OpenQuant
Getting Started Guide

Version 3.2.2, 2011-07-31

Kevin Jameson

1 Introduction

OpenQuant is an IDE (integrated development environment) application for designing and executing computerized quantitative trading strategies on either historical or live market data. OpenQuant is a simpler, more modern, and less expensive version of the original professional QuantDeveloper IDE.

In particular, the OpenQuant IDE has a modern look and feel, a better window docking system, better drag and drop capabilities, a simpler programming API, and an easy way of switching between Simulation Trading, Paper Trading, and Live Trading modes.

Here is a list of other SmartQuant documents that might be of interest to OpenQuant users. These other documents discuss general trading and strategy development topics. You might want to read the documents in the following order to maximize your learning speed.

1. **System Architecture Manual**. This manual introduces SmartQuant system terminology, major system components, and system configurations for solving typical user goals.

2. **Getting Started Manual**. This OpenQuant manual (the one you are reading now) shows you how to use the OpenQuant IDE to configure and execute a simple strategy on historical data stored in the IDE database. You can choose a financial instrument, run a strategy, and inspect the simulation results.

3. **Strategy Development Manual**. This manual shows you how to design and code user-defined strategies on user-specified financial instruments. The manual begins with an overview of trading system design, then moves on to SmartQuant system concepts, and finally discusses the SmartQuant code for several strategies in depth.

1.1 Intended Audience

This document is an introductory Getting Started document for the OpenQuant IDE, written at a level that is suitable for novice strategy developers. Experienced strategy developers will have no problems at all understanding the information provided here.

1.2 Goals of This Document

The main goal of this document is to show you how to use the new OpenQuant IDE to create and run computerized trading strategies on either historical or live market data, using either simulated or live trade execution modes. In particular, this document discusses the purpose and use of all IDE windows so you can use the IDE productively.

This document does not talk about the theory and classification of computerized trading strategies. Nor does describe how to write profitable and efficient trading systems using trend following, gap trading, volatility trading, or arbitrage trading strategies. For that kind of information, read the SmartQuant Strategy Development Manual.

After reading this document, you should have a good understanding of how the OpenQuant IDE works, and should be able to write and test your own trading strategies right away.

2 Installing the License File

The license file for the OpenQuant application is named *OpenQuant.license*. Copy this file into the SmartQuant *Framework/bin/* directory, normally located here in a typical default installation on Windows7:

C:\Program Files\SmartQuant Ltd\OpenQuant\Framework\bin\OpenQuant.license

Once OpenQuant can find the license file, the splash screen will not show a "Demo version" warning.

3 Touring the OpenQuant IDE Interface

This section provides an overview of the main elements of the OpenQuant user interface. The interface is intended to support the activities of

- Financial instrument analysis with technical indicators such as moving averages
- Strategy development in C# using the OpenQuant programming API, and
- Automatic trading in Simulation, Paper Trading, and Live Trading modes of operation

First, here is a screenshot of the entire IDE, showing menus, dockable toolbars (deliberately spread apart for visual separation), dockable side and bottom windows, and a chart of AMZN shown in the center window pane.

The OpenQuant IDE is modeled after modern software development IDEs such as Visual Studio—the main difference being that OpenQuant is intended for computerized quantitative trading strategy development, whereas Visual Studio is intended for general software development.

3.1 Understanding IDE Menus

This section introduces the IDE menus, one by one. A screenshot of each menu is followed by a brief description of the menu functionality.

3.1.1 The File Menu
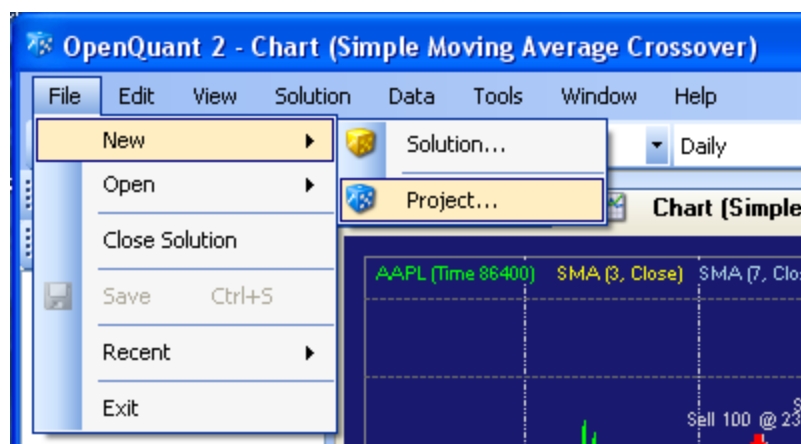
The File Menu provides the usual menu choices for creating new *solution containers* and *strategy projects*, opening existing solutions and strategies, and accessing recently used solutions and strategies. A *solution container* can contain multiple *strategy projects*. Each strategy within a solution can implement a different algorithm, using different instruments (tickers) and code blocks.

3.1.2 The Edit Menu
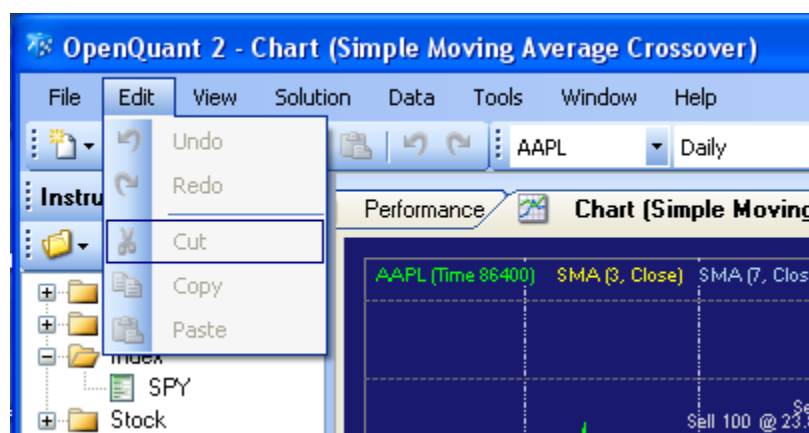
The Edit Menu provides the usual cut / copy / paste / undo / redo options for editing C# strategy source code files. (No choices are selectable in this screenshot because the main window is a candlestick chart of AAPL—you must have some text (code) selected before any of these choices will become active.)

3.1.3 The View Menu
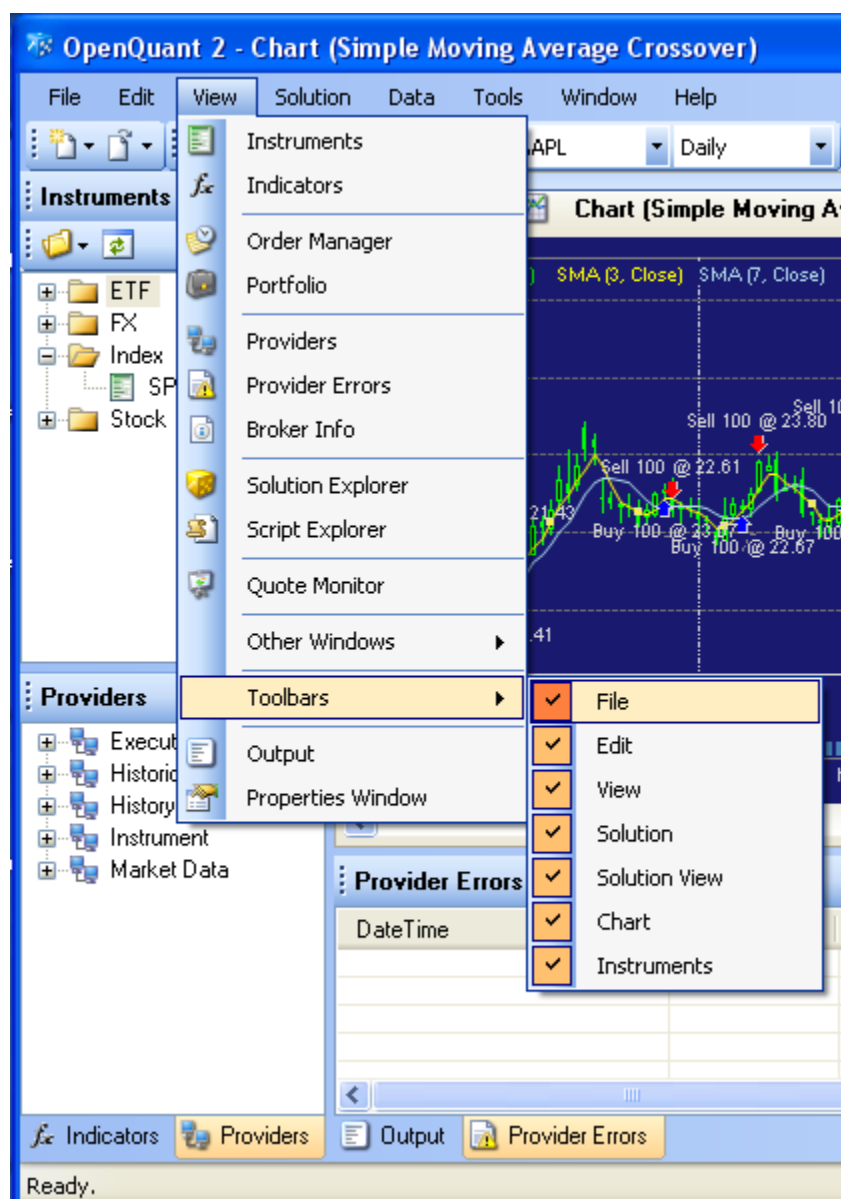
The View Menu provides choices for many different window panes and toolbars. Here is a brief description of the various window panes. Many of the window panes listed on the View menu are visible in the entire screenshot shown at the start of this section.

All of the docking panes on the left, bottom, and right side of the interface are modern "tear-off" windows panes. By grabbing the pane title bar with your mouse, you can tear off the panes and reposition them at other screen locations inside or outside of the main window boundaries.

The list below only gives a short introductory statement about each menu choice on the View menu. A more detailed picture and description of each window pane referenced by the View menu is shown later in this document.

- The Instruments window pane (top left) shows financial instruments that are defined for use by the IDE. The Instruments is visible underneath the menu in this screenshot. You must define your favorite instruments here before you can use them in a trading strategy.

- The Indicators window pane (bottom left) shows various technical indicators that you can drag and drop onto a chart of an instrument for analyzing instrument behavior. The Indicators tab is inactive in this screenshot.

- The Order Manager grid (center tab) helps you to view, enter, and cancel trading orders that are sent to the Execution provider that is responsible for executing trading orders.

- The Portfolio grid (center tab) shows all the trades for all the instruments that were executed as part of your strategy. A separate Portfolio is provided for each trading mode (Simulation, Paper Trading, or Live Mode). The Portfolio grid shows your open positions when you are in live trading mode, profit and loss for those positions, as well as Account values and Account transactions.

- The Providers window pane (bottom left) shows providers for trade Execution, Historical Market Data, Instrument definitions, and real time Market Data. The dockable Providers pane shares docking space with the Indicators pane in the bottom left of the screen.

- The Provider Errors window pane (bottom center) shows error messages that might arise from providers. Typical errors include connection failure messages, and data request failures. Check this window for information if you have trouble with providers.

- The Broker Info window pane (center tab) shows information about your current broker account, when you are paper trading or live trading with a real brokerage account. The center tab is empty if you are doing simulated trading.

- The Solution Explorer window pane (top right) shows key aspects of your trading strategy such as the size of bar data being used (Daily, 5-Minute, etc), the instruments being traded, and the C# source code and list of API software events that your strategy can utilize in its algorithms. To write some code for an event such as OnBarOpen, you double click the event name in this window to open up a code window in the center tab.

- The Script Explorer window pane (top right) shows a list of scripts (actually, C# code) that you might write to perform simple programmatic data management tasks or to test coding ideas that can use the OpenQuant framework. In other words, scripts are utility C# programs for doing non-strategy sorts of things. For example, you might write scripts to add instruments to the database, convert market trade data into bar data of various lengths, and so on.

- The Quote Monitor grid (center tab) displays incoming quotes for instruments of interest. Simply drag and drop instruments from the Instrument pane on to the Quote Monitor, and start the Quote Monitor to track instrument bid and ask prices in real time. If you are not connected to a market data provider when you drag an instrument on to the Quote Monitor, it will prompt you to establish a provider connection.

- The Other Windows menu choice currently has only the Start Page tab in the sub menu. The Start Page displays in the center tab, and shows a list of recent strategy solution links that you can click to load the referenced solution into the IDE for use.

- The Toolbars (all of which are tear-off and dockable) provide fast access to common functions. Current there are seven toolbars—File, Edit, View, Solution, Solution View, Chart, and Instruments. Each toolbar is shown and described later in this document. Several of the toolbars are displayed vertically along the right side of the main IDE window.

- The Output window pane (bottom center) shows printed output originating from the IDE and from print messages in your strategy. For example, compilation success messages appear here, as well as debugging messages from your strategy. Compilation errors are displayed in the Error List window, which shares docking space with the Output window.

- The Properties window pane (bottom right) shows the properties of whatever object is selected elsewhere in the IDE. Read-write properties can be modified here. Property values such as strategy parameters can be modified here between strategy runs.

For practice, here is another screenshot of the main screen. Now that you know more about the various tabs (eg Script Explorer) and where they are located (see the many vertical Toolbars on the right side of the main screen), see if you can identify the function of each window pane.

3.1.4 The Solution Menu
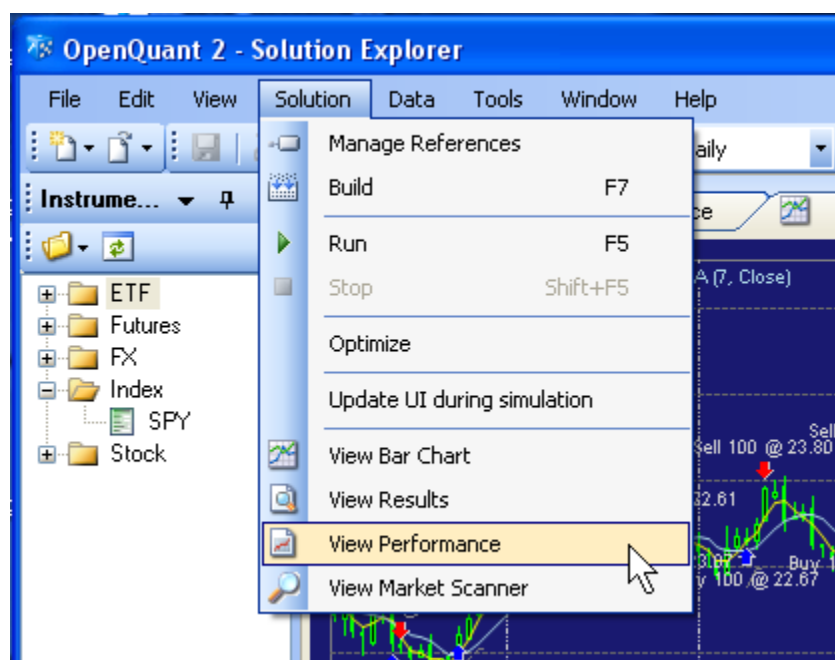
The Solution Menu provides functions for managing references to code libraries, and for building, running, pausing, stopping, and viewing the output of your strategy in various ways (chart, results, performance, and market scanner).

If you want to watch the bars and trades flash by on your screen as your strategy runs, you can tell the IDE to update the chart display during the simulation. Updating the IDE also means to update all other windows that show dynamic information—portfolio positions, account values, and transaction values. Updating the IDE slows down simulations significantly, so the default is not to update the IDE.

The Optimize choice displays the Optimization dialog so you can optimize various strategy parameters. To optimize, you must code optimization parameters (described later) and specify an optimization range and increment size for the parameters. Then OpenQuant loops and runs completely new simulations, one for each possible parameter set. A summary of results is displayed at the end of all runs.

The Solution menu also provides functions to view the following displays (Solution Views). Each of these views is also available on the Solution View toolbar on the right side of the main screen.

- The bar chart of instrument prices and indicators
- The results of your strategy (bar chart, trade points, and transactions)
- The performance of your strategy (statistics on average profit per trade, equity charts, etc)
- The code for your strategy (so you can examine and edit the code)

3.1.5 The Data Menu

The Data Menu provides many useful functions for importing or downloading various kinds of data for use in your trading strategies.

- Using the CSV or TAQ choices on this menu, you can import historical instrument data in the form of CSV (comma-separated values) or as text files from a TAQ (trade and quotes CD).

- Using the Download choice, you can download historical market data from remote service providers such as ESignal.

- Using the Instruments sub-menu, you can download lists of instruments from a provider so you don't have to type them all in manually.

- Using the Realtime choice, you can download market data from a market provider.

- Using the Historical Data (beta) choice, you can download historical data from websites such as Yahoo, or from data service providers such as ESignal or Genesis.

- Finally, you can compress fine-grained trade data series information into bars of larger size—by time (1 second to 6 hours), by tick (1 to 1000 ticks), by volume (10 to 100,000), or by price range (.001 to 1).

3.1.6 The Tools Menu
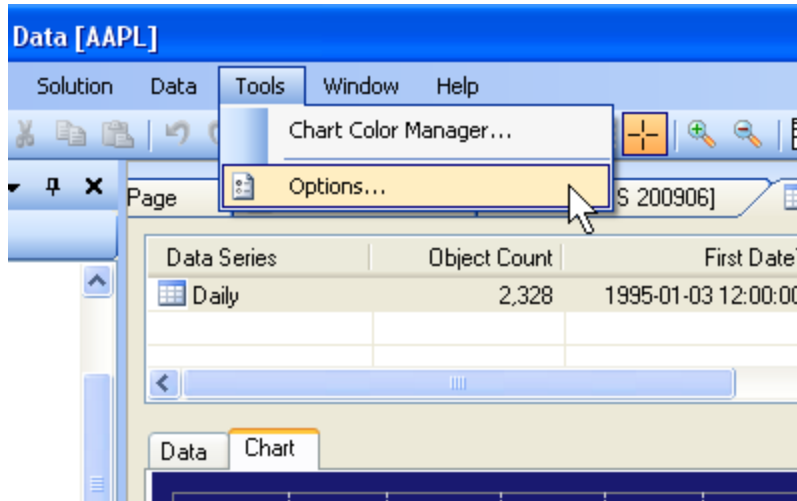
The Tools Menu provides functions for managing Chart Color Templates and for setting global application options.

To change the default Midnight Blue chart color, use the chart color manager dialog to create a new template with your own favorite colors, and set it as the default. If you upgrade the program to a later version, the upgraded program (at least as of Version 2.7.3) will not remember your choice of default template, so you will have to reset the default. The upgrade process does not destroy your templates, but it does forget which one that you defined as the default.

3.1.7 Using the Tools-Options Dialog to Solve Out of Memory Errors

The options dialog is mostly self explanatory, except for some General settings that can help to prevent "Out of Memory" errors. During a back testing simulation, OpenQuant stores all incoming bars, quotes, and trades in arrays in internal memory. The stored bar, quote, and trade values are required (1) for chart replay, and (2) for portfolio value calculations that involve the whole simulation. Chart replay is obviously useful when you want to scroll the bar chart back to the beginning of the simulation to see all bars in the whole simulation.

The portfolio statistics calculations are not always useful, but by default they are updated as each new trade arrives, and each separate value is stored in another internal array. These statistics can consume a lot of memory if you are processing a lot of trades, and are the source of many out of memory errors.

To avoid most out of memory errors, disable the calculation of portfolio statistics under the Backtesting section of the Options dialog. Not only will this free up lots of internal memory consumption, but it will also help your simulations to run noticeably faster because the statistics do not have to be calculated.

3.1.8 The Help Menu
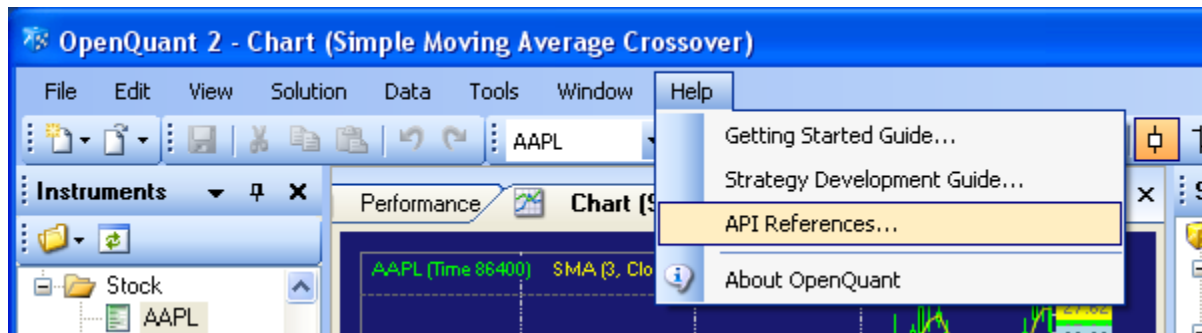
The Help Menu provides functions for accessing PDF documents (such as this one) and for searching and browsing API documentation in a searchable Help viewer window format.

3.2 Understanding IDE Toolbars

This section introductions the operations that are accessible through convenient toolbar buttons.

I have reorganized the toolbars in the screenshots below to separate them and put them all into two rows for documentation convenience. By default, they are organized in a different pattern that you can see in the pictures of the main screen, but of course you can click and drag the toolbars to arrange them to suit your own taste.

3.2.1 The File Toolbar
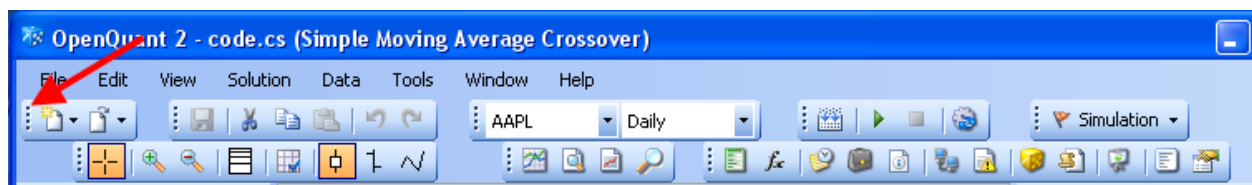
The File Toolbar provides only two buttons—one for creating New Solutions and Projects, and one for Opening existing solutions and projects.

3.2.2 The Edit Toolbar
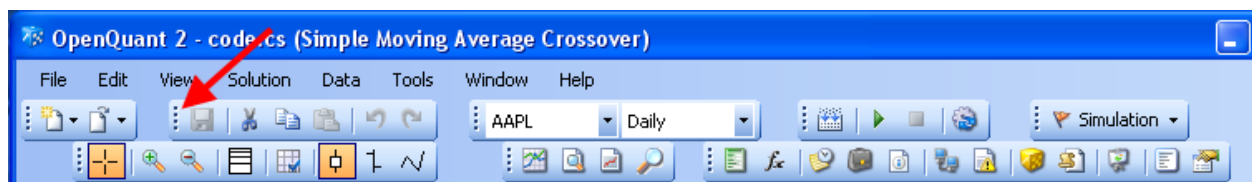
The Edit Toolbar provides buttons for Save, Cut, Copy, Past, Undo, and Redo operations.

3.2.3   The Instrument Toolbar
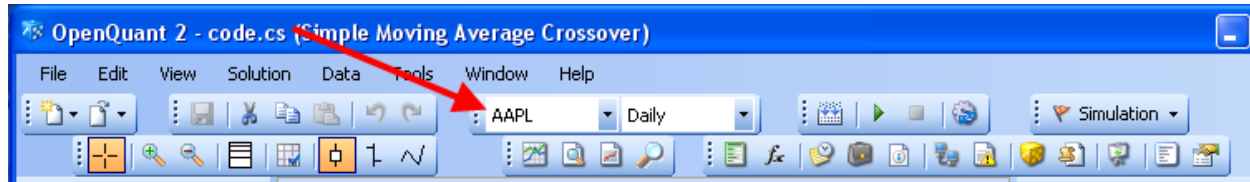
The Instrument Toolbar provides an easy way to choose an instrument and a bar size for viewing in the tabs of the center window. A single strategy can run against multiple instruments with multiple bar sizes, so you can use this toolbar to rapidly change instruments and bar sizes when you want to look at the results of a simulation for particular instruments and bar sizes.

3.2.4   The Solution Toolbar
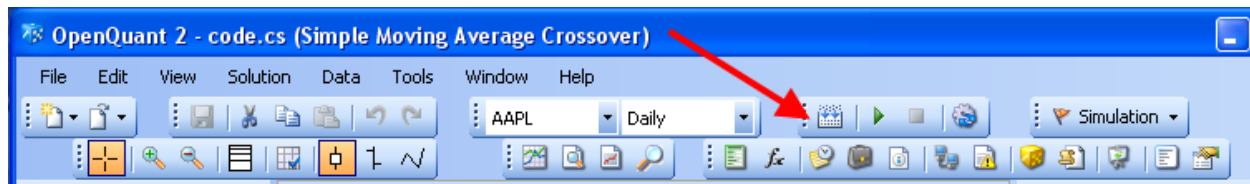
The Project Toolbar provides buttons for working on your strategy project:

- Build the strategy
- Execute (run) the strategy
- Pause execution of the strategy
- Stop execution of the strategy
- Update UI during simulation (shown highlighted)
- View strategy bar chart
- View strategy results
- View strategy performance
- View strategy code

3.2.5 The Trading Mode Toolbar
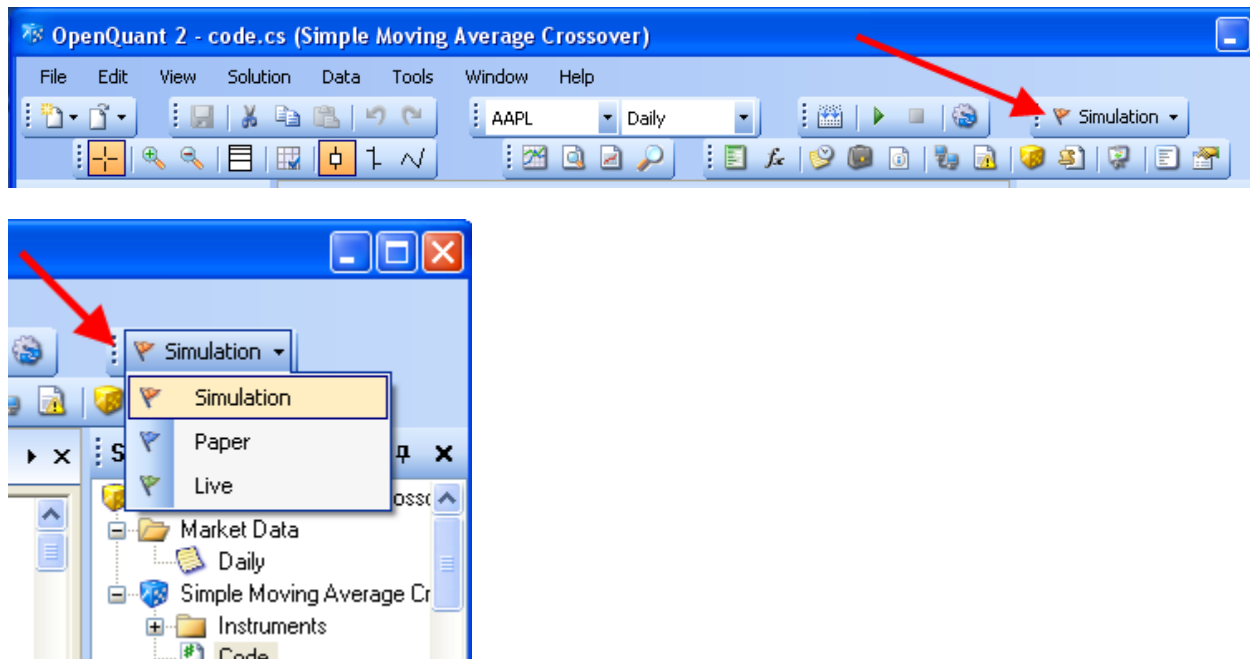
The Trading Mode toolbar provides three choices for selecting your trading mode:

- Simulation (historical market data, simulated trade execution)
- Paper Trading (live market data, live trade execution on a test account with fake money)
- Live Trading (live market data, live trade execution on a real account with real money)

3.2.6 The Chart Toolbar
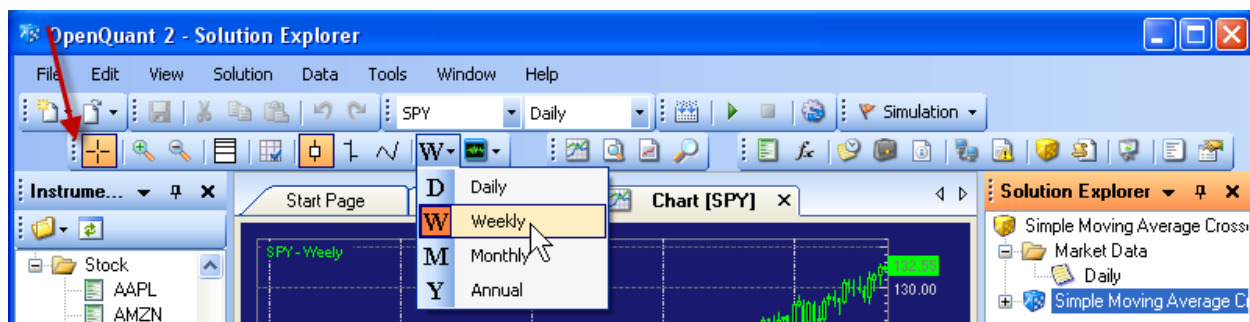
The Chart Toolbar provides buttons for controlling bar chart options:

- Show cursor as an arrow
- Show cursor as crosshairs
- Zoom in

- Zoom out
- Logarithmic chart scale for instrument prices (toggle on)
- Linear (Arithmetic) chart scale for instrument prices (toggle off, default)
- Trailing mode (default, chart scrolls old bars off left as new bars are added on the right)
- Fixed mode display (chart compresses instead of scrolling, and always shows all bars)
- Show instrument prices as candlesticks (toggle on)
- Show instrument prices as bars (toggle on)
- Show instrument prices as lines (toggle on)
- Select Daily, Weekly, Monthly, or Yearly (Annual) chart views
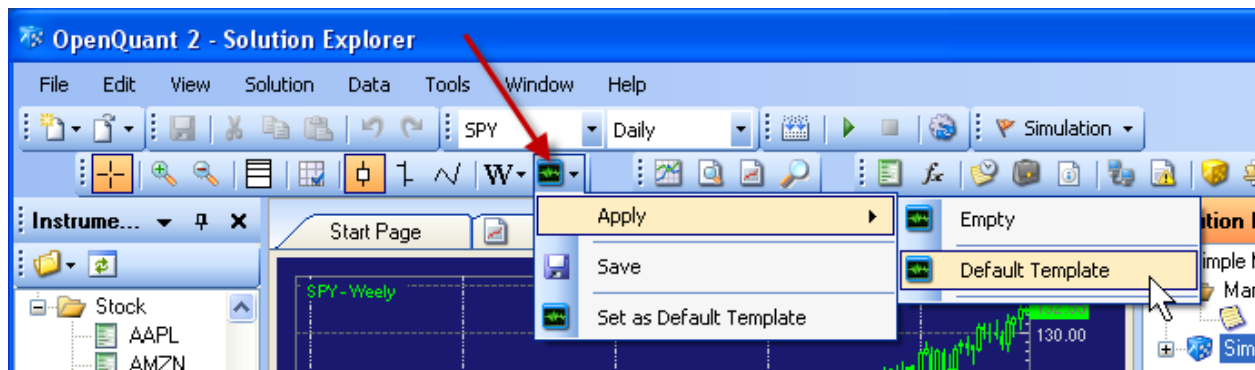
In addition, the chart toolbar provides menu choices for saving and applying your favorite chart templates. Configure a chart the way you like it and choose Save to save it. The next time you want to use it on a chart, choose Apply and select the chart template that you want to use.

3.2.7 The Solution View Toolbar
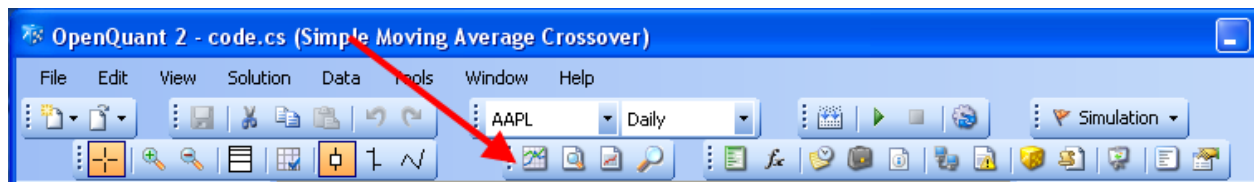
The Solution View Toolbar provides buttons to show various outputs of the current strategy:

- View Bar Chart
- View Results
- View Performance
- View Market Scanner

3.2.8 The Window Pane Toolbar
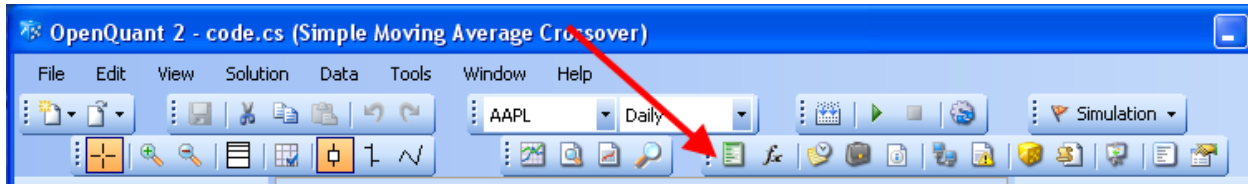
The Window Pane toolbar is usually located vertically on the right side of the IDE window. It provides buttons for displaying the various window panes and tabs mentioned above:

- Instrument pane (top left)
- Indicators pane (bottom left)
- Order Manager tab (center)
- Portfolio tab (center)
- Broker Info tab (center)
- Providers pane (bottom left)
- Provider Errors pane (bottom center)
- Solution Explorer pane (top right)
- Script Explorer pane (top right)
- Quote Monitor (center)
- Output pane (bottom center)
- Properties pane (bottom right)

3.3 Working with Dockable Window Panes

This section summarizes the functions of the various window panes in the OpenQuant IDE. All window panes are "tear-off and dockable," which means that you can tear off and relocate the window panes to suit your preferences in visual display structure. Any window that can be torn off can also be floated over top of the IDE, simply by not re-docking the window pane.

Here is a list of things you can do with dockable window panes:

- To auto-hide a dockable window, click the pin icon to unpin (sideways) the window. The window will slide into the edge of the IDE, and will leave a tab showing.

- To show a window temporarily, move your mouse over the tab to pop out the window.

- To show a window permanently, click the pin icon to pin (upright) the window.

- To tear off a window, click and drag the title bar. Pay attention—there is a difference between using the TITLE BAR of a window, and using the TAB of a window when there are multiple tabs in one pane.

- If you drag and drop a TITLE BAR, you will see a PARTIAL set of drop indicators that will let you select the WINDOW PANE to hold the dropped window. The dropped window will occupy the whole window pane, and will show up as a TAB inside the window pane.

- But if you drag and drop a TAB, then you will see a full set of drop indicators that will let you drop the tab in the upper / lower / left / right sections OF THE SAME WINDOW PANE. You will have to practice to see the different kinds of drop indicators, and their effects.

- To re-dock a window, tear it off and drop it (your mouse arrow) on one of the 8 re-docking indicators that appear. The four "outside" docking indicators located near the top, bottom, left and right sides of the main window will cause the tear off window to occupy the whole edge of the IDE.

- The four "inside" docking indicators will cause the tear off window to share the associated section of the center IDE window (e.g. dropping the tear off window on the top inside indicator will split the center pane with a horizontal line, and will show the tear off window in the top half of the center pane).

- In all cases, the new area to be occupied is indicated by a light blue shadow.

- To float a window, tear it off and drop it anywhere on a docking indicator.

- To return a floating window to its previous docked location, double click its title bar.

3.3.1 The Instrument Window Pane
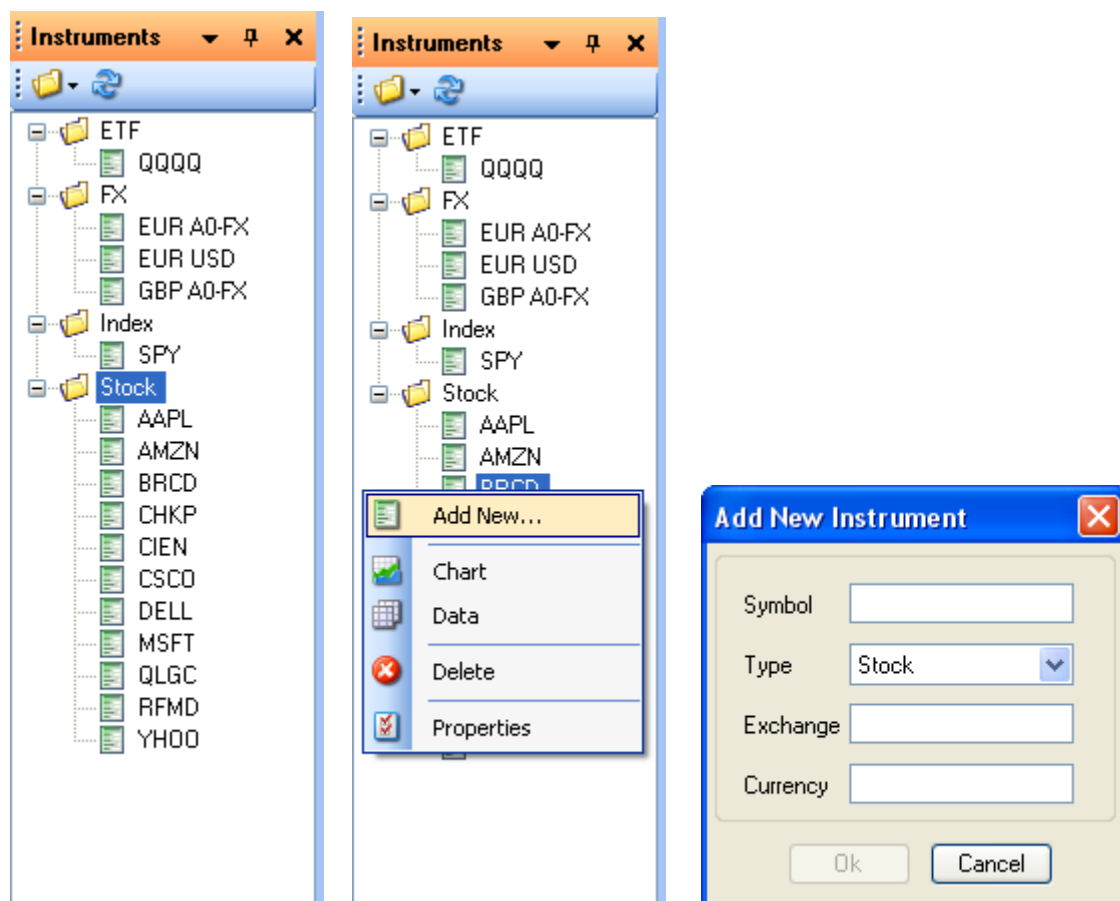
The Instrument Pane is where you define the financial instruments that interest you, either for technical analysis using charts and indicators, or for use in your trading strategy. Here is a list of things you can do with instruments in the Instruments pane:

- To define an instrument, right click in the Instrument window somewhere (or on the type of instrument that you want to create, e.g. "Stock" above), and define the name and type of your instrument in the dialog that pops up (shown above). The dialog will provide appropriate data fields for you to complete. The instrument type is used to organize instruments of the same type into folders, so that all common stock instruments appear in the stock folder.

- Once you have defined the name of the instrument, you must also define properties of the instrument in the Properties window. To define properties, select an instrument in the Instrument pane, and then look in the Properties pane. You must define—at a minimum—the instrument Currency and the Exchange (where the instrument is traded).

- To view a chart of the instrument, select Chart from the right-click context menu.

- To view the trade or bar data of the instrument, select Data from the context menu.

21

- To view properties of the instrument, select the instrument (left click) or right-click and select Properties from the context menu. Either way, you must set properties in the Properties pane—you don't get a pop-up dialog by using the context menu.

- To add an instrument to the quote monitor, drag and drop the instrument name from the Instrument pane into the Quote Monitor window. Or you can drag and drop the whole folder of instruments. (There is no other way of performing this operation with menus—drag and drop is the only way.)

- To delete an instrument from the Instrument pane or the Quote Monitor, right click the instrument name and choose Delete.

3.3.2 The Providers Window Pane
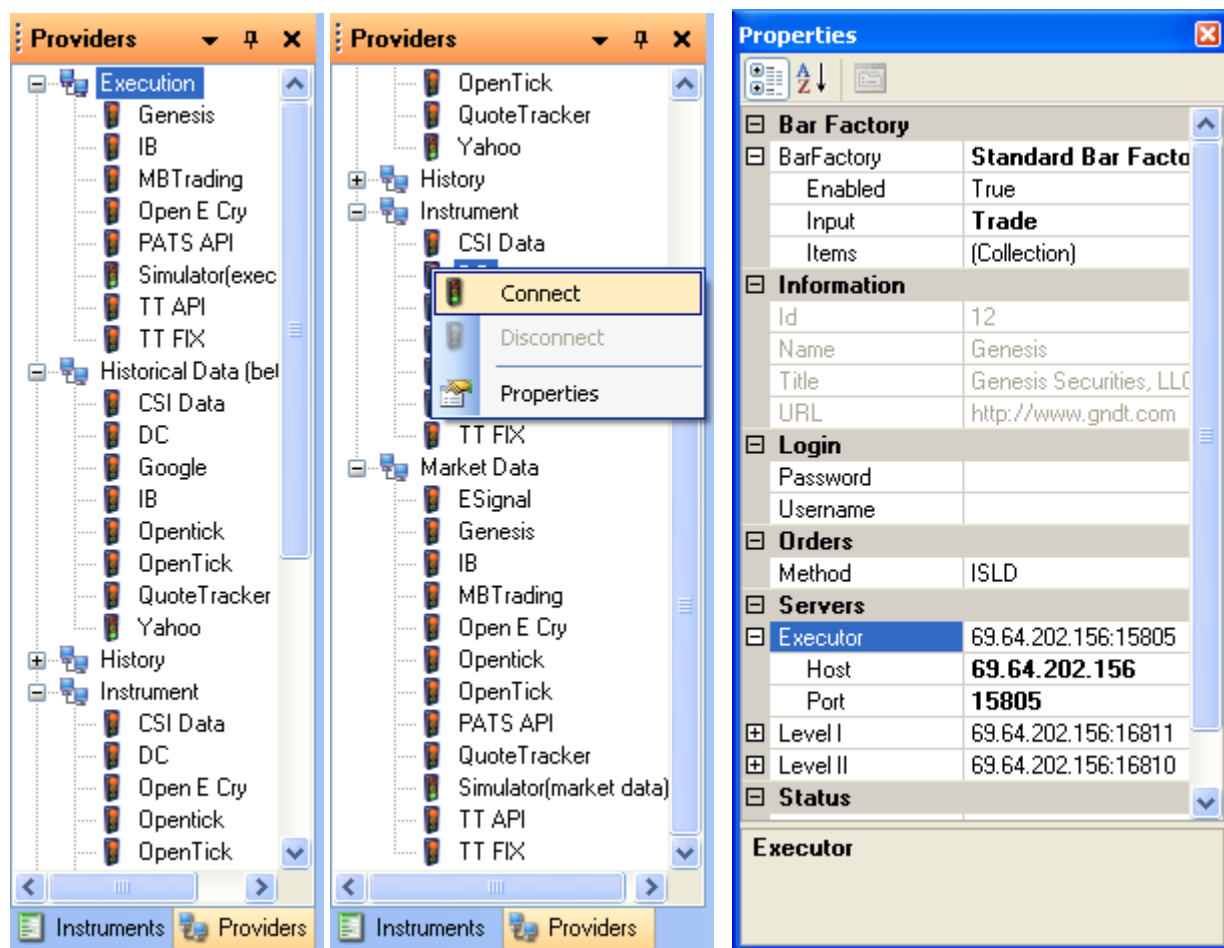
The Providers pane lists the available providers for trade Execution, Historical market data, Instrument definitions, and real time Market Data. The DC (Data Center) name refers to a separate SmartQuant product. All other names refer to non-SmartQuant vendors.

- To set Provider properties, either select the provider name (left click) or select Properties on the context menu (right click). Either way, you must set up connection properties such as login names and IP addresses in the Properties window pane.

- To avoid connection errors, you should probably set up connection properties before you try to connect (except for the Yahoo provider, which has its properties predefined). You can use Yahoo as a test case for downloading free historical market data, since Yahoo is predefined in the IDE, and does not require any more information from you.

- To connect to a provider, right click its name and select Connect. Look for error messages in the Provider Errors window pane (bottom center) if the connection does not work. If the connection works, the Red provider icon will turn Green.

3.3.3 The Indicators Window Pane
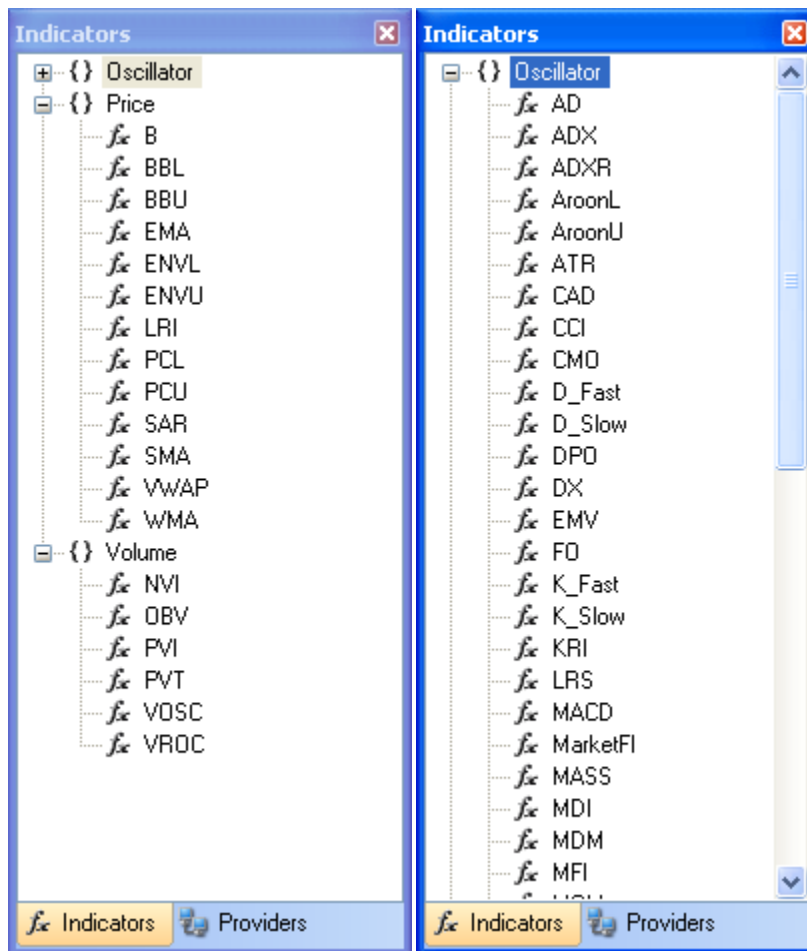
Here are two screenshots of indicators available in OpenQuant2. There are three categories of indicators—oscillators, price indicators, and volume indicators. Each indicator is explained in the online reference documentation.

The indicators pane is quite limited—it has no tooltips, no right-click context menu, and no visible responses to mouse clicks with your mouse. But the pane is still useful. Here is what you can do with indicators in the Indicators pane:

- To read up on the definition and function of a technical indicator, read the API documentation under the Help menu. Select Help, API References, and then (in the left side of the help viewer Contents pane) select the OpenQuant.API.Indicators help tree. This will show the documentation for all the available indicators.

- To apply a technical indicator to a chart, right-click an instrument name, select Chart to show it, then drag and drop a technical indicator onto the chart. This will bring up a dialog box to help you configure the indicator. The screenshot above shows the dialog box for an Exponential Moving Average indicator. This dialog lets you choose a name, a data instrument and series to apply the indicator to, the color and width of the indicator line, and the period of the moving average.
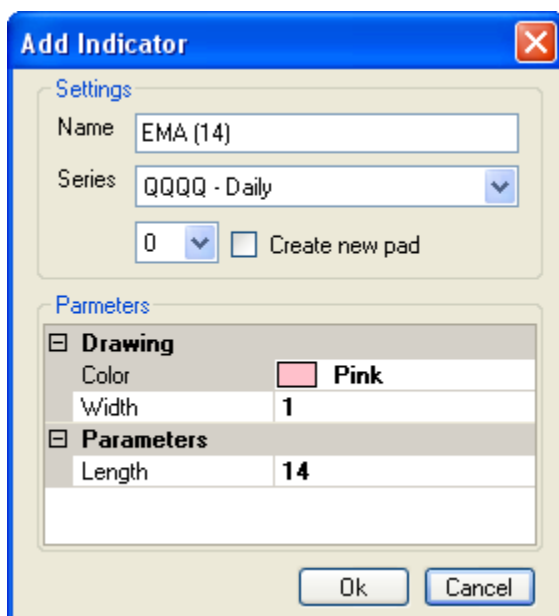
- To modify the properties of an indicator after you have applied it to a chart, left-click the indicator line in the chart—this requires precision mouse movements sometimes—and then modify the indicator properties in the Properties pane. Or right-click the indicator line on the chart, and select Properties from the context menu. Beware—if you use the right-click method, you don't get to use the dialog box to modify properties—once the indicator is on the chart, the only way is to use the Properties window pane.

- To display the indicator on a separate section of the bar chart (not on top of the price bars), either select the desired "pad" (chart section) number in the spin box, or check the checkbox to create a new chart section (pad). The new chart pad will be added to the bottom of the chart.

3.3.4   The Output Window Pane
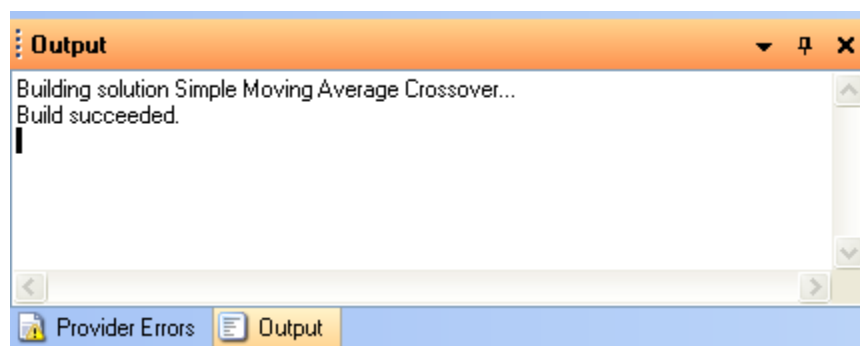
The output window pane is straightforward—it displays text output information generated by the IDE, or debugging text generated by your strategy. The example above shows the output that occurs when you press the "Build" button on the Project Toolbar when the "Stock 2% Gap" strategy is visible in the Solution Explorer window pane.

3.3.5   The Provider Errors Window Pane
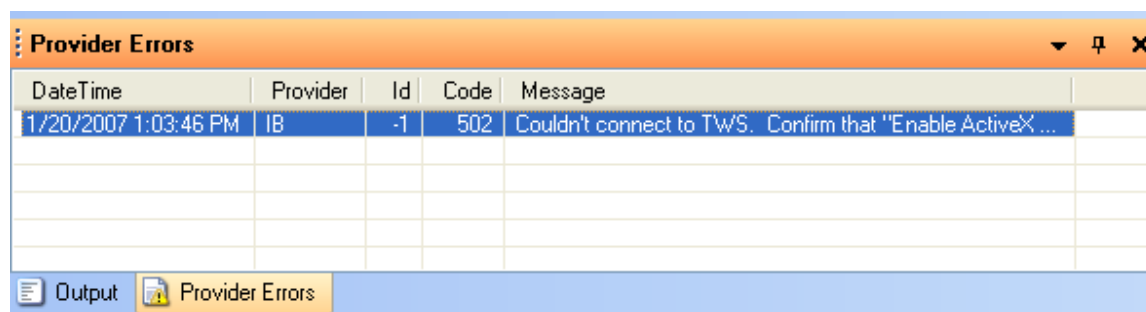
The Provider Errors window pane displays a grid of error messages (if any) from various providers. In the example shown above, the error message says that the IDE could not connect to the IB (Interactive Brokers) TWS (Traders Workstation Software). The TWS software requires that you enable a particular TWS option ("Enable ActiveX…") before TWS will accept incoming connections such as the one from the IDE.

3.3.6 The Properties Window Pane

Properties	
⊟ Account	
Account	
DefaultAllocati	Account=
⊟ Bar Factory	
⊞ BarFactory	**Standard Bar Fa**
⊟ Debug	
ConsoleOutpul	False
LogFileOutput	False
OverwriteLogF	False
⊟ Information	
Id	4
Name	IB
Title	Interactive Brokers
URL	http://www.interac
⊟ Settings - Execution	
Transmit	True
TreatSellShort	True
⊟ Settings - Historical Data	
BarSizes	1sec,5sec,15sec,3
DataRange	DaysAgo
DataType	Bar, Daily
MaxConcurren	-1

Name

Properties	
⊟ Appearance	
Currency	
Description	
Exchange	
Symbol	QQQQ
Type	ETF
⊟ Appearance (alternative)	
AltExchange	
AltSource	
AltSymbol	
⊟ Derivative	
Factor	0
Maturity	
PutCall	**Put**
Strike	0
⊟ Display	
PriceFormat	**F2**
⊟ Industry	
Group	
Sector	
⊟ Margin	
Margin	0

AltExchange
Instrument alternative exchange

| Properties	
⊟ Exit	
CrossoverExit	**True**
OCAExitEnab	**True**
StopExitEnab	**True**
⊟ OCA	
LimitOCALeve	**1.05**
StopOCALeve	**0.98**
⊟ Parameters	
Qty	**100**
⊟ Reporting	
ReportEnable	**False**
TestingPeriod	**day1**
⊟ SMA	
Color1	☐ 255, 255, 0
Color2	☐ 173, 216, 23
Length1	**3**
Length2	**7**
⊟ Stop	
StopLevel	**0.05**
StopMode	**Percent**
StopType	**Trailing**

Color1
Color of SMA1

The Properties Pane allows you to view and modify properties of elements that are selected (highlighted) in other IDE windows. Three example screenshots are shown above.

- The first example shows properties of the IB Interactive Brokers execution and market data provider. If you look closely, you can see the provider name, title, computer host name, TCP/IP port number, and connection status in the properties grid.

- The second example shows properties of the QQQQ (Nasdaq 100) ETF instrument. The currency is USD, but the Exchange has not yet been specified. Recall that the Exchange field must be specified for Live Trading mode, so the IDE knows where to send orders.

- The third example shows properties for a strategy (in this case, the "Simple Moving Average Crossover" strategy that ships with the IDE). Parameters for the strategy behavior include one for the type of trade exit to use (exit on crossover, exit on OCA (one cancels all), exit on stop loss) and another property for the Qty (number) of shares to trade ("100"). Each kind of strategy that you write will have a different set of properties.

- When you write your own strategies, you can code Parameters into the C# code that will appear in this same Properties window, so that you can modify and tune them between runs of the strategy, without recompiling the strategy code each time.

### 3.3.7 The Solution Explorer Window Pane
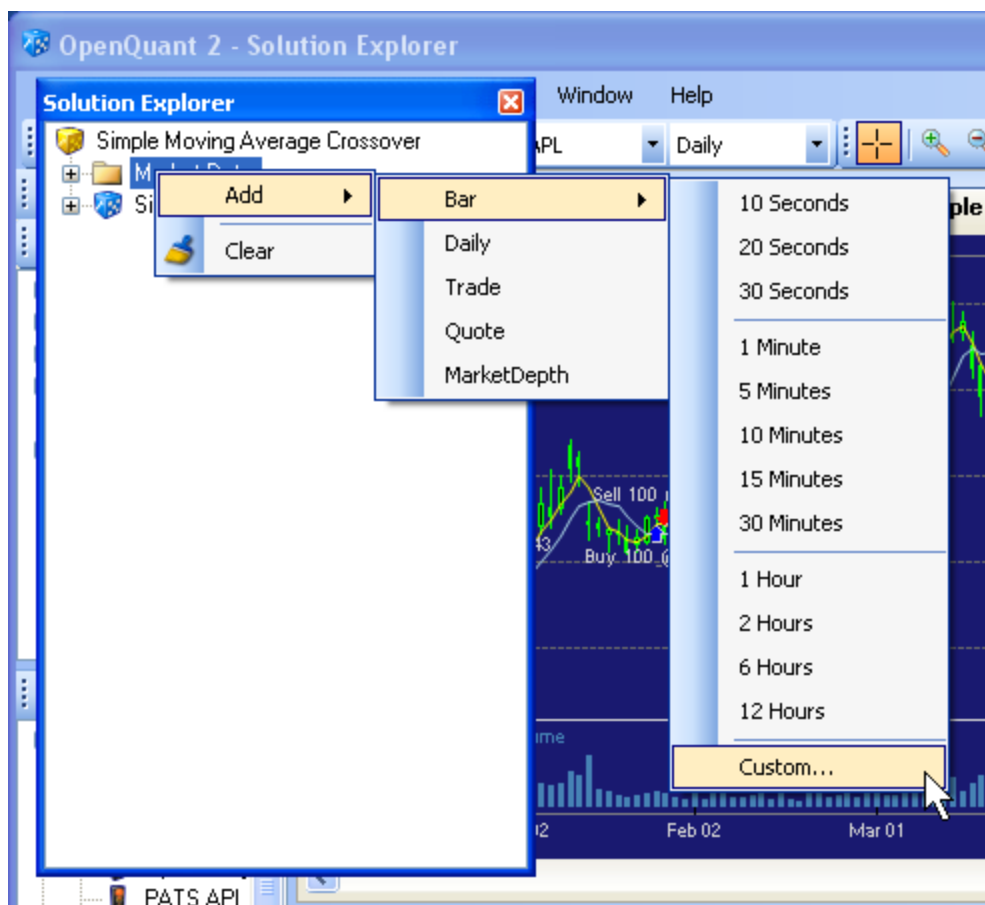
The Solution Explorer pane shows the structural elements of your strategy, such as the instruments that the strategy trades, the bar size(s) that it uses, and the C# code and events that control its behavior. (The screenshot above shows a floating explorer window to put the cascading menus over top of the chart.) Here are some of the things you can do with the Solution Explorer pane:

- Right click the strategy name for choices to Build, Run, and view strategy results (chart, results, performance) and code.

- Right click the Instruments folder to add or remove instruments.

- Right click an instrument to set its properties, or to remove it from the strategy. Removing an instrument from the strategy does not remove the instrument from the Instruments pane.

- Right click the Market Data folder to add input data types (Bar, Trade, Quote) or input bar sizes (1, 5, 15 minutes, or custom bar sizes).

The normal way to add bar sizes in simulation, paper trading, or live modes in OpenQuant is to use the Solution Explorer pane, right click the Market Data folder, Add a new data series (shown in the screenshots above), and choose a bar size from the pop-out menus (these pop-out menus are shown in a screenshot a following section).

The screenshot below shows the dialog that appears when you choose *Custom…* from the bar size menu shown in the previous picture. This dialog lets you specify custom bar sizes for your strategy. Bars can be manufactured from Trade data if you check the checkbox. Otherwise, bars will be read from Bar data series directly (no manufacturing is required, because the IDE is reading bars, not trades.)
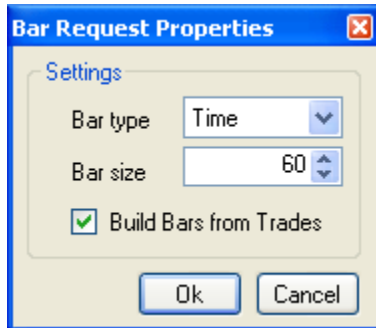

- Right click the C# code icon to edit the strategy code.

- Double click, or right click, an Event name to jump to that location in the code. Like Visual Studio, the OpenQuant IDE will generate template code for the event handler if it does not already exist.

3.4   Working with the Center IDE Window

The center window pane of the IDE contains a tab control for displaying bar charts, portfolio grids, and code file editors. Each type of information is displayed on its own tab. Tabs can be torn off and docked in various ways to create side-by-side chart layouts, above-and-below layouts, or other layouts that suite your preferences. Lots of experimenting is required to understand the docking indicators that show up when you tear off a window pane. But the good news is that creating side by side displays in the center window only is a little easier.
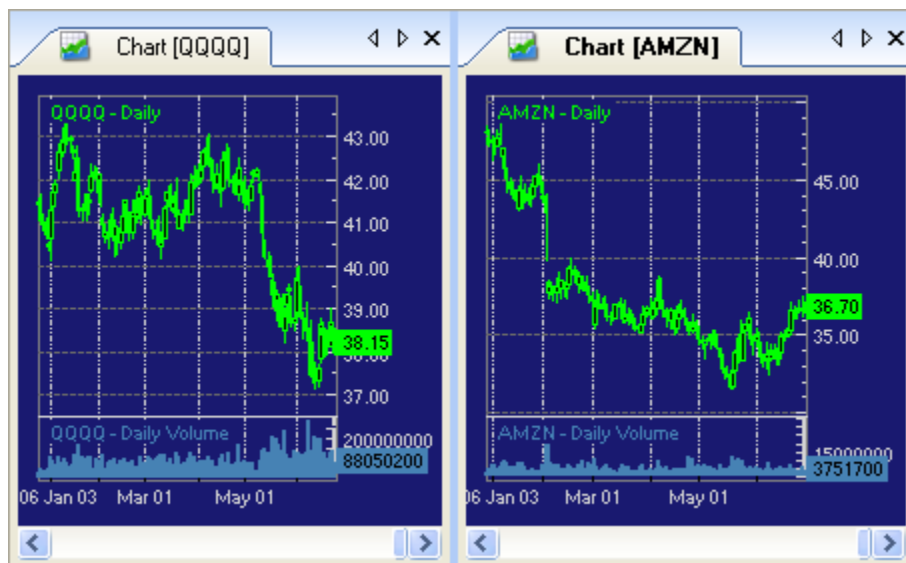
Each tab can be torn off and floated over the IDE. Tabs can also be docked on the sides and bottom of the IDE, with the Instruments pane, the Output pane, or the Solution Explorer pane. (But the Instrument pane and its friends are not allowed to dock in the center window.) Double click the title bar of a floating tab to dock it back into its previous docking location.

3.4.1   The Start Page Tab

29

The Start Page Tab lists the names of recent projects, their most recent modification date, and their file system path location. Click on a link to open that particular project.

- To hide the Start Page Tab, click the X at the top right corner (this applies to all Tabs).

- To make it visible again, use the View / Other Windows / Start Page menu.

3.4.2 The Quote Monitor Tab
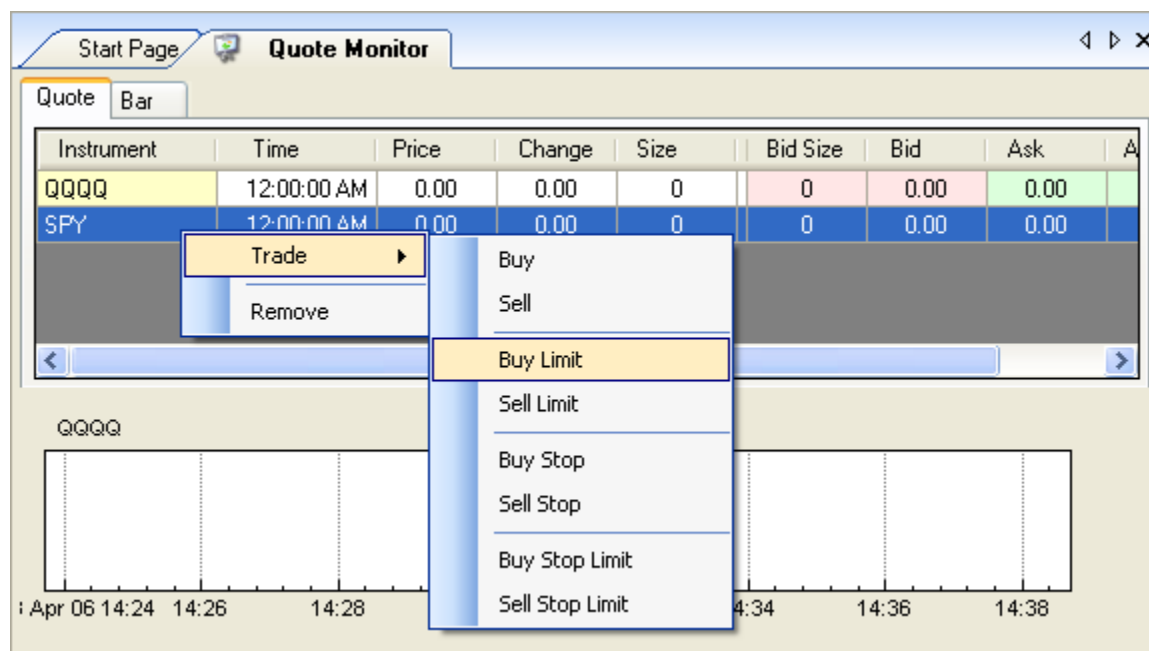
The Quote Monitor Tab monitors incoming bid/ask and trade prices for instruments that you drag and drop onto the Quote Monitor Tab. It helps you to see if your connection to a market data provider is actually working. If it is, the quote prices will change; if the connection is not working, then no price movement will be visible (in which case you should check the Provider Errors window for error messages).

- To add an instrument to the Quote Monitor Tab, drag and drop an instrument symbol from the Instruments pane on to the Quote Monitor Tab.

- To remove an instrument from the Quote Monitor list, right click it and select Remove.

- To make a trade from the Quote Monitor, right click the instrument line and select the type of trade that you want to make. A pop-up dialog will appear to collect information for the trade order. Once you have entered the order, you can see the results of your trade in the Order Manager window (transaction status) and the Portfolio Manager window.

3.4.3 The Instrument Chart Tab
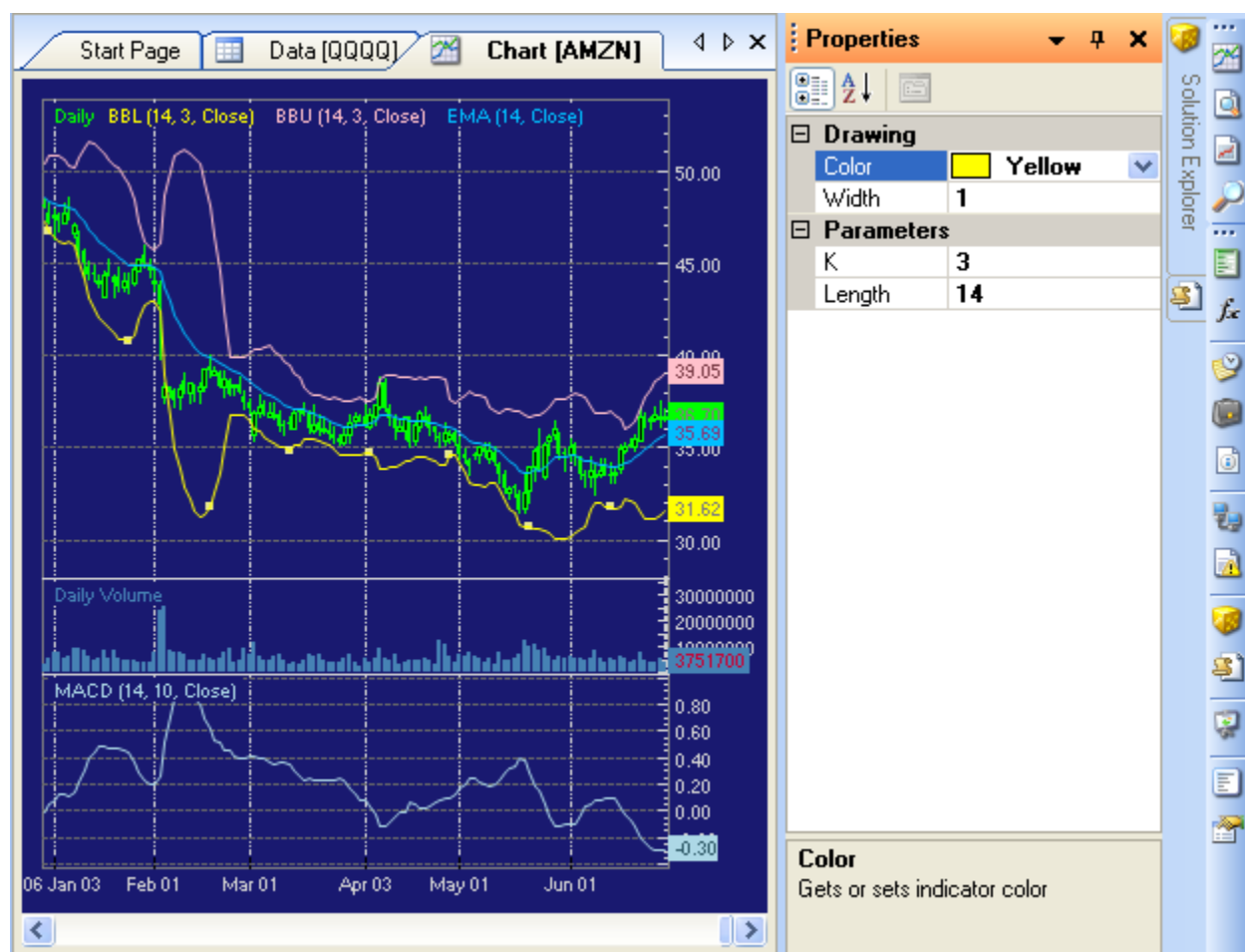
To show a chart of the bars for an instrument, right click an instrument in the Instrument pane, and choose the Chart option from the popup dialog. This shows a data series for the instrument only—it does not show a chart of strategy buy/sell signals applied to the instrument. (To see buy/sell strategy signals, right click on a strategy in the solution explorer window, and choose View Chart, or use the view chart toolbar button on the right side toolbar).

The Chart Tab shows a chart of instrument prices and technical indicators. The example above shows a chart of AMZN price and volume, plus several technical indicators. The Chart Tab also shows indicators that are coded into your strategy with Draw(..) statements, as well as trade symbols as trades are made during Simulation, Paper Trading, and Live Trading executions. So you can watch your strategy display indicators and make trades in real time, all on this chart.

- The first indicator BBL (Bollinger Band Low, in yellow) is a lower Bollinger band indicator, using a 14-day period with a K value of 3, and based on the closing price value of the bars in the chart. To change the properties of this indicator, left click on the line, and the associated properties will show up in the properties panel (see the Properties pane above).

- The second indicator listed at the chart top is an upper Bollinger Band (BBU) indicator, in pink.

31

- The third technical indicator (this time in its own chart section or "pad") is a MACD (Moving Average Convergence Divergence) with periods of 26 days (slow MA) and 12 days (fast MA), based on closing bar prices.

- To remove an indicator from the chart, click the line until you see some "handles" show up, as shown on the yellow BBL line above. Then right click the line, and choose the Delete option.

As you have seen previously, it is possible to create multiple chart tabs, and float them or rearrange them into side-by-side or above-below displays to suit your preferences.

3.4.4 The Instrument Data Series Tab
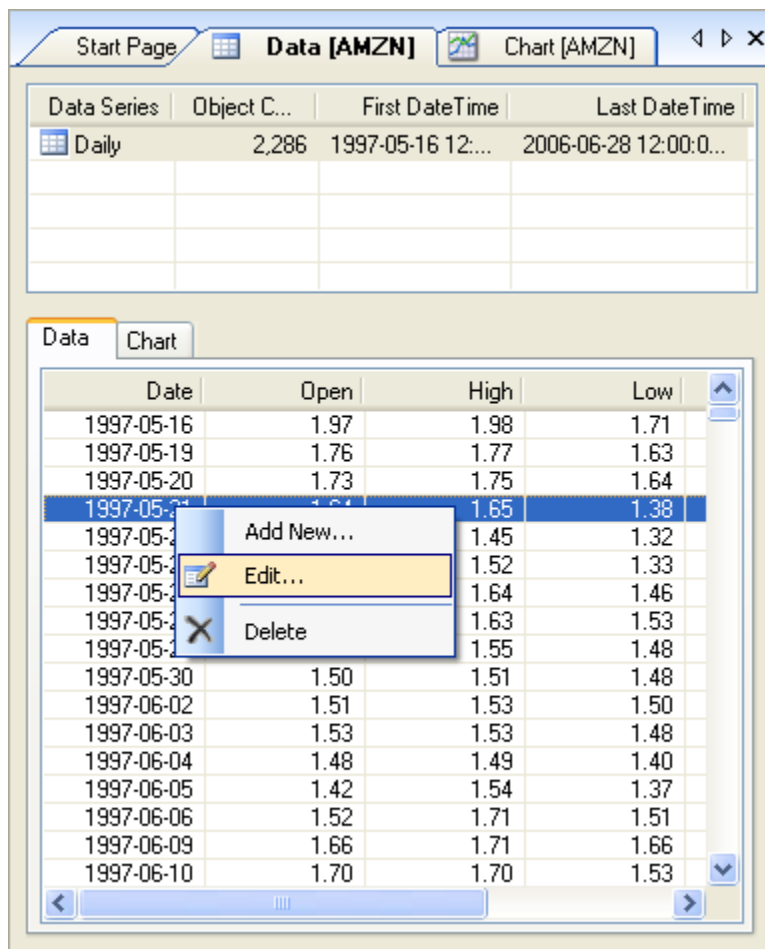
To show the objects (bars) in a data series for an instrument, right click an instrument in the Instrument pane, and choose the Data option from the popup dialog.

- Double click a bar series (Daily, 5-minute, etc) to load individual bars into the grid at the bottom of the window.

- Double click a particular bar in the bottom window to see a dialog that will let you change the OHLCV (open, high, low, close, volume) for the bar.

- Right click in the bottom window to see a dialog that will let you add a new bar to the series, edit a bar in the series, or delete a bar from the series. The right click menu is shown in the picture above.

3.5 Working with Strategy Views

There are several different ways to investigate the results of your strategy. You can see trades on a bar chart, a grid listing of transactions in your portfolio account, an overall equity performance curve for the strategy, and individual trading results by individual instrument. Strategy views can be accessed from a right click menu (as shown in the next screenshot), or from the solution view icons on the right side vertical toolbar.

3.5.1 The Strategy Bar Chart View
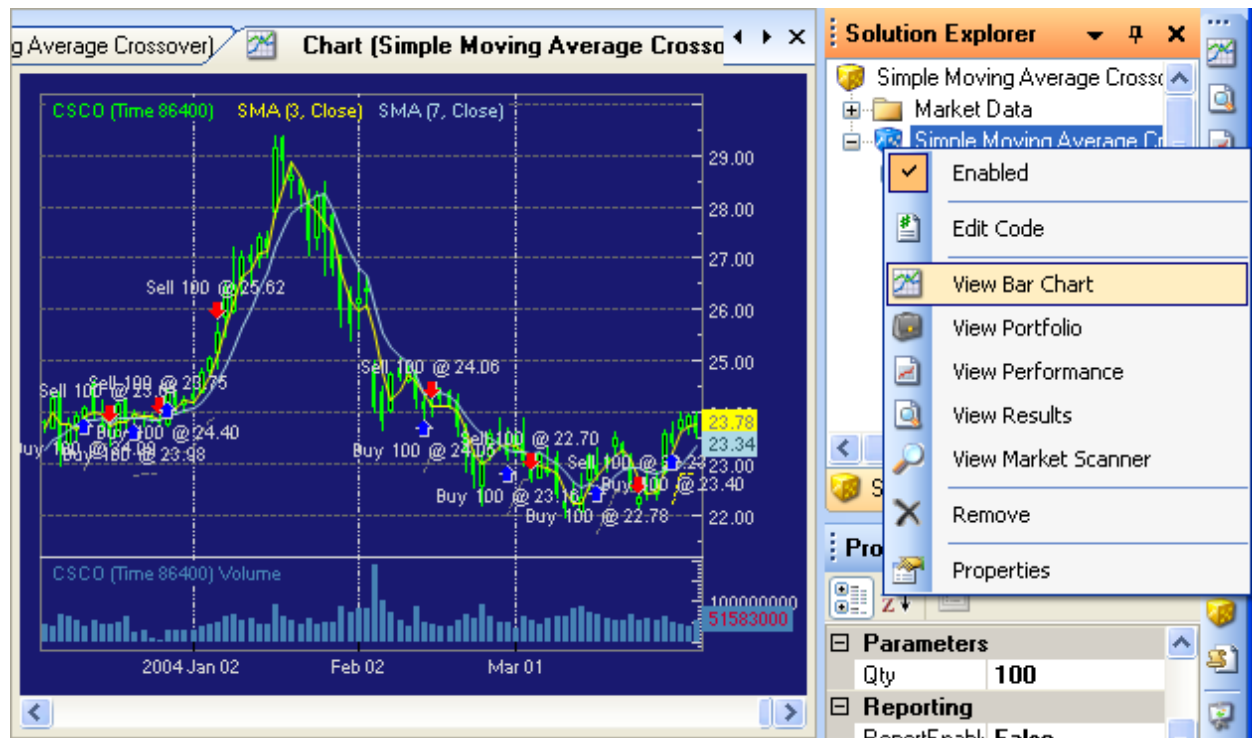
The Strategy Bar Chart View shows strategy buy / sell signals on top of the Instrument bar chart view. This is a convenient view for seeing where your strategy issues buy and sell signals, either in profitable places or unprofitable places (sigh).

To show different parts of the chart, you can move the scrollbar at the bottom of the window, or click somewhere on the chart and use your mouse wheel to compress or expand the chart. Here is a picture of a compressed chart made by moving the mouse wheel.
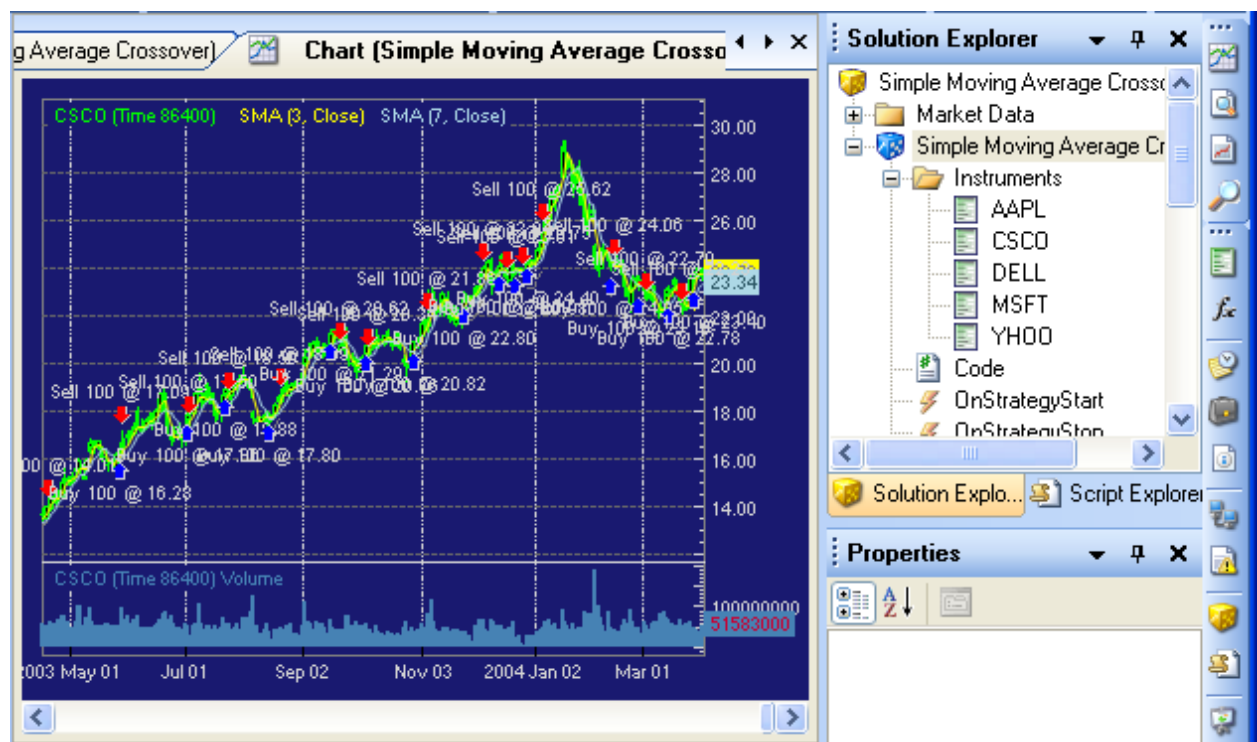
Once you have adjusted the chart scale by moving your mouse wheel, there is no way to disable the mouse wheel action, and no way to return the chart to its original scale. Instead, you must close the whole tab and open a new one (which will open up a new chart with the default horizontal scale).

3.5.2 The Strategy Portfolio View

The Strategy Portfolio View shows how your strategy affects your portfolio. This is the view that you can use to see exactly what trades your strategy made, on what instruments, at what times, and at what prices (and profits).

The main use of the Portfolio Tab is to see your open positions during Paper Trading or Live Trading modes. But you can also look here (in the Transactions window) for specific trades that lost money, so you can go and debug your strategy to see what went wrong.

3.5.3 The Strategy Performance View
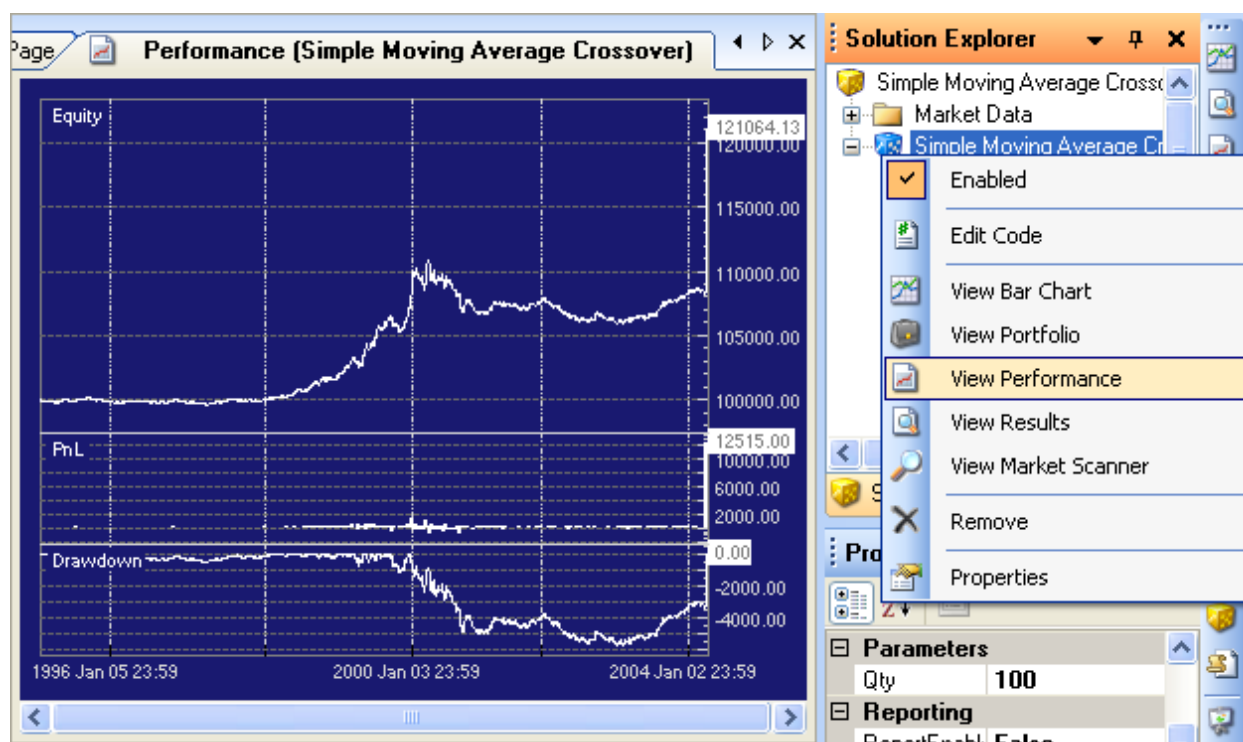
The Performance Tab shows several performance curves for the strategy.

- The Equity curve shows how portfolio equity varies during the strategy. Ideally, this line should rise continuously and smoothly (that is, with low volatility) from bottom left to top right. This particular strategy started with 100,000 dollars in 1996 (starting cash, defined in the strategy Properties window that was shown earlier in this document). Then it earned about 2,000 dollars during the peak Dot-Com years (circa 2000), and then lost money into the bottom of the dot-com crash (circa January 2002). Finally (and thankfully!) the strategy recovered its losses and ended this simulation run with 1,204.50 dollars profit (which is about 1.2% over 8 years, or "not so good" :-).

- The PnL (profit and loss) curve shows the per-trade profit and losses. Ideally, this line should show consistently bigger bars above the line (profitable trades) and consistently smaller bars (or none) below the line (losing trades). As you can see from the bars, there were some huge losing trades (> $1000 loss) around January of 2000.

- The Drawdown curve shows the drawdown (loss of equity) of the strategy, including trades in progress. Ideally, the drawdown curve should be flat or shallow, indicating that the strategy did not have to dip into capital too much during its trades. The problem with deep drawdowns is that they are psychologically and emotionally very disturbing, and often cause traders to bail out at the very bottom of a trade, when they can't take the losses any more. In the screenshot shown above, the maximum drawdown was nearly $3,000—3% of capital—and lasted from late 2001 until mid-2003, or for the best part of a year. Three percent loss is not bad at all for a loss, and probably

36

would have been easy to take for the year. But if the 3% had been 30%, the drawdown would be more difficult.

3.5.4   The Strategy Results View
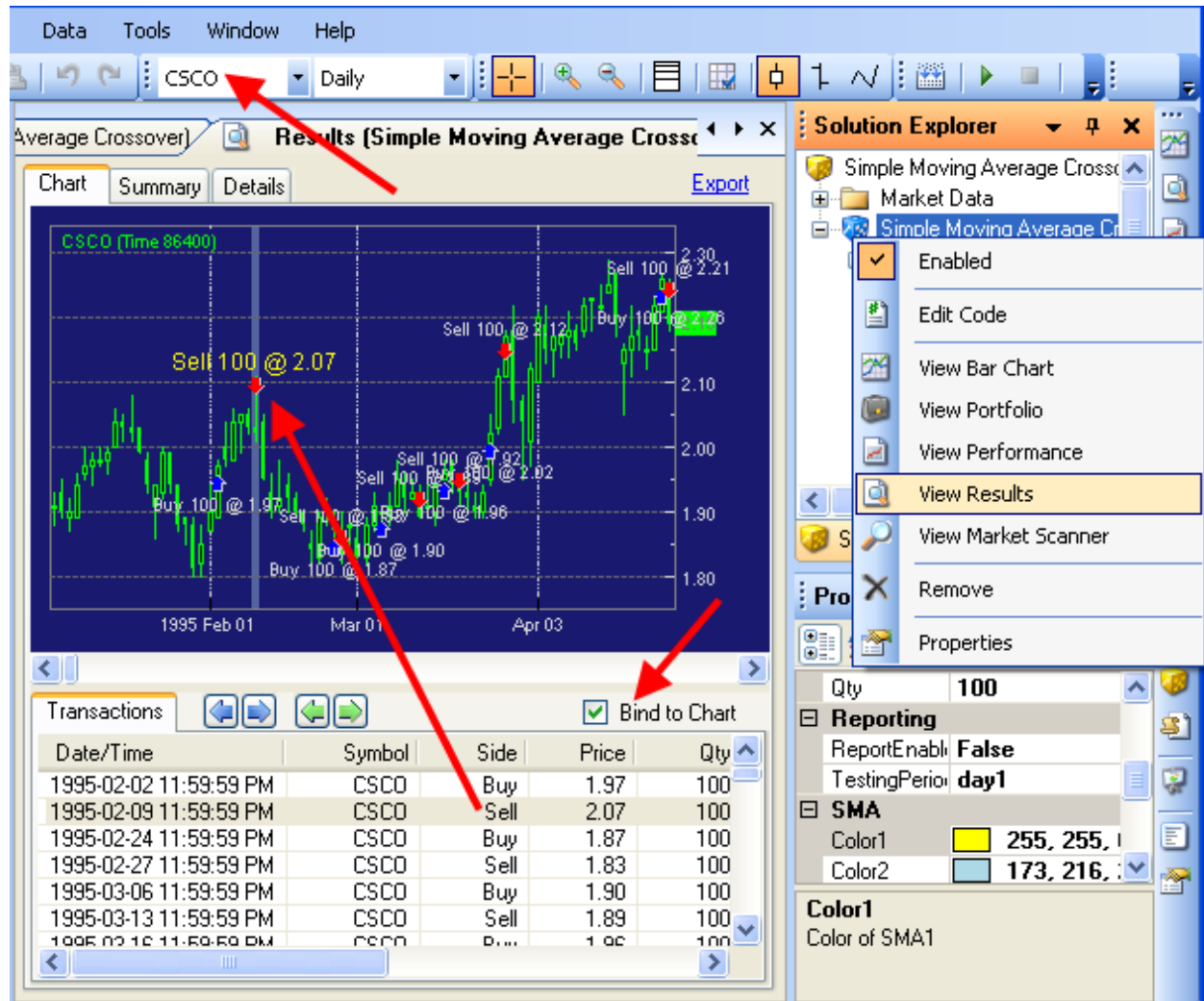
The Results Tab is a useful tab for stepping through the trades that your strategy has made.

- The Results Tab is not shown on the View Menu or the View Toolbar. Instead, to see the Results Tab you must right click the strategy name in the Solution Explorer pane, and select View Results from the context menu.

- You can adjust the resolution of the chart by left-clicking the chart and scrolling the mouse wheel to zoom in or out on the X axis timeline of the chart.

- You can shift the chart left or right by using the bottom scrollbar on the chart.

- The red arrow in the picture above shows you how to examine the strategy trading results for a single instrument within a strategy. (Recall that a single strategy can trade multiple instruments—so you need a way to look at the strategy behavior instrument by instrument.)

- You can lock (bind) the trading transactions to their locations on the chart by checking the "Bind to Chart" checkbox (top right of transaction window). This way, you can double click a transaction, and the chart will jump to that transaction and display the trade that was made, in the context of chart prices. This is a really nice feature.

- You can jump to the next (or previous) transactions using the Blue and Green arrows at the top of the transaction window. The Blue arrows take you to the next transaction, no matter what Instrument was traded. The Green arrows take you to the next transaction in the same instrument that is shown by the current transaction. So use the Green arrows to follow trades in a single instrument, and the Blue arrows to follow all trades in all instruments in chronological order.

3.5.5 The Strategy Market Scanner View
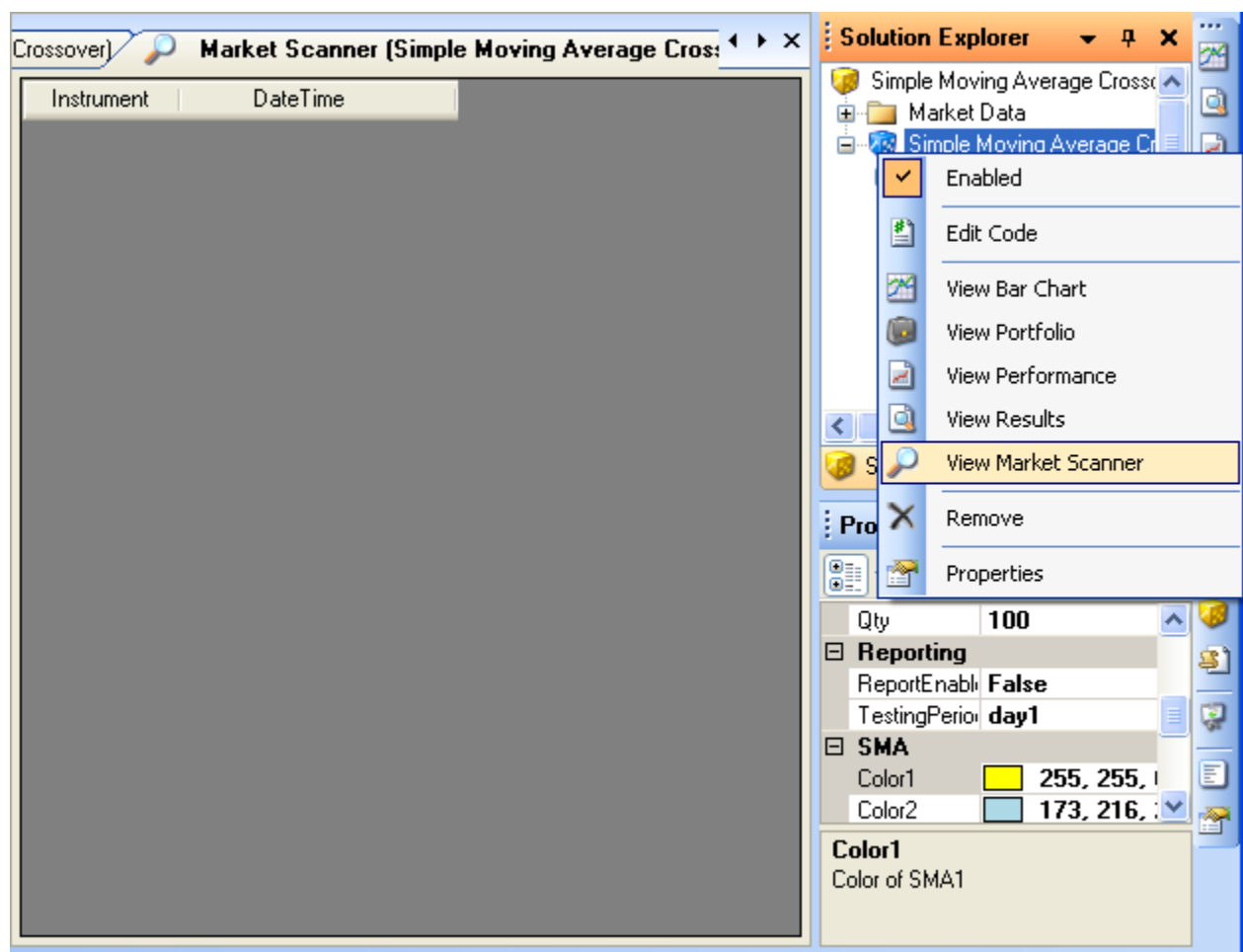
The Strategy Market Scanner screen shows various pieces of information from your brokerage account. The information shown varies with each different broker.

38

This brings us to the end of the tour of the main IDE menus, toolbars, window panes, and strategy result panes. Now we can move on to actually using the IDE to create and run a strategy against an instrument.

4 Acquiring Market Data

This section talks about the different kinds of market data that you can use, sources of that data, and how to import or download the data. OpenQuant ships with a small amount of preconfigured market data, but you will probably want to obtain your own (more recent) market data to test out your strategies.

4.1 Types of Market Data

Market data can be classified into several different types for convenience of understanding.

- Historical data is instrument data from the past—good for back testing strategies, but not very effective for making money on future trades. In contrast, real time market data is more current, and can be used to make reasonable trades in the present.

- Precisely speaking, all market data is historical, because by the time you can capture a trade price and volume at the exchange, the price is already part of history (if only a few microseconds "in the past" behind the present moment in time).

  The only question that matters is whether the "real time" data that you can obtain is sufficient for your strategy. For example, if your strategy only needs daily bars, then free 20-minute delayed prices would be good enough. But in contrast, if your arbitrage trading program needs to trade prices 150 milliseconds after they happen in order to make a profit, then even a 1 second delay makes the data useless to you.

- **Quote data** consists of pairs of bid / ask prices that people are willing to pay to buy or sell an instrument. The presence of a bid-ask pair does not mean that trading will always occur—if the spread is too big, no trades will occur until the market makers step in to facilitate some trades.

- **Trade data** consists of single prices (and volumes) at which trades actually occurred. These are the normal prices that are used to make up an OHLCV (open, high, low, close, volume) Bar.

- **Bar data** is manufactured by OpenQuant from Trade data. To pick a bar size for your strategy, you should use the right-click menu (shown previously) on the Market Data folder in the Solution Explorer pane. That menu lists several different bar sizes that you can chose from (or you can create custom bar sizes as shown in the dialog below.
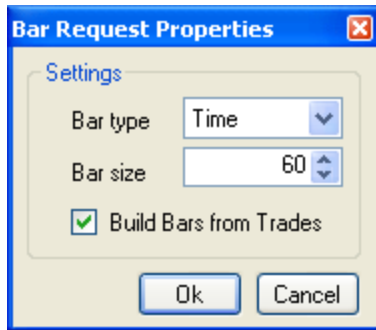
- TAQ data is an abbreviation for Trade and Quote data, and is commonly associated with the DVDs of actual NYSE TAQ data that you can buy from the NYSE. These DVDs contain real time TAQ data for all the stocks and options traded at the NYSE, and are available by monthly subscription. (They are for institutional traders, and are expensive.)

4.2   Importing Market Data

Computerized quantitative trading strategies can consume very large quantities of trade and quote data, so it is important that the software that processes the data is fast and high performance. For this reason, SmartQuant wrote their own high performance database to store trade data.

When you import or download market data, it is converted into the format used by the high performance SmartQuant database. Here is a screenshot of the data import wizard for importing data from a CSV (Comma Separated Values) or TXT (text) file.
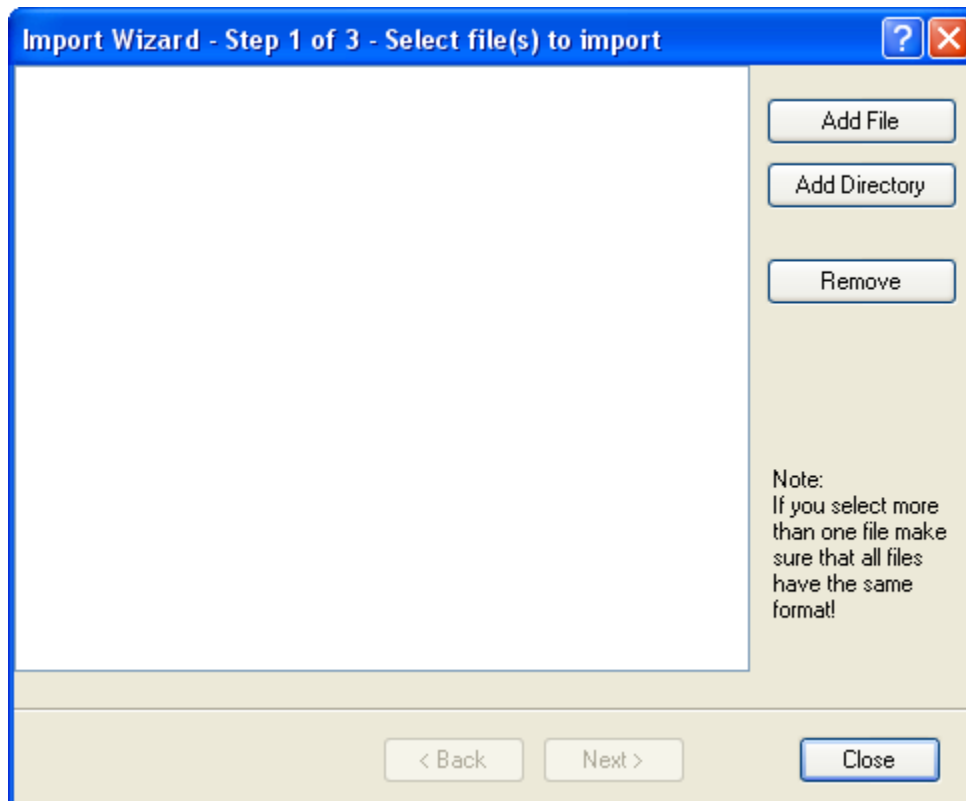
- To import data from CSV (comma separated values) and TXT (text) files, use the Data / Import menu on the IDE. This will pop up a data import wizard dialog that will help you to perform the import operation.

- To import data from a TAQ CD, use the Data / Import menu on the IDE to get a different dialog box that is designed specifically for importing from the TAQ CDs.

- To download market data from a network data provider, use the Data / Import menu on the IDE to get a wizard dialog sequence for downloading. Choose your desired data provider from the list, and continue with the wizard. You should probably try to download free daily bars from Yahoo first, to see how downloading works.

- Downloading data also automatically imports the data and stores it in the SmartQuant high-performance database. There is no need for you to import after downloading.

4.3   Viewing a Data Series

Right click an instrument in the instrument pane to view the underlying data series or chart. This right-click menu is also the mechanism for adding a new instrument to the instrument pane.
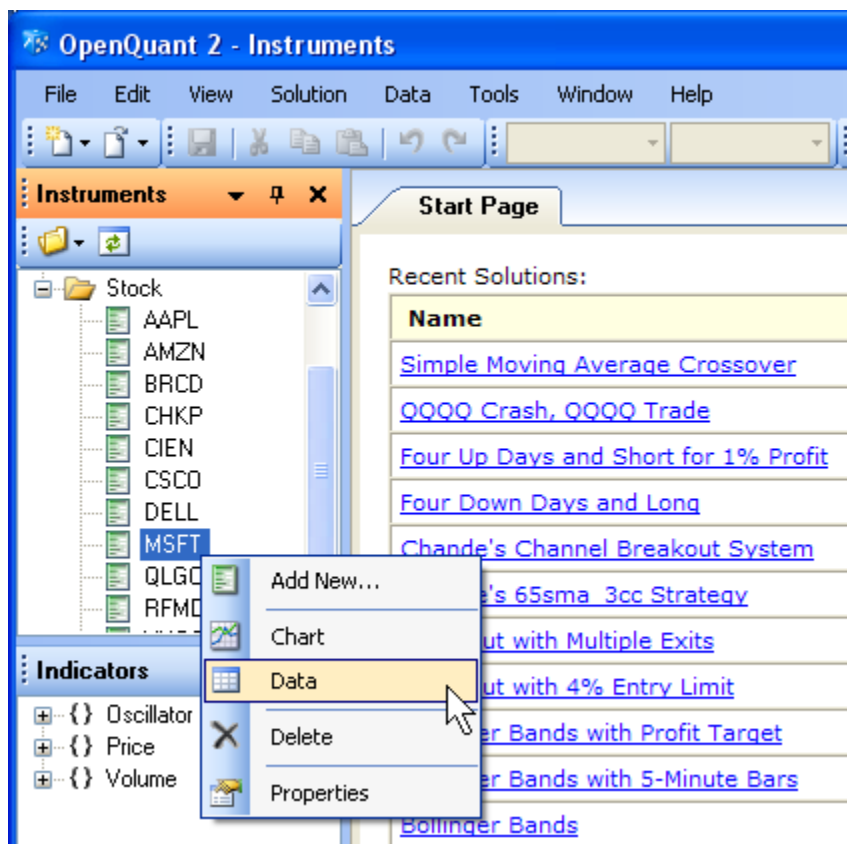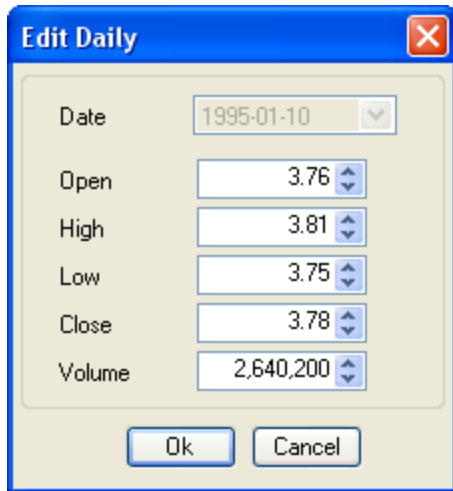
Here is a screenshot of the various preloaded data series for the MSFT instrument. You must left click a series to select it before the corresponding bar data will appear in the bottom window.

Data Series	Object Count	First DateTime
Bar 5 min	9,224	2006-05-01 2:00:00 PM
Daily	2,329	1995-01-03 12:00:00 AM
Quote	27,517	2001-12-04 7:30:59 AM
Trade	40,769	2001-12-04 9:30:00 AM

Data | Chart

Date	Open	High	Low
1995-01-03	3.81	3.81	3.73
1995-01-04	3.74	3.77	3.69
1995-01-05	3.77	3.77	3.67
1995-01-06	3.71	3.80	3.71
1995-01-09	3.77	3.78	3.70
1995-01-10	3.76	3.81	3.75
1995-01-11	3.78	3.82	3.76
1995-01-12	3.79	3.84	3.77
1995-01-13	3.84	3.91	3.84
1995-01-16	3.92	3.99	3.90
1995-01-17	4.01	4.03	3.98
1995-01-18	4.02	4.04	3.98
1995-01-19	3.99	4.02	3.93
1995-01-20	3.90	3.94	3.80
1995-01-23	3.78	3.90	3.75
1995-01-24	3.90	3.91	3.84
1995-01-25	3.82	3.86	3.78

4.4   Editing Stored Data

- To add a new bar to the series, right click anywhere in the bottom data window, and choose Add New… from the pop-up context menu.

- To edit a particular bar from the series, double-click the bar line, or right-click the bar that you want to modify and choose Edit from the pop-up context menu.

- To delete stored data series entries, select the bars that you want to delete, right click them, and choose Delete from the pop-up context menu.

4.5   Analyzing Data with Indicators

The procedure for analyzing stored data with technical indicators was described above under the Instruments pane and the Indicators pane. Here it is again:

- Right click an instrument name in the Instruments pane, and show Chart.

- Drag and drop an Indicator name from the Indicators pane to the Chart Tab.

- Configure the new indicator for the first time using the dialog box that appears (shown below). Be sure to check the "Create a new pad" checkbox if you want to show the indicator in a new chart section at the bottom of the chart.
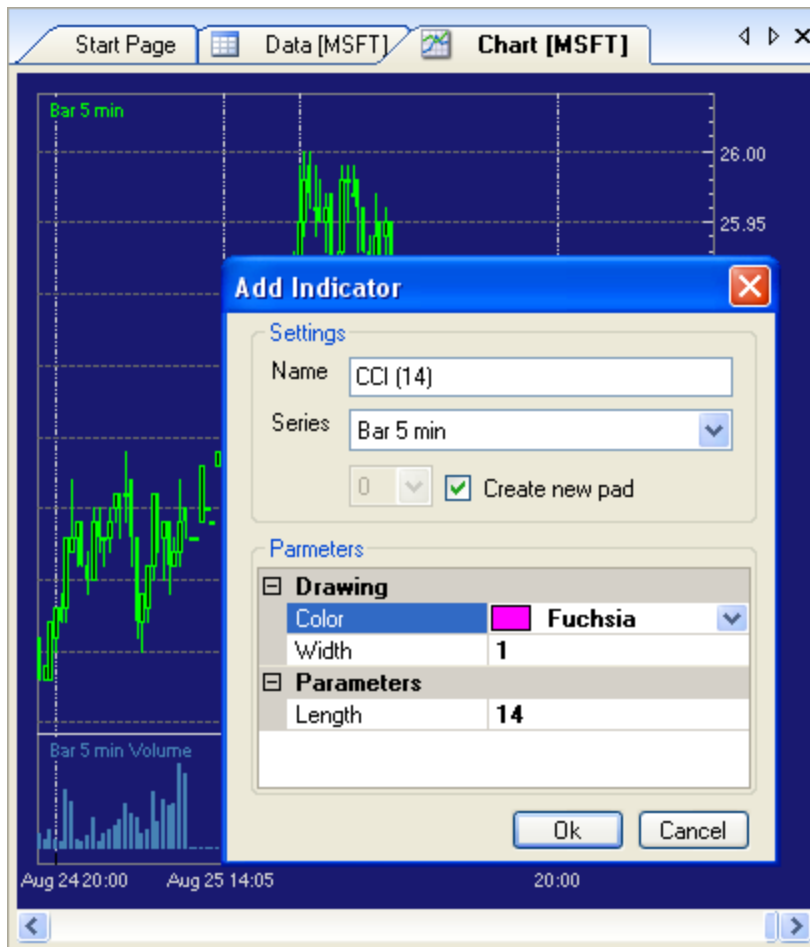
- Edit indicator properties by left-clicking the indicator line in the Chart Tab. This can be tricky for narrow lines. Once you have the indicator line selected, you will see several selection handles on the indicator line, and the corresponding indicator properties will show up in the Properties pane for editing.

- To delete an indicator, select it in the chart window (as just described), then right-click it and choose Delete from the pop-up menu. The indicator will leave a blank area on the chart. There is no way of removing the blank area. Instead, delete the whole chart window tab from the center window pane (use the upper right X icon), and rebuild the chart again by dragging and dropping indicators.

5 Running a Preconfigured Trading Strategy

Finally, we can start to talk about running the preconfigured trading strategies!

5.1 Running Your First Strategy
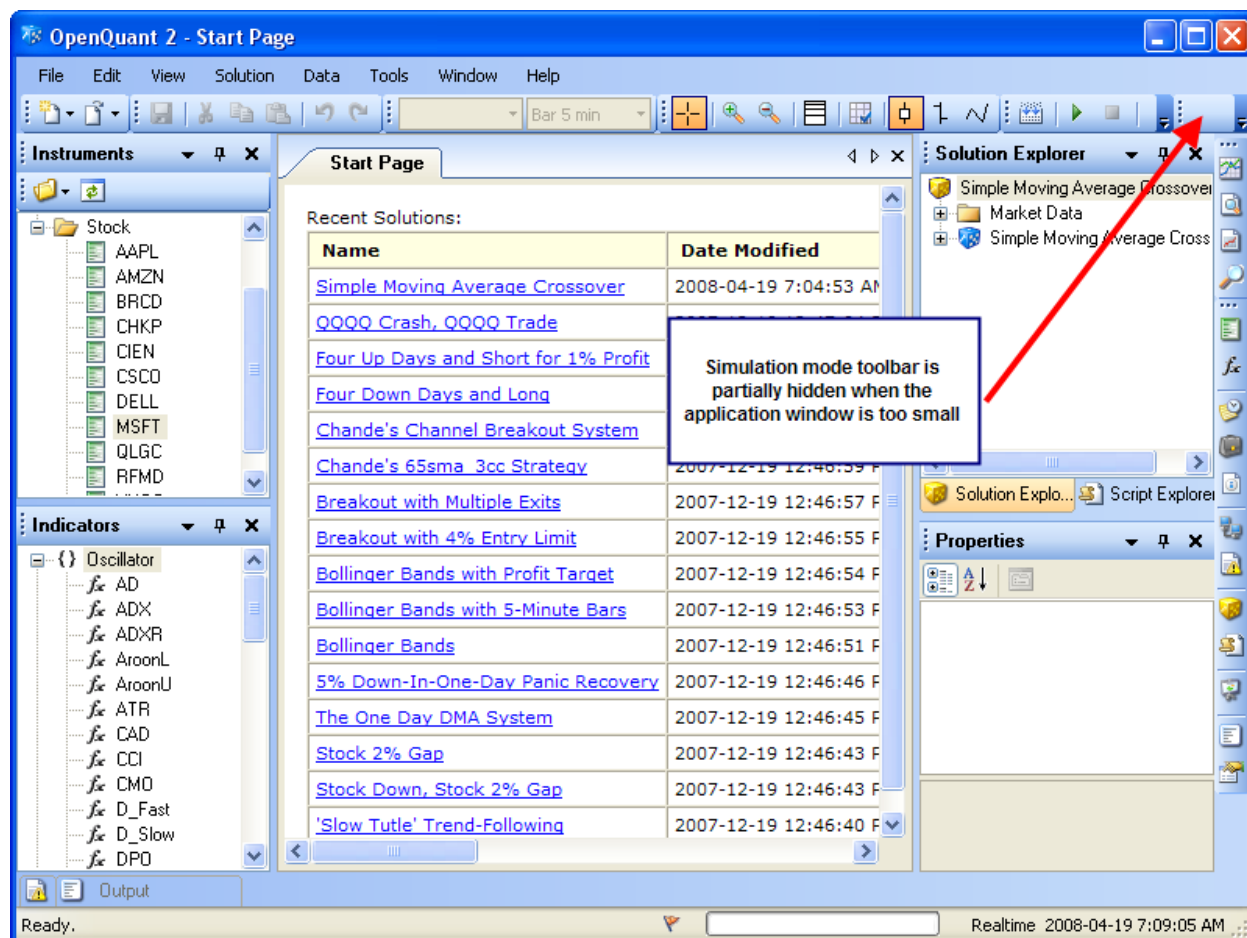
To run your first strategy, run one of the preinstalled strategies. Show the start page with View / Other Windows / Start Page, then click on the Simple Moving Average Crossover Strategy. That should load the strategy into the Solution Explorer pane. If it is not visible, use the View / Solution Explorer menu to make it visible.

- Use the Trading Mode toolbar to select Simulation mode. The toolbar might be hidden or partially hidden if you have set the application window size too small (like I did for the screenshots in this document, to make the screenshots consistent in size, 800x600). If you can't see the Trading Mode toolbar, drag some toolbar handles (the vertical four dots on each toolbar) to make two rows of toolbars. Select simulation mode when you can see the toolbar.
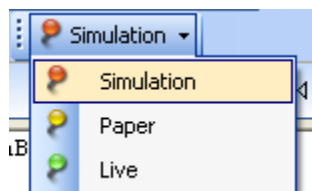
- To run the strategy, click the Run arrow icon (tiny light green arrow) on the Solution toolbar (about 1 inch to the left of the long red arrow in the screenshot above). Or right-click the project name "Simple Moving Average Crossover" and choose Run from the context menu. (Now that you know which icon means Run, you can use the Solution Toolbar to run the strategy next time.)

- Once the strategy run has completed, a performance chart will show up in the center pane.
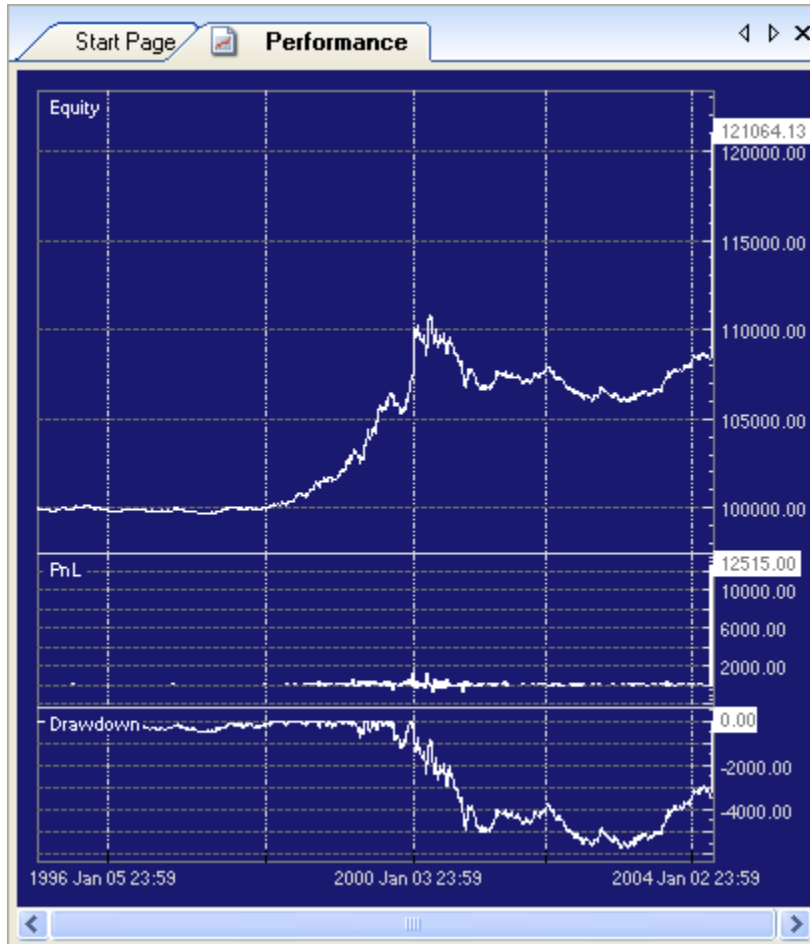
Previous parts of this document have already showed how to analyze strategy results, so you can inspect the results of this strategy for yourself. Remember that you must use "View Results" on the Strategy context menu (shown earlier) to show the Results tab. Recall that the Results tab locks the trades on the chart to the trade transactions in the bottom window of the Results tab.

Be sure to have a careful look at the Performance curves (Equity, PnL (Profit and Loss) and Drawdown) for all of the preconfigured strategies. They should show you that writing a good strategy that performs well is not as easy as it looks!

6 Creating New Solutions, Projects, and Strategies

This section summarizes the steps required to create completely new *solution containers* and *strategy projects*, without going into the theory or details of good strategy design (this document is only a Getting

Started Guide). See the other SmartQuant PDF documents for a deeper discussion of trading strategy design.

To avoid confusion concerning the meaning of the word "project", this document always refers to *solution containers* and *strategy projects*, or to (short names) *solutions* and *strategies*. For example, this document will never call a solution a project. In fact, this document tries to never use the word "project" at all, since it is so confusing.

In contrast, the application software uses the terms Solution Project ("New Solution", "New Project") to refer to *solution containers* and *strategy projects*. Now imagine that you named your first solution "My New Project." When you say "my new project" to someone or in a document, are you referring to the solution or to the strategy? (As you will see, if you happened to create the new solution and project simultaneously with the File Toolbar New Project button, names and terminology get even more confusing!)

It will be much easier for you to learn to think in terms of solutions and strategies and to avoid thinking about projects—the project terminology is too confusing for a Getting Started guide.

The software allows one solution container to contain multiple strategies, because it is often convenient to run multiple simultaneous strategies within one simulation run. The following screenshot shows how you can enable individual strategies by using their right-click menus.
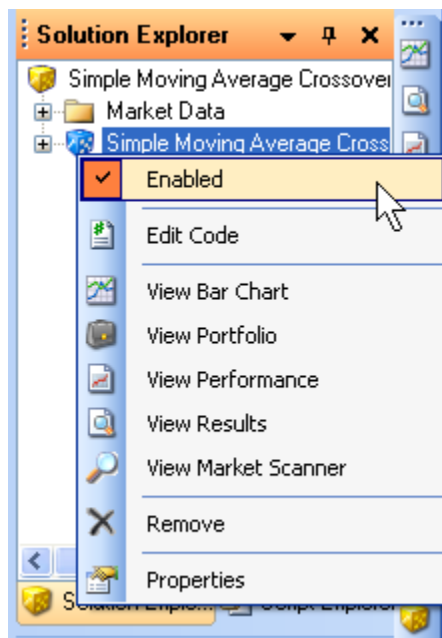
6.1   Solutions, Strategies, and Projects

The current OpenQuant menus enable you to do three kinds of actions easily:

- (Recommended) Create a new solution container (use the *File / New / Solution* menu). This method is recommended, and creates a new (empty) solution container that can have a name that is clearly distinct from any strategy name.

- (Recommended) Create a new strategy project inside of an open container (right click the solution name). This method is recommended because it clearly separates *solution* creation actions from *strategy* creation actions, and makes it easier to treat solutions and strategies—and their names—as completely separate objects (which they are, within the software).

- (NOT Recommended) Create both a solution and a strategy simultaneously (use the confusing *File / New / Project* menu). This method is NOT recommended in this Getting Started Guide, because it makes it too easy to confuse solution names with strategy names, to easy to confuse solution projects with strategy projects, and too easy to pick a solution name that is coupled to a strategy name.

  It also makes it too easy to think that you can use an existing solution name in the dialog box to put a new strategy into an existing solution container (which is NOT possible).

  Furthermore, since renaming solutions and strategies is not that easy to do in OpenQuant (not all references to the old name are automatically removed from all software objects), it is important that you get your solution and strategy names right the first time around. For all these reasons, this Getting Started Guide recommends that you create solutions and strategies separately, and *think* that way, as described by the two previous paragraphs.

6.2   Creating a New Solution

To create a new strategy, use the *File / New Solution* menu or the New Solution button on the File Toolbar. Enter a solution name in the dialog that pops up. New solutions are stored in the "*C:\Program Files\SmartQuant Ltd\OpenQuant\Solutions*" folder.

If you ever want to delete a solution, you must delete the solution folder at the end of that pathname. One of doing the deletion from the OpenQuant application is to choose File / Open / Solution, which will bring up a standard Windows file open dialog. Then right click the folder you want to delete and choose delete from the popup context menu.

Deleting a folder does not remove all references to the solution or strategy (such as the references on the Start Page).
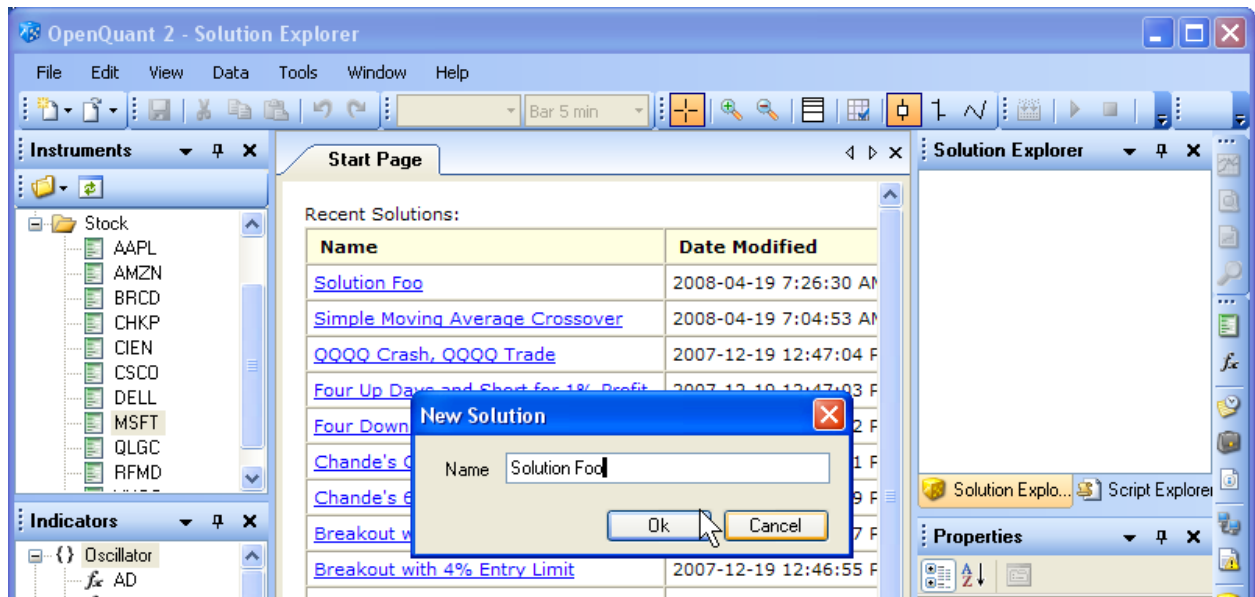
- A new (blank) Solution will be created in the Solution Explorer pane.
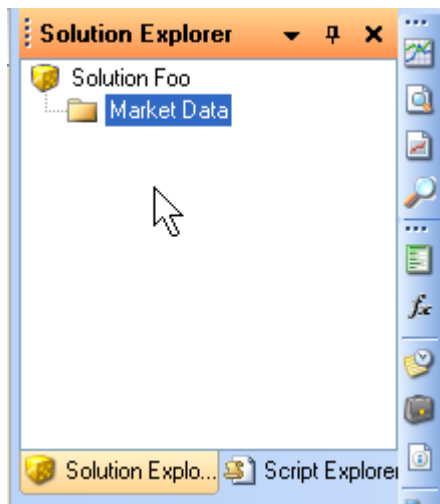
- As you can see from the screenshot above, the new blank solution has no Strategies, no Market Data types, and no Instruments in the solution. We must define all of those things before we can run a strategy. The screenshot below shows what the window should look like after all the missing pieces have been defined.
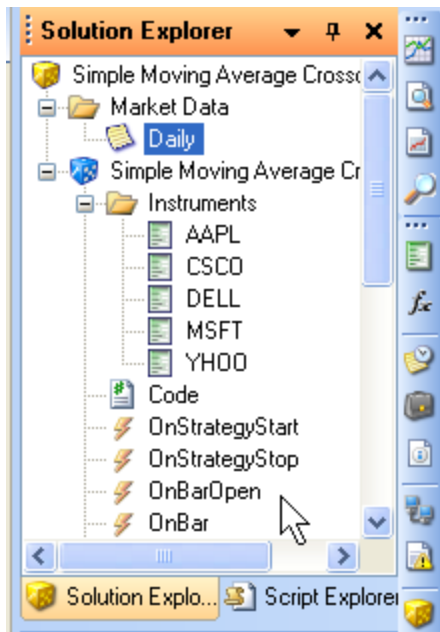
6.3   Adding a New Strategy to a Solution

- To add a new strategy to the current (open) solution, right click the solution name and choose *Add / New (Strategy) Project* from the context menu.
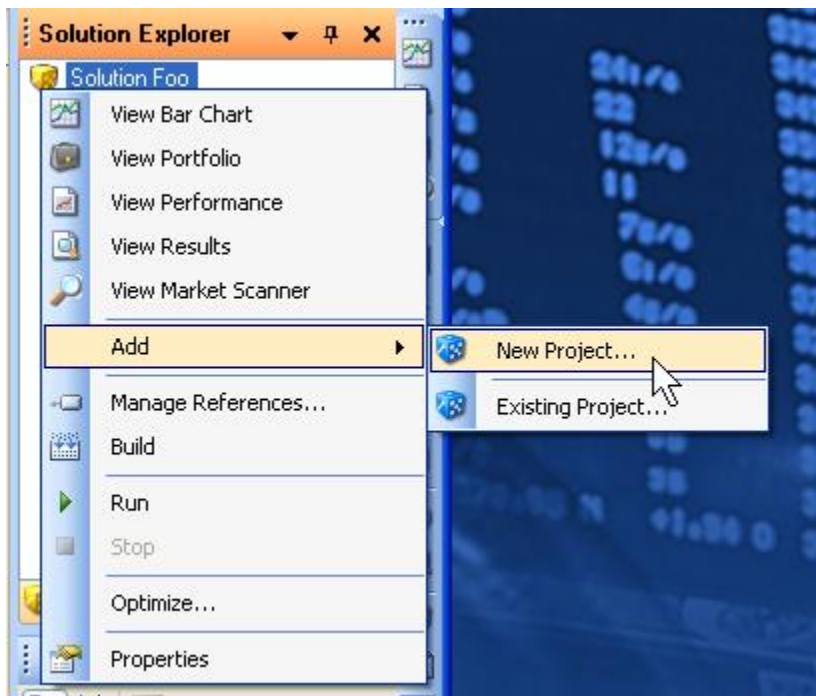
Notice that the context menu also gives you the choice of adding an existing strategy project to the current open solution. This software organization—which stores solutions and strategies in separate folders on the computer disk—enables multiple solutions to share the same single copy of a strategy.

- Then fill in the dialog box that appears to create your new (blank) strategy. Notice how the solution name is automatically filled in for you, since you are using the recommended way of adding a strategy to a solution (right click the solution name).
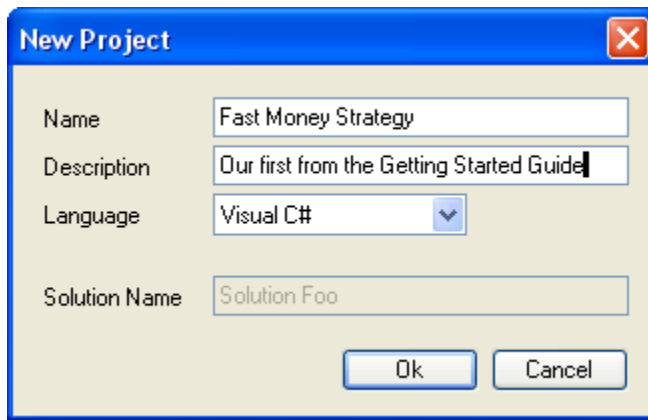
- Your new strategy will now appear in the Solution Explorer pane. But the strategy still doesn't have any instruments defined, and the solution doesn't have any market data types defined (Daily, 10 minute bars, etc). We'll add both of those next.
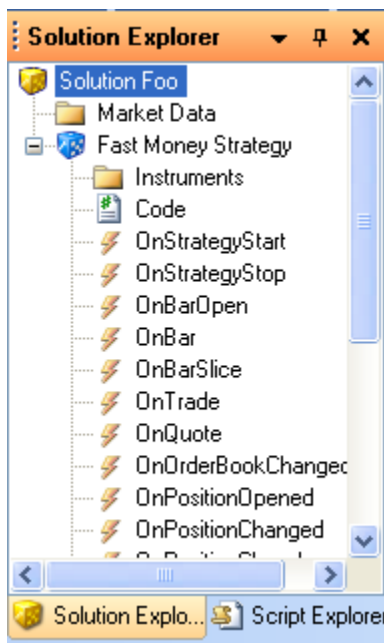
6.4 Adding an Instrument

- To add an instrument with menus, right click the Instruments icon below the *strategy* name in the Solution Explorer pane, and select "Add an Instrument." Choose whatever instrument you like. We picked AMZN in this example.
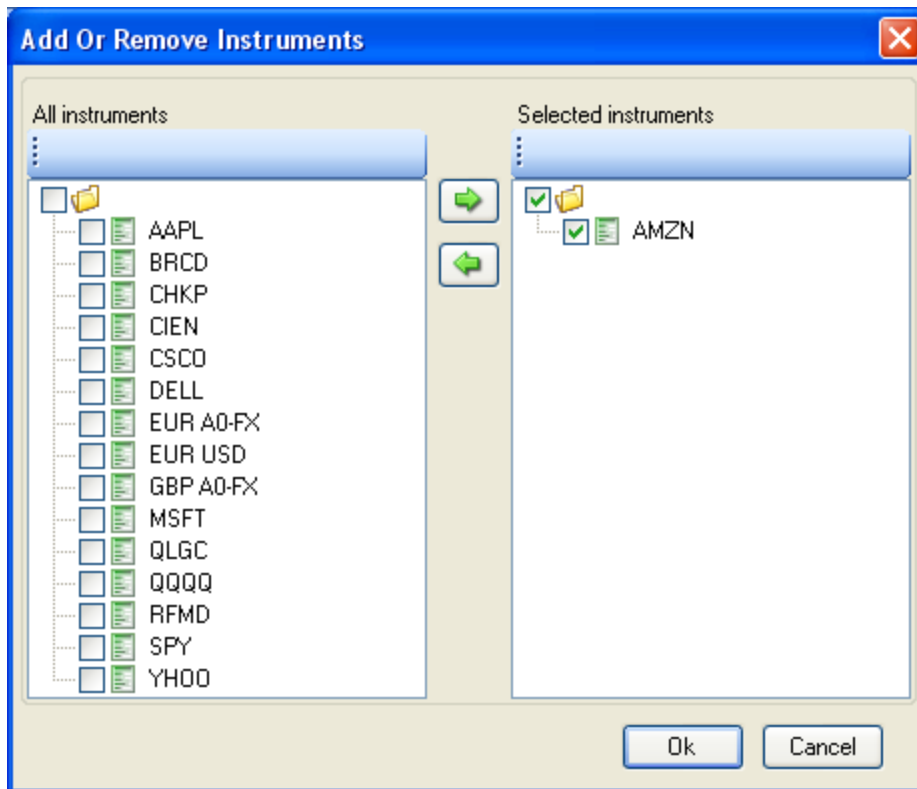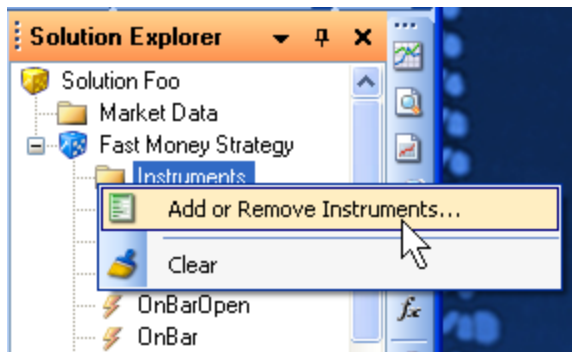
- To add an instrument using drag and drop, drag an Instrument from the Instruments pane and drop it somewhere within the current strategy within the Solution Explorer pane. This method is fast and easy, but it does not let you add and remove multiple instruments easily.

- Either way, you now have an Instrument defined for your strategy, and you can see the Instrument properties in the Properties pane. Normally you would access the instrument properties by working with instruments through the Instruments pane, but you can also edit them by selecting an instrument within a strategy. Be aware that changing instrument properties within one strategy affects the underlying instrument definition for all strategies.
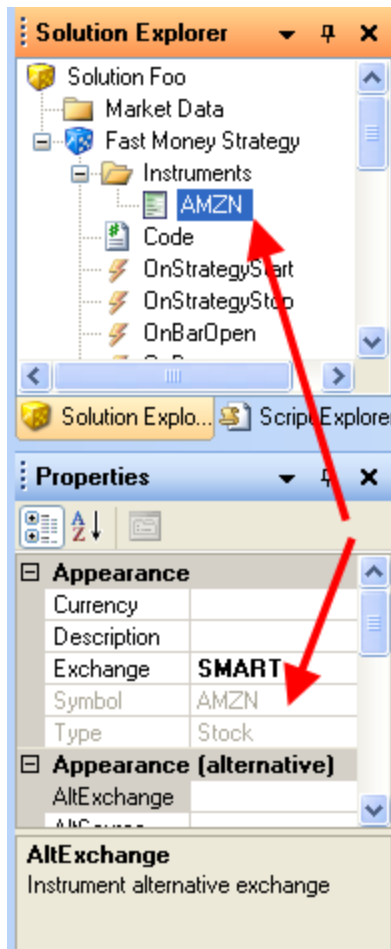
6.5 Choosing a Market Data Type

- To choose a market data type, right click the Market Data icon in the Solution Explorer pane. Then choose *Add / Daily*. We want to choose Daily bars because the IDE has a preconfigured data series for AMZN already installed.
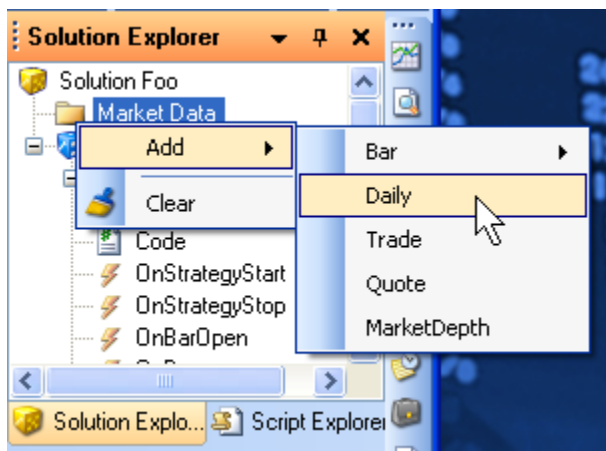
- At this point, the Solution Explorer should display both AMZN and Daily bars.
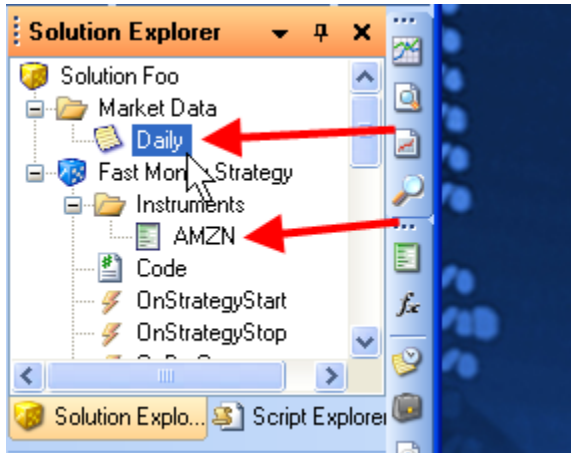
6.6 Viewing Your Strategy Code

- To view your new strategy code (it will be almost empty), double click the C# code icon in the Solution Explorer pane. This will display your C# code in the Code Editor tab.

- Add some code of your choice to make the strategy run. Since this document is only a Getting Started guide, it will not show you how to code a strategy. Instead, cut and paste the code from one of the preconfigured strategies into your strategy, so that you can get your strategy to work.

- Once you have copied code from another strategy, your strategy should be ready to run. Click the Build icon on the Project Toolbar, or right-click the strategy name in the Solution Explorer pane and choose Build from the context menu.
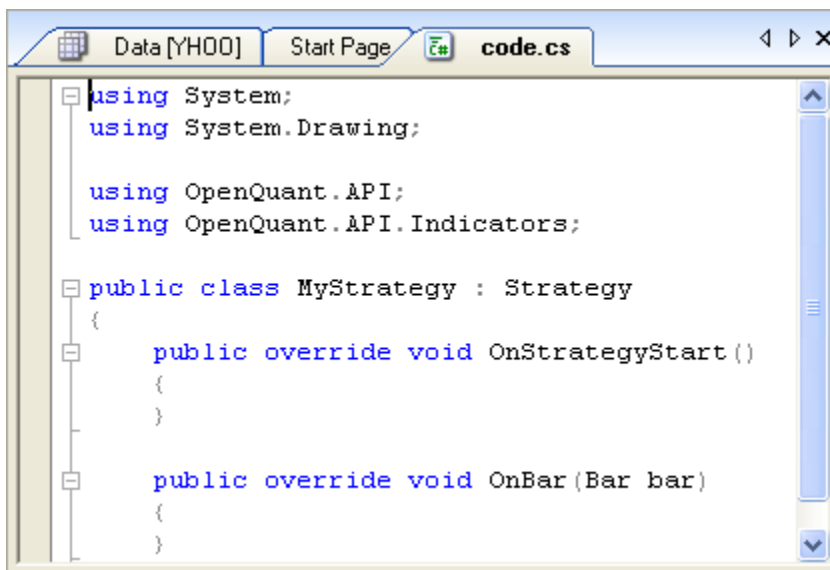
6.7 Correcting Code Errors

- If you make a coding mistake and try to build the project, you will see error messages in the Error List Pane. The Error List pane was not shown previously in this document, because there is no way for you to make it appear unless you generate a coding error. It does not appear on the View menu or View Toolbar.

- A picture of the Error List pane is shown below. It shares docking space with the Output and Provider Errors panes. If you double click an error line in the Error Pane, the editor will move the code editor cursor to the source code line that generated the coding error. After you fix the error and rebuild (or Run) the project successfully, the main Output window will again get the focus, and the Error List pane will lose focus and will be inactivated.
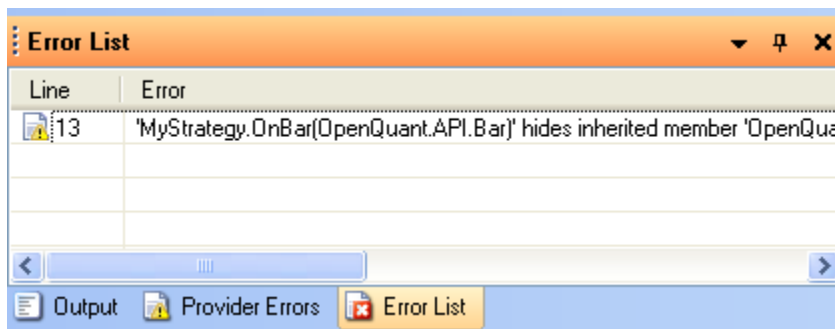
6.8 Code Editor Capabilities

- The code editor in the IDE is capable of color syntax highlighting, automatic formatting, and code completion suggestions, as the following screenshot (of code completion) shows.
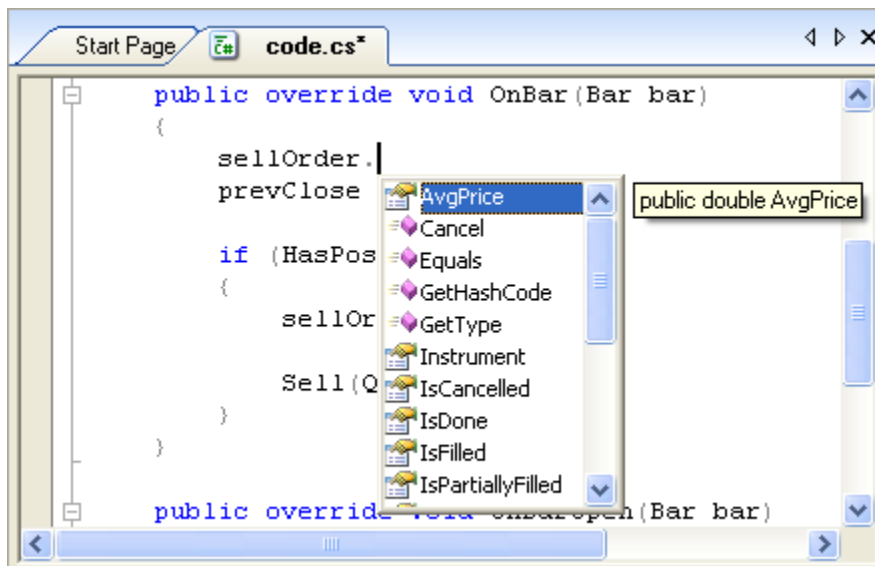
6.9   Running Your New Strategy

- To keep things simple, let's cut and paste the strategy code from the Simple Moving Average Crossover strategy into the new Fast Money Strategy that we created above.

- Click on the Start Page tab in the center window (or show it using the View menu), click on the Crossover link to load the Crossover solution, click the Code icon, click inside the code editor window, type Ctrl-A to select all the crossover code, and click Ctrl-C to copy the code. Now let's go back to our own strategy and paste the code there.

- Click on the Start Page tab, click on the Fast Money Strategy link, click the code icon in the Solution Explorer pane, click within the code editor window, type Ctrl-A to select all the (empty) code in our strategy, and finally type Ctrl-V to paste the crossover code into our own code editor window. If you can see a pile of code, then we can move on. Otherwise, repeat these steps until you get the code, or use your own cut and paste method.

- Because we have used a preconfigured instrument (AMZN) and a preconfigured bar size (Daily), there is no need for further configuration in order to run the new strategy.

- Make sure that you are in Simulation mode (use the Trading Mode toolbar)

- Run the strategy as usual (click the little green Run arrow) and examine the Performance chart that appears. Here is what the chart for AMZN using the Fast Money Strategy looks like (really it's the code from the Simple Moving Average Crossover strategy).
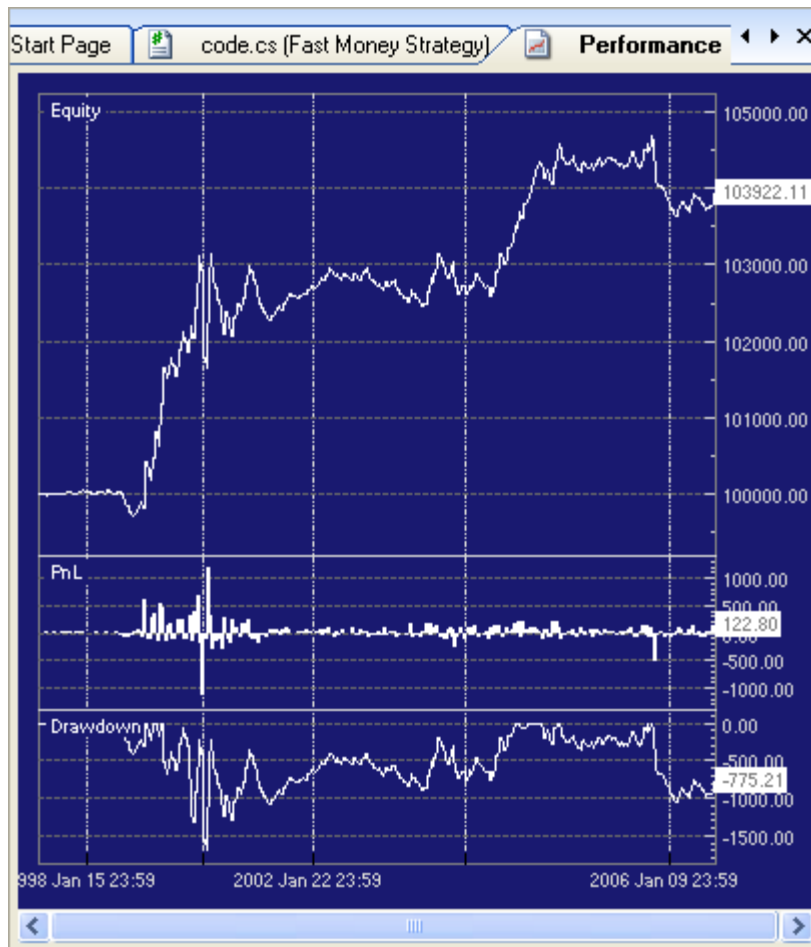
- As you can see, the crossover strategy worked OK between 1998 and 2006—the equity curve climbed quickly during the dot-com era (1998-2001), oscillated a bit during the dotcom crash, but then recovered a bit in the 2003-2005 time frame. But the strategy did not earn much money—it started with 100K in 1998, and ended up with about 104K in 2006. Only 4K for 8 years is not that good—but the good news is that the strategy did not LOSE a pile of money during that time. Hopefully your strategies will perform better.

- You should try to modify the strategy code by changing the names of some parameter variables or parameter text descriptions. Then look for your changes in the Properties pane after you rebuild the project. That will convince you that you can add strategy parameters and have them show up in the Properties pane with all the other parameters.

6.10 Setting Initial Simulation Values

For Simulation mode only, you can specify particular start and finish dates for a strategy execution, regardless of what trading mode you are using. Of course, it doesn't make much sense to specify dates when you are live trading, because you are always trading in the present.
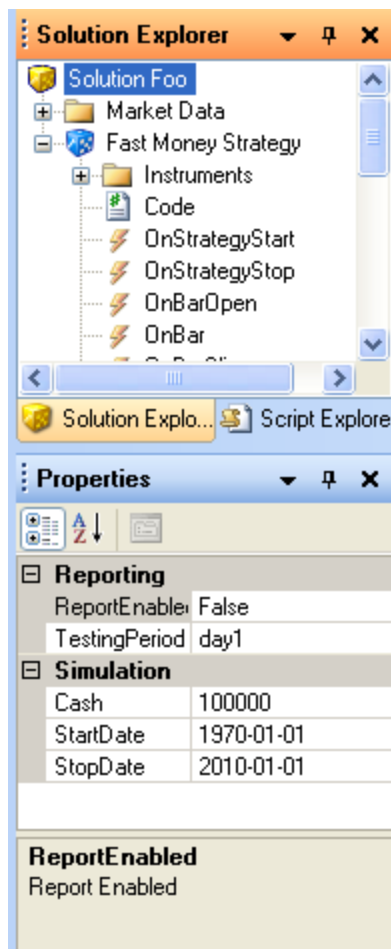
- To set initial simulation values, select the *solution* name in the Solution Explorer pane. This action will load the simulation properties into the Properties pane. (If you mistakenly select the *strategy* name, you will load the strategy parameters into the Properties pane, not the simulation run parameters.)

- You can also specify the amount of starting cash that the strategy gets to spend. In the example above, the strategy is blessed with 100,000 dollars of initial capital. (You would probably want to be _very_ sure you had a good strategy before you traded that much capital in a live trading mode.) The starting Cash is used in Live Trading as a starting point for equity, and return calculations. (But of course the Cash you specify in the IDE is not the same real Cash that resides in your brokerage account.)

6.11 Making Results More Realistic

There are a variety of simulation parameters that you can change to make your simulations more realistic. Some of the most common parameters are commission costs, and price slippage amounts (the slippage between the market price when the simulation trigger fires, and the actual fill price that you might get from a trade execution provider).

- To choose the simulation execution provider, select the provider in the Providers pane. The little green dot on the icon indicates that the simulation execution provider is connected (active), since

we selected Simulation on the Trading Mode toolbar when we ran our new strategy a few paragraphs ago (above).
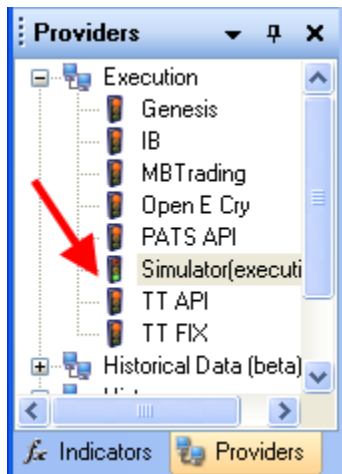
- Here is a screenshot of some of the parameters for the Simulation execution provider.
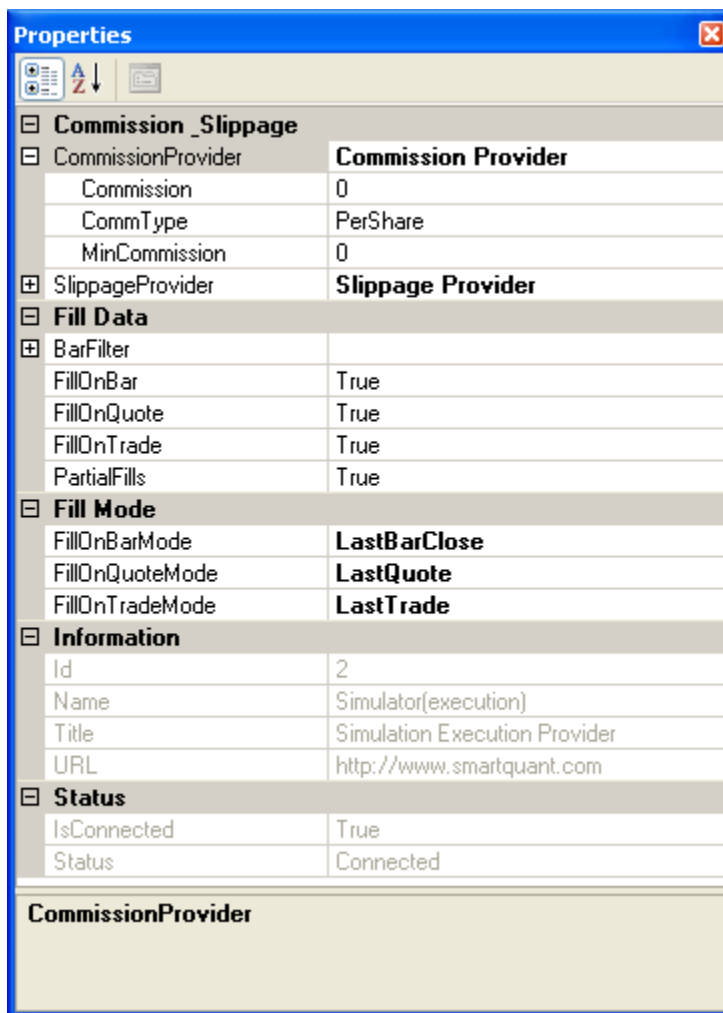
- For simulation mode (and the Simulation Execution Provider) only, make sure that you have specified reasonable values for Commission and Slippage charges in the Properties pane for the Simulation execution provider. This is a good thing to do because it helps your strategy to report more realistic profit results. A significant amount of profit in a computerized trading strategy can be consumed by commissions and price slippages during order fills. (If you are using a Live execution provider, the broker will automatically use their actual values for commissions and price slippage in your live transactions, so there is nothing for you to configure there.)

- For Simulation mode only, select the type of Order Fill that you prefer for Bars, Trades, and Quotes. The defaults are to fill orders on the close of the bar, or on the most recent Trade or Quote (if you are using Trade or Quote data directly).

7 Configuring a Strategy for Paper or Live Trading

You have already seen most of what you need to know in order to create and run a strategy, and to analyze its results and performance behavior. The only remaining topic to discuss is how to connect your strategy to a live market data provider and a live trade execution provider.

Switching from simulation mode (which we have been using so far) to paper trading or live mode is not difficult, as the following paragraphs will show.

7.1 Configure Trading Mode Options
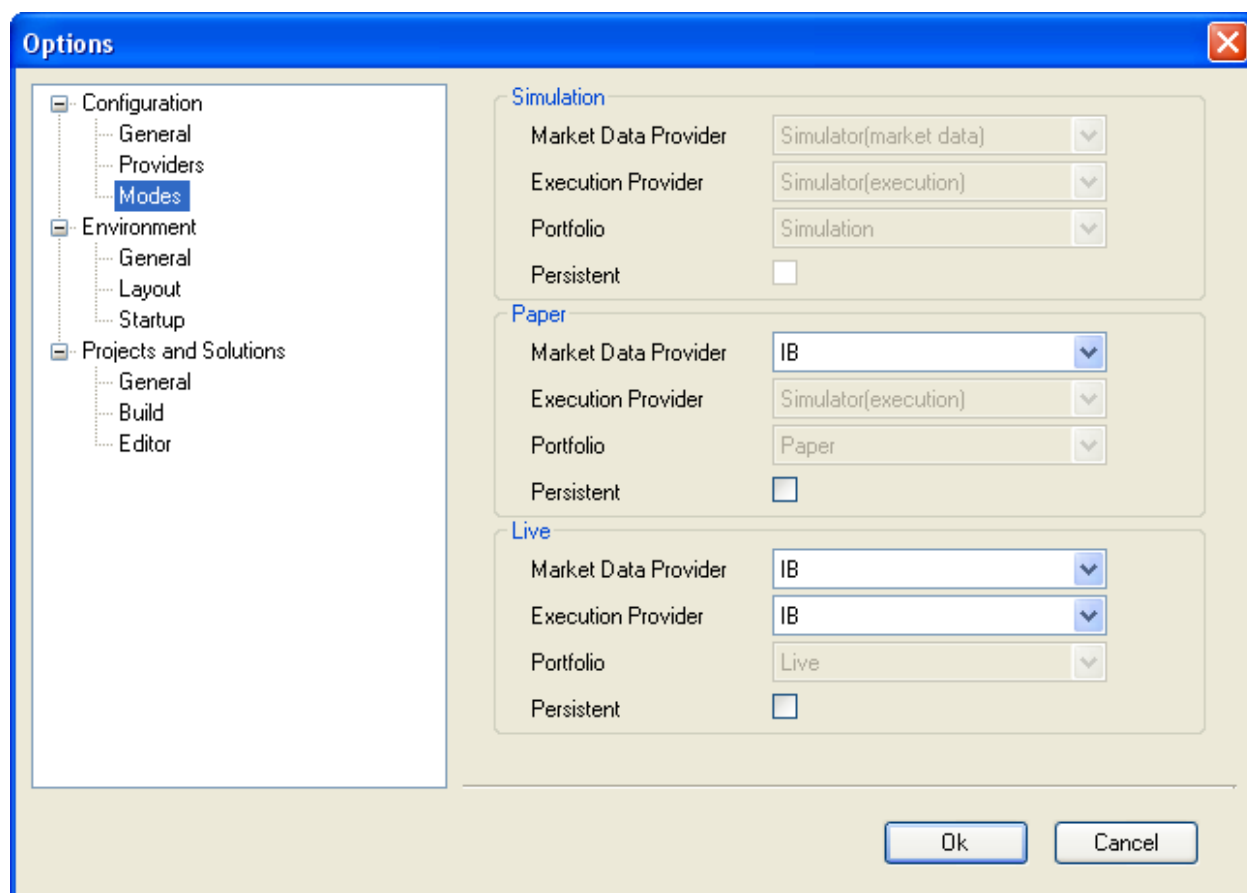
- The Tools / Options / Configuration / Modes dialog is shown above. It specifies which providers will be used for each of the three trading modes.

- For Simulation Mode, you cannot set any options. This mode connects your strategy to a simulated market data provider, and to a simulated trade execution provider. Results are stored in the "Simulation" Portfolio (recall that there is a separate portfolio for each trading mode.)

- For Paper Trading mode, you can choose your favorite (live) market data provider. The IDE will connect your strategy to a simulated trade execution provider. Results are stored in the "Paper" Portfolio. In this example, the data provider is IB Interactive Brokers.

- For Live Trading mode, you can choose both the market data provider and the live trade execution provider. In this example, both have been set to IB Interactive Brokers. Results from live trading are stored in the "Live" Portfolio.

7.2 Choose a Trading Mode
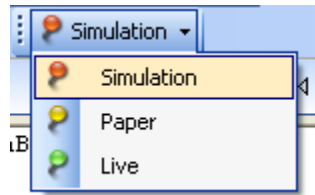
- Choose a trading mode using the Trading Mode toolbar, as shown above. When you select one of the modes, the configuration values (explained above) for that mode are used to connect your strategy to the desired providers and portfolio database.

7.3 Check Provider Connections

- When you switch trading mode to something other than Simulation mode, it is a good practice to check for successful connections to the providers for the newly selected trading mode. That way you can be sure that all connections are working before trying to run your strategy with live data or live trade execution.

- Use the View / Providers and View / Provider Errors menus to show both the Providers pane and the Provider Errors pane. If a provider is properly connected, its icon in the Providers pane will be Green.

7.4 Check for Provider Errors

- Check for Provider Errors even if the provider icon is green. This is because the Green state only covers the provider connection, not the messages or content sent across the connection. For example, if you forget to specify an Exchange property for an instrument, a live execution provider will return an error message saying that it doesn't know where to trade the instrument.

7.5 Check Data Flow with the Quote Monitor

- Once you have defined matching bar sizes, and have successfully connected to a live market data provider during trading hours, check for a flow of market data by using the Quote Monitor to make sure that your strategy has access to input trade and bar data.

- View the Quote Monitor with the View / Quote Monitor menu. Drag and drop the instrument(s) used by your strategy on to the Quote Monitor window. Watch to see if prices change. If they do, you have data flow for your strategy.

7.6 Check Live Execution Provider Functions

- Check that all your instruments have at least a currency and an exchange value specified (perhaps your trade execution provider will require other fields as well).

- Check for a good connection to the live execution provider.

- Check that you can successfully send trading orders to your live execution provider using the Quote Monitor. Right click an instrument line in the Quote Monitor to send a trade order to the execution provider. Some execution providers such as IB Interactive Brokers provide test accounts that are identical to real accounts, except that they do not use real money. These accounts are useful for testing your strategy in a realistic environment.

- Check that you can cancel an order too, if possible.

8 The Scenario Feature

OpenQuant includes a new Scenario feature that enables users to programmatically run strategies in a sequential fashion, under the control of scenario code contained in the *scenario.cs* code file. Each new project created in OpenQuant is created with a skeleton scenario file that simply runs the new strategy. Here is a screenshot showing where you can find the scenario file, and a picture of its default contents.
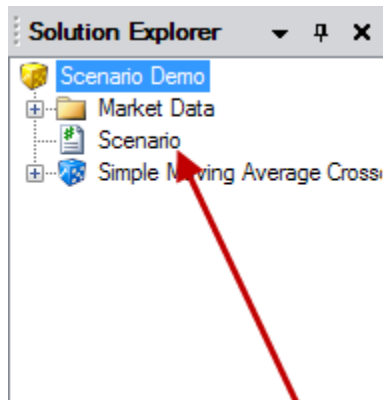
8.1 The Default Scenario File

Here are the default contents of a scenario file. The default code simply starts the current strategy, but you can add more code to do many other useful things.

```
using System;
using OpenQuant.API;
using OpenQuant.API.Engine;

public class MyScenario : Scenario
{
    public override void Run()
    {
        Start();    // run the current strategy
    }
}
```

8.2 The Scenario Demo Solution File

Here is the *scenario.cs* file from the solution Scenario Demo. As you can see from the code, this scenario code does the following things:

- Runs the default strategy solution
- Removes the daily bars and all instruments, and adds 1 minute bars for MSFT
- Changes two parameters, and runs the strategy again
- Prints some PnL (profit/loss) results from the second run
- Changes some strategy parameters again
- Runs the strategy a third time, and prints the results again.

As you can see, the scenario file makes it easy to code complex sequences of strategy runs with different parameters, enabling you to write complete and complex back testing or optimization scenarios for your strategy solutions. Since the scenario file resides at the solution level, it can reference all the projects within the solution.

```
// from Scenario Demo solution
using System;
using System.Collections.Generic;
using System.Text;
using OpenQuant.API;
using OpenQuant.API.Engine;
using OpenQuant.API.Indicators;

public class
MyScenario : Scenario {

public override void
Run () {
    // start solution

    Console.WriteLine ("Running solitiom with default settings");
    Start ();
    Console.WriteLine ();

    // remove Daily request
    Solution.Requests.RemoveDaily ();

    // add 1 min bar request
    Solution.Requests.Add (BarType.Time, 60, true);

    // get project object
    Project project = Solution.Projects[0];

    // clear project instrument list and add MSFT
    project.Instruments.Clear ();
    project.AddInstrument ("MSFT");

    // start the solution a second time, and print statistics
    Console.WriteLine (string.Format (
        "Running solution with sma1 length = {0}, sma2 length = {1}",
        project.Parameters["Length1"].Value,
        project.Parameters["Length2"].Value));
    Start ();

    Console.WriteLine (
```

```csharp
    string.Format ("Portfolio value = {0}, PnL per Trade = {1}",
        Solution.Portfolio.GetValue (),
        Solution.Statistics.PnLPerTrade.ToString ("F2")));


    // Run the solution a third time
    // change project parameters and start the solution again
    Console.WriteLine ();
    project.Parameters["Length1"].Value = 10;
    project.Parameters["Length2"].Value = 28;
    Console.WriteLine (string.Format (
        "Running solution with sma1 length = {0}, sma2 length = {1}",
        project.Parameters["Length1"].Value,
        project.Parameters["Length2"].Value));
    Start ();
    Console.WriteLine (
        string.Format ("Portfolio value = {0}, PnL per Trade = {1}",
            Solution.Portfolio.GetValue (),
            Solution.Statistics.PnLPerTrade.ToString ("F2")));
}}
```

8.3   Scenario File Algorithm Possibilities

Here are some code examples that show possible ways of implementing strategy actions. This code has not been tested, and is provided as a guide to general syntax only.

8.3.1   Configure a Solution Object

```csharp
    // configure solution object
    Solution.StartDate = new DateTime(1995, 1, 1);
    Solution.StopDate = new DateTime(2001, 1, 1);

    Project myproject = Solution.Projects[0];
    myproject.AddInstrument(instrument);
    myproject.Parameters["Length"].Value = 14;
    Start();
```

8.3.2   Do Batch Back Tests
- Read parameter sets from a file
- Loop over all parameter sets
    - Install new parameters
    - Start();
    - Write out results

8.3.3   Do Walk Forward Back Tests
- Loop over calendar intervals within the data set
    - Set up the in-sample interval within the data set
    - Run strategy multiple times to optimize parameters
    - Set the out-of-sample interval within the data set
    - Run the optimized strategy on the out-of-sample data set

8.3.4 **Do Monte Carlo Back Tests**

- Read original data series
- Create and define a derived Monte Carlo (MC) data set
- Loop over all MC data sets
    - o Clear out data in the historical database
    - o Install the MC data set into the historical database
    - o Run the strategy
    - o Store the results
- Build up the MC distribution from all results

8.3.5 **Do Continuous (Additive) Back Tests**

- Set run interval (Date1, Date2)
- Run the strategy
- ResetOnStart = false; // keep existing performance results so far
- Set new parameters
- Set interval (Date2, Date3)
- Run the strategy again // results are added to previous results

8.3.6 **Run Back Test to Load Indicators then Switch to Live Trading**

- Set interval (Date1, Date2);
- Run a simulation strategy (eg, to load up indicators)
- ResetOnStart = false;
- Start (Mode.Live)

8.3.7 **Do Custom Optimization of Strategies**

- Loop over many parameter sets
    - o Install a set of optimization parameters
    - o Run the strategy, then fetch the final result
    - o Objective = Solution.Portfolio.GetValue();
- Choose the best optimized parameters
- Start(); // run strategy one more time with the best parameters

8.3.8 **Produce Custom Report Outputs**

- Run the strategy
- Write out interesting results:
    - o Solution.Portfolio.GetValue()
    - o Statistics.AnnualReturn
    - o Statistics.Duration
    - o Statistics.FinalWealth
    - o Statistics.LongTrades
    - o Statistics.LongTradesPnL

8.4 Scenario File Code Examples

8.4.1 Select and Load Liquid Instruments

```csharp
// get reference to strategy project
Project project = Solution.Projects[0];

// clear project instrument list
project.ClearInstruments();

// add most liquid stocks to instrument list
foreach (Instrument instrument in InstrumentManager.Instruments) {
    if (instrument.Type == InstrumentType.Stock) {
        BarSeries series = DataManager.GetHistoricalBars(instrument, BarType.Time,
86400);
        if (series.Count != 0 && series.Last.Volume > 50000000) {
            Console.WriteLine("Adding " + instrument);
            project.AddInstrument(instrument);
        }
    }
}


// start backtest
Start();
```

8.4.2 A Brute Force Optimization Loop

```csharp
 // set in-sample data interval
Solution.StartDate = new DateTime(1995, 1, 1);
Solution.StopDate = new DateTime(2001, 1, 1);

// get reference to strategy project
Project project = Solution.Projects[0];

// define variables
int best_length1 = 0;
int best_length2 = 0;
double best_objective = 0;

// brute force optimization loop
for (int length1 = 3;length1 <= 7;length1++) {
    for (int length2 = 3;length2 <= 7;length2++) {
        if (length2 > length1) {
            // set new parameters
            project.Parameters["Length1"].Value = length1;
            project.Parameters["Length2"].Value = length2;
            // print parameters
            Console.Write("Length1 = " + length1 + " Length2 = " + length2);
            // start backtest
            Start();
            // calculate objective function
            double objective = Solution.Portfolio.GetValue();
            // print objective
            Console.WriteLine(" Objective = " + objective);
            // check best objective
            if (objective > best_objective) {
                best_objective = objective;
                best_length1 = length1;
                best_length2 = length2;
```

```
                    }
                }
            }
        }

        // print best parameters
        Console.WriteLine("BEST PARAMETERS");
        Console.WriteLine();
        Console.WriteLine("SMA1 Length = " + best_length1);
        Console.WriteLine("SMA2 Length = " + best_length2);
        Console.WriteLine("Objective = " + best_objective);

        // run strategy with the best parameters on out-of-sample data interval
        project.Parameters["Length1"].Value = best_length1;
        project.Parameters["Length2"].Value = best_length2;
        Solution.StartDate = new DateTime(2001, 1, 1);
        Solution.StopDate = new DateTime(2005, 1, 1);
        Start();
```

8.4.3   Run Simulation on Historical Data and Switch to Live

```
        // first run simulation from historical data up to "now"
        this.Solution.StartDate = new DateTime(1996, 1, 1);
        this.Solution.StopDate = Clock.Now;
        Start(StrategyMode.Simulation);

        // make sure all objects: portfolios, orders,
        // indicators and strategy state remain the same
        // on next strategy start
        this.ResetOnStart = false;

        // now start solution in Live mode
        Start(StrategyMode.Live);
```

9   OpenQuant and Visual Studio

Some people prefer to use Visual Studio for their main editing environment, especially if they use a external library source files to support their strategies. This section shows how you can combine OpenQuant with Visual Studio to improve your strategy development environment.

This section only shows how to edit files in Visual Studio. You cannot use the VS debugger to debug your strategies that are run under the OpenQuant IDE, because the code is being run under the IDE, and not through Visual Studio.

9.1   Edit Strategy Files in Visual Studio

The main technique to use is to create a VS project file, and then add your code files to the VS project—*being careful to add your code files as links to the VS project.*

This approach keeps the physical strategy file where it must reside within the SmartQuant folder hierarchy (C:\Program Files\SmartQuant Ltd\OpenQuant\Strategies\ Stock 2% Gap*.cs). But it also allows the code file to be part of a VS project.

To link an OpenQuant code file to a Visual Studio project, follow these steps:

- Create a new strategy project using the IDE. The new strategy project should be located within the OpenQuant file system (see the pathname above). If you can find your new strategy code file, you have a strategy code file to edit.

- Create a new VS project and add DLL references to the OpenQuant DLL libaries that you need (located at C:\Program Files\SmartQuant Ltd\OpenQuant\Bin*.dll). Now you have a VS project that can include the code file that you created in the previous step.

- Right click the VS project (or a folder within the project), choose Add -> Existing Item, and browse to your OpenQuant strategy code file in the OpenQuant folder hierarchy (e.g. C:\Program Files\SmartQuant Ltd\OpenQuant\Strategies\ Stock 2% Gap*.cs). Now we come to the key action that makes this approach work.

- Instead of clicking the "Add" button in the file browsing dialog in the previous step, be careful to click the tiny little arrow beside the Add button, and select "Add as Link." Here is a picture of what the button looks like, indicated by the red arrow.
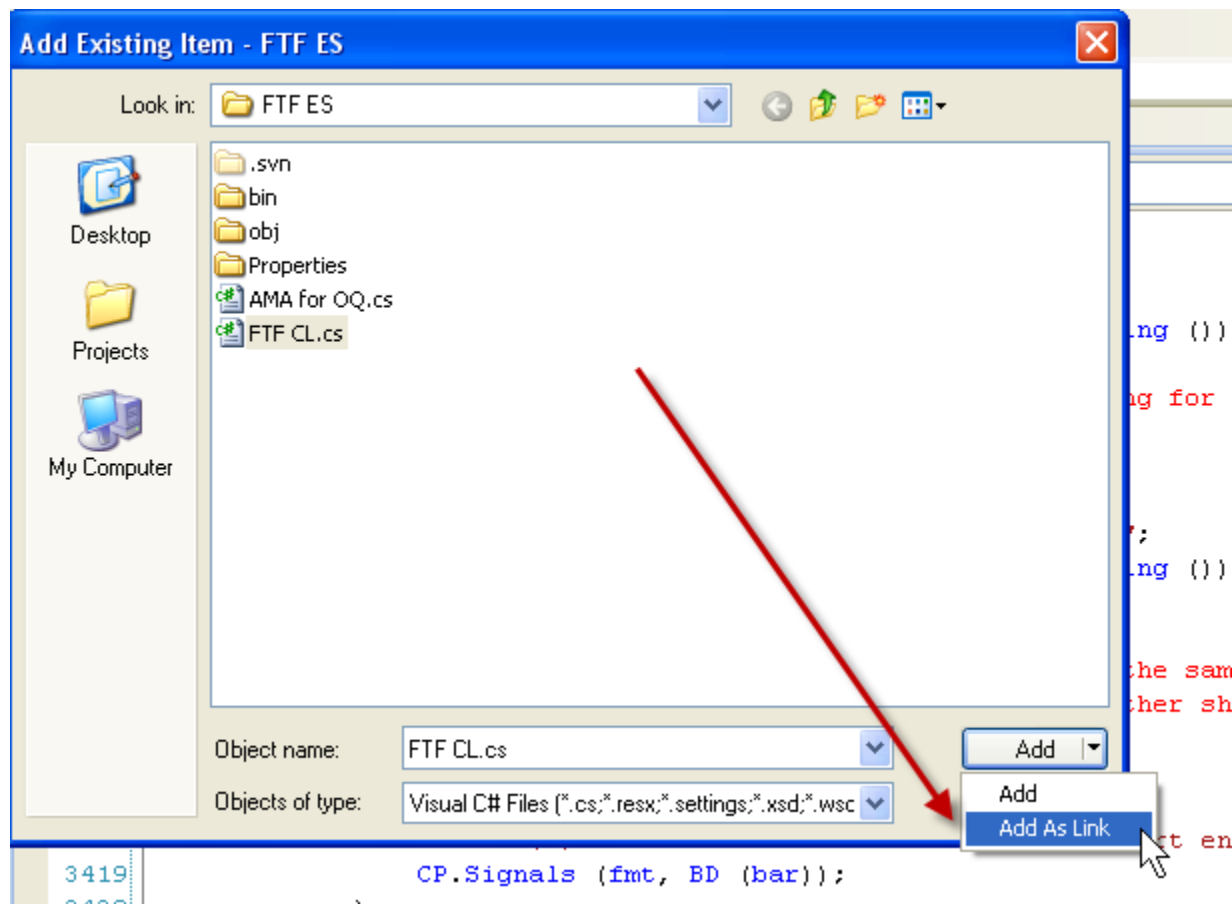
- The dialog shown above is from Visual Studio (VS). The contents of the dialog are OpenQuant code files that I want to add to my VS project.

- After you add an OpenQuant file to the VS project as a link, you can edit the file in the VS editor, then type Alt-TAB to switch from VS to OpenQuant, and run the strategy code that you just edited. The process works quite well, without too much hassle. You get to use all your favorite VS editor key bindings and editing power, at the minor cost of a few ALT-TAB key sequences to flip back and forth between applications.

10 Conclusion

That's all for this Getting Started Guide document. Now you can spend many happy hours writing and running profitable trading strategies. It's certainly a good feeling when you see good results from a computerized trading strategy simulation. See the Strategy Development Manual for some introductory information on strategy writing.

And because this is a Getting Started Guide, I can offer this last piece of common sense advice—before switching to LIVE trading with real money, it's probably a good idea to spend lots and lots of time back-testing your strategies using multiple different time periods, using multiple different instruments, and using multiple different stress-testing scenarios (like the crash of 1987 (Dow down 23% in ONE DAY), the recessions of 1990-1991, the dotcom rise and fall of 1998-2003, the recovery until 2007, and the recent market corrections (Jan 2008 – Apr 2008).

Finally, if you find out later that you need more strategy development power than is provided by OpenQuant, more powerful products are mentioned on the SmartQuant web site.

11 Revision Notes
- 2009-10-15 – Initial version created from previous OpenQuant2 version.
- 2011-07-26 – Added doc and code examples for scenario.cs file