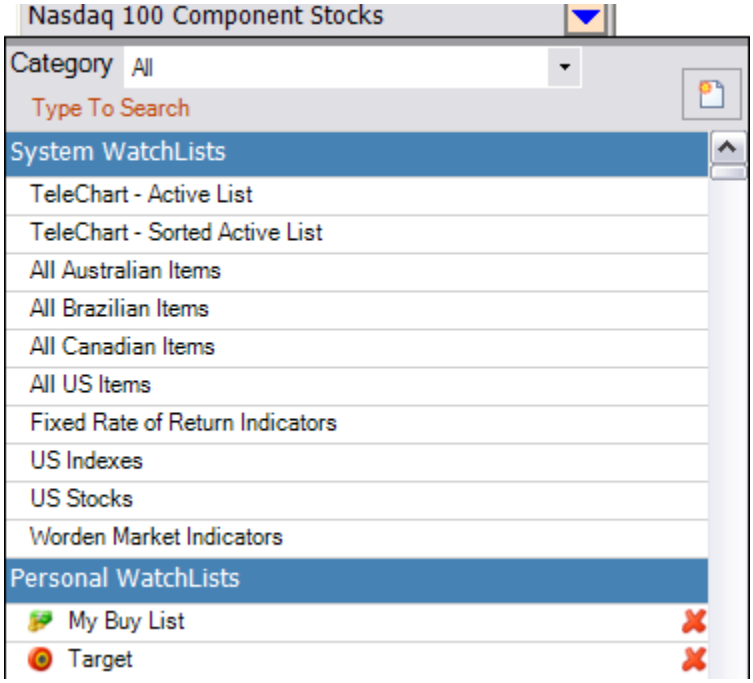
WatchList sub tabs

The WatchList sub tabs allow you to display symbols from different WatchLists other than those selected in the WatchList picker. For example: it can show the stocks in

the selected symbols industry or sub industry, it can show the stocks in your TD AMERITRADE account, your favorites WatchList, or any other WatchList in the system.

WatchList Picker

The WatchList picker is at the top of the WatchList tool, it displays the selected list of symbols. You can click on the name of the WatchList and a drop-down will display with all the available WatchLists in the system.
To filter the list, simply type the name of the list you wish to load.
Example: To change to the Standards and Poor's 500, click the WatchList dropdown and start typing “Standards”. Alternatively, you can click the category filter at the top of the drop down list to filter by category
Click on the name of the list to set it as the active list of symbols.



Symbol List

The symbol list displays the symbols in the currently selected WatchList. It can have 1 or more columns to display additional information about each symbol. You can select a symbol by clicking on the symbol in the list. To select multiple symbols, hold down the CTL button while clicking.

To customize the appearance of the WatchList, right click on the main list area and select properties.

Columns

Sometimes you need to see tabular data about a symbol. While a chart is good at showing you data for an individual stock over a period of time, the WatchList can show you data for multiple stocks for a specific period in time.

Like everything in blocks, there are multiple ways to load columns into your WatchList. The first option is dragging and dropping from an existing indicator on your chart. The second option allows you to load a pre-saved column and the third option allows you to create a column and it's calculation from scratch.

Types

There are 8 types of columns you can load into your WatchList.

Text
A Text column displays Textual data. Examples would be: Symbol, Company Name, Exchange

Numeric
A numeric column displays a floating point (decimal) number. Examples: Price, Entry Price, Net Change.

Whole Number
A whole number displays an Integer value (no decimal places). Examples: Shares owned, Days in Position

Date
A date column displays date data

T/F
A T/F column displays a Boolean value (True or False)

Flag
A Flag column displays an image for a symbol that belongs to a WatchList. For more information on the Flag column see page 44

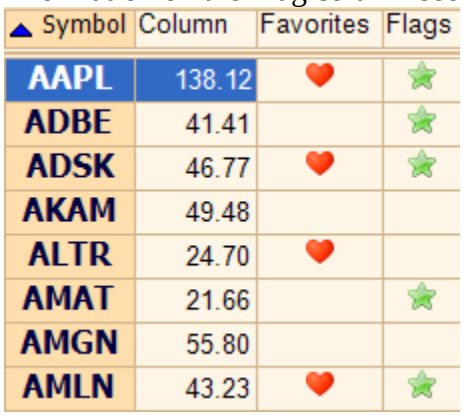


Figure 42 – Flag Columns

Mini-Chart

A Mini chart column displays a plot in a miniature chart in the WatchList. This is a really cool feature.

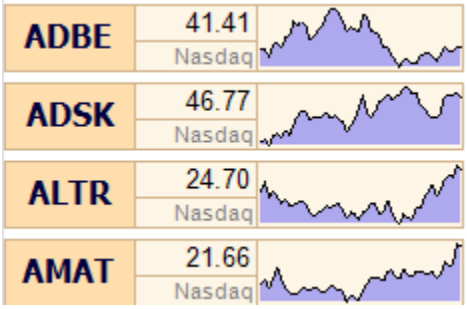


Figure 43 – WatchList with a Mini-Chart

Column Properties

To customize the appearance of a column, right click on the column header and choose properties.

Calculate in Background	Yes/No performs the calculation on a background thread. Default: Yes
Cell Background	The background color of the cell
Cell Border	The border color of the cell
Cell Font	The cell font for this column
Cell Text	The cell text color for this column
Column Span	The number of horizontal columns to span
Header Alignment	Left/Right Horizontal Text alignment for the header
Header Background	Background color for the header
Header border	Border color for the header
Header Font	Font for the header
Header Text	The color to use on the header text
Highlight When Selected	True/False. Show the highlight color on this cell when the row is selected
Minimum update interval	The amount of time (hours:minutes:seconds) to wait before updating the column value
Name	The unique name of the column
Number Format	The format string to apply to a numeric column
Recent Change Color	The color to use when flashing on a recent change
Recent Change highlight mode	Cell/Dot/Backcolor/Forecolor/None. Set this value to highlight the cell when the value changes.
Row Span	The number of vertical columns to span

Table 6

Calculate in Backgound	**Yes**
Cell Alignment	**Right**
Cell BackGround	**OldLace**
Cell Border	**Tan**
⊞ Cell Font	**Arial, 10pt**
Cell Text	**Black**
Column Span	**1**
Header Alignment	**Left**
Header BackGround	**OldLace**
Header Border	**Tan**
⊞ Header Font	**Arial, 10pt**
Header Text	**Black**
Highlight When Selected	**True**
Minimum update interval	**00:05:00**
Name	**Column.2**
Number Format	**0.00**
Recent Change Color	**Orange**
Recent Change Highlight M	**None**
Row Span	**1**

Figure 44 – Column Properties

Creating a Numeric or Mini-Chart column by Dragging and Dropping from a Chart Study

By far the easiest way to make a column is to drag and drop from an existing study on your chart. Position your mouse over the indicator you wish to view as a column, hold down the left mouse button (but do not release the button), while keeping the left mouse button down, move your mouse to the column header section. Your mouse pointer will turn to a + icon to show you the correct drop zone. Once your mouse changes, release the mouse button. You will be presented with the Create Column dialog (Figure 45).

Choose the column type you wish to create. The options are Last value or Mini-Chart. The last value will show the latest value for the plot. The Mini-Chart will show a small version of the plot (the last 50 bars) for every symbol in the WatchList. Choose the column name in the Name field. The *Limit Data used for Calculations* checkbox is used to increase the column performance.

TIP: A column can only display one data point. Most indicators need a minimum amount of data to perform the calculation (A 50 bar moving average requires 51 bars of data). To get the best performance and use the least amount of resources as possible, choose the lowest data limit needed for the column. For instance, if you're showing the last price on the chart, you can limit the data to 1 bar. If you're showing a 50 bar moving average, you need 51 bars of data.

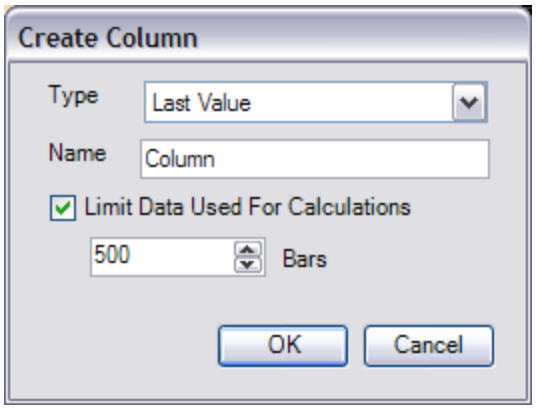


Figure 45 – Drag and Drop, Create Column

Loading a Saved Column (from the Library or My Computer)

There are a number of pre-configured columns available for you to load into your WatchList. To add a new column, click in the empty section of the column header and choose Load Column. You will be presented with the familiar selection dialog (Figure 46). The tabs along the left from top to bottom are: Library, My Computer and Recent. Type the name of the column you wish to load to filter the list. Left Click on the column name to load it

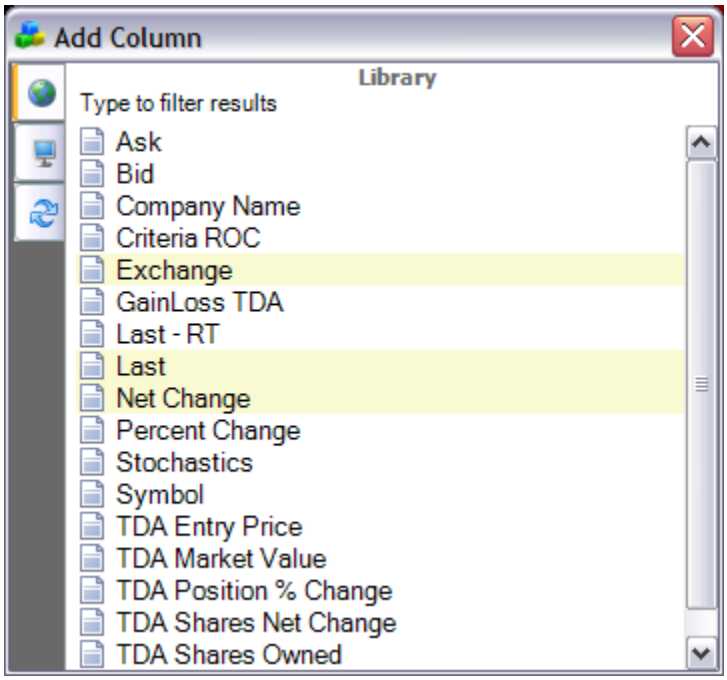


Figure 46 - Add Column

Column Quick Edit

If a column has any quick edit fields (to adjust moving average periods, time frames or any other parameters) right click on the column header and select Quick Edit.

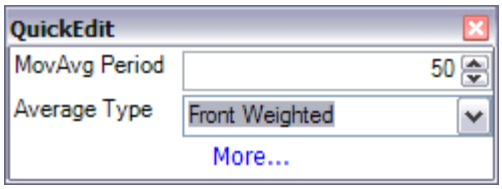


Figure 47 – Column Quick Edit for a Moving Average Column

Column Sets

By default, there is one "row" of columns. If you want to use more vertical space for more column data you can add additional column sets. Figure 48 has two column sets (The bottom column set has the exchange under the latest price column)

To add an additional column set, click in the empty space in the column header (to the right of all your existing columns) and choose *Add Column Set*.

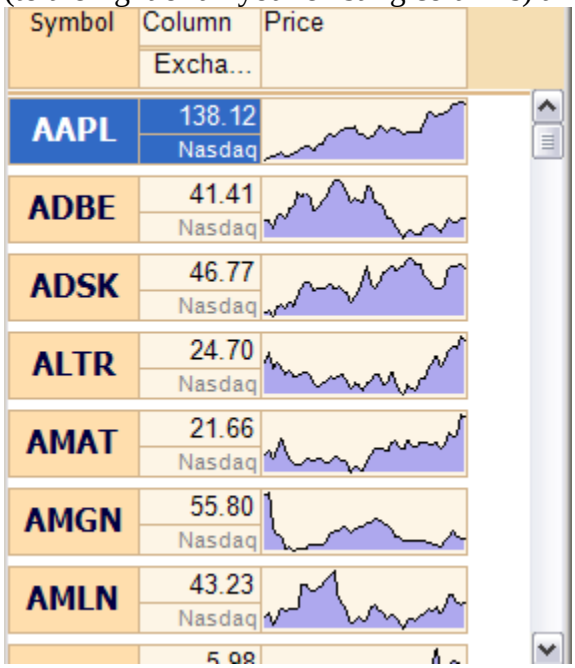


Figure 48 - Column Sets

Column Positioning

If you have more than one column set (stacked columns) you can control how a column is positioned in the column sets. Figure 48 above has two column sets. Columns by default have a column span of 1 and a row span of 1, meaning they take up one column in the grid. In Figure 48, the symbol column has been set to span 2 rows, and therefore you can see that the exchange and latest price columns are both to the right of the symbol column. The Mini-chart is also set to span 2 rows.

If you wanted the exchange column to be displayed under the mini-chart column, you would set the column span on the exchange column to 2.

To choose row and column span, as well as to move a column left or right, right click on the column header for the item you wish to adjust. Choose the *Column Position* sub menu.

Flag Columns

Flag columns are used to display membership of a symbol in a WatchList. They can also be used to add/remove a symbol from a WatchList. For instance if you add a flag column of the DJ-30, any symbol that is in the DJ-30 will have a flag icon in the column, all other symbols will not have any icon in the flag column.

If a flag column is set for a personal WatchList (one which you maintain the symbols in) you can add/remove symbols from the WatchList by double clicking on the flag column. For more information on Personal WatchLists, see page 45.

Scanning and Sorting

You can scan a WatchList and filter the results using the Strategy tool. The Strategy tool is detailed in chapter 4.

WatchList Sorting

Sorting a WatchList is as easy as left clicking on the column header. Depending on the complexity of the column calculation, it may take up to a few seconds for the WatchList to re-sort. To reverse the sort direction, simply click on the column header again. If your column data is updating frequently (on the tick) you may wish to adjust the Minimum update interval on the column properties (See Column properties for more details).

Visual Sorting

A Mini-Chart column is sorted visually instead of numerically. Figure 49 shows a Mini-Chart column of stock price visually sorted. The symbols on the top of the list their price is at the top of the value scale, while the items at the bottom of the list their price is near the bottom of the scale. Mini-Charts by default are displayed with the last fifty bars, so using the example in Figure 49, the stocks at the top are near the top of their 50 bar high, while the ones at the bottom are closer to a 50 bar low. Since Mini-Charts can be constructed out of any indicator, you can use this to sort on all sorts of ideas.

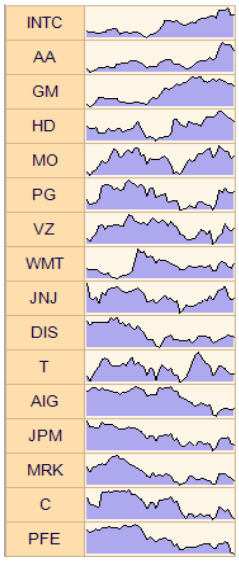


Figure 49
Visual Sorted Column

Example: Add Stochastics of Price to your chart (see page 22 for adding studies to your chart) Drag and Drop Stochastics onto your WatchList Column Header (page 41). Click on the Column header for the new Mini-Chart Column. Stocks at the top of the list should be near the top of the range (overbought). Tops at the bottom of the list should be near the bottom of the range (oversold)

WARNING: There is one caveat to the example above. By default the chart only displays the last 50 bars. If the last 50 bars of the chart were all oversold (below 20) then it could end up at the top of the list because the chart scale is relative to those last 50 bars. It's always a good idea to verify the Mini-Chart by viewing the main chart with the full indicator values.

WatchList Tabs

WatchList tabs give you a way to quickly change your list of symbols. Figure 50 shows the 4 WatchList Tabs that come pre-configured in the Personal Chartist workspace. The *Main List* tab displays the symbols selected from the WatchList drop down. To the right of main list is the Industry (*Ind*) tab that shows the list of symbols in the Industry of the active symbol. The *Sub* tab displays the list of symbols in the sub industry for the active symbol.

The last tab is the TD AMERITRADE positions list. If you have a TD AMERITRADE account, and you have logged in to TD AMERITRADE, your positions will be listed in the WatchList by clicking the TD tab button. (For more information on TD AMERITRADE in Blocks, see chapter 11)

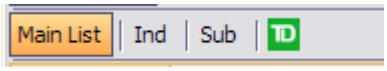


Figure 50 - WatchList Tabs

Creating a new tab

To create a new WatchList tab, click the *Add WatchList Tab* button. Select *Add WatchList* tab. You will be presented with all the WatchLists in the system, type the name or select the WatchList from the list.

Creating a new tab for new WatchList

If you want to create a new personal WatchList and a tab for the new WatchList you can do it in one step by clicking the *Add WatchList Tab* button and selecting *Add Tab for New WatchList.* See the next section for instructions on creating a new Personal WatchList

Personal WatchLists

Personal WatchLists are lists of symbols that you maintain. You can keep different sets of symbols for any reason. There are a few different methods for starting the process creating a new WatchList.

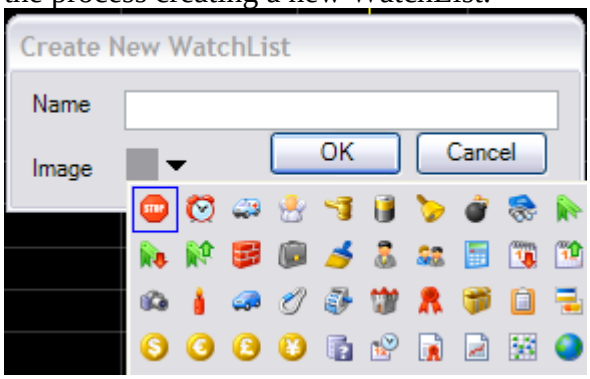


Figure 51 - Create New WatchList

Creating with Flag Columns

By far, one of the easiest methods to create a new WatchList and populate it with symbols is to create a new Flag Column for new WatchList. Click in the empty space in the column header and choose *Add Column – Flag for new WatchList*. You will be presented with the dialog in Figure 51. Type the name for your new WatchList. You can optionally select an image to represent your new WatchList by clicking the down arrow and choosing from the list of images. Click ok and a new column will be added to your WatchList display. To add a symbol to your new personal list, simply double click in the flag column. To remove a symbol, simply double click in the flag column on the icon.

Creating From WatchList picker

You can also create a new WatchList from the WatchList picker drop down. Click the WatchList drop down and in the upper right corner there is the *Create New WatchList* button. You will see the dialog in Figure 51 where you can choose the name and an image to represent your WatchList.

Creating from new WatchList tab

You can also create a new Personal WatchList (and a WatchList tab at the same time) by clicking on the *Add WatchList Tab* button. Select *Add Tab for New WatchList* and type the name and image for your new WatchList

Adding stocks to a list

You can manually add stocks to a Personal WatchList by right clicking in the WatchList display area and choosing *Add Symbols Manually* from the menu. From here you can type the symbol name or the company name into the search field to add it to your list (See figure 52).

You can also manually add symbols to a WatchList by highlighting one or more symbols in the list (hold down the CTL button and click on the symbols to multi-select). Right click on the WatchList display and choose *Add to Another List*. From the sub menu choose *Current Symbol, Selected Symbols or All Symbols*.
All Symbols will make an exact copy of the current WatchList you are viewing.
You can also add symbols to a WatchList by using a Flag Column. See page 45 for more details.

Figure 52 - Add Symbols To WatchList

Removing stocks from a list

To manually remove symbols from a Personal WatchList, right click on the WatchList display area and select *Remove From List.* On the sub menu you can choose to remove the current symbol or all the selected symbols (hold down CTL while clicking on the symbols to select more than one)

Adding symbols from an Excel spreadsheet (or other external document)

To add one or more symbols to your WatchList from Excel (or any other text document) simply highlight the symbol names and select copy (CTL-C or right click, copy). To paste them into your WatchList, right click in the WatchList display area and choose Paste.

Exporting to Excel

To export your list of symbols simply right click in the display area and select *Copy – Current Symbol*, *Selected Symbol* or *All Symbols.* This will copy the symbol(s) to your clipboard where you can paste them in any document with the paste function.

Editing and Deleting Personal WatchLists

You can rename or change the icon for your Personal WatchList by opening the WatchList Picker drop down and right clicking on your WatchList name. Select *Change Name*

To delete your WatchList, hit the red X to the right of the WatchList name (Figure 53)

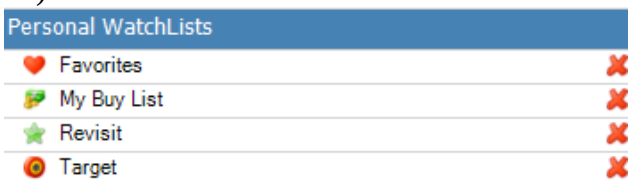


Figure 53 - Personal WatchLists

Chapter 4:

Strategy Scanning & Testing

Scanning and Testing Overview

The Blocks software has powerful and unique scanning and back testing tools (collectively called Strategies). One unique feature is the ability to back test on an entire WatchList and give the performance summary for each symbol to find stocks that work best with a given strategy. Another unique feature is the ability to shape conditions based on a sequence of events. These sequences are AND THEN conditions that allow one event to follow another event over a given period of time.
The major elements of the Strategy Tool are:

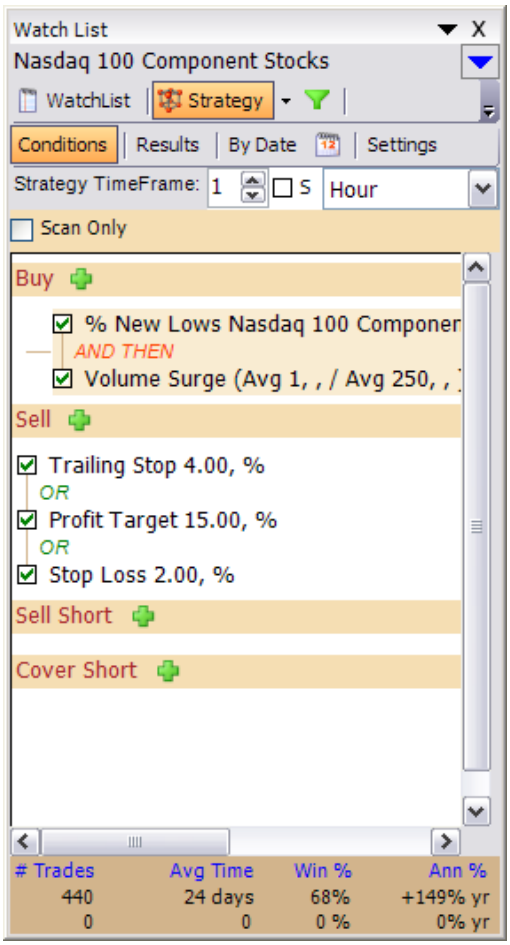


Figure 54 – Strategy Tool with Buy / Sell conditions

Condition Editor

The Condition Editor lets you set your scan conditions or your buy, sell, sell short and cover conditions as well as the strategy timeframe. These conditions can be concatenated into Boolean expressions using AND logic, OR logic or AND THEN logic. They can be nested into a series of AND, OR and AND THEN to create much more complex logic trees.

Strategy Results

Once you have run a scan or back test, you can view the results of your strategy. The results tab lists the symbols that are currently passing your scan or entry conditions.

Results By date

This displays the symbols that are passing your scan or entry conditions on a specific date. By default the date is controlled by the date pointer on the Bar Chart. Move the date pointer to any date to see the list of stocks that pass your conditions on that date.

Settings

The settings tab lets you change the scan and test settings

Performance

The performance tab lets you view a series of reports about your Strategy. Included reports are: Summary, By Symbol, All Trades, and Trades for the selected symbol.

Performance Snapshot
The performance snapshot is available at the bottom of the WatchList tool to display a quick snapshot of the following fields: #Trades, Average trade time, Win %, Annualized Win %.

Loading a Pre-Configured Strategy

The Blocks software comes pre-configured with a few strategies in the Library. You can open one of the pre-configured strategies as an example to build your own custom strategies. You can also open your own saved strategies from the same menu. Click the down arrow button next to the *Strategy* tab on the WatchList tool. You will be presented with the menu in Figure 55. Select Open Strategy. The Load Strategy dialog (Figure 56) will display. Click on the *System Library* button to view the pre-configured strategies, or to load your own saved strategies, click on the *My Computer* button. Type the name of the strategy to filter the list, or browse the available strategies. Click on the name of the item to load it.

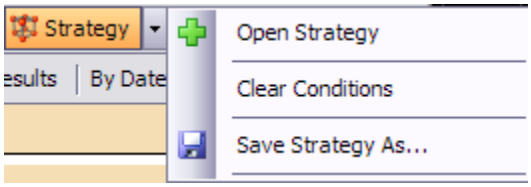


Figure 55 - Strategy Menu

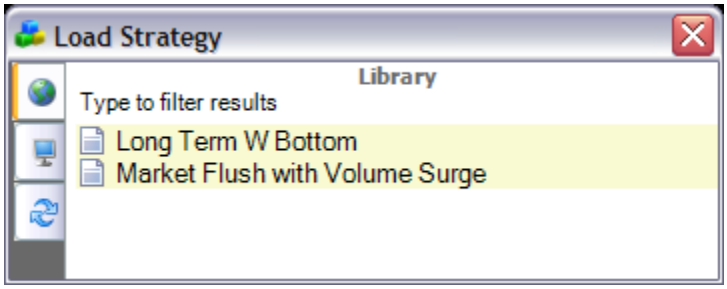


Figure 56 – Load Strategy Dialog

Condition Editor Overview

The Condition Editor is where you configure your entry and exit conditions (or your scan conditions in scan only mode).

Scan Only

Scan only is used when you simply want to filter the list of stocks based on a set of Boolean conditions. To set scan only, click the Scan Only checkbox at the top of the condition editor. This will remove the buy, sell, sell short, cover conditions and give you one set of Strategy conditions.

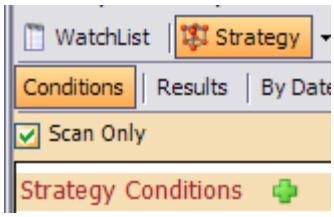


Figure 57 - Scan only

Conditions

A Condition is a set of calculations and logic that produce a True or False value for a given stock for a given timeframe (daily, minute). When a strategy is running for a symbol, the symbol is fed into the condition and the logic produces a True or False for that given symbol on every bar of the scan.
Conditions can be grouped together to form additional Boolean logic. When all conditions are met (TRUE) then the symbol passes the scan on that given bar.

Loading Pre-Saved Conditions

The Blocks library has many conditions to get you started building your own custom strategy. To add a condition, click the *Add Condition* button. In Scan Only mode (see above) the *Add Condition* button is next to the *Strategy Conditions* label. If you are not in scan only mode, click the *Add Condition* button next to the action you want the condition to trigger (Buy, Sell, Sell Short, Buy to Cover). Figure 58 shows *Add Condition* next to the buy action of a strategy.

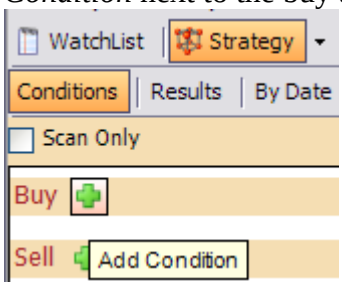


Figure 58 - Add Condition to Buy Action

And / Or / And Then

If you wish to have multiple conditions trigger an action, you can use the logical operators AND, OR, AND THEN.

AND requires that both conditions are met at the same point in time for the action to occur (Figure 59). To AND two conditions together, add the first condition then right click on the condition or click the *Add Condition* button to the right of the first condition and select *AND Condition* .

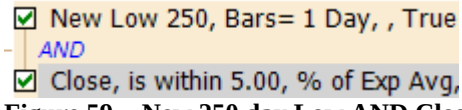


Figure 59 - New 250 day Low AND Close within 5% of its Exponential moving average

OR requires that either action be true at the same point in time for the action to occur

To OR two conditions, add the first condition and the right click on the condition or click the *Add Condition* button to the right of the first condition and select *OR Condition*

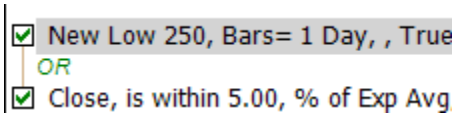


Figure 60 – New 250 day low OR Close is within 5% of it's Exponential Moving Average

AND THEN requires that one condition occur AFTER a previous condition has occurred. The length between the first and second condition is variable and is defaulted to 100 bars.

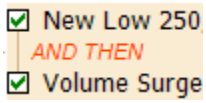


Figure 61 – New 250 day low AND THEN a volume surge within 100 bars of the low.

To create one or more AND THEN conditions, add the first condition and the right click on the condition or click the *Add Condition* button to the right of the first condition and select AND THEN. To configure the length of the AND THEN sequence (the number of bars that the sequence must occur) right click on entire AND THEN sequence (the box around the conditions will highlight when you move your mouse over it) and select properties. Edit the span in bars property to the number of bars you want the AND THEN sequence to occur.

AND THEN sequences also have the properties in Table 7.

Last condition must be true for test date	True/False – For the sequence to be true, the last condition in the branch must be true on the day that it is being tested.
Minimum # of bars between conditions	The minimum number of bars that must occur between conditions for the sequence to pass
Span in Bars	The number of bars in which the entire sequence of AND THEN must pass otherwise the sequence will return false.

Table 7 – AND THEN Sequence properties.

Nesting Conditions

Conditions can be nested to form more complex Boolean logic. You can have many AND, OR, AND THEN sequences in combination. Each branch in the condition tree will be surrounded by a box that is highlighted when your mouse passes over the branch.

Figure 62 shows two sets of conditions that must be true in order for a symbol to pass the scan. The top set says that the PE ratio must be greater than 20 OR that the

symbol's capitalization is greater than 10 million. If neither of these is true, then the symbol will not be included, no matter what the rest of the conditions evaluate to. Because they have an OR relationship, if either one is true, then the scan will continue.

The next part of the scan in figure 62 shows another set of conditions that have an AND THEN relationship. First the symbol must reach a 250 day high, AND THEN it must have a volume surge within the next 100 bars. If this condition never happens, then the scan will be false for the symbol that is being tested. Because the bottom AND THEN sequence also has an AND relationship with the top branch (PE ratio or Capitalization) both the top and the bottom half must be true for a symbol to pass the scan.

Figure 62 - Nested Conditions

Creating New Conditions

The Blocks condition library comes pre-loaded with many conditions you can use for creating your Strategy. The conditions are highly configurable and many of the parameters and calculations can be modified from the existing conditions.
If you want to use your own custom logic, you can create a new condition following these steps:
Click on the *Add Condition* button and select *Add Condition – New Condition* (Figure 63)

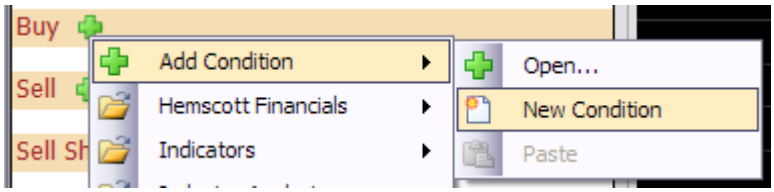


Figure 63 – Creating a New Condition

Your new condition (surprisingly labeled *Condition*) will now be added to the condition editor. Double click on your new condition to open the block diagram. Your new condition diagram will look like figure 64. A condition is provided the symbol to test, and consumes a series of dates and Booleans (true/false). For the symbol provided your condition should set a true on every day that your condition passes, and false on every other day.

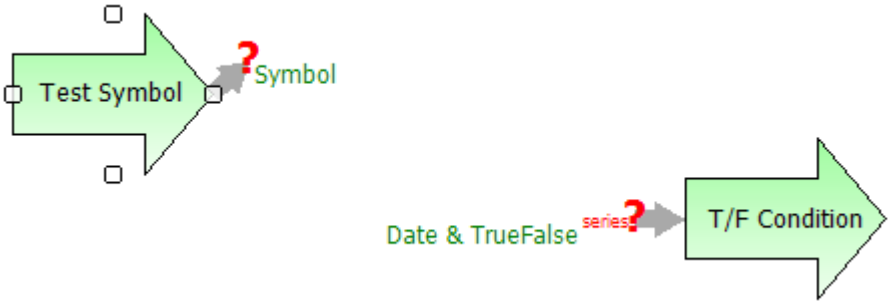


Figure 64 – New Condition empty diagram

EXAMPLE:
To create a condition that tests for the a symbols daily net change is greater than 0, inline the Prices block, net change block, and greater than value block between the test symbol and the T/F condition. (Figure 65)

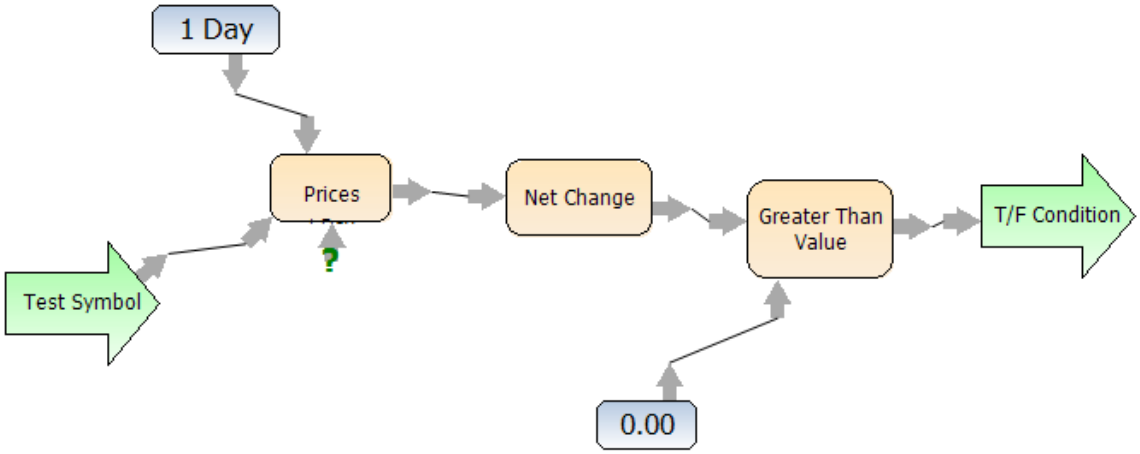


Figure 65 – Condition Price Net Change is greater than 0

Saving Conditions

Once you have created a new, or customized an existing condition you can save your condition to use in a different strategy at a later date. Simply right click on your condition and select *Save As…*

Configuring a Strategy

A Strategy has many settings you can configure to customize the scan or test to your liking. To view or change the strategy settings, click on the Strategy Button on the WatchList and then choose the Settings tab on the far right. You will be presented with the list in Figure 66.

Auto Run

Auto Run WatchList when Conditions Change will automatically rescan the WatchList when any parameter or condition is changed/added/removed. (Not recommended)

Auto Run Symbol when Conditions change will update your charts buy/sell or true/false markers for the active symbol while changing your conditions. This allows you to tweak your settings on the current symbol to dial in a pattern you are looking for.

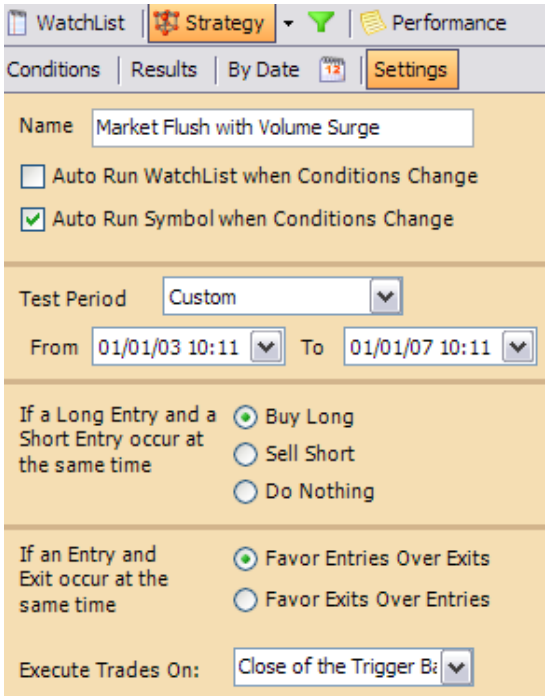


Figure 66 – Strategy Settings

Test Period

The test period allows you to choose how much history is tested. Using the *Custom* option you can set the *From* and *To* dates. Setting this value to a shorter timeframe will speed up the time it takes to run a test. This can be helpful on large WatchLists with complex conditions. The default value is *All History*.

Long and Short Signals

If a Long entry and a Short entry occur at the same time, you can customize what you want the Strategy to do. The options are: Buy Long, Sell Short, Do Nothing.

Favor Entries

If an Entry and Exit occur at the same time you can customize what you want the Strategy to do. Strategies default to favor entries over exits because most likely if you exit the symbol, your buy condition will still be valid and you'll enter again.

NOTE: This setting is important because it may affect your exit signals. For example, if you use a trade length sell condition (say 25 days) your trade may not exit in the timeframe you expected if the buy signal is still true on the 25th day. If you are

getting exit signals that look like they are much too late, change this setting to Favor exits over entries and verify your logic.

Execute Trades On

This setting lets you choose where the trade will open or close. Setting this to *Close of the Trigger Bar* will execute on the close of the same date that the conditions pass. Setting this to *Open of Next Bar* is a more realistic setting as the trigger may have been the actual close value and you could not possibly execute the trade on the same day.

Running a Strategy

By default, a Strategy is always running on the active symbol while you are configuring the conditions (see Auto Run in previous section). This lets you shape your conditions and get immediate results for the current symbol. Once you are happy with your conditions, you can run the strategy on the entire WatchList by clicking the Run Strategy button to the right of the strategy tab. While your strategy is running, a progress % will be displayed next to the run button. Once your strategy is complete it will update all the reports and results.

Strategy Results

There are many reports and results that are available once you have run your strategy. In scan only mode you are most likely looking for a list of symbols that pass the scan on a particular date, while in test mode you will have a large set of reports and data on each individual trade as well as the strategy as a whole.

Results

The results tab (to the right of the Conditions tab) will show your list of symbols that are currently passing the scan right now (scan only) or are currently in an open trade. Click this tab to get the list of symbols. Figure 67 shows a list of symbols in the results tab that are currently passing a strategy. A chart for the selected symbol (SBUX) is shown on the right and the entry date is highlighted by the buy marker. Notice the shaded background that is filled to the far right of the chart.

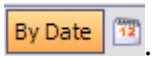
Figure 67 - Strategy Results

By Date

The *By Date* tab shows your Strategy results by a specific date. By default, it uses the date from the date pointer on the chart. Sliding the date pointer backwards and

forwards in time will change the list of symbols that pass the Strategy on the date selected by the date pointer. (for more information on the date pointer, see page 35).

Figure 68 shows a list of symbols that pass a Strategy on 12/31/2006. Notice the yellow vertical date pointer set to that date on the date scale. The active symbol (ESRX) is displayed on the chart and notice how the buy and sell markers enclose the date pointer.

You can change the date that is used in the By Date list by clicking the calendar icon to the right of the *By Date* tab 

. This will bring up a diagram and allow you to choose the tool, or block to provide the date for the *By Date* results. For more information on block diagrams see chapter 6.

Figure 68 – Results *By Date*

Performance

Once you have run your Strategy for the current WatchList (see page 56) you will have many reports available to you. There is a quick performance summary at the bottom of the WatchList that displays the number of trades, average time in all the trades, the win% and annualized % for both the current symbol, and the WatchList (See bottom of figures 67,68 and 69).

The *Performance* tab has all the performance results for your current Strategy. There are four sub tabs under *Performance* that you can use to access each of the different reports.

The *Summary* tab summarizes the results of your Strategy to give you and overall picture of your performance results. The *By Symbol* tab gives you a summary of the Strategy for each symbol in the list. The All Trades tab gives you a break down by every trade and the last tab (Active Symbol) gives you a break down of all the trades for the currently selected symbol

Summary

The Summary report shows you the following fields for Winners, Losers and the Total (see figure 69).

 # Trades – The number of trades executed by this Strategy

Return/ Trade – the average return%

Trade/Time – the average number of days you were in a trade

Best Exit – The best gain/loss% you could have hoped to achieve with this strategy had you exited at the highest point in each trade

Draw Down – The amount you gave back from the best exit in the trade

Worst Exit – The worst exit you could have made had you exited at the lowest point in each trade

Winning % - The percentage of winning trades you made with this strategy

Figure 69 – Performance Summary

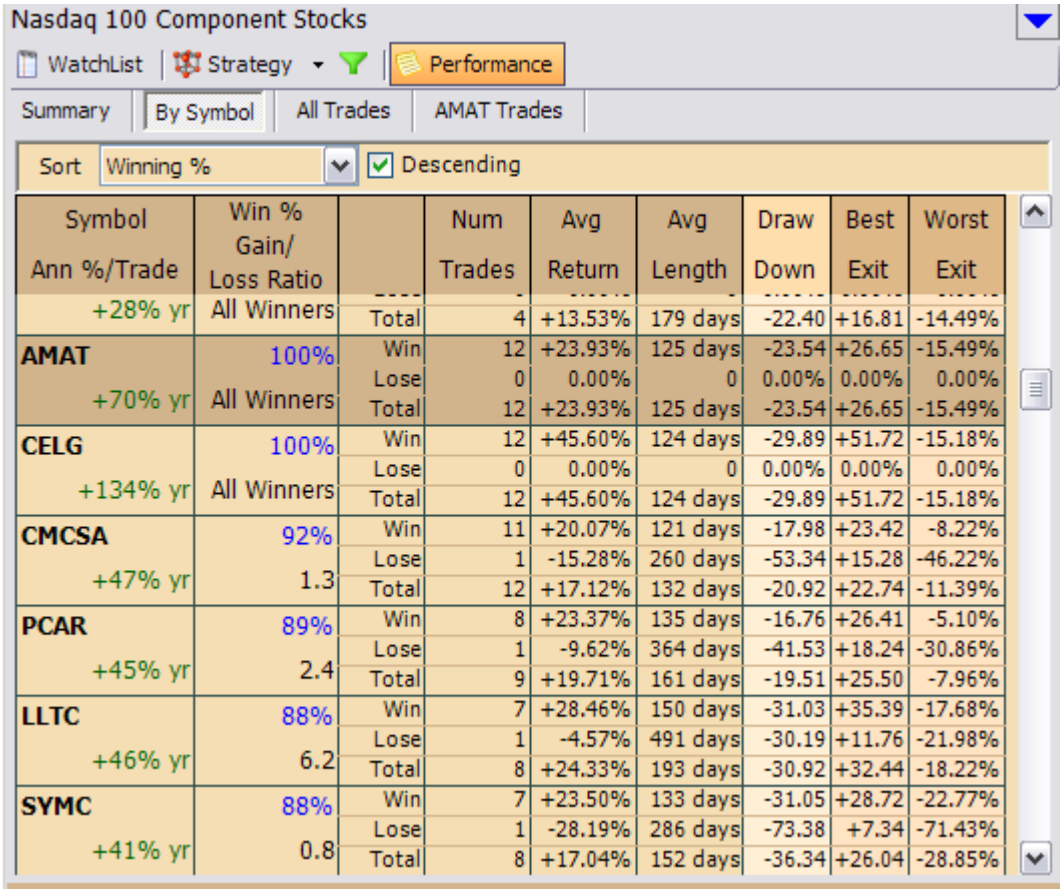
Gain/Loss Ratio – The ratio of Return/Trade for winners vs. losers A ratio of 2:1 would mean your winning trades gained twice as much as you lost on your losing trades (Example:. If your return/trade for winners was 50% and your return/trade for losers was 25% then you would have a 2:1 Gain/Loss ratio)

Ann Return/Trade the averaged Annualized return per trade. Each trade is annualized using the following formula: ((((100 + % change) / 100) ^ (1 year / time held)) * 100) - 100

Scan performance – The bottom of the summary shows the number of conditions tested and the amount of time it took to run the strategy for the WatchList along with the number of conditions/second

By Symbol

The Performance results By Symbol displays a summary of all trades on a symbol by symbol basis. You can sort the results using the sort dropdown at the top of the list. Figure 70 shows the performance results by symbol



Symbol Ann %/Trade	Win % Gain/ Loss Ratio		Num Trades	Avg Return	Avg Length	Draw Down	Best Exit	Worst Exit
+28% yr	All Winners	Total	4	+13.53%	179 days	-22.40	+16.81	-14.49%
AMAT	100%	Win	12	+23.93%	125 days	-23.54	+26.65	-15.49%
		Lose	0	0.00%	0	0.00%	0.00%	0.00%
+70% yr	All Winners	Total	12	+23.93%	125 days	-23.54	+26.65	-15.49%
CELG	100%	Win	12	+45.60%	124 days	-29.89	+51.72	-15.18%
		Lose	0	0.00%	0	0.00%	0.00%	0.00%
+134% yr	All Winners	Total	12	+45.60%	124 days	-29.89	+51.72	-15.18%
CMCSA	92%	Win	11	+20.07%	121 days	-17.98	+23.42	-8.22%
		Lose	1	-15.28%	260 days	-53.34	+15.28	-46.22%
+47% yr	1.3	Total	12	+17.12%	132 days	-20.92	+22.74	-11.39%
PCAR	89%	Win	8	+23.37%	135 days	-16.76	+26.41	-5.10%
		Lose	1	-9.62%	364 days	-41.53	+18.24	-30.86%
+45% yr	2.4	Total	9	+19.71%	161 days	-19.51	+25.50	-7.96%
LLTC	88%	Win	7	+28.46%	150 days	-31.03	+35.39	-17.68%
		Lose	1	-4.57%	491 days	-30.19	+11.76	-21.98%
+46% yr	6.2	Total	8	+24.33%	193 days	-30.92	+32.44	-18.22%
SYMC	88%	Win	7	+23.50%	133 days	-31.05	+28.72	-22.77%
		Lose	1	-28.19%	286 days	-73.38	+7.34	-71.43%
+41% yr	0.8	Total	8	+17.04%	152 days	-36.34	+26.04	-28.85%

Figure 70 – Strategy Performance Results By Symbol

All Trades

The *All Trades* report will display every trade that passed the conditions in your strategy.

Current Symbol

The *Current Symbol* tab will display the trades for the active symbol

Trade Markers (Strategy Trades)

In the Personal Chartist workspace, your chart is pre-configured to display trade markers for entry and exit signals generated by a strategy. These buy and sell Trade markers will show up for the active symbol. See Figures 67 and 68 in the previous section for an example of Trade Markers. You can load the trade markers from the library on to your own custom charts if you create your own or accidentally remove them from the chart by selecting the Strategy Trades study from the *Add Study* button.

Equity Lines

In addition to the Trade Markers on the chart, you can plot additional equity information about your Strategy to compare how it performs over time. All of the Strategy studies are available in the library under the *Add Study* button. All Equity lines are calculated with a starting value of $100. If you are not in any trades for any bar, then the APR value on Strategy Settings tab is used to calculate the gain.

Strategy Equity Line

The *Strategy Equity Line* is based on being equally weighted in all symbols in your strategy. It produces a line of your equity over time as your Strategy simulates trades with the buy and sell conditions. When this line is moving up, you're gaining equity, and when it's down, you're losing. Needless to say, you want this moving in the positive direction

Strategy % in Line

The *Strategy % in Line* study shows you the percentage of stocks that are in a trade at that point in time.

Strategy Count Line

The Count Line shows you the count of stocks that are currently in a trade at that point in time

Strategy Buy & Hold Equity Line

The Buy and Hold equity line simulates buying and selling all the symbols in the list over the period of the entire strategy test. Each point on the plot would be your equity if you sold at that point in time. All items are equally weighted.

Strategy Equity Line vs. Buy & Hold Comparison

This study shows you the Equity line vs. the buy and hold equity line on the same pane. It allows you to see if your strategy equity underperforms or outperforms a buy and hold strategy.

Strategy Performance vs. Buy and Hold

Like the Strategy Line vs. Buy and Hold Comparison, this line compares your Strategy vs. a Buy and Hold strategy. Instead of comparing the equity, it compares how

much it is outperforming or underperforming. Values greater than 1 and it is outperforming a buy and hold strategy, while values less than 1 it is underperforming.

Symbol Performance Equity Lines

In addition to viewing the performance of the entire strategy, there are symbol equity lines to compare how an individual stock performs over the life of the strategy test. There are 3 symbol studies available: Strategy Symbol Equity line, Strategy Symbol Buy & Hold and Strategy Symbol vs. Buy and hold.

Exporting results to Excel

From the Strategy Settings tab, you can choose to export the results of your strategy to a CSV (Comma separated value) file that can be viewed in Excel or imported into any other program that can read CSV files.

To export the results of a Strategy, check the *Export Results after calculating Strategy* checkbox. You can choose from 3 reports to include in the CSV file: Summary, By Symbol Results, and All Trades. Change the export path by clicking on the "…" button to the right of the export path label. Run your scan with the *Run Scan* button and your results will export to the specified folder location.

Saving a strategy

Once you have configured a Strategy you can save it to run again in the future by using the Save Strategy option. Click the down arrow to the right of the Strategy button to open the Strategy Menu (see figure 71). Click the *Save Strategy As…* option. Provide a name for your strategy and click ok.

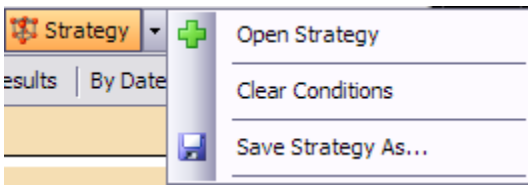


Figure 71 – Strategy Menu

Part 2 - Customizing Blocks

Chapter 5: Workspace Customization

A Workspace is simply a collection of Tools that have been organized into a layout. Each Tool saves its current configuration when you save a workspace. Tools "run" inside a Tool Window that can be positioned anywhere on your screen or relative to the other Tools in the workspace. Examples of Tools are: WatchList, Bar Chart, data labels, command buttons and Web Browsers.

Managing Tool Windows

Tool windows can be in one of 4 states: Docked, Floating, Maximized or Hidden. By default when you add a Tool window it is in the floating state. In this state you can drag it around your Windows Desktop and can resize it by dragging the edges. You can move it onto multiple monitors by dragging it across the screen to the appropriate place. You can dock the tool into a frame by using the drag and drop docking option (see Moving/Docking Tools). You can always maximize a tool by double clicking on the Tool Window header (where Tool name is listed). From the maximized state, you can return it to its previous position by double clicking on the header again, this will return it to docked or floating (whatever its was previously set to before maximizing). You can hide windows from view and return them to view by using the *Window* menu option. This will list all the Tool Windows in your workspace.

When you save a workspace, all the Tools will save their current location and return to those positions when the workspace is re-loaded.

Adding Tools via Start

The Personal Chartist Workspace comes pre-configured with 3 Tools (WatchList, Bar Chart and Google Finance Browser). You can learn more about the Chart and WatchList in Part 1. There are additional tools that come pre-configured to add into your workspace. Some of these tools include: Web browsers that load web pages for the active symbol (ala Google Finance Browser), Point and Figure charts, Million Dollar Bar Chart, Overbought/Oversold pie chart and more.

To Add a Tool to your workspace, click the 

button on your toolbar. You will be presented with the Tool library in figure 72. Type the name of the tool to filter the list or browse the sub categories to find a tool. Click on the tool name to add it to the workspace. You can load Tools you have saved by clicking on the My Computer icon from the drop down list and browsing to the file you wish to load.

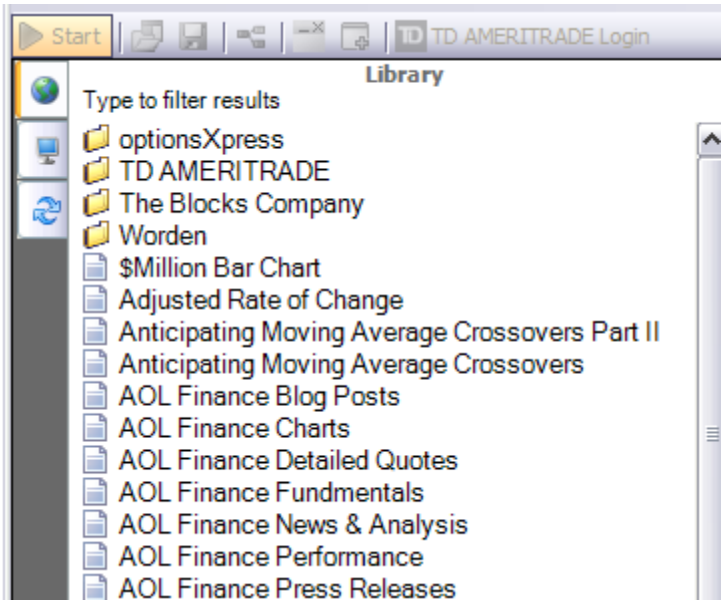


Figure 72 – Tool List

Moving / Docking Tools

The tools in your workspace can be rearranged to any layout you choose. Simply move your mouse to the top of the tool and your cursor will change to a hand on the window header. Hold down your left mouse button and "drag" your mouse to move the tool.

Tools can be "docked" into the workspace. To dock a tool, simply move it over an existing tool (or an empty workspace) and hold it over the tool for a second. The docking "hot spots" (see image right) will appear. To dock a tool above or below an existing tool, move the mouse over the top or bottom image in the "hot spot" cross. To dock it left or right, move the mouse over the left or right images. To tab your tool with an existing tool, move the mouse over the center image in the hot spot. Once you have chosen your new dock location, release the left mouse button to "drop" your tool in its new location.

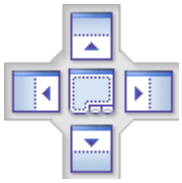


Figure 73 shows a chart tool with docking hot spots (notice the cross in the middle of the chart). To see a demonstration on how to move and dock tools, see the video at http://tinyurl.com/ypyoe5

Figure 73 – Chart tool with Docking "Hot Spots"

Hiding & Deleting Tools

When you are finished with a Tool, you can delete it from your workspace. This will stop all calculations that are being performed on the tool and will break any linked data that is being read from the tool (for more information on linking data, see page 83). Hiding a tool is similar to delete in that it will be removed from your view, but it will continue to run in the background and can be returned by using the *Window* option from the main menu. To hide or delete a Tool, click the X icon in the upper right corner of the Tool. You will be presented with the dialog in Figure 74.

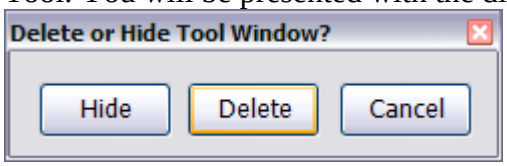


Figure 74 – Hide or Delete Tool Window

Adding additional Frames

A "Frame" is a part of the workspace where you can dock your tool windows. By default workspaces have one frame (the main working area with the File menu and the toolbar with the Start button). If you wish to group tools into additional frames (for instance to view on a second monitor) you can add more frames to your workspace by clicking the *Add Frame* button from the main toolbar. This will add an empty frame (figure 74) that you can drag your tool windows into and dock into position.

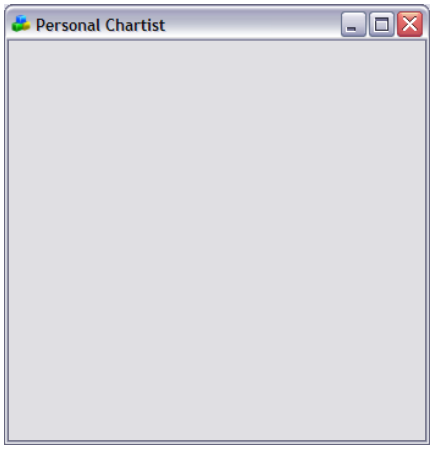


Figure 75 – An Empty Frame

Saving Custom Workspaces

Once you have positioned your tools and configured your charts WatchLists and personal settings, you can save the workspace to load it for future reference. To save a workspace, simply click on the Save button on the main toolbar.

Saving a Customized Tool

You can save Tool Windows with your own custom settings to be loaded from the Start button. For example let's say you've configured a chart with your favorite indicators. You can save this chart to load into any workspace. To save a tool, simply right click on the tool header and select *Save As Tool Window* from the menu.

Creating a Tool from scratch

The tools loaded from the Start button are pre-configured with some settings but you can also create new Tool windows from scratch with the File – Create New Tool Window option. Creating tools from scratch are beyond the scope of this book but there are many to choose from. If you're more the adventurous type you can browse the list and create some composite tools from those that are available. Some more interesting tools are: LED meter, Speedometer, Horizontal Toolbar, Knob, Command Button.

Hot Keys and Commands

Each Tool can have a series of command that can be mapped to a hot key. For example, the WatchList tool has a Next Symbol command that will change the active symbol to the next one in the list. A workspace also has a series of commands that can be mapped. For instance you can map a hotkey to show/hide a tool window.

Viewing the Hot Keys

The Personal Chartist workspace come's pre-configured with some hotkeys. To view the hotkeys click the System – Hot Key Mapping. This will bring up the Hot-Key manager (figure 76) that lists the current hotkeys and the commands they are mapped to and the target tool. The target tool differentiates between multiples of the same tool in a workspace (for example if you have two WatchList tools)

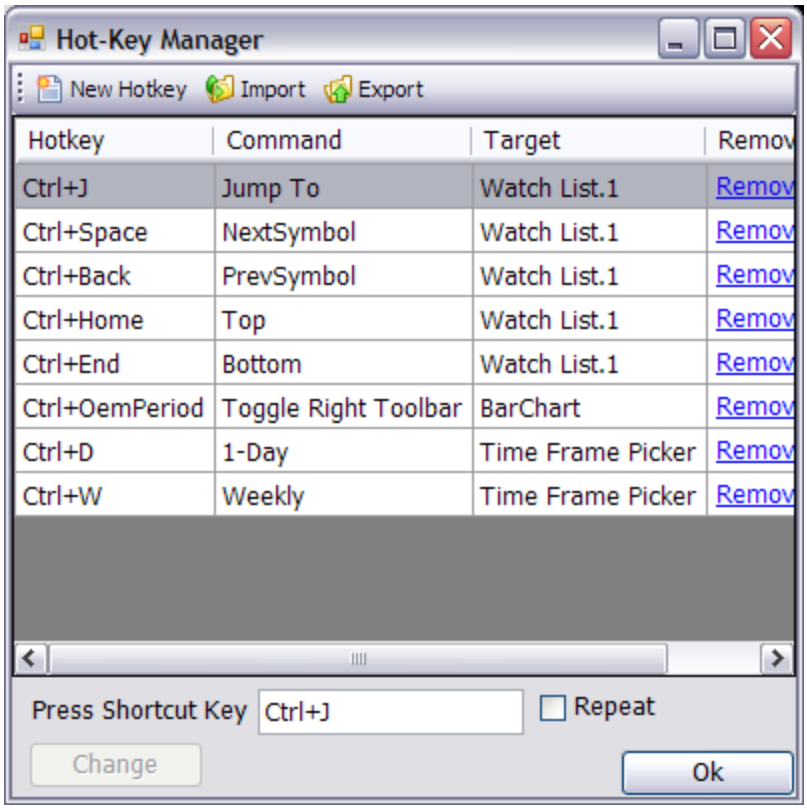


Figure 76 – Hot-Key Manager

Changing an existing hotkey

To change an existing hotkey, simply click on the command in the Hot-Key Manager, highlight the current hotkey in the *Press Shortcut Key* box (bottom of the form) and press the new hotkey (or hotkey combination) you would like to use. Hit the *Change* button and your new hotkey will be mapped to the command.

Mapping a hot key

To map a new hotkey, click the *New Hot Key* button. You will be presented with a list of all the tools in the workspace and all the commands that can be mapped. Browse to the command you wish to add a hotkey, put your mouse cursor in the *Press Shortcut Key* box and type the hotkey you would like to assign. Click the *Map* button to assign the new hotkey.

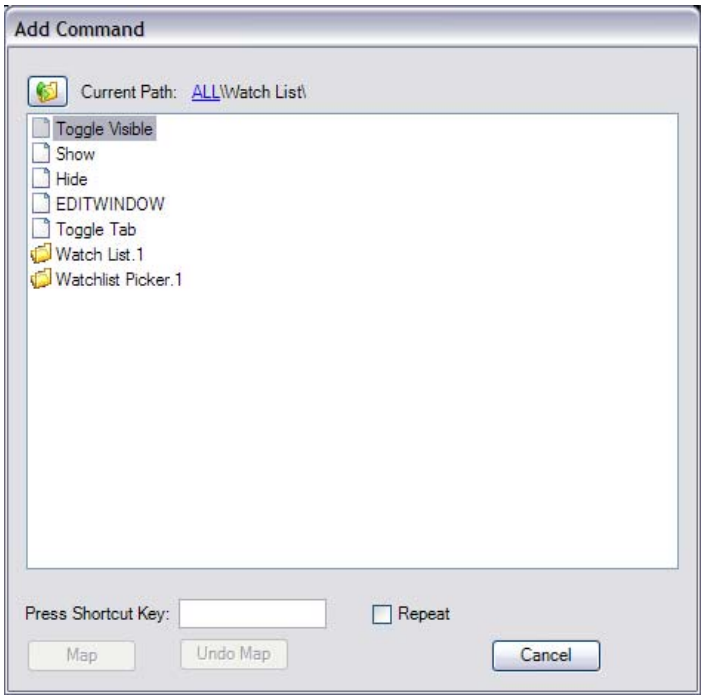


Figure 77 – Add Hotkey Mapping

Saving/Loading a set of hotkeys

When you save a workspace, all your hotkey mappings are also saved with the workspace. Additionally you can save your current hotkeys from your workspace to load them into a different workspace or share them with a friend.
To export your hotkeys, click the Export button from the HotKey manager. Select a file name and click save. To import a set of hotkeys, click the import button and browse to the file with the hotkeys to import

Figure 78 – Import and Export hotkeys button

Issuing a Command from a Command button

Besides issuing a command from a hotkey, you can assign a command to a command button. Command buttons are generally placed on a toolbar (see the top of the Chart in Personal Chartist). Using the properties of the command button, you can change the Command property and pick the command to execute when the button is clicked.

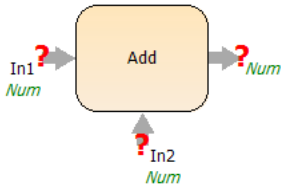
Chapter 6: Introduction to Block Diagrams

The magic behind the Block software is in the Block Diagrams. Diagrams control nearly every calculation that occurs in the system. From deciding what list of symbols to display, to the latest price, company financials, strategy conditions, custom indicators, time frames, data feeds, plot color, nearly everything is controlled through a block diagram. They're easy flow diagrams that read like a flow chart. While they may look intimidating at first, they're actually quite easy to read, use and modify.

What this means to you as the end user is you can "get under the hood" and take a peek at the "guts" of the program. As you get more comfortable reading block diagrams you might even modify one to change a calculation to your liking. Before you know it you'll be able to create diagrams from scratch and then you'll be able to harness the real power behind the Blocks software.

What is a block?

A block is an item that works with data. Blocks have connectors (represented as arrows) to consume and/or supply data. They are also known as *code blocks* as they are made up of fragments of code. For more information on code blocks and writing your own blocks in C# or VB.net, see chapter 10.



What is a diagram?

A diagram is a collection of blocks, with their inputs and outputs connected to create a data flow. Figure 79 shows a block diagram that adds 5 plus 4 and outputs it to a data display.

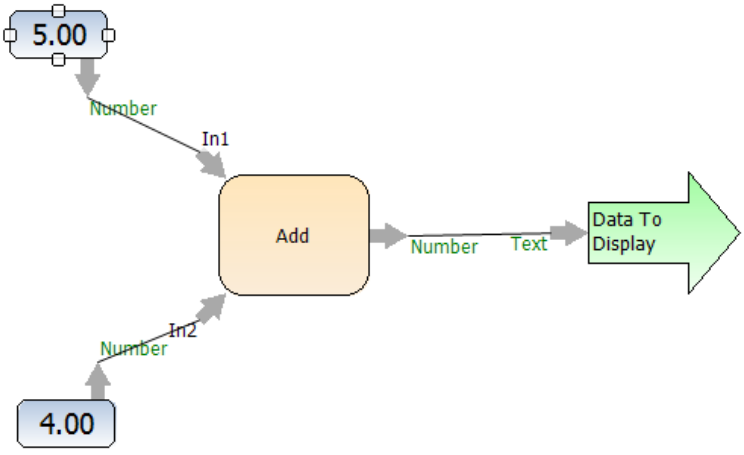


Figure 79 – A Block Diagram.

What uses diagrams?

Anything that consumes or supplies data does so through a block diagram. Nearly every diagram has at least one block that consumes data. For more information on consuming block diagrams, see page 72.

Adding a Block to a diagram

To add a block to a diagram, right click on the white space of the block diagram and choose *Select Block...* You will be presented with the blocks library (figure 80)

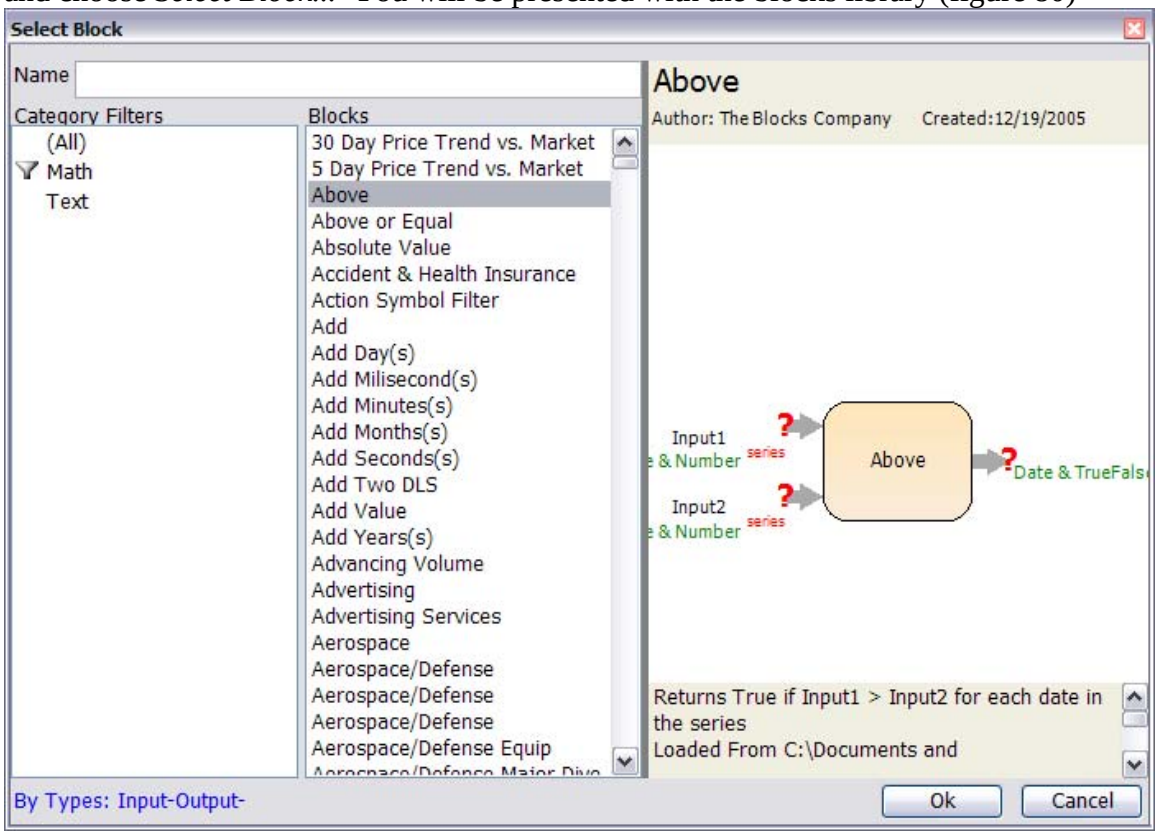


Figure 80 - Block Library

The blocks library lists all the blocks that are available to add to the diagram. You can filter the list by typing the name of the block in the Name field. You can get a preview of the block by clicking on the block name and a preview will be displayed in the right hand preview pane. A description of the block will also be presented in the preview pane to help you determine the functionality the block provides. Once you have selected the block to add to your diagram, click the ok button.

Selecting a Block by valid connections

When adding a block to a diagram, you are most likely going to connect it to an existing block. You can kill two birds with one stone (adding and connecting) by dragging off the connector you wish to connect to, and dropping in the white space of the block diagram and choosing *Select Block & Connect*. This will filter the block library by displaying only blocks that will provide or consume data of the connector type you

dragged from. Once you select a block it will add it to the diagram and connect the two blocks together.

Connecting Blocks

Connectors

Blocks either provide and/or consume data. The data is fed into the block by an input connector (an arrow pointing into the block) and is provided by the block by an output connector (an arrow pointing out of the block). Each connector describes a type of data that the block is consuming or supplying.

Common Connector Types

In order for blocks to connect to one another, they need to understand the same type of data. The most common connection types are:

Symbol

Also know as the DataID, this describes a set of data. This is used to display a stock symbol, or select the stock symbol to use when loading prices data from a Prices block.

Symbol Series

Also known as a SymbolList, this data type describes a series of Symbols, more commonly referred to as a WatchList.

Text

Also known as a String, this is any textual data.

Integer

This data type represents a whole number from negative 2 billion, to positive 2 billion. Integers inherit from Text so anything that takes a Text input will also accept an integer

Number

Also known as a Single, this is a decimal number with single floating point precision ranging from negative 3.402823e38 to positive 3.402823e38. Numbers inherit from Text so anything that will accept Text will also accept a number.

Date

A standard Date and Time data type. To connect to a Text input, use the Date Text Format block.

T/F

Also known as a Boolean data type, represents a True or False value.

Date & Number Series

Also known as DateLineSingle, this connector type is a series (array) of dates and (decimal) numbers [Single point precision]. This is the most common connector type for drawing a line on a chart or describing values over a period of time.

Date & Bar Series

Also known as DateBarSingle, this type is similar to a date and number series, only instead of one value per date there are 4 values: Open, High, Low and Last. This data type is used to describe values over a time period and is most common in stock prices. This data type is compatible with Date & Number series so anything that will consume a Date & Number Series will take a Date & Bar series and use the Last (close) value. (NOTE: if you wish to use the open, high or low, from a DateBarSingle in a DateLineSingle connection, you need to use the Bar Open, Bar High and Bar Low blocks between the DateBarSingle connector and the DateLineSeries connector.)

Date & T/F Series

Also known as a DateLineBoolean, this type represents a series of dates and T/F values. This data type can be used for chart markers, scan and testing conditions and any other date Boolean logic.

Date & Color Series

Also known as a DateLineColor, this type represents a series of dates and colors. This can be used to draw a plot in different colors based on any conditions.

Consuming Block Diagrams

Every diagram exists because there is some item that wants to consume data. All diagrams have one or more block that represents the eventual data to be consumed. The most common consumers of data in the stock market are described below

Plots

Plots are lines on a chart. Numeric plots consume a Date & Number series. Figure 81 represents a diagram for a stock price chart. Notice the *Prices* block is fed a *Time frame* and a *Symbol* and outputs the price data to the *Numeric Plot*. The *Numeric Plot* (right green arrow) is the ultimate consumer of the diagram and represents what is plotted on the chart. Also in the diagram is a green circle with the label Color. This is a secondary consumer of the diagram that provides the color to use when plotting. The color block will take a single color, or a Date & Color series that defines a color for a given day.

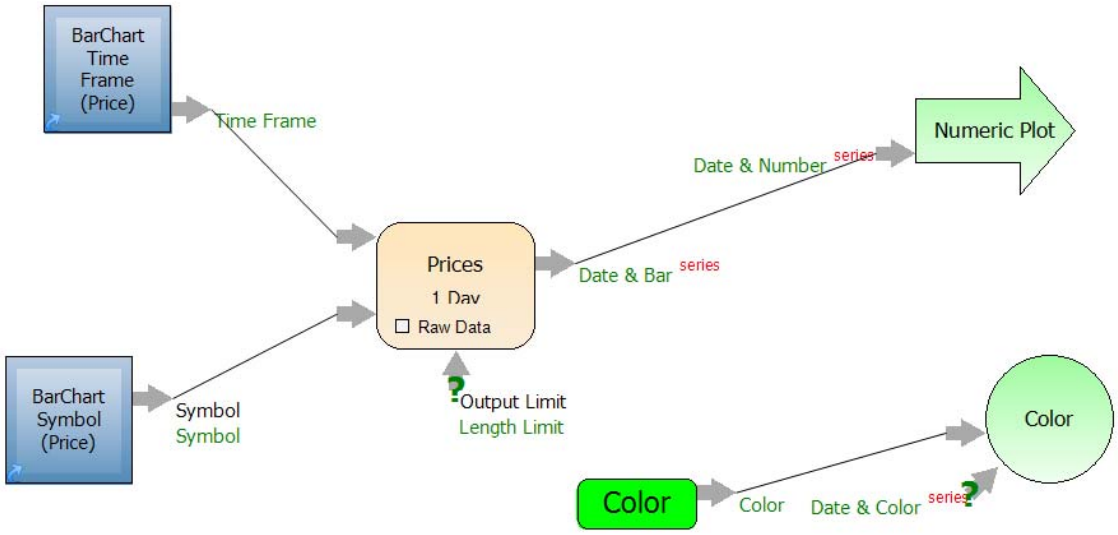


Figure 81 – A diagram for a Price Plot.

Columns

Columns represent data in a tabular format on a WatchList. Depending on the type of column, (Numeric, Boolean, Text) they consume data for a given symbol to display. Figure 82 represents a Text Column that displays the company name. This diagram provides a symbol (column symbol block) and consumes text (Column Data block).

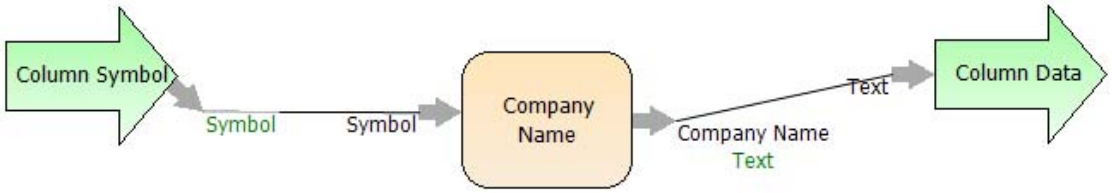


Figure 82 – Company Name Column

Strategy Conditions

Strategy Conditions are much like columns in that they provide a symbol to the diagram, but instead of consuming textual or numeric data, they consume Date & T/F series. If a condition is to pass on a given date, then it will have a True value for that date, otherwise it will be false. Figure 83 shows a condition to check if a 10 bar moving average is crossing down through a 50 bar moving average. On every date that they cross, there will be a True value, otherwise it will be false. Notice this diagram provides a test symbol block and consumes a Date & T/F series in the T/F condition block.

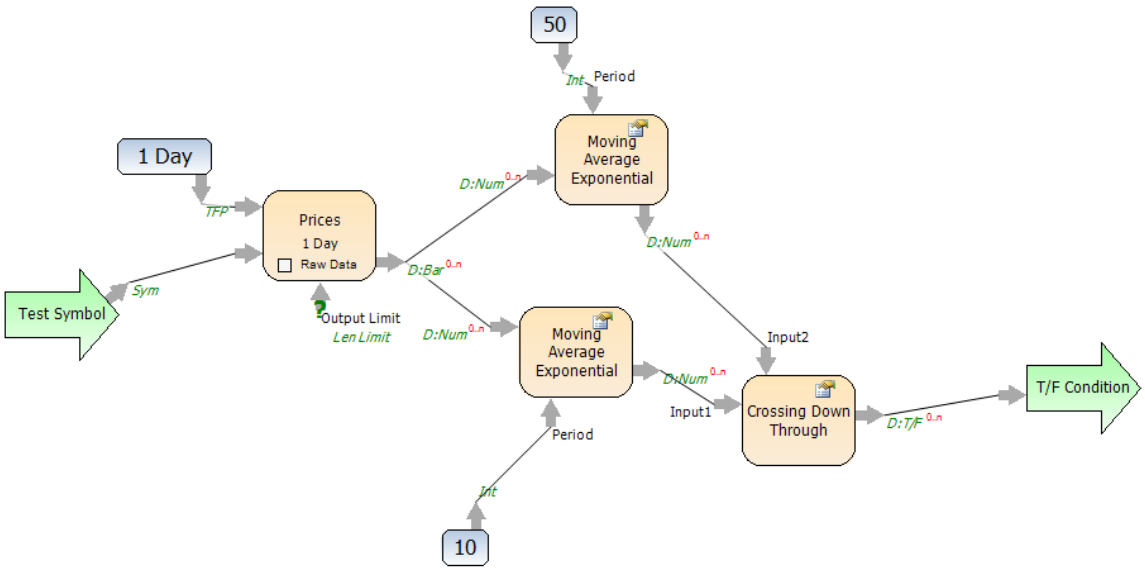


Figure 83 – A Moving Average Crossing Condition.

Chapter 7: Working with Block Diagrams

The best way to get comfortable with the concept of a block diagram is to simply use one. First we'll go over with the basics of manipulating the blocks in a diagram and then we'll dive right in and do an exercise to make a simple calculator.

Moving Blocks in a Diagram

To change the position of a block, simply click and drag the block to a new location. The connectors on the block will rearrange to find the shortest distance to its connection.

Moving Connectors

Connectors always try to find the shortest distance to the block it's connected to, but sometimes this results in overlapping connectors. To move a connector, right click and drag the connector to a new location around the block.

Moving multiple blocks

You can move multiple blocks by selecting more that one block with the lasso. To select multiple blocks, hold down the CTRL button, click in the white space of the block diagram and drag the mouse to "lasso" the blocks together. Selected blocks will have 4 white boxes along the edge to indicate they are selected. Once you have selected multiple blocks, simply drag any of the selected blocks to move them all at the same time.

Connecting Blocks

To connect two blocks, simply drag from the connector of one block and drop it on the connector of the second block. While dragging, valid target connectors will highlight in blue.

Changing the results of a calculation

You can change the results of a calculation by adding or removing blocks from the diagram and changing their connections. Sometimes simply reversing the order of blocks will change its mathematical precedence and result in a different calculation.

Changing the data source

All data in a diagram must either be provided by a block or be provided by another tool in the Workspace. Most data sources are either stock prices, or lists of symbols, or fundamental criteria. To view the available data sources, right click on the block diagram and choose *Data Sources* from the menu. Tools can also provide links into their own data. For instance the WatchList tool provides a selected symbol that can be used in diagrams that require a symbol. A plot on a chart will also expose its own data to be used in the calculation of another plot, say for moving average.

Linking to other tools

The ability for diagrams to consume data, and the ability for tools to expose their data allows data to be linked in the workspace. This gives the ability for a WatchList to change the active symbol and have the price plot load the correct prices for that symbol. For more information on linking tools, see chapter 9

Block Diagram Exercise: Making a simple calculator

Our calculator will display the results of a calculation, so let's create a new tool window to display our data. Click *File – Create New Tool Window*, Select *Data Display* and click ok.

A new Data Display tool window will be added to your workspace. Right click on the data display and choose properties. This will open up the block diagram for the data display.

Your block diagram should have the blocks in figure 84. You may need to rearrange the blocks by dragging them around the diagram.

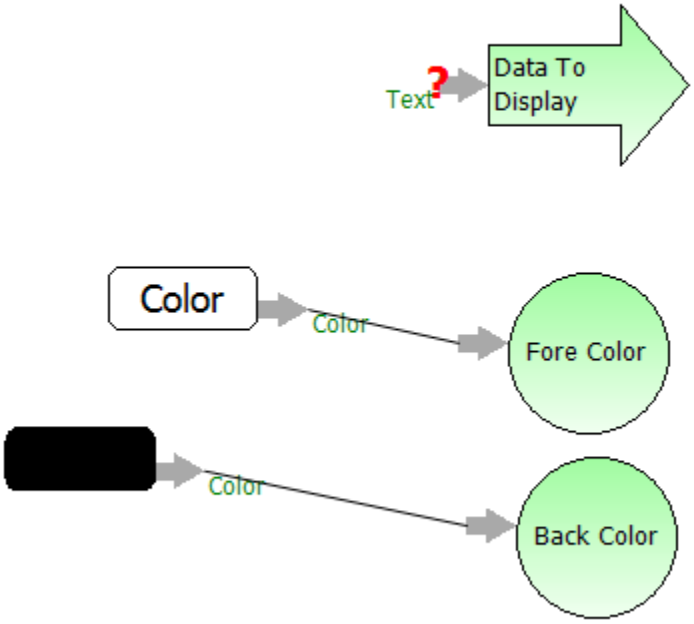


Figure 84 - Empty Data Display Diagram

You'll notice the Data display consumes 3 types of data, the text to display, the foreground color and the background color. This allows you to programmatically control all the display elements from within the block diagram.

The first thing we want to do is some basic addition, so let's select the *Add* block from the library. To do this, right click in the blank space (white background) of the diagram and choose *Select Block…* Type *Add* to filter the list of blocks and choose the *Add* block by clicking on it in the list and choosing ok.

Your diagram should now have a block in it like figure 85

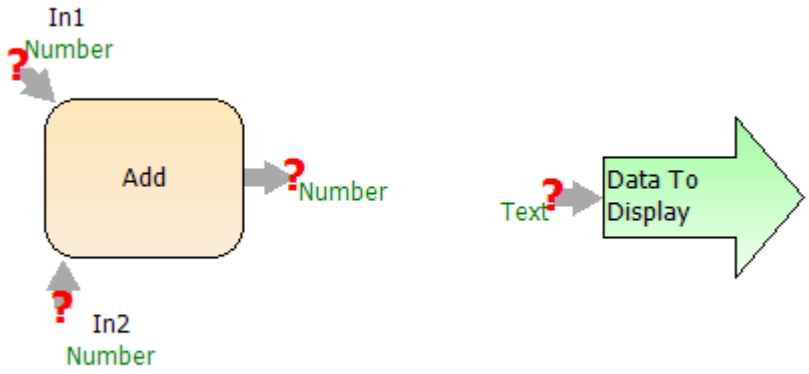
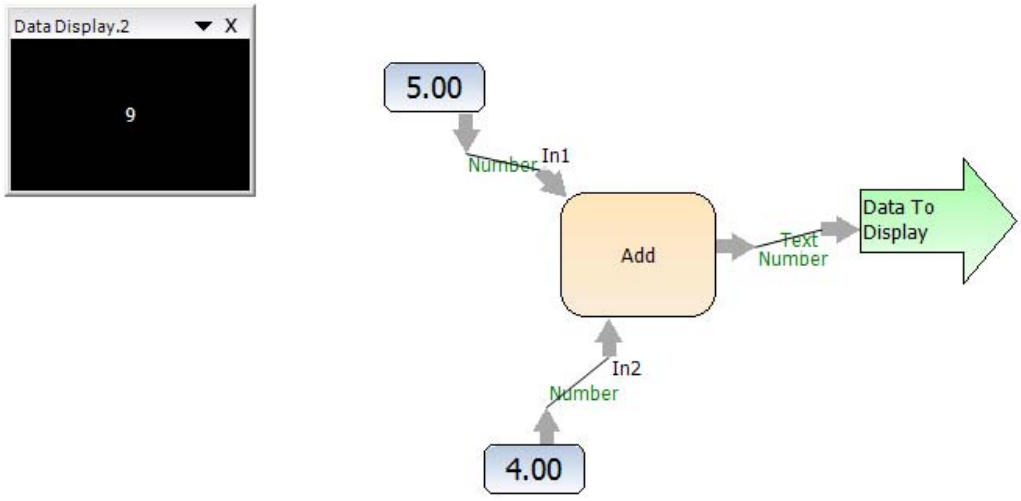


Figure 85 - Data Display with *Add* Block

Connect the add output connector (arrow coming out of the block) to the Text input connector on the green *Data To Display* block. When you click and drag off the output, the text input connector should highlight in blue. "Drop" the connector onto the input by releasing the mouse button. The two blocks should now have a line between them indicating a connection.

Now we need to provide two numbers to our addition block. This time, let's drag off the connector labeled *In1* and drop in the white space of the diagram. This will bring up a menu, click on *Select Block & Connect.* The block library will now be filtered by blocks that provide a numeric output. Type *number* to filter the list and select *Parameter: Decimal Number*. This will add a blue block with the number 0.00 connected to *In1*. Repeat this process for *In2*. Once both blocks have been connected to the *Add* block, your data display should read 0. Double click on one of the parameter blocks and change the value to a different number. You should notice that as you change the values of your parameter blocks your data display should update with the sum of the two values.



You can continue to modify this block diagram to use more math blocks and create a more complex calculator. You can also substitute Numeric Selector tools for the parameter blocks and link the Numeric value from the tools into the diagram. For more information on linking tools, see chapter 9.

Part 3 – Advanced Block Diagrams

Chapter 8: Parameter Blocks & QuickEdit

Parameter blocks provide user defined data to a block. In the Simple Calculator example (page 76) we used two *Parameter:Decimal Number* blocks to provide values to add. Most parameter blocks are drawn with a light blue background and display their current value in the block. There are parameter blocks available for many data types.

Changing a Value

To change the value of a parameter block in a block diagram, simply double click on the block. A dialog (figure 86) will appear. You can change the value of the parameter block by editing the bottom item labeled *Value*.
Parameter values can also be changed by Quick Edit. Many items expose a quick edit dialog to change the value(s) of parameter blocks. For more information on using quick edit, see the next section.

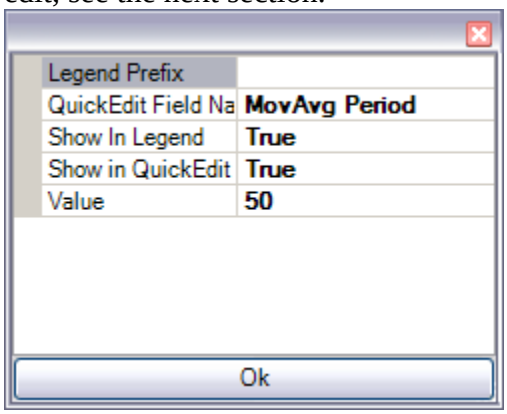


Figure 86 - Edit Parameter value

QuickEdit

Quick Edit is a dialog that allows the user to change the values of a block diagram, without opening the block diagram. Parameter blocks can be "promoted" to display in the quick edit. Double click on a parameter block and set *Show In QuickEdit* to true. You can edit the label that is displayed in the quick edit by setting the *QuickEdit Field Name.*

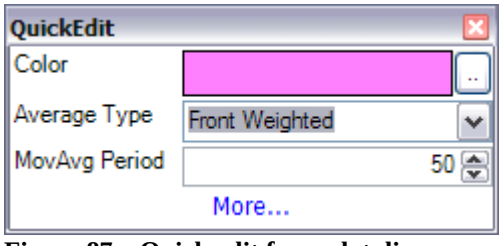


Figure 87 – Quick edit for a plot diagram

QuickEdit Replace

In addition to changing the value of a Parameter block, you can use QuickEdit to actually replace blocks in the diagram. This is great way to reuse the same logic in a diagram, but change one part of the calculation without having to build a new diagram from scratch.

One great example of QuickEdit Replace is to change the Average type when using the moving average Study. Traditionally there are 3 different ways to calculate a moving average: Front Weighted, Simple, and Exponential.

To enable QuickEdit Replace on a block, right click on the block, and select *Enable QuickEdit Replace*. Your block will now have a new property icon in the upper right corner (Figure 89).

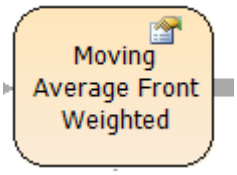


Figure 88 – Moving average with QuickEdit Replace enabled

QuickEdit Replace Properties

You can control how the QuickEdit replace displays and the blocks it allows you to replace with the QuickEdit Replace Properties. Click on the Property icon on the block (that has QuickEdit Replace enabled).

Quick Edit Field Name – This is the label that is displayed on the QuickEdit dialog

Replace From – Library or Manual List. Library will allow all blocks in the library that meet the valid connector types. Manual list lets you specify what blocks are displayed to be replaced.

Blocks List –Manual List mode allows you to choose the blocks that will be displayed in the Replace field. Use the Add and Remove buttons to change this list.

Quick Edit Choice Name – (Optional) This allows you to set the label of the block when displayed in the quick edit replace dropdown

Legend – (Optional) allows you to set what the chart legend will display if this block is selected for quick replace.

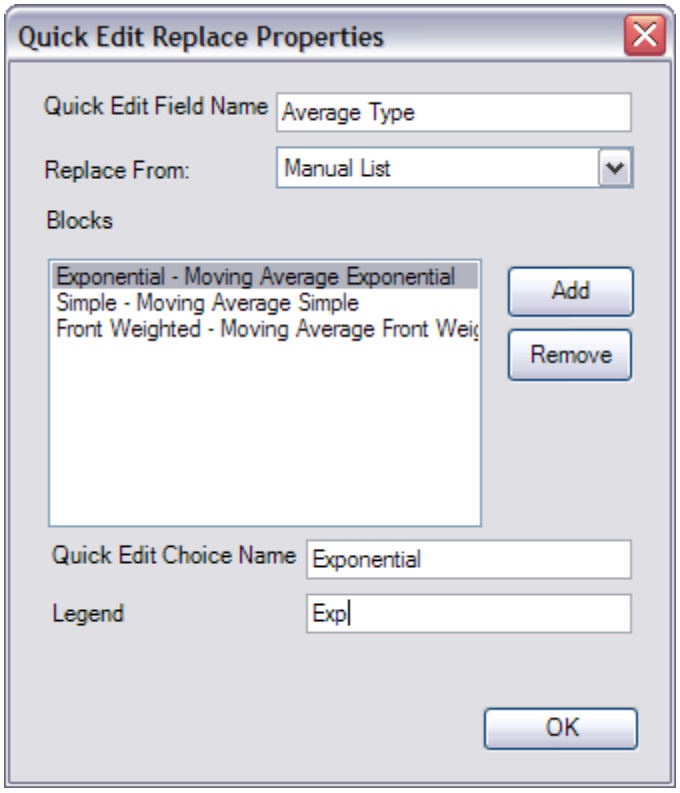


Figure 89 – QuickEdit Replace Properties

Figure 90 is a QuickEdit Replace Properties for the moving average study. It has 3 blocks that it allows you to select in the QuickEdit dialog: Moving Average Simple, Moving Average Exponential and Moving Average Front Weighted. Each of these blocks has been given a shorter QuickEdit Choice Name: Exponential, Simple and Front Weighted. If you came up with your own moving average calculation and wrote a custom code block, you could add it to this list by clicking on the Add button and selecting your custom code block from the list. See page 91 for more information on writing custom code blocks

Chapter 9: Linking Diagrams (to Tools and other diagrams)

Linking to Tools and Diagrams

Blocks can be connected to one another in a block diagram. Sometimes the data you need for your diagram has already been calculated in another diagram in the system. Other times you need data that is not in a diagram but is exposed by a Tool. The *Time Frame Picker* tool exposes a timeframe connector to be used in a block diagram; this allows the diagram to calculate based on the selected Time Frame. If multiple diagrams are connected to the same Tool, then they will be synchronized using the same Time Frame.

Linking a block to a Tool will normally will replace the need for a Parameter block. Nearly every data type that is consumed by a tool "exposes" its data to be used in other diagrams. You can "link" your blocks to these exposed data sets thereby re-using the calculation from another diagram and "daisy chaining" the diagrams.

For example, a Moving Average study on a chart can link its data from a Price study to create a data chain. When the price plot diagram recalculates, the moving average diagram will also recalculate using the new price data.

Figure 91 is a Block Diagram for a Stock Price plot. Notice the Prices block in the middle of the diagram. It is connected to two blue blocks on the left hand side of the diagram. These two blocks are blue, because they are "Linked" externally from the block diagram. The Top blue block is labeled *Time Frame Picker;* this prices plot is reading its timeframe from a tool named *Time Frame Picker*. The bottom blue block is providing a Symbol for the prices block; it is getting that symbol from *Watch List 1 Selected*.

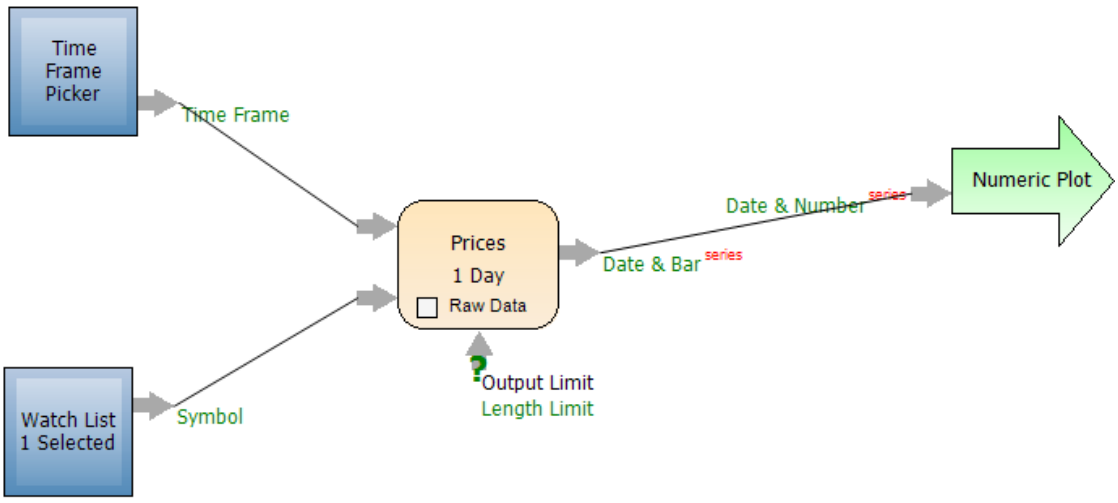


Figure 90 – Price Study with Links to WatchList 1 and Time Frame Picker

To link to another Tool or to change where a block is connected for its input, simply change the input connector to the desired item. Left click and drag off the input connector and drop it into the white background of the diagram. Choose *Link From Another Tool.* The sub-menu (figure 92) will be filtered to display all the Tools and

diagrams in the system that provides the appropriate data for the connector you're connecting to. Select the item from the sub-menu and it will connect.

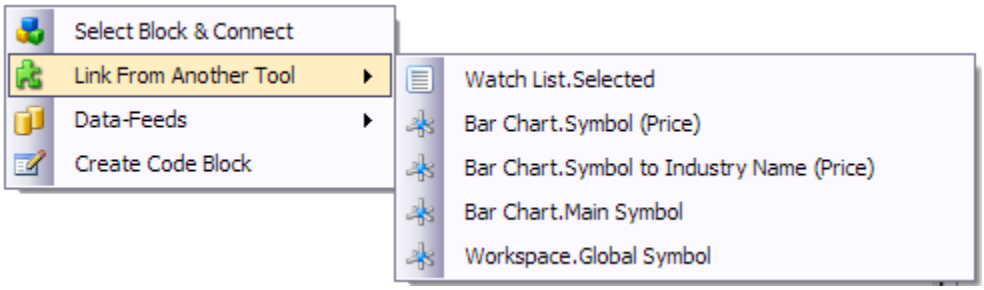


Figure 91 - Link from another Tool

Alternatively, you can also click and drag from the input connector, but instead of dropping it onto the white background, many tools support drag and drop right onto the Tool surface. To change the Symbol in figure 93 to a different WatchList, you can simply drag off the Symbol input connector and drop it on the WatchList Tool in your workspace. It will change the input connection and you will now be reading the prices data for the symbol selected in the newly connected WatchList. Notice the drag and drop line (figure 93) extending from the input connector on the prices block to the cursor over the WatchList Tool. This same drag and drop technique can be used to link to data in other studies on a chart or from a timeframe picker tool.

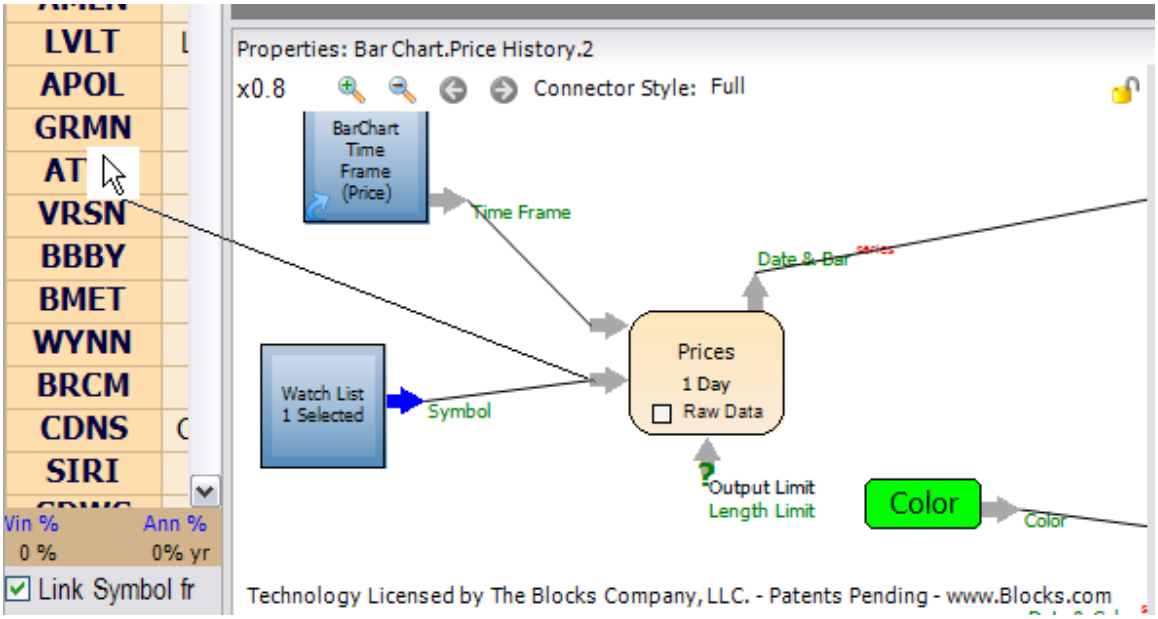


Figure 92 - Link to WatchList, drag and drop

Navigating Block Diagrams with Links

When you link a block to an external source, it has a blue shaded block representation. Some of these are links to other diagrams. When you are linked to another diagram, you have a shortcut indication in the lower left corner (figure 94) while if you're linked to a tool that simply provides a value (no diagram) then there is no shortcut link (figure 95).

If you are linked to another diagram, you can view the diagram by double clicking on the block with the shortcut (figure 94). This will show the diagram for the data source you are connected to. You can continue to "walk" a diagram link structure until you come to the ultimate source of the data. For stock charts this will usually end up at a symbol provider tool and a timeframe provider tool.

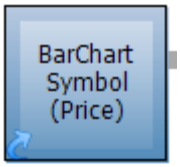


Figure 93 – Link to another Diagram (note Blue shortcut icon in lower left corner)

Figure 94 – Link to another Tool (no shortcut icon, cannot follow link)

When you have navigated to another diagram, you can return or "go back" to the previous diagram by using the backwards and forward buttons at the top of the diagram window (figure 96).

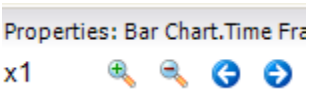


Figure 95 – Back and forward buttons when browsing data links.

Loading diagrams with links to other tools

When a Tool (or a tool part, like a study or scan condition) is saved that has links to other tools, a reference is saved so that it can reconnect when it is loaded. This reference is saved by the Name of the item you are connected to. If the item does not exist in the workspace when you re-open the Tool, it will attempt to connect to an appropriate connection within the workspace level. If it cannot determine what to connect to, you may be prompted with a dialog and be presented with valid connections for your missing link.
Alternatively, missing links may be converted to a Window Input.

Window Inputs

A Window Input is a way to change a connection inside a diagram, without opening the diagram. Any input connection can be "promoted" to a Window Input. Example: The Bar Chart Tool has a window input to change the Symbol source for the bar chart. This lets you change the Tool that the bar chart is using for its active symbol without opening a diagram.

If a tool has a Window Input, then the Window Input icon will appear in the upper right corner of the Tool Window (next to the X icon).

To change the connection for a window input, click the Window Input icon to display the list of Window Inputs. You can change the connection by using the sub menu

(figure 97) or by dragging from the Window input name (Symbol in figure 97) and dropping on the tool you would like to connect it to. For more information on drag and drop connecting, see the previous section on Linking tools and Diagrams on page 83.

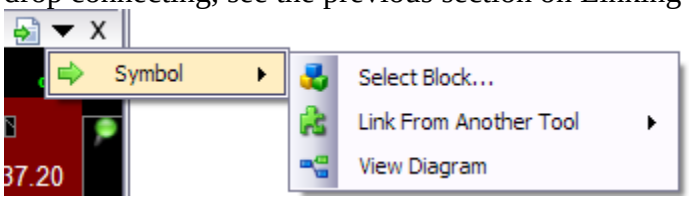


Figure 96 - Symbol Window Input

When working with your own diagrams, you can promote block inputs to window inputs by right clicking on the input connector and selecting *Expose as Window Input*. You will be prompted for the label to use on the window input. Input connectors that are exposed as window inputs are highlighted in orange and have the Window Input: label next to them (see figure 98).

To remove a Window Input, simply right click on the input connector in the diagram and choose *Remove Window Input*.

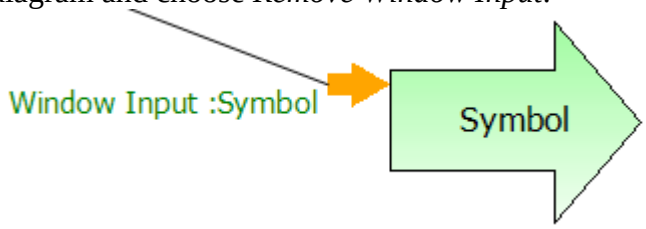


Figure 97 - Symbol Block with a window input connection

Patch Panels

Patch panels make organizing your workspace easier and allows you to re-use calculations without having to duplicate the diagram in many places. You can use them simply as “global” diagrams. A Tool Window, some tool parts (like the chart pane) and the Workspace can all have patch panels. Patch panels have a unique name in the workspace and are referenced in a diagram by this unique name.

Patch panels are utility diagrams that allow you to “patch” data. They’re really useful for synchronizing multiple diagrams in one central location. Think of them like the old telephone patch that an operator would use. One caller would be “patched” to the receiver. The only difference is that in Blocks, you can have multiple people listening to a patch panel.

Example:

The Bar Chart tool that comes with the Personal Chartist workspace has two patch panels: Symbol and Actions.

Let’s examine the Symbol patch:

A stock chart generally deals with a symbols price data and aggregate data based on the symbol. We know that charts are made from one or more studies (plots) and each study is made from a diagram. By default the Bar Chart gets its symbol from the WatchList tool. Each plot that needs a symbol is therefore getting its symbol from the

WatchList Tool. They can each be directly connected to the WatchList for their symbol. But what happens if we add another WatchList Tool to our workspace and want the chart to use the new WatchList? If each of the plots were directly connected to the WatchList, we would have to go change each and every diagram to use the second WatchList tool. This is where patch panels come in.

Patch Panel Uses

Instead of connecting each diagram of a study (plot) directly to a WatchList, we can create a Tool Window patch panel or a Pane patch panel. This patch panel will be a "pass through" diagram. It will consume a symbol and also provide a symbol. Each plot in the chart will connect to the patch panel symbol and the patch panel symbol will in turn be connected to the WatchList. This way, if we want to change the symbol for the chart, we only have to adjust it at the one location and all the other plots will be updated with the new symbol.

You can even promote the input in the patch panel up to a Window Input to make changing the symbol for the tool quite easy and this is, in fact, what the Bar Chart tool does.

Patch Panels can also be added to panes in a chart. This lets you apply the same principal on the pane level. Each plot in a pane can connect to the pane level patch panel so an entire pane can be changed without having to change each individual plot, but not needing to change the symbol for an entire chart.

You will notice that the Price History study in Personal Chartist is connected to a patch panel. Open the diagram for Price history (double click on the price history plot) and notice it is connected to *BarChart Symbol (Price)* [figure 99]. This is the name of a patch panel on the Price History Pane. We can navigate to this diagram by double clicking on the *BarChart Symbol (Price)* block.

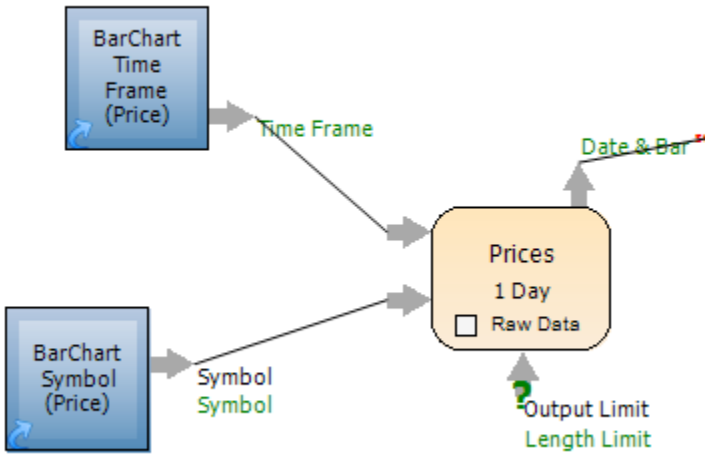


Figure 98 – Prices Diagram connected to two patch panels BarChart Symbol (Price) and BarChart Time Frame (Price)

The *BarChart Symbol (price)* patch panel (figure 100) is named as such to differentiate it from the *BarChart Symbol* patch panel which is a tool level patch panel.

This one is for the Prices pane, so it has the (price) part on the end. Patch panels can be named anything you choose when you make them (names can be changed from the Name entry of the property dialog).

You will notice that this patch panel provides a symbol, but also has two legend display blocks. These are special blocks that the Chart tool uses to display items in the legend area of the pane. This patch panel not only provides a symbol back out of the diagram, but displays the company name and the textual symbol representation in the legend. You will notice that this patch panel is getting its symbol from the *Bar Chart Main Symbol* diagram. This is a tool level patch panel and you can navigate to that diagram by double clicking on the blue block labeled *Bar Chart Main Symbol*

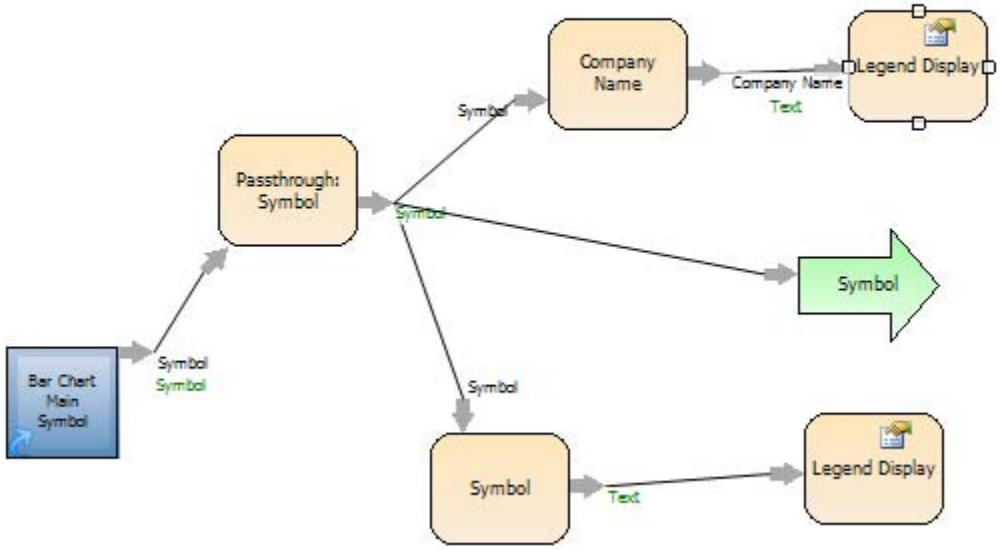


Figure 99 – BarChart Symbol (Price) Patch Panel.

Working with Patch Panels (viewing, adding, deleting)

Currently Patch panels are tied to specific data types (see page 71 for more information on data types). You can create patch panels for the following data types:

Symbol
Date & Number Series
Date & T/F Series
Number
Whole Number
Symbol List
Time Interval
Time Frame

You can access Tool window Patch Panels from the Tool Window drop down menu ▼ icon (upper right corner of the window). Panes level patch panels can be

access by right clicking on a pane. Workspace level patch panels can be accessed from the *Workspace* menu in the main menu toolbar at the top of the program.

Each of these menus will contain a Patch *Panel* sub menu. This will list the current patch panels (if any) and allow you to Add/Remove patch panels. Click on the Patch Panel name from the sub menu to view the diagram for that patch. Use the delete menu to remove a patch panel.

Choose the *Add Patch Panel* option to create a new patch panel. You will be asked for the data type the patch panel will consume/expose (figure 101). Select the data type and click ok. Your diagram will have a block to provide the patch along with the properties of the patch panel. You should choose an appropriate name to be able to identify your patch panel when linking to it.

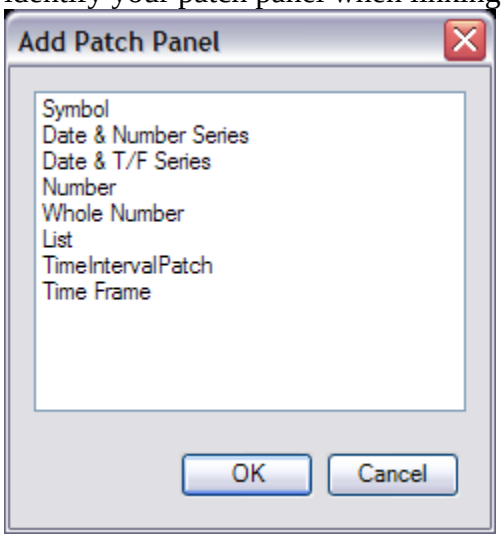


Figure 100 – Add Patch Panel

Patch Panels are great when you need a diagram to share data, or wish to partition (encapsulate) logic for multiple calculations.

Part 4 – Code Blocks

Chapter 10: Introduction to Code Blocks

Code Block Overview

Code blocks come in two forms in the Blocks software: Pre compiled (dll assembly) or source code. Many of the core blocks in the system are pre compiled, but a handful (around 20 or so) are compiled on the fly at application startup. You can view these existing code blocks as examples to get started on your own custom code blocks.

All code blocks are .net classes. They inherit from a specific base class. The base classes are templates that provide specific input and output connectors along with helper methods and functions to access the input data and provide the output data.

When you create a code block you choose the input and output template you want to create and thus choose the base class that your block inherits. You can write as many functions, methods and properties as you choose and can write any delegate, call back, and private classes as needed.

The Code Block Editor

The Blocks software includes a full featured code editor that mimics Microsoft's Visual Studio. The code block editor can always be accessed from the *System – Code Block Editor* menu option. Figure 102 shows an empty code block editor. The code block editor behaves like any Tool window in the workspace. You can have multiple code windows open at any time.

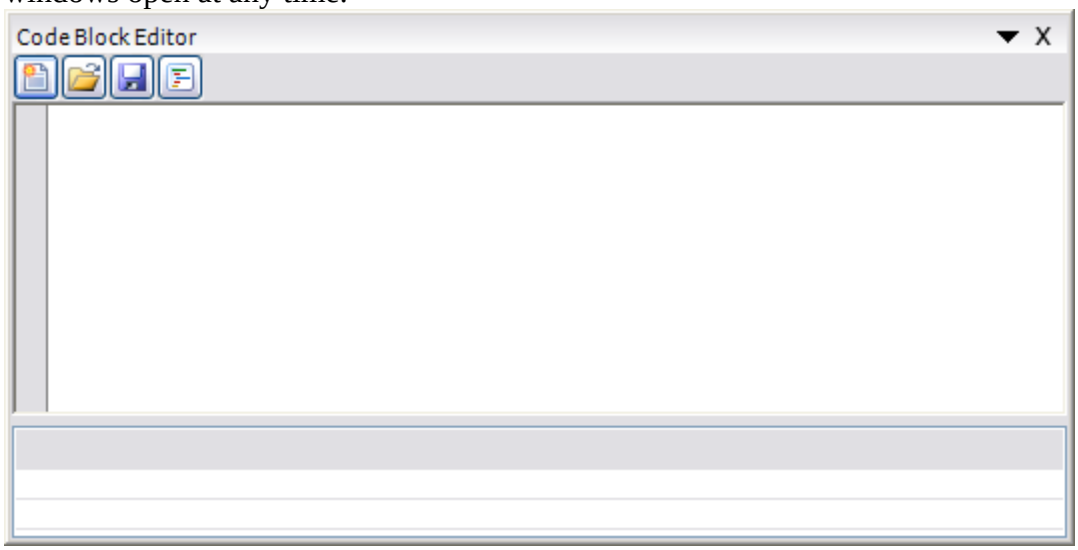


Figure 101 - Empty Code Block Editor

Opening a code block

You can open an existing code block by opening the code block editor and choosing the Open Code Block button from the toolbar. This will show you a file browser and you can navigate to the source code file.

Alternatively you can right click on a code block in a diagram and choose edit code. This will open the code block editor with the source code for the selected block.

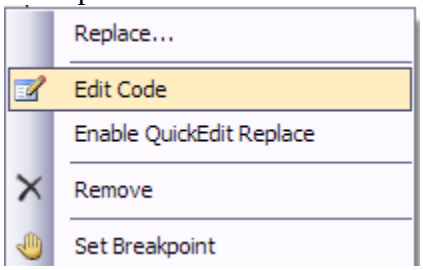


Figure 102 – Code Block Menu

Creating a new code block

Create a new block code block by opening the editor and choosing the new code block button. This will present you with all the available code block templates and give you a preview of the code block input and output connectors (figure 104).

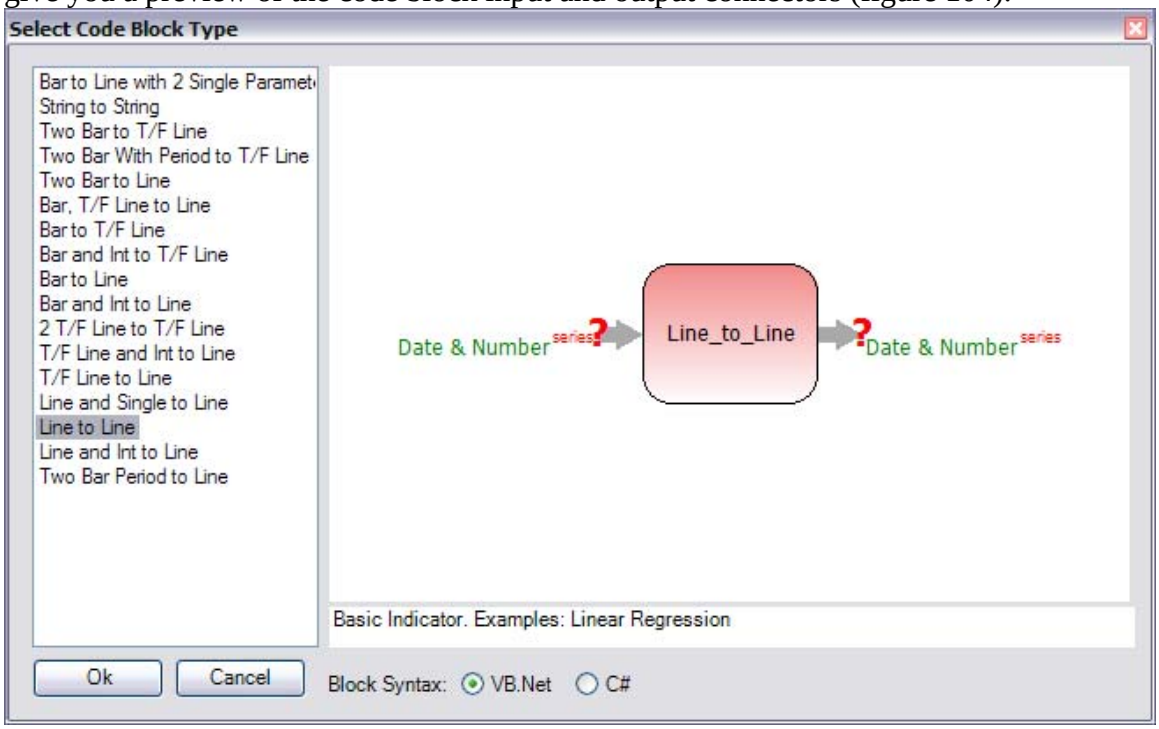


Figure 103 – New Code block template selection

Figure 104 shows the available templates at the time of this printing, but there are constantly new templates added. If a set of connectors is not available for the type of block you are attempting to create, you can make a request on the support forums.
You can choose your language at the bottom of the dialog; choose between VB.net or C#

NOTE: When saving code blocks, do not mix c# and vb.net source code files in the same folder. Code blocks are compiled on a folder by folder basis.

Choose your code block template, your syntax and click Ok. Provide a name for your code block (figure 105) and click ok; this will be the filename not the name that is presented in the block list.

Figure 104 – New Code Block name

The editor will stub out your class and you can fill in the calculate method to perform your custom calculation.

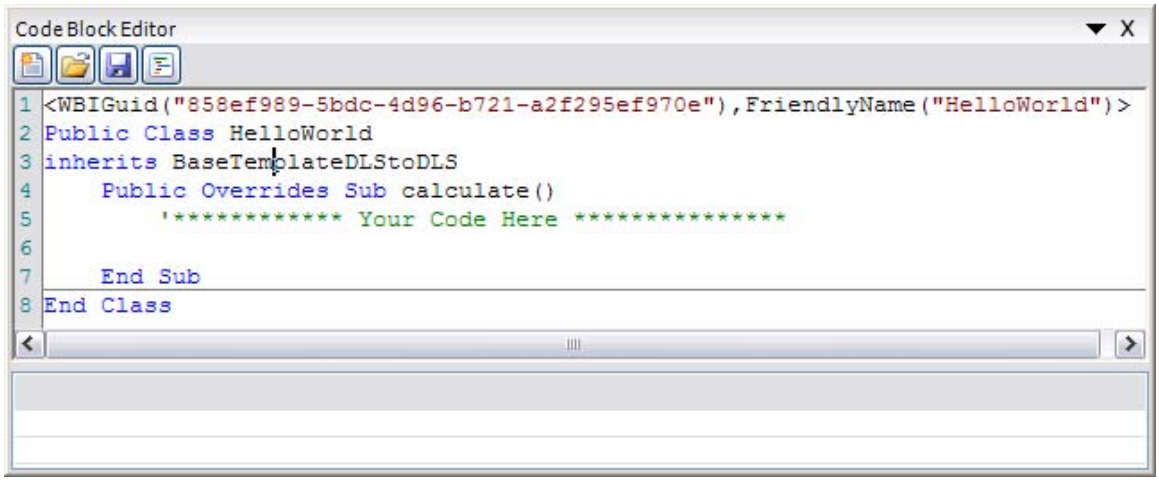


Figure 105 – New Code Block source code.

Compiling/Saving a code block

The code block editor behaves much like the Visual Studio 2005 editor and has syntax error highlighting. You can check your syntax with the Compile button. This will validate your syntax and provide a list of errors at the bottom of the editor. Double click on the error to jump to the line in the code that is invalid. Only once your block compiles successfully without any errors will you be able to save it. To save, simply click the save button from the toolbar.

Hello World: Writing your first code block

A picture says a thousand words and writing *hello world* is the de facto standard for an introductory tutorial in programming, so that's where we'll begin. The goal for our hello world example will be to take an existing plot on a chart and add 5 to the value on each date.

- Start by opening the code block editor – *System – Code Block Editor*.
- Click the New Code Block  button.
- Select the Line to Line template. We are taking a line of data (dates and numbers) and outputting a line of data (dates and numbers)
- Click on the VB.Net syntax option (you can use C# but we'll use VB.Net for this example)
- Name your block HelloWorldDemo.

Your code block editor should now contain the following code:

```
<WBIGuid("0cfe77b4-562a-445d-a59e-123fa28eea9a"), FriendlyName("HelloWorldDemo")> _
Public Class HelloWorldDemo
inherits BaseTemplateDLStoDLS
    Public Overrides Sub calculate()
 '************ Your Code Here ***************
     End Sub
End Class
```

Let's quickly review what each item in the source code. The class starts with two custom attributes, the `WBIGuid` and the `FriendlyName` attribute. `WBIGuid` is the unique identifier for your block, the system uses this identifier (not the class name or the friendly name) when referencing your block.

NOTE: Every block must have a unique `WBIGuid`. It is important that when you copy existing code that you let the system generate a new `WBIGuid` for you by choosing the new code block option. You should never reuse the same `WBIGuid`, doing so could cause the wrong code to be used in a calculation.

The `FriendlyName` attribute is what is displayed in the Library and in the diagram. You can edit this to include spaces and have a friendlier name.
You can include a description in your code block by adding a `ClassAuthor` attribute to your class. For more information on the class author attribute see page 114.

The next line of your source code has class declaration followed by ***`inherits BaseTemplateDLStoDLS`***. This defines the class you are inheriting from which provides you the helper code to access the input(s) and output(s) of your class.

The last block of code is the `calculate` method. This is where we're going to put our code to add 5 to the incoming line.

Everything we need to access our input(s) and output(s) is wrapped in a helper class named `CodeBlock.` You can browse the available methods and properties by using code browsing feature similar to Microsoft's Visual Studio Intellisense™. In visual basic syntax you can browse the available calls by typing `Me.CodeBlock` or `MyBase.CodeBlock`. In C# you would type `this.CodeBlock` or `base.CodeBlock.` From here, you can view all the available methods and properties you can call.

The goal of our block is to add 5 to the value of the incoming line. To do this we need to loop through all the incoming values. We can do this with the vb.net For loop syntax. Type the following code into the code block editor under the `Your Code Here` comment:

```
For i as integer = 0 to me.CodeBlock.InputCount – 1

Next i
```

This statement starts at a 0 based array index and loops through all the incoming items. If you are new to programming or the concept of 0 based arrays, the table below shows the structure of a *DateLineSingle* (Date & Number Series). This table illustrates an array with 5 items in it (`InputCount`). The first item in the array is at Index 0 and the last item is at index 4. Because we start at index 0 our loop in the example above needs to run from 0 to 4 or 0 to `InputCount`(5) – 1.

Input Value	56.50	57.01	55.14	56.10	57.46
Input Date	1/1/1980	1/2/1980	1/3/1980	1/4/1980	1/5/1980
Array Index	0	1	2	3	4

Now that we have the code to loop through each of the items, we simply need to output our new values. Add the additional line of code between the `For` line and the `Next` line:

```
AddToOutput(CodeBlock.InputDate(i),CodeBlock.InputValue(i) + 5)
```

This line reads in the incoming data and adds a new value to our output. Let's review. The first part of this line calls a helper function: `AddToOutput. AddToOutput` is expecting two values, a date and a number (single). Since we're not modifying the date that is coming in, we can simply pass the current date value to our output. We can get the date value through `the CodeBlock.InputDate` call. This returns the date at the specified index. The second parameter on `AddToOutput` takes the value for the specified date. We can get the value through the `InputValue` property for the given index. Since we wanted to add 5 we simply add + 5 to the end of the call.

Below is the complete source code:

Intellisense is a registered trademark of Microsoft Corporation

```
<WBIGuid("aadaa9a0-5ef3-4265-b2a0-503b65577370"),FriendlyName("HelloWorldDemo")> _
Public Class HelloWorldDemo
inherits BaseTemplateDLStoDLS
    Public Overrides Sub calculate()
            '************ Your Code Here ***************
     For i As Integer = 0 To Me.CodeBlock.InputCount - 1
            Me.AddToOutput(Me.CodeBlock.InputDate(i),Me.CodeBlock.InputValue(i) + 5)
     Next i
     End Sub
End Class
```

Click the Compile button to check your syntax. Once everything compiles correctly, click Save.

Leave the code block editor open (or dock it in the workspace). Let's create a new plot to view the results of your code block.
Right click on the top pane on the chart in the Personal Chartist workspace and choose *Create New – Plot*. Give your new plot a name, and choose to scale with Price History (figure 107)

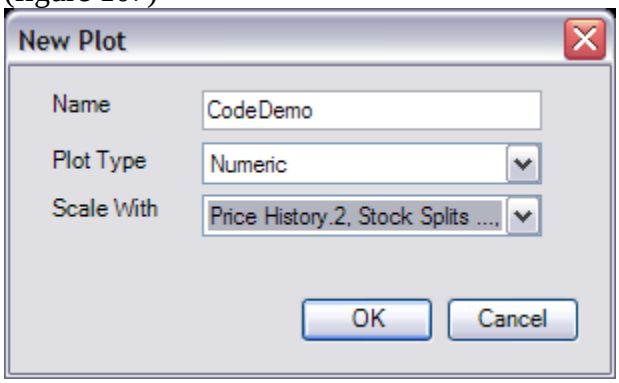


Figure 106 – New Plot, scale with Price History

A block diagram will open up to create your new plot.

Drag off the input connector for the plot and drop it in the white space of the block diagram. Choose *Select & Connect Block*. Type the name of your code block (HelloWorldDemo) to filter the list and select your block. Your diagram should look like figure 108.

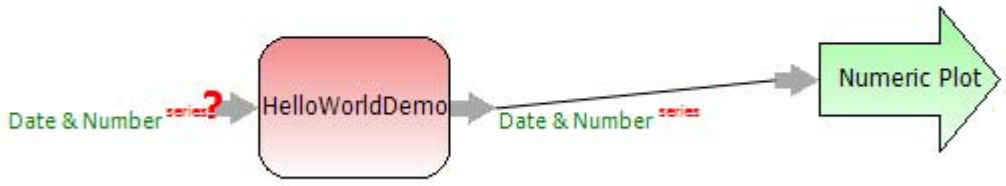


Figure 107 – Hello World code block in diagram.

Connect the input of your block to the Price History plot: Click and drag off the input connector and drop it in the white space. Choose *Link From Another Tool – BarChart.PriceHistory.2.* Your diagram is now complete; you can close the diagram window.

At this point you should have a line that is $5 higher than every point on the price history plot (figure 109).

Figure 108 – Bar Chart with CodeDemo Plot

Let's modify the source code to subtract 5 instead of add 5. Bring the same code block editor to the foreground (click on the Tool Window or choose *Windows – Code Block Editor*). Modify the `+5` to a `-5`. Click save and wait a few seconds for your new code to compile. Notice the plot will detect the change in the source code and recalculate automatically updating the plot on the chart.

There are many more code examples for you to view by opening the code block editor, choosing the *Open Block* button and browsing to the *Worden* folder.

Debugging code blocks with the debug log

Blocks comes with a Debug Log Tool (add via Start button). You can write to this debug log by using the `Log.Debug("My Message")` or `Log.Info("My debug info").` You can also use the Log.Exception("My Message",exception) to write a caught exception to the debug log. To avoid spamming the debug log in loops, use the mod operator. The following code example displays Count and the loop index every 100 times through the loop.

```
For i as integer = 0 to InputCount -1
    if i mod 100 = 0 then
         Log.Info("Count " & i)
    End if
Next
```

The output for the above code with an item of 501 elements would read:

```
Count 100
Count 200
Count 300
Count 400
Count 500
```

Debugging code blocks with Debug.Assert

You can use the built in `System.Diagnostics.Debug.Assert` calls to raise an Assert message when a condition is false. For more information on using `Assert`, see the MSDN article at http://msdn2.microsoft.com/en-us/library/e63efys0.aspx

Debugging code blocks with Visual Studio 2005

If you have Visual Studio 2005 with the JIT debugger installed (non express edition) then you can debug your code using the built in debugger. There are two ways you can step into your code. Option one is the built in breakpoint option. This will set a one time breakpoint (it will rest once the breakpoint is hit) that is called two lines before your Calculate routine.

To set the built in breakpoint, right click on your code block in a diagram and choose Set Breakpoint. The next time your block is called to

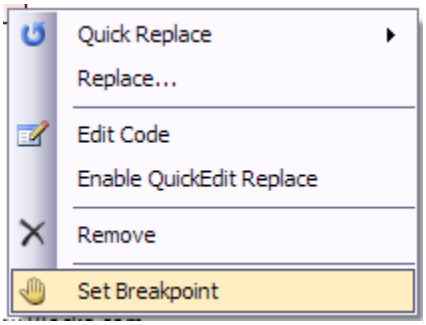


Figure 109 – Set breakpoint

update the calculation, a JIT debugger window will appear (figure 111)

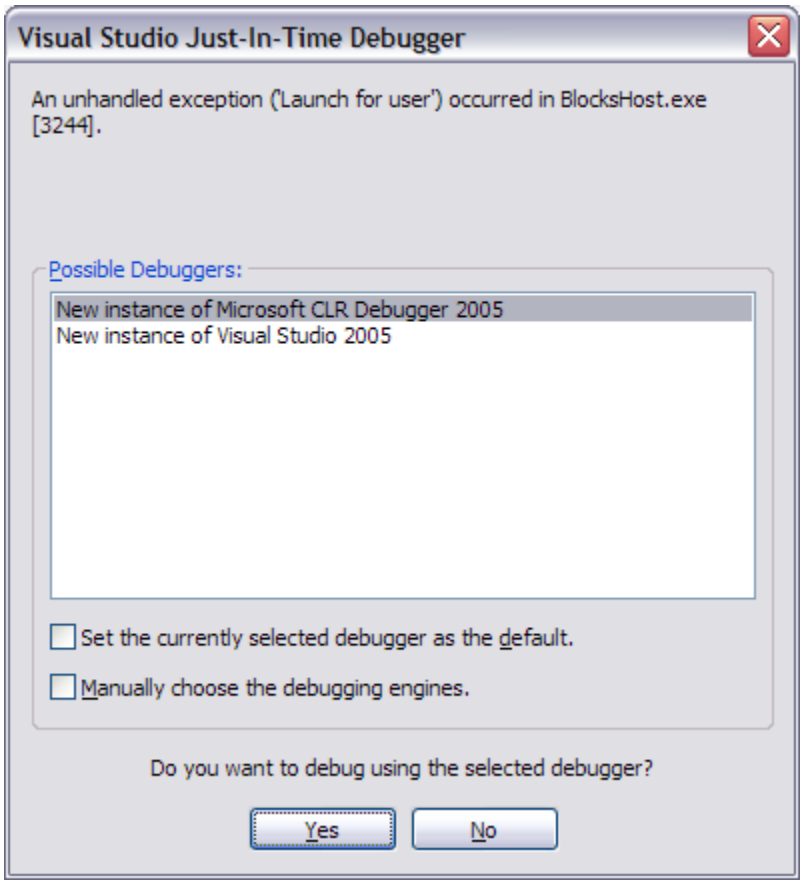


Figure 110 – Just-In-Time debugger.

Choose *New Instance of Visual Studio 2005* and click *Yes*.
Visual Studio will load and you will see a dialog box that says *No symbols are loaded…*(figure 112). Click Ok.

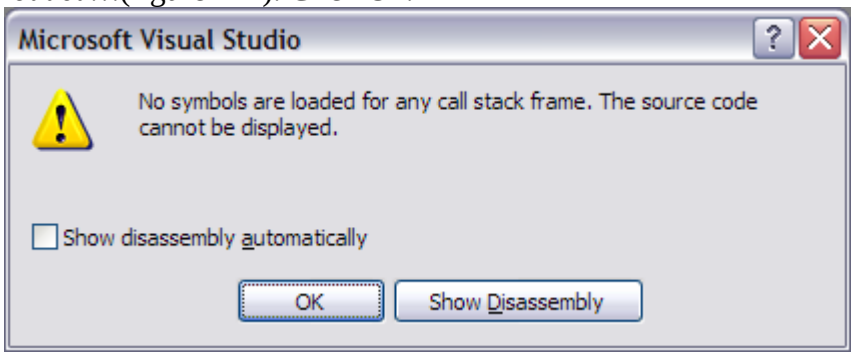


Figure 111 – No symbols are loaded

You now need to select *Step-Into* twice. You can use the step into hotkey or choose *Debug – Step Into* twice (if the No symbols loaded dialog appears again, simply click ok). You should now be on your calculate method (figure 113)

```
Imports WBI.Blox.Attributes
Imports LokiAppInterfaces
Imports WBI.CommonBlocks
Imports WBI.Loki.CommonBlockDataTypes
<WBIGuid("aadaa9a0-5ef3-4265-b2a0-503b65577370"),FriendlyName("HelloWorldDemo")> _
Public Class HelloWorldDemo
inherits BaseTemplateDLStoDLS
    Public Overrides Sub calculate()
        '************ Your Code Here ***************
        For i As Integer = 0 To Me.CodeBlock.InputCount - 1
            Me.AddToOutput(Me.CodeBlock.InputDate(i),Me.CodeBlock.InputValue(i) + 5
        Next i
    End Sub
End Class
```

Figure 112 – Source Code debugging.

Once the call stack is in your class, you can set regular break points and use any of the Visual Studio debugging features (Autos, Locals, Immediate window etc). When you are finished debugging, you can choose the *Debug - Continue* option (F5 hotkey).

NOTE: Any changes you make to your code file in the Visual Studio debugger are NOT updated in your blocks source code. You must edit your code in the built in editor and not in the JIT debugger. Any breakpoints you set in the JIT will no longer be set if you edit your code block as the class will be recompiled.

In addition to using the built in Set Breakpoint option, you can add the following line of code anywhere you want your class to break:

```
System.Diagnostics.Debugger.Break
```

NOTE: Do not put a break inside a loop as your code block will constantly load a user defined breakpoint. Instead, add a test condition (if then) and put your break inside the test condition.

Part 5 Additional Tools and Customization

Chapter 11: OneTouch™ Trading

OneTouch™ trading is an on-chart trade ticket that displays your current position(s) with the number of shares and the net gain, and any pending orders for the selected symbol you have on the table. New orders can be placed directly on the OneTouch™ Trading and existing orders can be modified.

OneTouch™ can currently be used with a TD AMERITRADE account or with an optionsXpress account. You can currently only trade equities through the on chart trading tool but both optionsXpress and TD AMERITRADE have additional Tool Windows you can add to a workspace to make different types of trades.

To access your brokerage account in Blocks, you must enable API access on your accounts. This is handled by the TD AMERITRADE or optionsXpress websites and you must log in to your accounts on those websites to enable them. Enabling API access is beyond the scope of this document.

Figure 113 – OneTouch™ Trading On chart order ticket.

You can open the on-chart trading tool by clicking on the Trade 

button on the chart toolbar. You will need to log into your brokerage account. *Brokers – Login* or use the Login buttons on the main toolbar or OneTouch™ Trading tool.

You can toggle the OneTouch™ tool by using the Pin icon in the upper right corner. There are 3 states it can be in: Fully expanded, Full Sliders and Mini Sliders.

Fully Expanded lists the full trade ticket and lets you modify all the parameters of the order.

Full sliders hides the full trade ticket but shows all the open orders and positions as well as ticket sliders for new orders.

The Mini sliders view (figure 115) is more conservative on screen real estate and only shows your open positions and pending orders.

Figure 114 – Mini sliders

Viewing Current Positions and Open Orders

There are many WatchLists available to you to track your account positions including: Current Positions and Pending orders. To view all the available WatchLists, simply click the WatchList Picker drop-down and type TD AMERITRADE or optionsXpress to filter by WatchLists from those brokerages.

Once you have logged in to your account, the on-chart trade tool will list any positions and any orders you currently have in the selected symbol directly on the right hand side of the value scale. If you are logged in to multiple brokerages the icon for the broker will appear next to each item.

Open orders are listed in a light cyan blue color. The screenshot on the right shows an open sell stop limit order for 25 shares that will trigger at 42.47.

Current Positions are listed in a light Tan color. The screenshot on the right shows we are Long 25 shares and have a current net gain of $258.50

You can control the look and feel of the OneTouch™ trade by right clicking on the tool and selecting properties. By default, it lists each individual position from each brokerage, but you can combine the positions for display purposes by setting the *Show Positions as Summary* property to true.

Placing a OneTouch™ Trade

To place a new entry order, simply click anywhere on the background of the on-chart ticket tool. A new Buy order slider will appear under the mouse. You can drag the order to any price to set the limit. You can change the order action and order type by clicking on the Buy drop down; from there you can pick buy, sell, sell short, buy to cover, limit, market, stop limit, stop market. To adjust the number of shares, you can open the Trade Order Entry (white strip to the right of the slider) and choose the number of shares and set more advanced options for the trade.

When you are ready to place your order, simply click the lightening bolt icon or the Place Order button. Once your order is received, a blue slider will appear on the OneTouch™ tool.

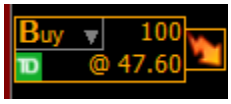


Figure 115 – On-Chart order ticket

Modifying an existing Limit Order

You can change the limit price of an existing order by simply clicking the blue slider to the new limit and clicking the replace  button. This will cancel your previous order and place a new one at the new limit price.

Canceling an Existing Order

To cancel an open order, simply click on the  next to the blue on-chart pending order.

Other Trading Tools

Each brokerage also has an accompanying tool that can be loaded into the workspace for viewing and managing your account. You can add the TD AMERITRADE or optionsXpress tools by clicking the *Start* button on the main toolbar and typing TD AMERITRADE or optionsXpress to filter the list. These tools can be used in conjunction with the OneTouch™ trading tool.

Chapter 12: Diagram Examples

In this chapter, we move beyond a reference manual and into a guide for practical applications of Block Diagrams.

Coloring with Boolean (T/F) Logic

Many of the diagrams in the Blocks software have color input blocks that allow you to use the code blocks to decide how to display foreground and background colors. The most common application of color logic is on plots to determine the plot color, WatchList columns to determine the fore or back color of a cell or on a data label.

Figure 117 is a diagram for a volume bars study on the Bar Chart that draws green bars on up days, and red bars on down days. The majority of the blocks in this diagram are performing the coloring logic. The bottom left part of the diagram simply takes the volume for the selected symbol and time frame and outputs it to the Numeric plot.

The coloring portion of the diagram is the *Prices*, *Net Change*, *Greater than value* and *T/F color* block. Following the logic from left to right, we get the *Prices* for *BarChart Main Symbol* and run it through the *Net Change* block, which will calculate the net change on every bar (day) for the provided symbol. Next, *Net Change* is run into a *Greater Than Value* block, which tests that the net change is greater than 0 (the 0 is coming from the blue parameter block).

On every bar that net change is greater than 0, it marks it true otherwise it's false. We now have a Date & T/F series coming out of *Greater Than Value*. We need to convert that into a Date & Color series for it to plot a color, so we use the *T/F Color* block. The *T/F Color* block simply assigns a color to a true value and a different color to a false value. The *T/F Color* block is then connected to the *Date & Color Series* input on the *Color* block and the bars now have a color for every date.

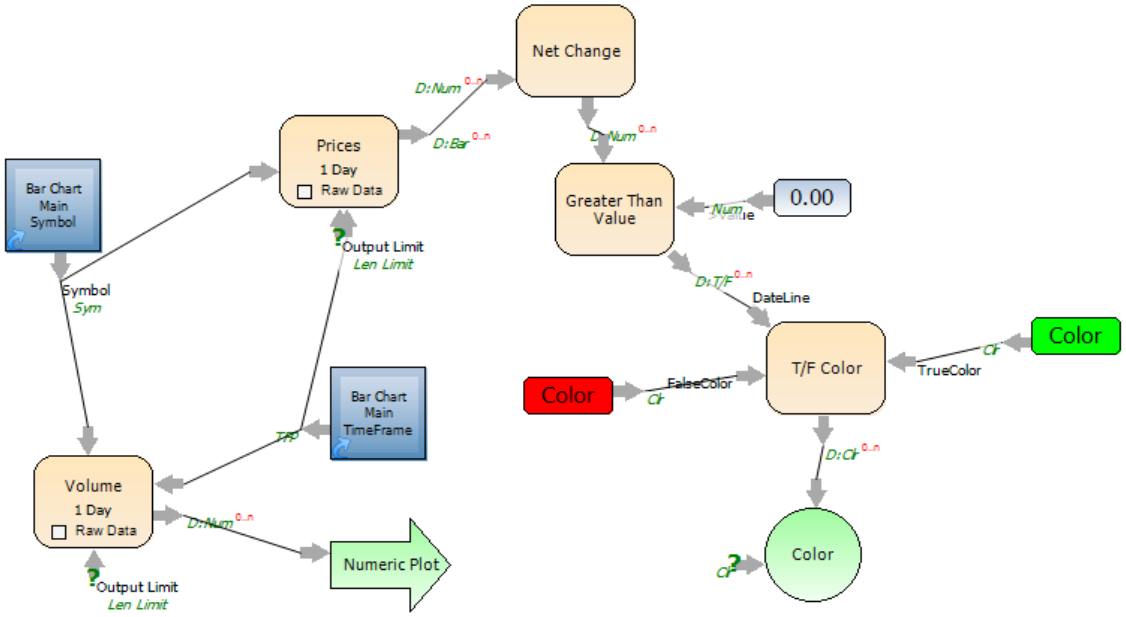


Figure 116 – Volume Bars conditional coloring

You could continue to add more conditional coloring to this diagram by using a *True Color Replace* block. *True Color Replace* takes a *Date & T/F series*, a *Date & Color Series* and a *Color*. It outputs a new *Date & Color series*. They way this works is the *Date & T/F series* defines the new condition you want to color. Any true value in this input will replace the color in the *Date & Color series* with the new *Color*, and any false value will leave the color the same.

You could chain as many of these *True Color Replace* blocks as needed to create a multi-conditional, multi-colored diagram.

Linking (borrowing) color from an existing plot

Each plot not only exposes the data it uses for drawing, but it also exposes the Date & Color series it uses to color the line. This allows you to link color logic across studies. To borrow the color logic from another plot, simply drag off the Date & Color series input on the color block and select *Link From another Tool*, or drag it directly to the plot on the chart and release the mouse. Figure 118 shows the price history plot with linked color logic from the volume bars. Figure 119 shows the diagram to produce Figure 118.

Figure 117 – Prices plot with linked color from Volume Bars.

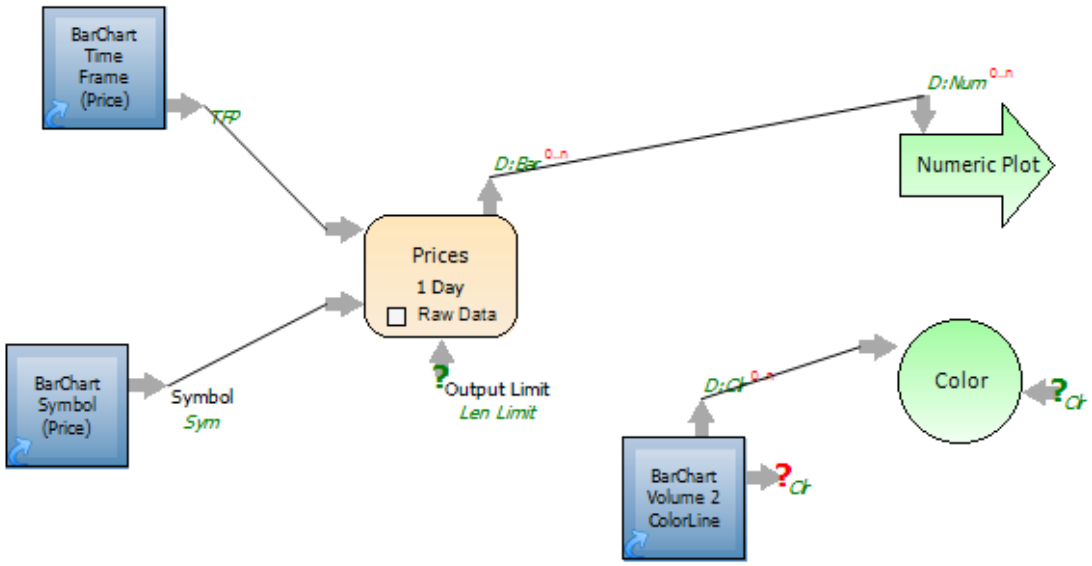


Figure 118 – Block Diagram for Figure 118 Price History Plot. Linked to diagram in 117

Chapter 13: Workspace Customization and System Options

System Options

You can configure your Blocks software environment from the *System - Options* menu.

Startup Options	• Show Open Workspace – Prompts for a Workspace to open (Default option) • Last Workspace – The last opened workspace • Specific Workspace – the workspace specified in the Startup Workspace field.
Startup Workspace	The workspace to load at startup if Startup Options set to Specific Workspace
Authentication Timeout	The number of seconds to wait when making a call to a remote server. Users with slow internet connections may need to increase this value
Always Save Workspace	If checked, will not prompt for the Save Changes to workspace, instead it will always attempt to save.
Auto Backup Workspaces	If checked will keep up to the most recent 20 versions of a workspace. Versions are tracked by file name not by the file path. Backup files are stored in My Documents\Block Files\Backup
Enable Code Block Debugging	Compiles user code blocks (source code) with debugging symbols enabled
Show New Patch Notes at Startup	Displays new patch notes
Force Check for new files	Performs a check for a program update at next start. All instances of Blocks and the Blocks Data Downloader must be closed for an upgrade to occur.

Other System Settings

In addition to the System – Options menu, you can set additional settings under the System menu:

System - Always On Top: Forces the main application frame (window) to always be on top of any other application. Very useful when combined with a semi transparent window (Opacity setting)

System – Opacity: Sets opacity (transparency) for the main application frame

System – Header Style: Changes the header style of the tool windows.

Multiple Monitors

Blocks supports multiple monitor systems. By default, any floating Tool Window can be placed on any monitor by simply dragging the window to the appropriate monitor (see page 63 for more information on working with Tool Windows)

If you wish to dock multiple Tool Windows in a second monitor, you can add a new Frame and drag it to the secondary monitor. For more information on additional Frames, see page 65.

If a Tool or Frame has been saved on a secondary monitor, and the monitor is no longer available when the workspace loads, the window will be repositioned on the main screen.

Chapter 14: Blocks Tips & Tricks

Table Tool

The Table tool is an invaluable tool for verifying data. The Table tool will display date and value series or Date and T/F series in a tabular format. You can load multiple columns of different data types and link them to indicators on charts or to any other diagram that provides Date & Number or Date & T/F series. The date of the table can also be synchronized by linking the date column to a date provider. The table tool can be vertically or horizontally oriented. Figure 119 shows a Table tool below the chart set in the horizontal orientation that is date synchronized with the end of the chart and shows the values at each date in the Price History.

Figure 119 – Chart tool with synchronized table tool below.

To configure the data for a column, right click on the Value header column and select properties, this will open the block diagram for the column. From here you can create a new diagram, or link to an existing. To view the values of a study on the chart, click and drag from the input connector and drop it on the plot you would like to view in the table.

To add additional columns to the table, right click on the table and select *Add*. You can choose from the following three types: *T/F Column, Date Column, Value Column.*

To customize the table display or to synchronize the date of the tool with another date, right click on the background of the Table Tool and select Properties. This will bring up a diagram to set the scroll date. You can link this to another tool (like the Chart Start date, End date or a Date Pointer). You can also set the properties for the Table Tool from here including background colors, cell colors, fonts and grid orientation

For an example of a configured Table, load the Time & Sales tool from the Start button.

Figure 120 – Table Tool with Custom Colors

Converting *Series* connectors to *Single* values

Many of the code blocks use a *Series* connector type. For example the prices block outputs a *Date & Number Series*. Some times you don't need the entire series, but instead need one specific value from the series. There are a few helper blocks you can use to extract a value from the series:

Last Value in Series – This block simply takes the last (latest date) in a *Date & Number Series.* When using this block, you can greatly enhance performance by limiting the amount of data that is calculated and fed into this block. For instance if you simply want the latest price of a stock, connect a parameter block to the Output Limit connector on the *Prices* block and set its value to 1. There is also a First Value in series if you wish to read the first value in a series.

Value for Date – This block takes a date and output's the value at that date in the series. The chart tool provides both the Start and End date for the chart and date pointer(s) on the chart will also provide a date. You can also use a date picker tool to choose a specific date. All of these date providers can be connected to the Date Input on *Value For Date*

Helpful Blocks

These blocks are great utility blocks that we find again and again in diagrams. Knowing about these can really simplify your diagram building.

Bar Offset – This offsets the data forward by the specified number of days so that the previous value is represented as the current value. This is a great block to use when you want to compare values from X number of bars ago. Use a value of 1 to get the value for

the previous bar. Using Bar Offset with a value set to 1 and a *Prices* block is the equivalent of the Worden PCF Formula: C1

Bar Open (High, Low, Close) - These sets of blocks take Date & Bar series data and extract the Open, High, Low or Close values to represent as a line. This is useful when you have a Prices block and wish to perform a calculation (say a moving average) of the highs. Figure 121 shows this example of *Bar High* in use.

Trim By Dates – This block is similar to *Value For Date*, but "trims" a line so that only the data between the two dates is output.

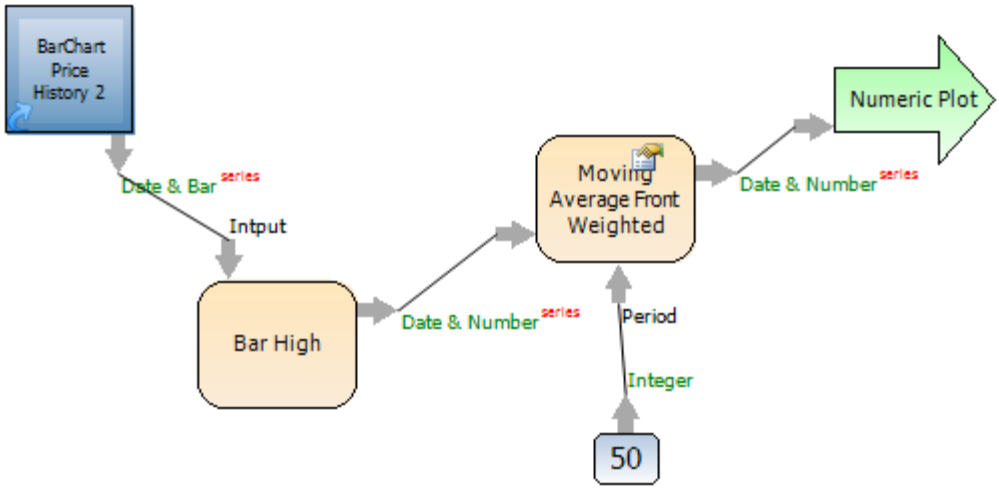


Figure 121 – Bar High block to perform a moving average of the highs

Time Frames and Bars

Traditionally in Technical analysis, time frame is though of as a specific data source you choose and everything is calculated off that data. Blocks takes a different approach to Time framing. A Time frame is simply a calculation of data based on a specified period. Blocks really only has three sources for stock data (tick, minute, daily) and every other time frame is calculated using a Time Frame Block to modify the data. The Prices block takes in a timeframe and will output the appropriate granularity for the supplied time frame. It does not output the selected timeframe; it simply picks the lowest possible data source for the provided time frame. You must run it through a Time frame bar builder block to convert it to the desired time frame.

Since converting data to a timeframe is a mathematical process, you can have non traditional timeframes (option expiration Friday, Presidential cycles). Since timeframe is a math function to Blocks, it treats stock data as Bars instead of daily data. If you are on a daily time frame you have daily bars, and a 2 minute time frame would be 2 minute bars. All of the traditional stock indicators calculate on bars, and are time frame independent.

This means you can build daily bars from minute data or build stochastics on minute data and then run the stochastics through a time frame block to convert it to weekly.

It also means you can build point and figure data (data based on reversals instead of time) and still apply technical indicators like moving average to them as the data is still represented as bars.

Building Blocks

The Blocks software is a modular software environment. You can get the most bang for your buck by building the smallest pieces possible and assembling them together for a larger calculation. It really embraces the principals of encapsulation and reusability that are some of the biggest benefits of Object oriented programming. If you are not familiar with the concepts of OO programming and you are writing your own code blocks you should try to familiarize yourself with the OO paradigm.

If you sit down to write a new code block, you might be able to accomplish the same calculation with some existing blocks. Browse the library or the knowledge base at http://kb.worden.com. If you do indeed need a new block to perform your calculation, keep it simple and performing one and only one job. This may end up causing you to write more than one code block to accomplish a calculation, but you will have more reusable pieces that you will not need to re-write in the future.

If you find yourself writing the same code over and over again in a code block, then you may need to abstract that code into its own block and connect it in-line to your other code blocks.

ClassAuthor Attribute

The `ClassAuthor` Attribute will let you define a few more details about your custom code block and provide a description for your code block. The attribute takes a Name, a Description and a Create/Edit date as string parameters. View a sample from the Worden Code folder for how to use the `ClassAuthor` Attribute.

Additional Resources

Worden Knowledge base: http://kb.worden.com
Worden Discussion Forums: http://www.worden.com/training
Blocks Website: http://www.Blocks.com
MSDN Website: http://msdn.microsoft.com