

Trading Blox User's Guide

© 2021, Trading Blox, LLC. All rights reserved.

Build: 5.4.x



Trading Blox User's Guide

By Traders... For Traders

Trading Software for Mechanical Systems Traders

© 2021, Trading Blox, LLC. All rights reserved.

*This user's guide and all its contents are copyright
2003-2021, Trading Blox LLC*

Trading Blox LLC
www.tradingblox.com
978-772-2620
customersupport@tradingblox.com

Table of Contents

Foreword	0
Part I Welcome	1
Part II Using Trading Blox	4
1 Operating Overview	7
2 Installing and Using Trading Blox	10
Part III Main Screen Overview	13
1 Main Menu Groups	16
File Menu Groups	19
Home Menu Groups	23
Chart Menu Group	34
Results Menu Group	41
Tools Menu Group	46
Help Menu Group	53
Quick Access Tool Bar	57
2 Suite Selector List	60
3 System Selector List	64
4 Simulation Parameter Editor	67
5 Log Window	73
6 Auxiliary Window	77
7 System Information	82
8 Trade Chart	86
Part IV Global Suite Parameters	98
1 Allocation Manager	100
2 Simulation Parameters	102
Entry Day Retracement	116
Reducing Optimization Time	118
3 Walk Forward Parameters	131
Walk Forward Testing	133
4 Futures Parameters	144
Slippage Percentage	148
5 Stock Parameters	151
6 Forex Parameters	156
7 Broker Direct	157
Broker Selection Settings	160
8 Equity Manager	167
9 Fees Parameters	174
10 Capital Adds Draws	176

Part V Preferences Editor Menu	180
1 Data Folder and Data Options	183
2 General	193
3 Monte Carlo	199
4 Single Test Reporting	201
5 Small Simulation Reporting	203
6 Large Simulation Reporting	205
7 Trade Chart Colors	208
8 Trade Details	210
9 Blox Editor Colors	213
10 Changing Default Settings	216
11 Trading Blox Parameter Settings	221
Part VI System Editor Information	233
1 System Editing	236
Print System Scripts	244
2 System Blox Modules	250
3 Available Blox Modules	254
Part VII Blox Basic Editor	258
Part VIII Historical Data Setup	275
1 Forex Dictionary Editor	291
Forex Editing Details	294
Interest Rate Files	295
Forex Symbol List	296
2 Futures Dictionary Editor	297
Dictionary Controls	299
Market Specifications	301
Symbol Data Details	306
3 Stock Dictionary	310
CSI_Stock-Symbol_Num	314
Part IX Portfolio Editor	317
1 Sample Portfolios	325
Part X Historical Testing	328
1 Parameter Stepping	331
2 Specifying the Test Dates	335
3 Order Filter	336
Part XI Reporting Information	338
1 Graphical Reporting	341

2 Monte Carlo Reporting	343
3 Text Reporting	345
4 Trade and Equity Logs	346
Part XII Importing-Exporting Suites	348
1 Importing Suites	350
2 Exporting Suites	353
Part XIII Summary Results	355
Part XIV Test Results	360
1 Stepped Parameter Performance	363
2 Multi-parameter Surface Chart	365
3 Parameter Stepping Graph	367
4 Log Equity Graph	369
5 Linear Equity Drawdown Graph	370
6 Monthly Returns Graph	372
7 Monthly Returns Distribution Graph	373
8 Total Risk Profile Graph	374
9 R-Multiple Distribution Graph	376
10 R-Multiple Contribution Graph	377
11 Yearly Performance Summary	378
12 System Parameter Settings	380
13 Instrument Performance Summary	384
14 Trading Performance and Statistics	386
Part XV Log Files	394
1 Trade Log	396
2 FilteredTradeLog	398
3 Daily Equity Log	399
4 Monthly Equity Log	400
5 PrintOutput.csv	401
Part XVI Monte Carlo Simulations	402
1 Equity Curve Graph	407
2 Returns Graph	408
3 Sharpe Graph	409
4 MAR Graph	410
5 R Squared Graph	411
6 Drawdown Graph	412
7 Drawdown Length Graph	413

8 Confidence Level Report	414

Part XVII Built-in Systems and Blox 415

1 Money Managers	417
2 Risk Managers	419
Correlation Risk Manager	421
Correlation Risk Manager - Check Fills	423
Group Risk Manager	426
Total Risk Manager	428
Unit Limiter	432
3 Portfolio Managers	434
4 Chandelier Exit	436
5 Gap Against Exit	437
6 ADX System	438
7 ATR Channel Breakout System	440
8 Bollinger Breakout System	443
9 Bollinger Counter-Trend System	446
10 Donchian System	451
11 Dual Moving Average System	453
12 MACD System	456
13 RSI Trend Catcher System	458
14 Stochastic System	461
15 Triple Moving Average System	466
16 Turtle System Rules	470
17 Walk Forward	481
18 Stats for Multi Parameter Charts	492
19 Top Ranked Stocks	493
20 Virtual System	494

Part XVIII Generating Orders 495

1 Broker Positions	499

Part XIX Quick Charts 503

Part XX Upgrade Changes 507

Part XXI Trouble Shooting 519

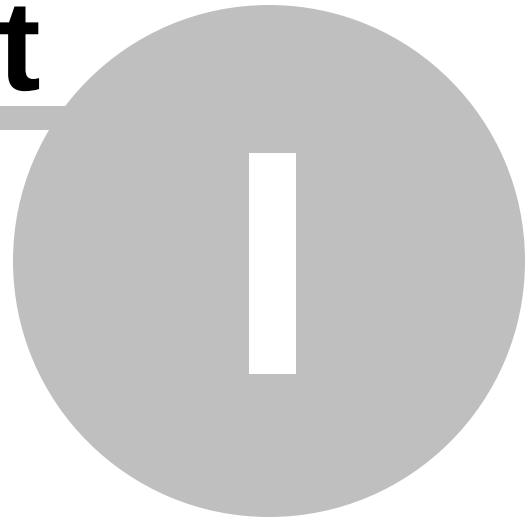
Part XXII Appendices 521

1 Backups	524
2 Batch Job	527
3 Configuration settings	534
4 Data Providers	540

5 Determining Big Point Value	541
6 Error Log and Output File	543
7 Folders	544
8 Forex Carry Calculations	545
9 Forex Dictionary Format	547
10 Futures Dictionary Format	548
11 Glossary of Terms	550
12 Moving the License	555
13 Portfolio File Format	556
14 Risk and Return	558
15 System Component Table	561
16 Technical Support	562
 Index	 563

Welcome

Part



Part 1 – Welcome

Trading Blox User's Guide

Program Version: 5.4.x

Help Version: Wednesday, March 16, 2022

The Trading Blox team brings together a unique blend of world-class software design talent and real-world experience in trading, system design, testing, and implementation. We are committed to developing software to help others fulfill their commitment to become better traders.

Trading Blox was conceived by traders for traders based on our observation that today's trading system software does not adequately address the needs of the user community at large. This is true not only for discretionary traders who are interested in learning to trade systematically, but also for systematic traders who don't have the tools at their disposal to help them sharpen their craft and aren't sure which way to turn.

Today's publicly available back testing software falls into two distinct categories. The first type offers a slew of indicators and advanced graphic capabilities, but only allows testing on a single stock or commodity at a time, and requires expensive add-on software to test across an entire portfolio.

Even then, this type of software is cumbersome and limited at best. It is limited most of all in its portfolio testing method because it doesn't allow for the influence of how a system might need to respond to changes in equity or performance at the end of each date, and before it sends orders to be tested in the next trade day's data. In real trading make portfolio and order decision each day. Add-on optional portfolio testing options blend equity from the tested results of each instrument in an attempt to simulate portfolio performance. This isn't a realistic approach most of the time when it is common to make decisions on which symbol will be allowed to trade and in which order it will be sequenced in analysis. There are many more difference between how Trading Blox test than how optional portfolio methods can report realistic portfolio results.

At the other end of the spectrum is software that allows portfolio-level back testing with money management, but which assumes significant programming skills on behalf of the user and only offers flexibility at the price of a steep learning curve. In short, with these products comprehensive testing is possible but painful.

This leaves a sizable gap between what trading system software does today, and what is truly possible. Let's face it: Until now, the ability to conduct comprehensive, full-featured back tests has been the exclusive province of a relatively small group of die-hard programmers and large-scale investment houses.

It doesn't have to be that way.

The team at Trading Blox is uniquely positioned to change this situation, and with the release of Trading Blox Builder II, Builder I, and Professional we've done just that. By making the Trading Blox technology available publicly, others will now have the opportunity to quickly develop a better understanding of how trading systems work in depth, in detail, and in the real world.

Trading Blox is world-class trading simulation technology that provides traders of all skill levels with access to an extraordinarily flexible and highly automated testing environment. By coupling a comprehensive, programming-free testing environment with an intuitive graphical interface, Trading Blox dramatically flattens the steep learning curve traditionally associated with trading system development. It helps the trader quickly gain the necessary level of confidence to successfully deploy and consistently trade a system.

While Trading Blox bridges the gap in existing offerings, it also provides capabilities that are not available in any other software. For example, you can easily instruct Trading Blox to test the pyramiding techniques and risk management rules that are an integral part of the Turtle Trading Rules, without writing a line of code. You can test a triple moving average system with sophisticated money management without a line of code. You can optimize those systems for different portfolios without a line of code.

At Trading Blox, our goal is to ensure that each and every user is completely satisfied with their entire Trading Blox experience, from purchase through trading. Moreover, we look forward to establishing long-term relationships with interested users, because it is from this group that many great ideas for future releases sprout. We value your feedback!

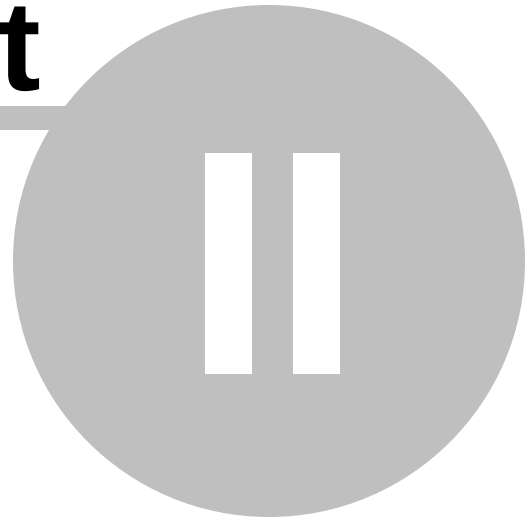
In addition to placing a high value on user input, everything we do is driven by the belief that quality is paramount. Every release is rigorously tested to ensure that every trade that should be taken is taken, trades which should not be taken are ignored, and that every single calculation is accurately performed and reflected in the results, down to the last penny.

If we can't produce quality software that helps traders become better traders, then we have no business being in business. With that said, we believe that the future looks very bright for you and all of us at the Trading Blox family!

The [Product Feature Page](#) explains the abilities of each of the Trading Blox product versions.

Using Trading Blox

Part



Part 2 – Using Trading Blox

Trading Blox is Windows-based software designed for those who want to use a realistic set of trading rules in a testing approach to determine entry and exit timing over range of dates, and long market history. In addition to the entry and exit timing, the control of which instruments, and their order and active position risk can also be discovered and evaluated.

Trading rules are as a set of scripted statements that create the orders and perform the other tasks a trader believes will generate a favorable result. When trading rules are written and saved, they create a way to test those trading ideas consistently for each symbol in the portfolio, and on each test in the range of dates the user determines.

Trading rules are the heart of a system. Systems are run from within a suite of a single system, or a group of many systems. Multiple system suites will produce a combined system performance report. During the testing of a multiple-system suite, the ability to enable all or just some of the systems to better understand which systems are helping, and which systems might need enhancements or removal.

Each system executing in a suite determines the trades that the rules in the system would have generated for each of the dates and each of the symbols in the portfolio. At the end of testing, all the results of those orders to enter and exit are calculated, summed and presented in an organized report. A long list of statistic can be enabled to show the viability of the suite's test results.

Test results can just create a Performance Summary, and, or create a list of active positions and new orders for the next trade day.

Trading Blox suites with multiple systems execute all the systems and its rules for the same date before it executes the next date. This test approach means all the system test all the symbols in its portfolio on the same test date. When all the systems in the suite have finished testing all the symbol on the same test date, the suite process allows the user to analyze the results of each system's testing for that date and make changes in any orders that each system created. This post same date testing and after order generation review of results is how we make decisions is deciding if we want to allow the collection of orders to be changed before they are allowed to be executed.

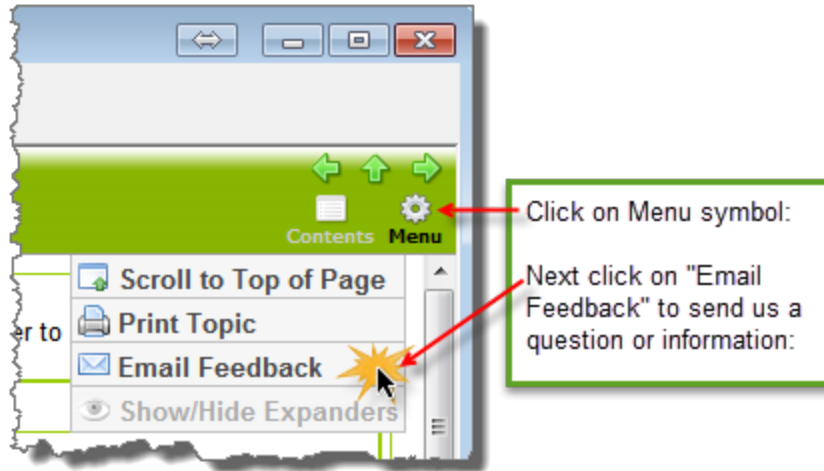
Systems can easily test symbols in each of the instruments classes of Stocks, ETF, Mutual Funds, Futures and Forex. Each system in a suite can process all the same class of instrument, or each system can trade a different class. Each system has a portfolio that contains the same or a different class of instruments.

Trading Blox is an unusually capable simulation engine capable of creating test results in the same way we would trade in real time. There are many option settings available that are explained in other sections of this User's Guide.

This Trading Blox user guide is in a period of significant change to accommodate all the features made available in version 5.4.x. Should you have a question or find an area that doesn't seem correct, please post your question in the Trading Blox Trader's Roundtable Forum. All Trading

Blox users can get access to the Trading Blox support section of the forum that is not made visible to public forum users that do not use Trading Blox.

An alternate method for sending feedback is to use the Email Option:

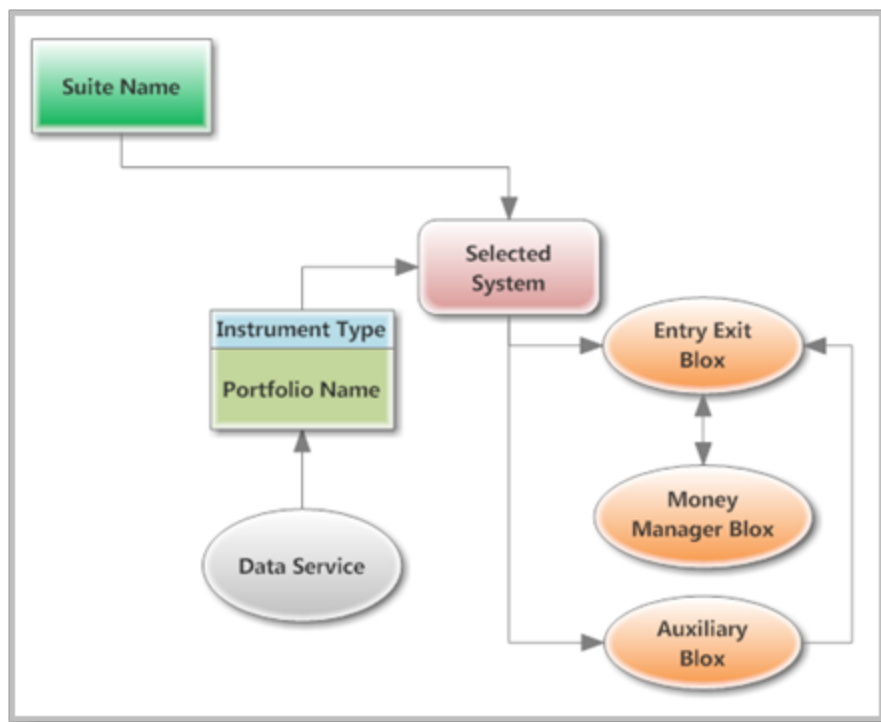


Edit Time: 9/13/2020 1:49:44 PM

Topic ID#: 233

Section 1 – Operating Overview

Trading Blox system construction begins when an existing suite is edited or a new suite is created. Suites can have one or more suites and they can also have their own Blox. Suites are the foundation component that allows systems to be tested. Suites can have a single system, or it can have multiple systems. When multiple systems are attached to a suite each system will be tested in alphabetical order of each system name selected to be tested in the selected suite name. To change the order in which systems are selected before the other systems, just change the names so the systems arrange their order when the system names are sorted alphabetical order.



Example of Suite to a simple System assigned Blox names

In the image example shown above, the structure shows a simple outline of the basic components that can be part of a system listing.

The top of a test is the suite name. Below that level there must be at least one system name. In the system structure there must be at least one blox name and a portfolio name. Other blox names can be added, and when a trading idea is to be tested there are other requirements (see below) . Trading idea examples available for testing are included during [installation](#). There are no practical limits on the number of suite or system names that can be added to Trading Blox.

Within a Suite, the selected system name(s) determines the testing process. Suite names with a selected system can be edited so that a different system or a group of systems can be tested. System names can also have their selection of blox changed so that edited system performs a test in a different way.

When working with one or more system names in a suite name, each system must have a portfolio name selected. If the test is to create results of a trading idea, that system must also have an amount of equity available to support the orders, and there must be a Money Manager

selected to size the positions when the orders are filled. Systems not intended to trade will still need access to a portfolio, which can be just one, or multiple symbols, but all systems used to generate after test performance must have a Money Manager.

Portfolio names determine what symbols to test for the after test information report. Trading Blox allows portfolios for three different instrument classes. Each instrument class supported allows systems to test different trade ideas for Futures, Forex, and Stock type instruments like Stocks, ETF and Mutual Funds. When multiple systems are selected in a suite, each system in a suite can test a different type of instruments using trading ideas for that type of instrument class. This also means when there are three systems selected in a suite name, each system can perform a test in each of three different instrument classes available.

When a system name is created, or edited, there is usually a trading idea in mind. When this idea is to perform trading results, the blox selected in a system will need to have blox names that can perform the task expected and can support some phase of the trading idea. To help the user get started with understanding how to test trading ideas, Trading Blox installs a sample suite name and a group of system names that represent some of the popular trading ideas. Each of the well-known trading systems installed with Trading Blox will use a different trading idea in its testing. Some trading ideas are more suited to one type of instrument class than another instrument class. All the supplied systems can be modified, and all will allow their blox to be viewed and edited.

Trading System Construction:

Each supplied system can be used as installed, or it can be changed and adjusted as needed. For example, each system's default parameters can be changed, or other Blox methods can be added or changed. Parameter values can be automatically changed when the Step option is made available near the parameter's entry field, or in the drop-down list. When Step is enabled the user can enter a starting and ending value and a value of how much to change on each step.

Once a particular combination of parameter values has been tested and found to produce acceptable results that system can be traded so the orders it generates can be used with a broker. Orders are generated when "Generate Orders" button on the main screen is clicked, or when the F7 key is pressed. With the click of the mouse, Trading Blox will generate a report of the current active positions and the next day's orders.

If any of a system's input parameters are changed those changed values will appear the next time that system is selected. Parameter setting values that appear at the time of a test are the values used to control how the system operates. However, some of these parameters have nothing to do with the system's design rules. For example, the test start and end date values entered, or the list of markets in the selected portfolio. Most other menu displayed parameters affect how the system works, such as the number of days used to calculate a moving average. Global parameters can effect a system performance result. For example, commission and slippage can reduce the profits and increase trading cost. Margin limits can limit how many trades are allowed, and amount of equity available can cause some orders to be rejected. Global parameters are saved with the Suite file name when a different suite name is selected or your exit the software.

At Trading Blox, we understand that any trading system is only as good as the consistency with which it is applied. By harnessing the speed and reliability of modern-day computer hardware,

Trading Blox significantly reduces the effort required to test and to trade a system, while eliminating the possibility of human error and missed trades. And therein lies its greatest strength: Trading Blox makes it easier to follow the trading systems you develop. At the heart of Trading Blox ability to help you build a successful trading strategy is its ability to test a large portfolio of instrument in a single test.

Supplied Suite:

- o Test Suite

Installed Trading Systems:

- o ADX System
 - o ATR Channel Breakout
 - o Bollinger Breakout
 - o Bollinger Counter Trend
 - o Chart
 - o Donchian w Rule Labels
 - o Donchian
 - o Dual Moving Average
 - o MACD System
 - o RSI Trend Catcher
 - o Stochastic System
 - o Triple Moving Average
 - o Turtle
-

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 180

Section 2 – Installing and Using Trading Blox

System Requirements:

Trading Blox is designed and tested to perform within any of the standard Windows operating systems like Windows 7, 8 and Windows 10. Trading Blox also runs with the Parallels, Bootcamp, or Virtual Box software on Intel based Macintosh computers. No additional software or equipment is needed other than the Virtualization Software and Microsoft Operating System. Trading Blox will run in a 32 bit operating system computer, and it can also run 32 bit and 64 bit versions in a x64 computer.

To run Trading Blox in a Intel based Mac, be sure to install Parallels or Bootcamp with your choice of Windows Operating System. Trading Blox can then be install in the installed Parallels or Bootcamp virtual windows environment.

A minimum screen resolution 1024 x 768 or larger is required at 96dpi. If using a larger size, such as 125% or 150% with a Windows larger screen resolution will be required. A minimum of 2 GB of RAM and a 2 GHz processor speed are recommended for normal testing. If you are testing with large stock portfolios or huge amounts of intraday data, we recommend at least 8 GB or more of RAM and a quad core x64 computer running the native x64 version of Trading Blox.

Trading Blox's Parameter Stepping feature allows for the concurrent testing of multiple distinct parameter value combinations. These kinds of tests finish significantly faster on high-performance machines. Ideally a machine would have 4 GB or more of RAM and a processor speed well over 3 GHz. Because Trading Blox uses is multi-threaded testing when it is performing stepped optimization test, a quad core or more computer is quite handy for large stepped tests.

Trading Blox is very fast, and it will run reasonably well on an older machines. However, the new version 5.x does not support XP or Vista.

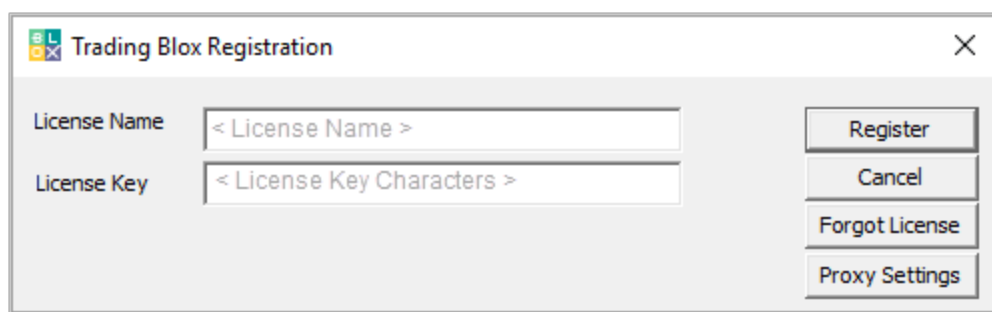
Installation:

Once a customer has purchased the software, they will automatically receive an email with a link to the download location, and the license keys required to install the software. The install takes a few minutes, and is completely automated. Trading Blox customers may download the Trading Blox installer from the [Trading Blox Web Site](#).

Clicking on the download link in the email will start the download of the installer. The installer will run when the download is completed, and the installation process is automatic by default. Once the installation is completed, the customer enters their License Name and License Key from email, and the program is up and running.

Trading Blox can be launched by clicking on the Start button to bring up the Start Menu, go to Programs | Trading Blox | Trading Blox.

When Trading Blox is first installed it will be necessary for the software to reach the internet servers at Trading Blox so the software can validate the license key code.



Registration Dialog before Proxy Area is Opened

Most installation that do not use a proxy server to reach the internet. When this is how you access the internet, there won't be any additional changes required to get the software key validate dialog to appear where the License Name and the License Key sent at the time of purchase can be entered into the above dialog when the software is run the first time.

When entering the License Name, be sure it is exactly how it is listed in the email. License name is shown just above the License Key character series in the email Trading Blox sent right after purchased. When License name is entered, the License Key must be entered. License Keys are a long series of characters. To ensure the key information is enter without any errors, make a copy License Key character series shown in the email. Then paste the copy key characters into the License Key field.

For most users the next step is to click on the "Register" button so Trading Blox can validate the license information.

If an error happens, it can be caused by one of three. License Name is different that the purchase name, there is a character missing or too many in the License Key field, or your internet connection requires proxy setting information so the network connected to the internet can allow Trading Blox to reach the internet.

If none of the above common problem areas is the issue, it is likely you need to also add Proxy information so the network will allow Trading Blox to pass through its security.

Proxy Settings:

When a Proxy server is being used for security, controlling which applications are allowed to access the internet, or for other reasons, it is likely that your network will need proxy information for internet access.

Proxy information details will be available from your network manager.

Required Information:	Replace These Values Proxy Server Values:
proxyServer	myProxyServerName This is the name administrators have given to your proxy server.
proxyPort	myProxyPortNumber

Required Information:	Replace These Values Proxy Server Values:
	This is an integer value that is assigned to the proxy server port made available to allow access.
proxyUserName	myProxyUserName Each user is usually given a different User-Name.
proxyPassword	myProxyPassword Each user is usually given a unique password to obtain access.

After the License Name and License Key fields show your license information, click on the Proxy Settings button to open the License Registration Dialog so the Proxy Setting fields become available. Entering the details are simple and they will be needed prior to being able to use Trading Blox if your network is controlled by a proxy server.

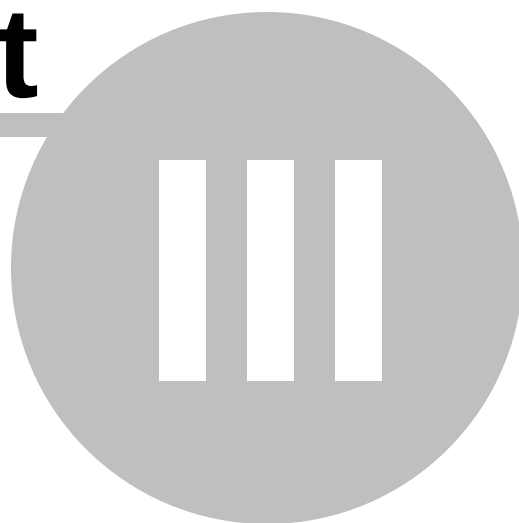
The screenshot shows the 'Trading Blox Registration' dialog box. It has a title bar with a close button (X). Inside, there are two input fields: 'License Name' with a placeholder '< License Name >' and 'License Key' with a placeholder '< License Key Characters >'. To the right of these fields are three buttons: 'Register', 'Cancel', and 'Forgot License'. Below these is a 'Proxy Settings' button, which is highlighted with a dashed border. The 'Proxy Settings' section is expanded, showing four input fields: 'Server' with placeholder '< myProxyServerName >', 'Port' with placeholder '< myProxyPortNumber >', 'User Name' with placeholder '< myProxyUserName >', and 'Password' with placeholder '< myProxyPassword >'.

Registration Key Dialog with Proxy Area Opened

Once the correct proxy information for you location is entered, Trading Blox should not have any trouble accessing the internet and validating the license key code.

Main Screen Overview

Part



Part 3 – Main Screen Overview

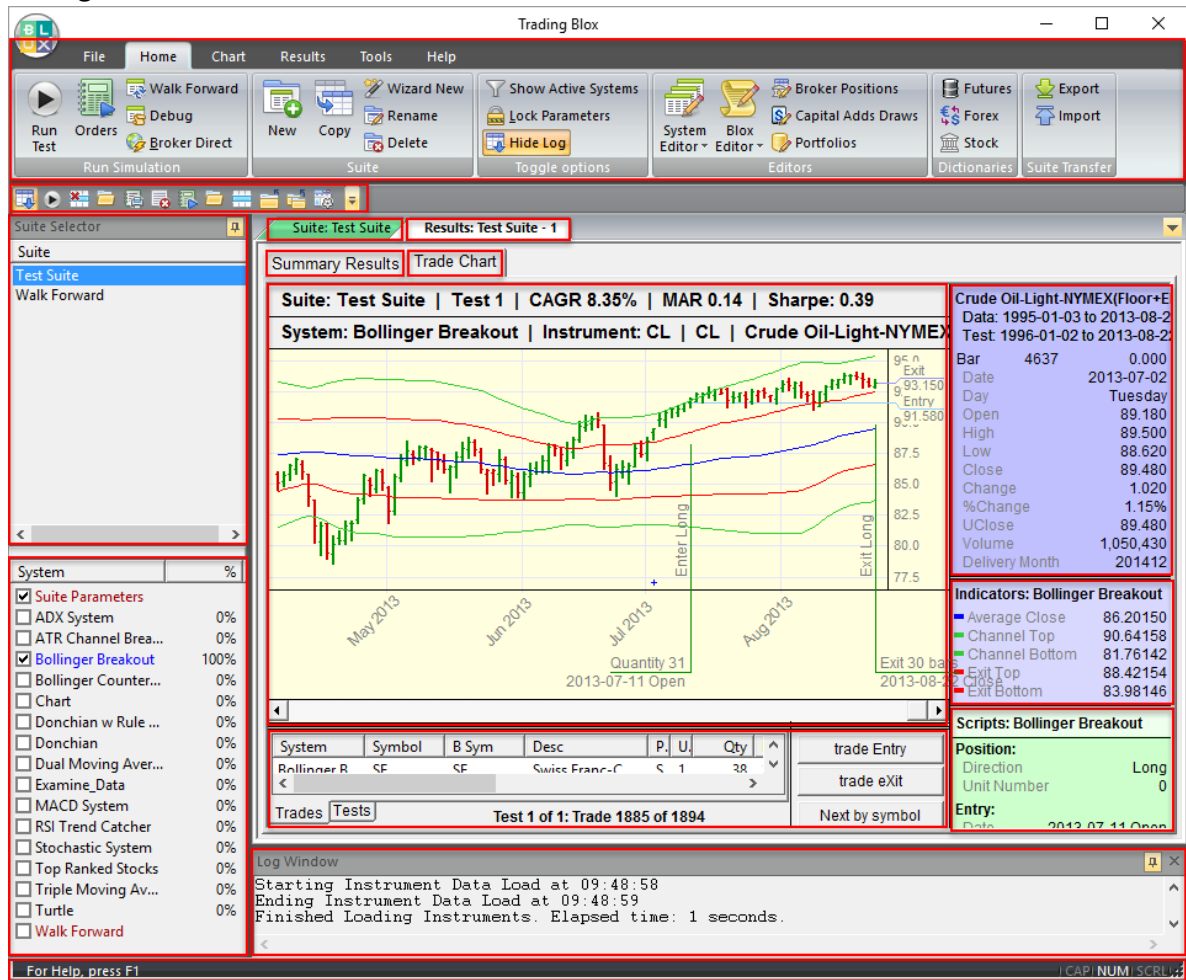
Trading Blox uses sophisticated graphical elements to allow users to easily test complex systems. Tests are performed by selecting or creating a new suite name. A variety of systems are installed during setup and can be used as a stand alone method, or with other systems in the same suite. Custom systems can be created by changing the list of blox modules assigned to a system. They can also be created or changed by creating new blox modules to use an idea not available in the provided strategies.

Systems are tested by assigning them to an existing or new suite. Suites can use systems that are designed to trade various instrument classes. For example, a stock trading system and a Futures trading system can both be in the same suite at the same time. Once a strategy has been tested and found acceptable, it can be used to generate orders for the next trading day.

When a strategy is tested, Trading Blox will generate a Summary Performance Report along with the selected reporting enabled in the [General Reporting](#) section of the program's [Preferences](#).

Testing and Order Generation are both controlled from Trading Blox main screen interface. This main screen area has seven areas where various aspects of the program can be accessed, controlled, and display test results. This next image is the software's main program menu and setting area. Each area outlined with a red box is a link to more information about that area. As the mouse moves over the image it will darken slightly and highlight the area where an active link is available. Clicking on a highlighted section will display a page where more information about that section is available.

- Image section areas are links to more information:



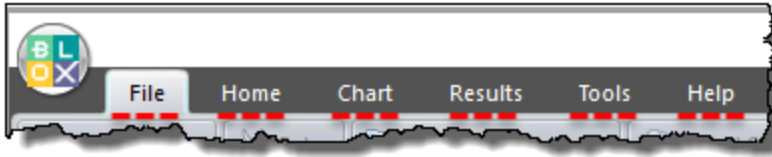
Trading Blox Main Screen with chart test information display.

Links:

[Main Menu Groups](#), [Active Suite and Test Results Area](#), [Auxiliary Window Output](#), [Suite Selector List](#), [System Selector List](#)

Section 1 – Main Menu Groups

Trading Blox version 5.x has simplified access to most of its main menu options by grouping them into ribbon styled group menus. There are six menu ribbon groups that are available. Any group can be displayed by clicking on one of the Top level ribbon tab controls:

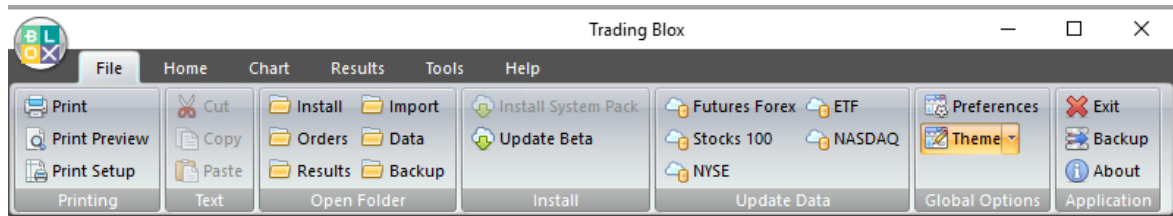


Main Screen Menu Access Tabs



Easy Access to user selectable menu controls.

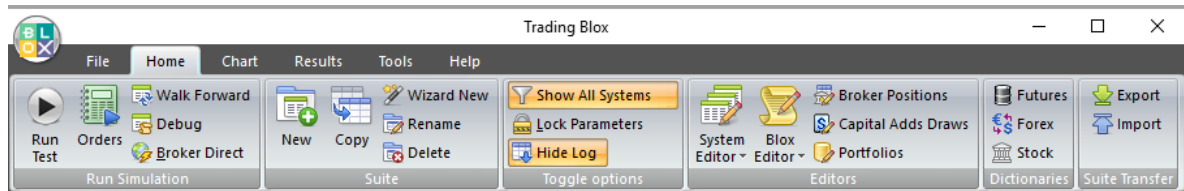
FILE Menu Controls: [^Top](#)



File Menu Control Group Areas

This group menu controls allows access to Trading Blox controls for printing, software folder access, updating data, software preferences, display theme colors, an installation backup, and information about Trading Blox.

HOME Menu Controls: [^Top](#)



Home Menu Control Group Areas

This group is the area where test and order simulations are executed and reported. Most user are likely to need the controls in this controls set.

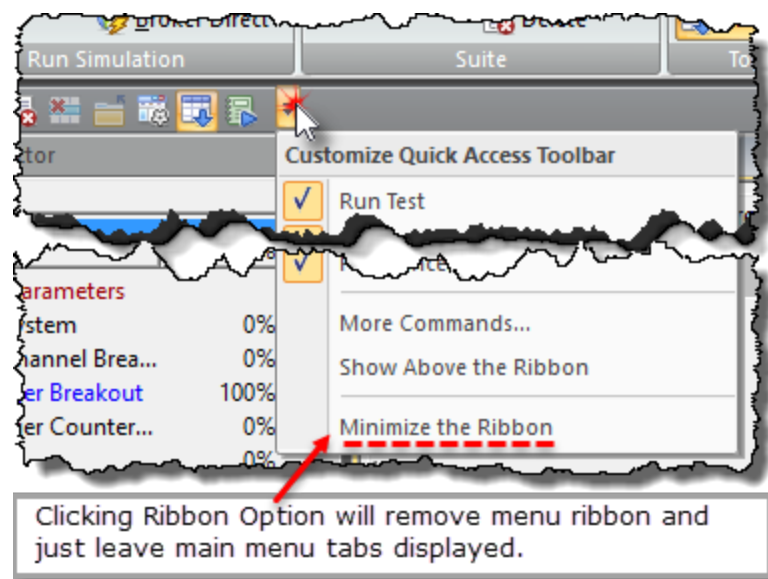
QUICK ACCESS Menu Icons: [^Top](#)



Quick Access Menu Area

Quick access is a user customizable tool bar. Access to this toolbar is available by clicking on the dark area's downward pointed triangle.

QUICK ACCESS Menu Icons: [^Top](#)



Click on toolbar's [Customize Button or Menu](#) image see more information.

CHART Menu Controls: [^Top](#)

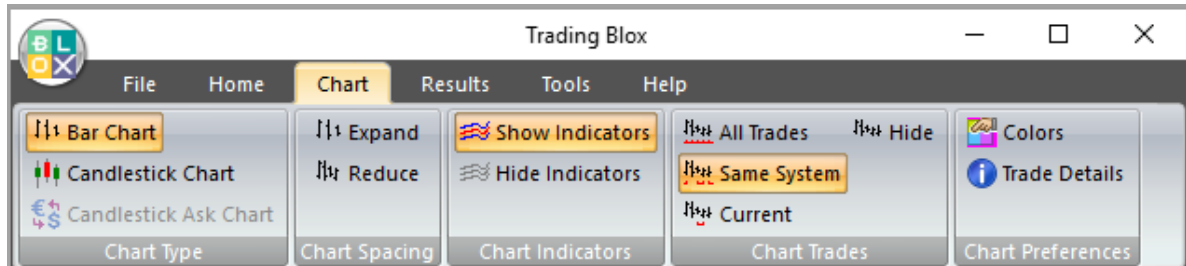
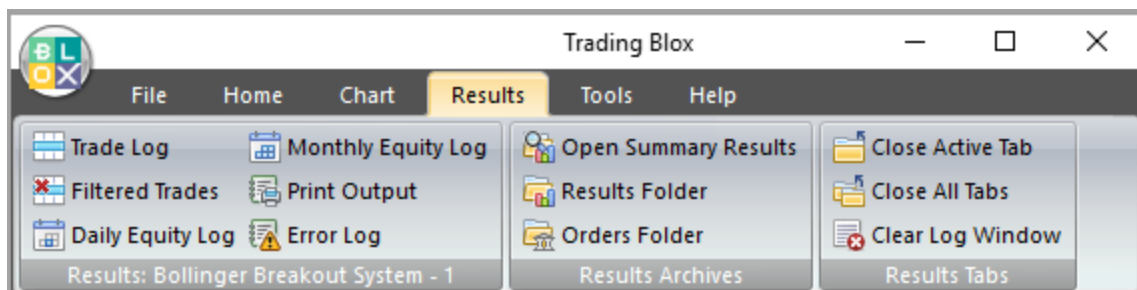


Chart Menu Control Group

This group is the area where after test charting and trades are displayed. It is also the most user controls that will be needed while a test report is reviewed.

RESULTS Menu Controls: [^Top](#)

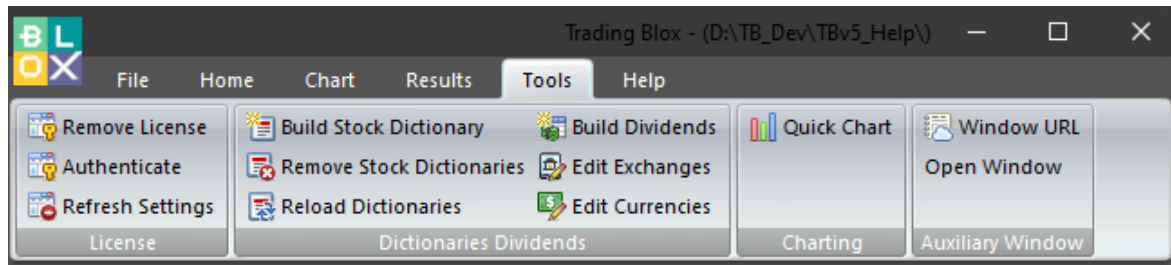


Results Menu Control Group

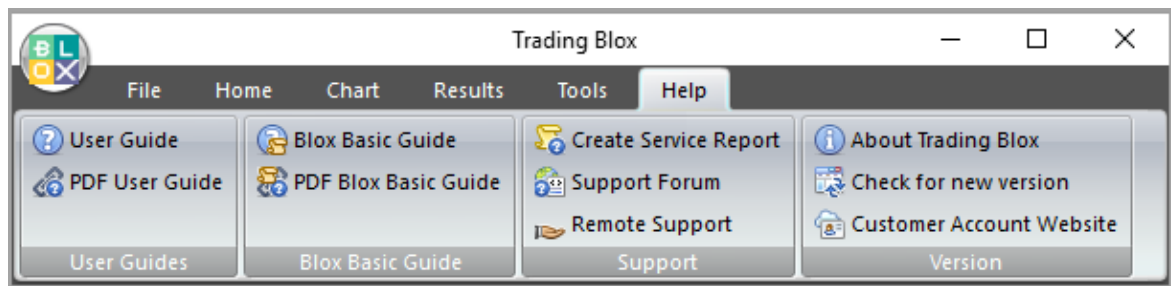
RESULTS Menu Controls: [^Top](#)

This group is the area to access current test logs and orders. Retrieve previous test results, and manage display changes like removing test tabs and clearing previous information from the main screen's Log Window.

Double-Click on the **"Results"** tab to open PDF order report

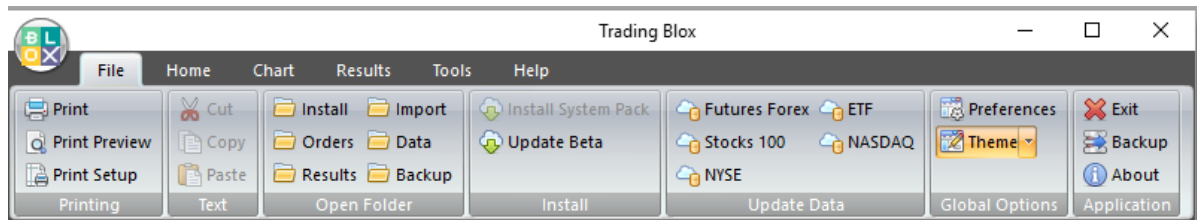
TOOLS Menu Controls: [^Top](#)**Tools Menu Control Group**

Tools menu group allows the user to disable the license, and reset Trading Blox settings to their installed default values. In Dictionaries and Dividends an updated Stock Dictionary can be created, or removed from the installation. There is access to the Dividend control that reads all the stock type files available in and under the specified stock folder so an update

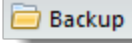
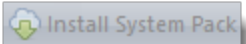
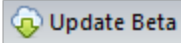
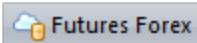
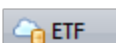
HELP Menu Controls: [^Top](#)**Help Menu Control Group**

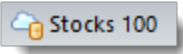
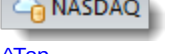
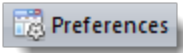
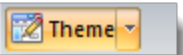
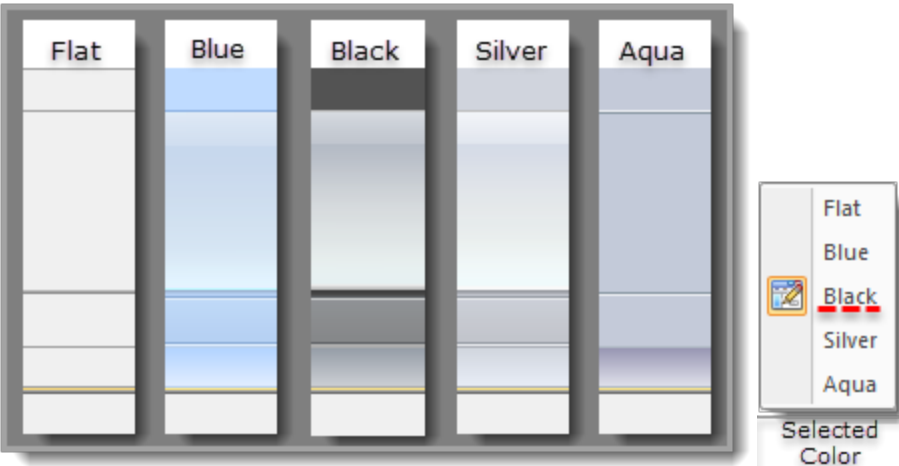
In this group access to the User Guide and Blox Basic Manual are made simple. In the Support section a service report can be created, and access to the Trader's Roundtable forum are possible with a single button click. Menu's version section will display the current version information, and allow the user to automatically update to the most recent release. Access to Trading Blox Web Site account area is just one click and the default browser will log the licensed user into their account page.

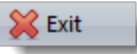
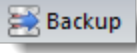
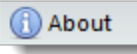
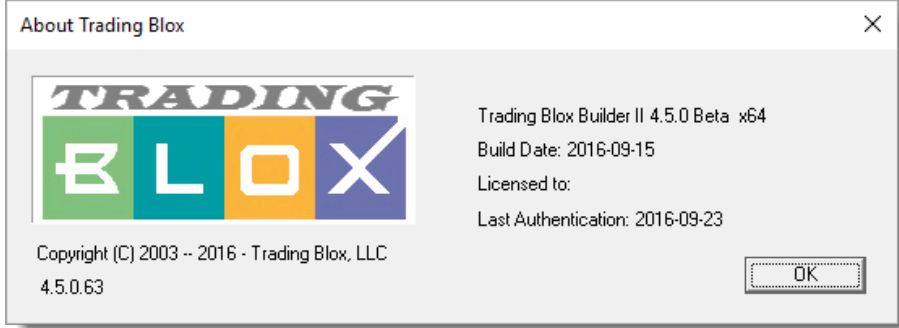
1.1 File Menu Groups



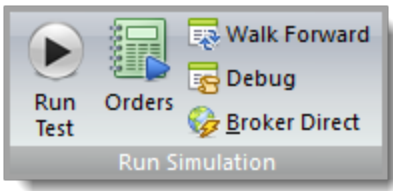
Printing: ^Top	Descriptions
	After a test report is displayed in the test results area, clicking on this Print button will display Windows' Printer dialog to print the report using the highlighted printer object.
	After a test report is displayed in the test results area, clicking on this Print Preview button will display Windows' Print Preview display screen showing how the test report will look after printing.
	After a test report is displayed in the test results area, clicking on this Printer Setup button will display Windows' Print Setup screen to enable print output adjustments.
Text: ^Top	Descriptions
	Temporarily Disabled.
	When text in blox parameter is highlighted, clicking this button will preserve a copy of the value in Windows' Clipboard area so it can be pasted elsewhere.
	When text is in Windows' Clipboard area so it can be pasted in a blox parameter field.
Open Folder: ^Top	Descriptions
^Top	Install opens to folder name where Trading Blox is installed.
^Top	Install opens to folder name where the Order files are located.
^Top	Install opens to folder name where test Results files are located.
^Top	Install opens to folder name where suite Import files are located.
^Top	This Data button opens a file dialog at the top level of Data folder where Trading Blox is installed. the Folder that is open will contain the folders for: Dictionaries, Dividends, Forex, Forex Rates, & Futures

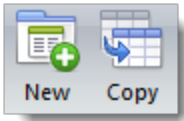
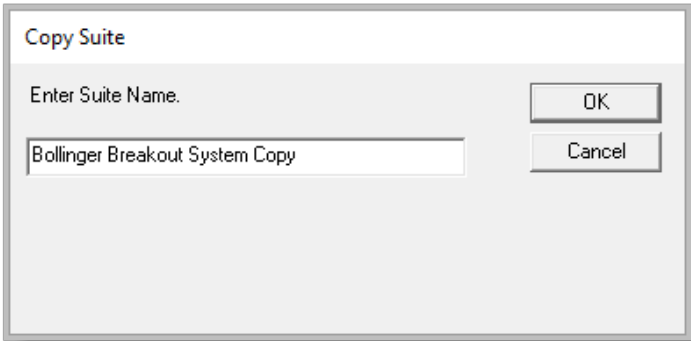
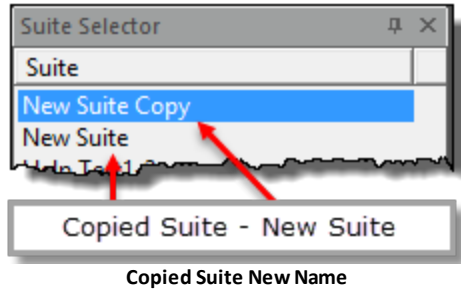
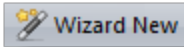
Open Folder: ^Top		Descriptions	
		Folder Names:	Purpose:
		Auxiliary	Broker Positions, Capital Adds & Draws, IB Fill Files
		Dictionaries	Dictionaries, Exchanges, Forex, Futures, Stocks and Currency Codes
		Dividends	Stock Dividend Files
		Forex	Forex Crossrate Data Files
		Forex Rates	Interest Rate files by Currency
		Futures	Futures Data Files
		Stocks	Stock Data File
 Backup ^Top		Install opens to folder name where Trading Blox the previously backed up files are located.	
Install: ^Top		Descriptions	
 Install System Pack ^Top		The system pack can be purchased on the Customer Login section of the website. When purchased, this menu will be enabled.	
 Update Beta ^Top		When this button is enabled, it will check the latest beta license release and will update Trading Blox to the latest point version. Beta Updates are not the same as the Help Menu's "Check for new version" process that only updates Trading Blox to its latest release version.	
Update Data: ^Top		Descriptions	
 Futures Forex ^Top		Courtesy Data is updated weekly and it is installed with the Trading Blox software. This data is made available to new customers for a period of 90 days, and to trial customers. Each data button provides an opportunity to update the sample data or the exit with no action.	
 ETF ^Top		Each button determines the type of data being updated. Sample data is suitable for testing and developing trading methods for each of the data types, but it will not be current enough to use for active trading signals and position price adjustments. All the buttons in the Update Data section will display an update dialog when a button is clicked. When the dialog appears it will show this information: "This will close your results windows and download and install new sample data into the Data folder. Would you like to continue?"	

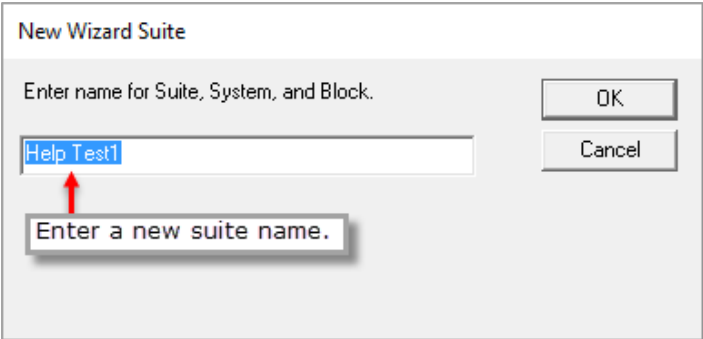
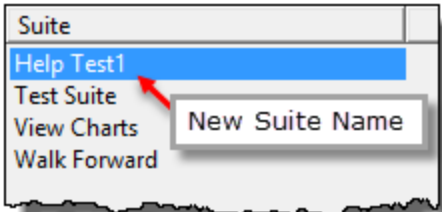
Update Data: ^Top	Descriptions
 ^Top	
 ^Top	
 ^Top	
Global Options: ^Top	Descriptions
 ^Top	This button will open the Trading Blox Preferences Editor Help topic for information about the settings all the software's operation settings. Topic in this section include: General Settings for: • Data Folders and Options Reporting Reporting Settings for: • Reporting General • Monte Carlo Single Test Reporting • Small Simulation Reporting • Large Simulation Reporting • Trade Chart • Trade Chart Colors • Trade Details • Blox Editor • Blox Script Colors
	Trading Blox theme color has five different options that can be selected when this Theme button is clicked: 

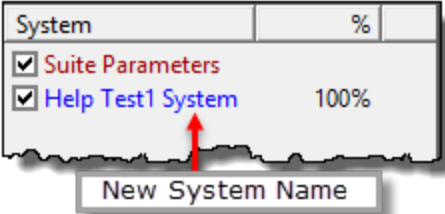
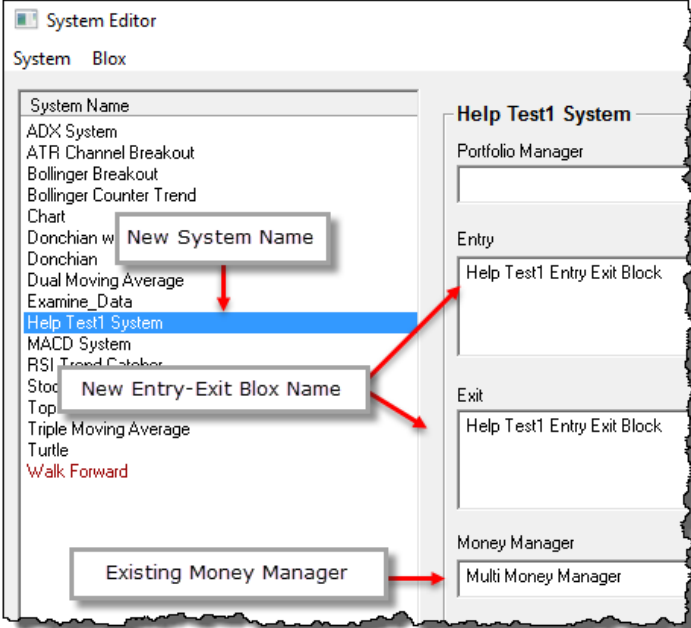
Application: ^Top	Descriptions
	Clicking this button will exit the program.
	This button will display the backup request dialog information" "This will run the user modifiable backup.bat utility in the background to backup and zip the selected files and folders. The backup will be then date and time stamped and placed in the backups folder. Would you like to continue?" When files are backup, the default process is to store the files in the Backup Folder. Advanced Users: It is possible change the backup.bat file, which executes the backup file process, to include other folders, or do other action like copy the blox folder to a Dropbox location or other location.
	Click this button to show your Trading Blox License and current software version information.  Trading Blox License Name and version information.

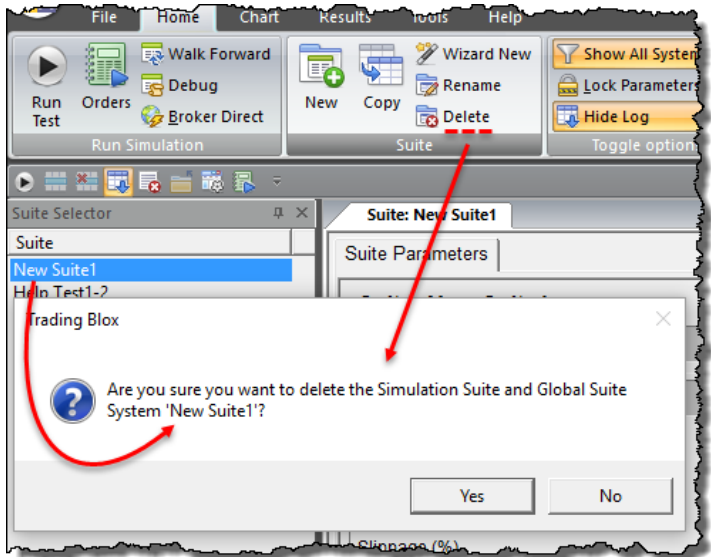
1.2 Home Menu Groups

Run Simulation: ^Top	Descriptions
 Run Test = (F5) Orders = (F7)	Use Run Test to run a normal simulation. Clicking on this control, or pressing the F5 key, will execute a test of the active suite's systems. Use Orders to generate orders. Clicking on this control, or pressing the F7 key, will generate a test and display a list of open positions and their orders. Walk Forward process uses the Preferences setting for Measure of Goodness. A built-in statistic or a custom statistic can be the goodness measures. Walk Forward Optimization information is available here: Walk Forward Parameters Using Walk Forward. Debug button in the Trading Blox version Builder II will stop execution at the first executable script line in each blox section. To continue execution in that script, press F5 to jump to the next debug breakpoint location, or press F11 to run the current line. Pressing F4 will open the Trading Blox Blox Basic Editor in script section and line location where F4 was pressed. Use the Broker Direct button to start the automated process that generates orders every bar. The first run is a plain simulation run, as it could be mid bar, but starting on the second run going forward, the Order Generation process will be run every bar as the new bar is created. Data is updated (if data connection type is selected), positions are synchronized between TB and the broker, and new orders are sent to the broker. Use Run Test to run a normal simulation. When the Broker Direct Preference Setting shows the Broker Connection is set to Data Only, or Data and Orders, then the data will be updated before the test is run. When the data is already updated, or static from an external source, then set the

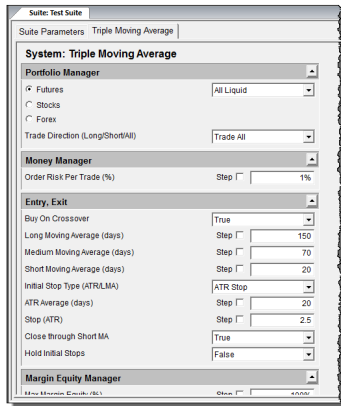
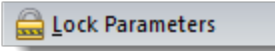
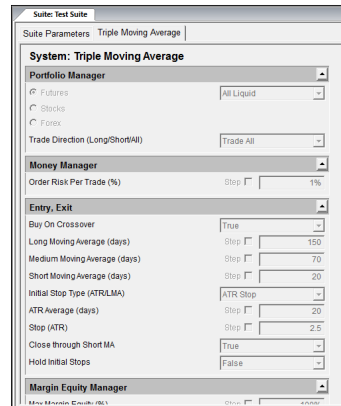
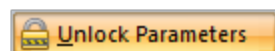
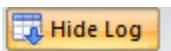
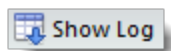
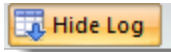
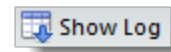
Run Simulation: ^Top	Descriptions
	Broker Direct Preference Setting show the Broker Connection is set to Offline. Use the Orders button to generate orders. When the Broker Direct setting Broker Connection setting is set to Orders Only, or Data and Orders, then the open positions will be synchronized and the orders will be sent to the broker.
Suite Options: ^Top	Descriptions
 ^Top	New: Clicking the New icon will display a dialog where you can enter the name of a new suite. This option only creates a new suite name. To create a New Suite, System and Blox modules, use Wizard New option. Copy: This Copy icon will display a dialog where the name of the selected suite is displayed with the "Copy" text characters added. Any name can be the new suite's name by editing the name shown in the Copy Suite name area. Selected name: "New Suite", click copy, dialog will show "New Suite Copy."  Copy Selected Suite Dialog  Copied Suite New Name
 ^Top	This Wizard option will create a New Suite name, a System Name, and an Entry and Exit blox module. It starts with a

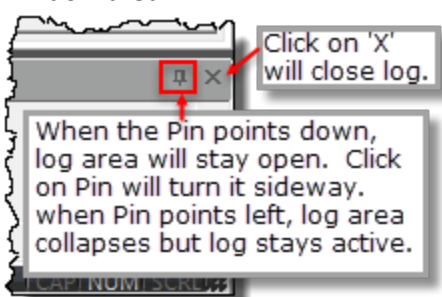
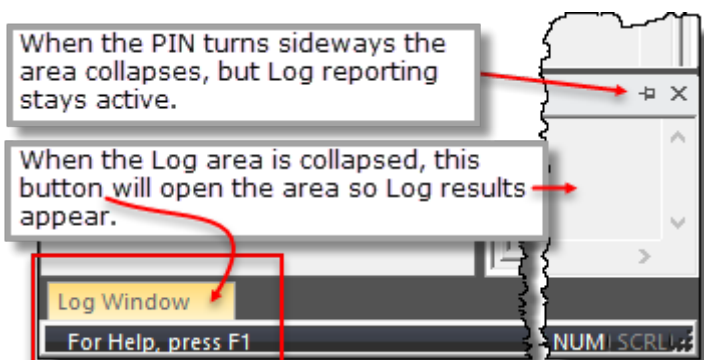
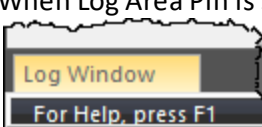
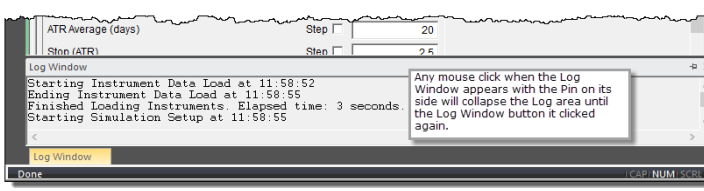
Suite Options: ^Top	Descriptions
	pop-up dialog where you enter the name for the new suite:  Wizard New Suite, System and Blox Module Name.  Wizard's New Suite Name.

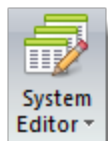
Suite Options: ^Top	Descriptions
	 Wizard's New System Name.  Wizard's New System Name Blox Modules. As soon as the "OK" button on the New Wizard Suite dialog, the entered name will appear in the Suite, and System name list. In the System Editor dialog there will be a two blox modules selected. One blox module will use the entered "New Suite Name", but it will have the suffix "EntryExitBlock" appended as its name. The second blox module will be the default "Multi Money Manager" that was installed when Trading Blox was installed.
 Rename ^Top	Renames any selected Suite name to the name entered.

Suite Options: ^Top	Descriptions		
	Rename Suite Enter Suite Name. Help Test1-2 Renaming a Suite from Help Test 1 to Help Test1-2 only changes the suite name. This is the same for new suites that use the "Wizard New" option to create a new Suite. Rename Suite Dialog		
 Delete ^Top	Deletes any selected Suite name when its pop-up when the deletion is confirmed.  Delete Selected Suite Dialog Method		
Toggle Options: ^Top	Descriptions		
 Show All Systems  Show Active Systems ^Top	This controls name changes between the two shown by the control names shown on the left. <table border="1" data-bbox="727 1528 1446 1682"> <tr> <td data-bbox="727 1528 1084 1682">Show All System will display all the system names that are available.</td><td data-bbox="1084 1528 1446 1682">Show Active Systems will only display the systems that are assigned to the active Suite.</td></tr> </table>	Show All System will display all the system names that are available.	Show Active Systems will only display the systems that are assigned to the active Suite.
Show All System will display all the system names that are available.	Show Active Systems will only display the systems that are assigned to the active Suite.		

Toggle Options: ^Top	Descriptions																																												
	<table border="1"> <thead> <tr> <th>System</th><th>%</th></tr> </thead> <tbody> <tr><td>☒ Suite Parameters</td><td></td></tr> <tr><td>☐ ADX System</td><td>0%</td></tr> <tr><td>☐ ATR Channel Brea...</td><td>0%</td></tr> <tr><td>☐ Bollinger Breakout</td><td>0%</td></tr> <tr><td>☐ Bollinger Counter...</td><td>0%</td></tr> <tr><td>☐ Chart</td><td>0%</td></tr> <tr><td>☐ Donchian w Rule ...</td><td>0%</td></tr> <tr><td>☐ Donchian</td><td>0%</td></tr> <tr><td>☐ Dual Moving Aver...</td><td>0%</td></tr> <tr><td>☐ Examine_Data</td><td>0%</td></tr> <tr><td>☒ Help Test1 System</td><td>100%</td></tr> <tr><td>☐ MACD System</td><td>0%</td></tr> <tr><td>☐ RSI Trend Catcher</td><td>0%</td></tr> <tr><td>☐ Stochastic System</td><td>0%</td></tr> <tr><td>☐ Top Ranked Stocks</td><td>0%</td></tr> <tr><td>☐ Triple Moving Av...</td><td>0%</td></tr> <tr><td>☐ Turtle</td><td>0%</td></tr> <tr><td>☐ Walk Forward</td><td></td></tr> </tbody> </table> Show All System Names <table border="1"> <thead> <tr> <th>System</th><th>%</th></tr> </thead> <tbody> <tr><td>☒ Suite Parameters</td><td></td></tr> <tr><td>☒ Help Test1 System</td><td>100%</td></tr> </tbody> </table> Show Active System Names	System	%	☒ Suite Parameters		☐ ADX System	0%	☐ ATR Channel Brea...	0%	☐ Bollinger Breakout	0%	☐ Bollinger Counter...	0%	☐ Chart	0%	☐ Donchian w Rule ...	0%	☐ Donchian	0%	☐ Dual Moving Aver...	0%	☐ Examine_Data	0%	☒ Help Test1 System	100%	☐ MACD System	0%	☐ RSI Trend Catcher	0%	☐ Stochastic System	0%	☐ Top Ranked Stocks	0%	☐ Triple Moving Av...	0%	☐ Turtle	0%	☐ Walk Forward		System	%	☒ Suite Parameters		☒ Help Test1 System	100%
System	%																																												
☒ Suite Parameters																																													
☐ ADX System	0%																																												
☐ ATR Channel Brea...	0%																																												
☐ Bollinger Breakout	0%																																												
☐ Bollinger Counter...	0%																																												
☐ Chart	0%																																												
☐ Donchian w Rule ...	0%																																												
☐ Donchian	0%																																												
☐ Dual Moving Aver...	0%																																												
☐ Examine_Data	0%																																												
☒ Help Test1 System	100%																																												
☐ MACD System	0%																																												
☐ RSI Trend Catcher	0%																																												
☐ Stochastic System	0%																																												
☐ Top Ranked Stocks	0%																																												
☐ Triple Moving Av...	0%																																												
☐ Turtle	0%																																												
☐ Walk Forward																																													
System	%																																												
☒ Suite Parameters																																													
☒ Help Test1 System	100%																																												
 Lock Parameters  Unlock Parameters ^Top	Simulation Suites can be locked. Locked suites disable all editing controls for the system parameters and settings and prevents the inadvertent changes to a system's settings. To lock a Simulation Suite, simply click on the "Lock Parameters" button. When a Simulation Suite is locked, the Toggle Group Lock Parameters button changes to an Unlock Parameters button. which can be used to unlock the Simulation Suite. Clicking on the Unlock Parameters button will unlock the parameter settings so they can be edited. Similarly, the Lock Suite menu item changed to Unlock Suite when the current suite is a locked Simulation Suite. Note that this does not prevent changes to the system or the blox, so to truly lock a suite a second copy of the entire Trading Blox folder should be used. An example would be for production order generation where changes could be disastrous.																																												

Toggle Options: ^Top	Descriptions
	 Suite Parameters Are NOT Locked. When Parameters are NOT locked, each parameter field shows black numbers on a white back ground and the Suite's Parameter control shows:   Suite Parameters Are LOCKED. When Parameters are LOCKED, each parameter field shows black numbers on a white back ground and the Suite's Parameter control shows: 
  ^Top	Default Log state appears at the bottom of the main screen. When the log windows is open, the Log button changes state and its label shows it can close the log.  Trading Blox Log Default State and Location.  When log does not appear or is moved, main screen area appears like this:  Trading Blox Log Area Closed, or Log is Floating.  When the log windows is closed, the Log button changes state and its label shows it can open the log. Trading Blox Log Window options allow the Log Window to capture information without the window being open all the time. In the upper right-hand side of the Log Window's Title bar is a small corkboard type Pin image that controls how an open Log Window was be displayed. When the Pin is pointing down, the Low Window will appear and stay open the entire time it is active.

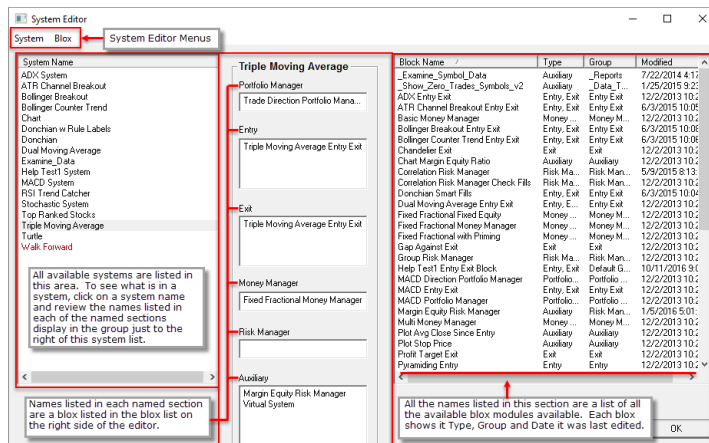
Toggle Options: ^Top	Descriptions
	When the Pin is pointing sideways, the Log Window will collect testing information, but the Log Window space it uses when it is open will be reduced so that only the Log Window title bar is visible. In this state, the Log Window name in the Log Window title bar name will be highlighted. When it is highlighted it becomes a button that will display the contents of the Log Window until the mouse click on the program screen outside of the open Log Window area.  Log Window Area Display Pin Control Details  Log Area Active when Pin turned sideways minimizes Log Area, but reporting stays active. When Log Area Pin is sideways and this button is clicked:   Any User activity outside of open log area when the Pin is on its side will collapse the Log area display again.

Editors: [^Top](#)F3 [^Top](#)

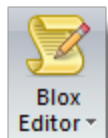
Descriptions

Trading Blox System Editor is where a new system can be added, edited or removed. Use this link to learn more about [System Editor Information](#).

System Editor Screen: (Click to Enlarge; Click to Reduce)



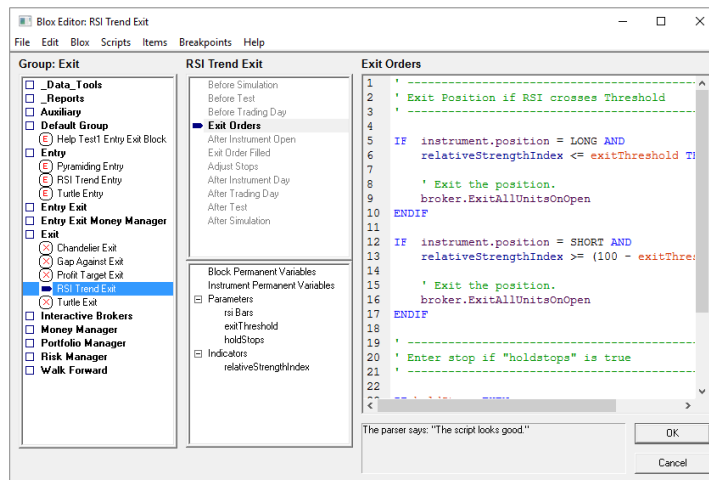
System Editor Area Descriptions

F4 [^Top](#)

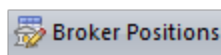
Not Available
Blox Editor Pro-3

Trading Blox Blox Editor is where new blox area created, edited, or removed. Use this link to learn more about [Blox Editor Information](#).

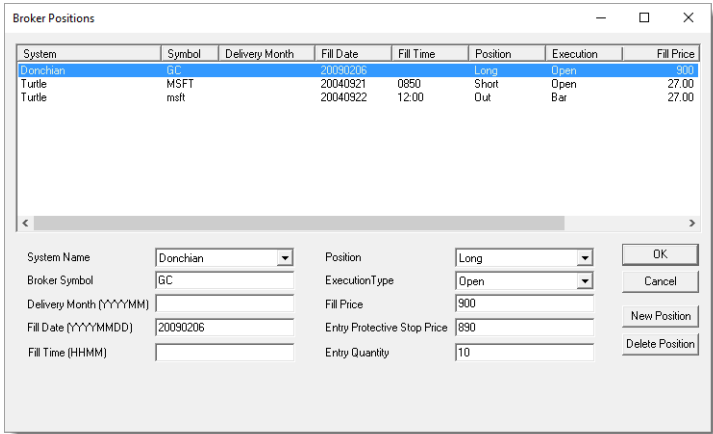
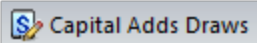
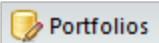
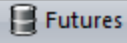
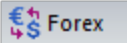
Blox Editor - Builder II: (Click to Enlarge; Click to Reduce)

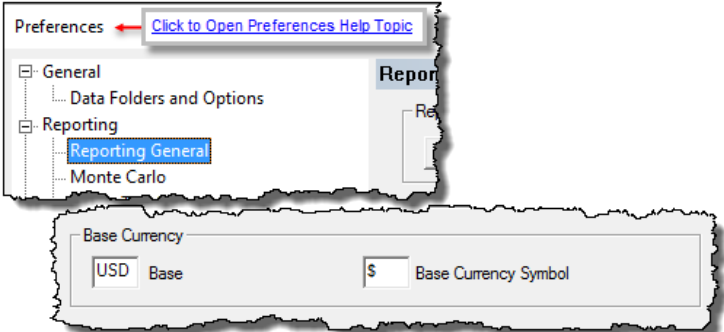


Blox Editor - Builder II, Blox Group, Blox Scripting Areas.

[^Top](#)

Trading Blox provides a Broker Positions Editor to enable a trader to use existing orders not created by the design of a Suite during a test or order generation. More information is available in [Broker Positions](#).

Editors: ^Top	Descriptions
	 Broker Editor Screen.
 ^Top	This control will display the Capital Adds & Draws Editor. This editor allows monetary amounts to be added, changed or earlier additions to be removed. Capital amounts can be used by the Global Parameter Fees control when that process is employed during a test. Capital Adds Draws will display information about how to use this feature.
 ^Top	Trading Blox allows users to create customized portfolios of futures, stocks, or forex data that can be historically tested, or used to generate daily entry orders. Each system has a portfolio manager section where you can define the portfolio that system will test or trade. This allows you to test multiple systems at once - each with a different custom portfolio. To create new portfolios, change the markets in an existing portfolio, or delete portfolios use the Portfolio Editor.
Dictionaries: ^Top	Descriptions
 ^Top	All Futures portfolios are dependent upon the built-in Trading Blox Futures Dictionary to provide the needed details required for testing and creating orders for the Futures trades. More information about this dictionary is available in the Historical Data Setup's Futures Dictionary Editor topic.
 ^Top	Trade testing and order creation for instruments that are traded on exchanges that are not in the same country as the money setting established in the Trading Blox Preference settings:

Dictionaries: ^Top	Descriptions
	 More information about the Forex Dictionary is available in the Historical Data Setup's Forex Dictionary Editor topic.
 Stock ^Top	More information about Stock Dictionaries is available in the Historical Data Setup's Stock Dictionary topic.
Suite Transfers: ^Top	Descriptions
 Export ^Top	Trading Blox method for the transfer of suites and all their components and settings into another installation of simple. Click on this topic Exporting Suites to see the steps explained.
 Import ^Top	Exported Suite zip file can use this function to import method that places all the exported files into their proper location. Information here Importing Suites explains the steps for importing suites and their component files.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 157

1.3 Chart Menu Group

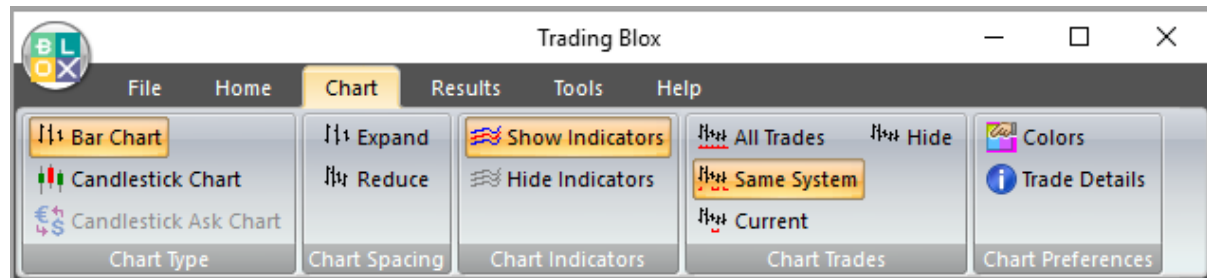


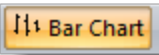
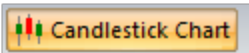
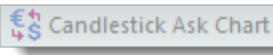
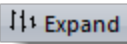
Chart Type: ^Top	Descriptions
 ^Top	Trading Blox standard Open, High, Low and Close colored bar appears when this option is enabled:  Open, High, Low and Close Colored Price-Bar Chart
 ^Top	Candlestick price bar charts will appear when this option is enabled:  Standard Candlestick Colored Price-Bar Chart
 ^Top	Forex Chart <pending>
Chart Spacing: ^Top	Descriptions
 ^Top	Spacing between each price bar is increased each time this button is clicked. When less bars like this are displayed, click this button to expand spacing and reduce how many price bars are displayed.

Chart Spacing: ^Top	Descriptions
	Small Spacing between each Price-Bar Wider Spacing between each Price-Bar
^Top	More price-bars displayed on a chart presents a wider historical view. Click this button to reduce the spacing between each price-bar until the amount of history needed is displayed. Wide spacing between Price-Bars limits how much history is displayed. Less spacing between Price-Bars increases how much history is displayed.
Chart Indicators: ^Top	Descriptions
^Top	Use this button to display indicators that were previously turned OFF: Button Changes Indicators turned OFF to Turned ON.
^Top	Use this button to turn OFF the displayed indicators:

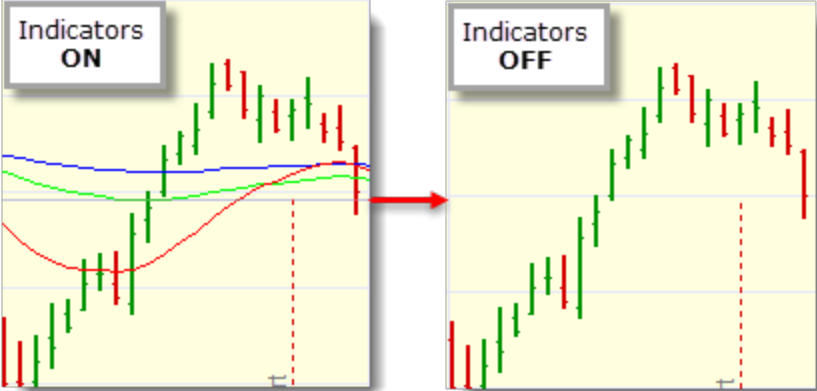
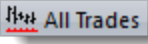
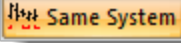
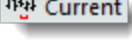
Chart Indicators: ^Top	Descriptions
	 Button Changes Indicators displayed to become hidden.
Chart Trades: ^Top	Descriptions
 ^Top	Suite test result trade positions can be shown in with all or the test created positions when this button is active.  Button shows all trades for this market.
 ^Top	Enable this button to limit test displayed trade positions to a single system.  Button shows all trades in this market this system.
 ^Top	When the current trade being examined is to be displayed, make this button active.

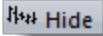
Chart Trades: ^Top	Descriptions
	 Button shows the current trades in this market this system.
 ^Top	To hide all positions, click this Hide button.  Button Hides all trades.
Chart Preferences: ^Top	Descriptions
 ^Top	Chart colors are controlled by this section in the software's Preference Editor.

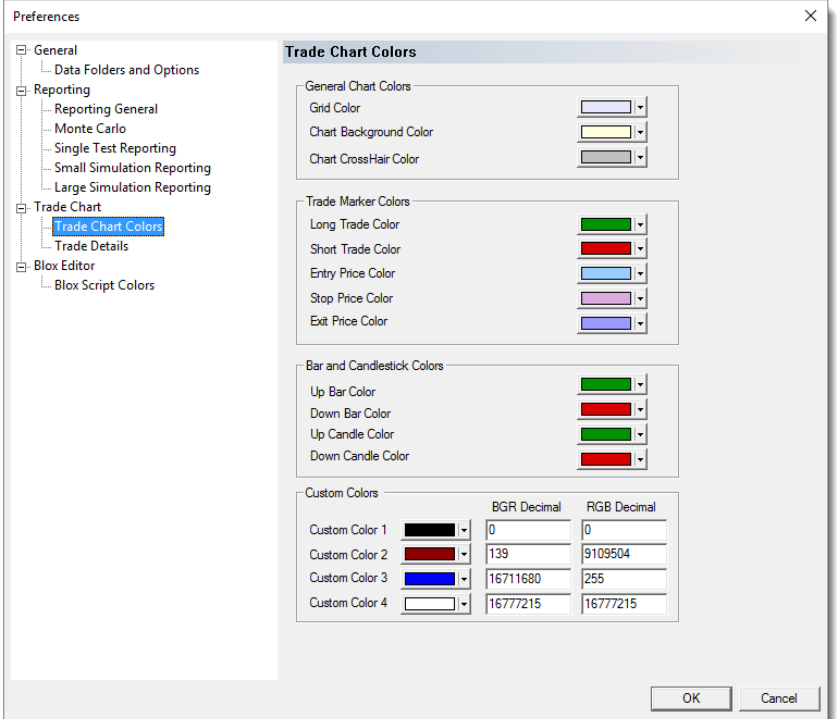
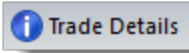
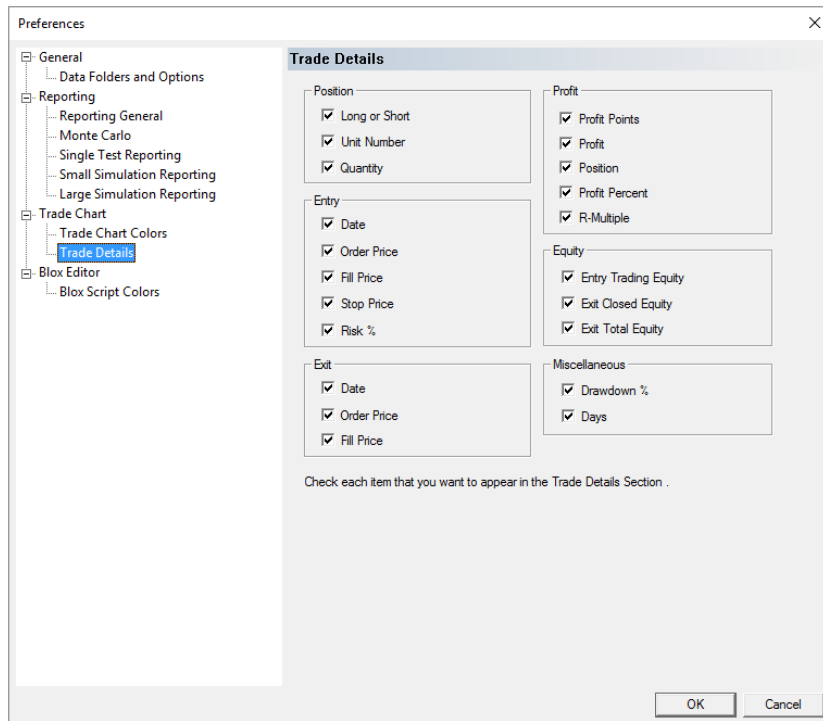
Chart Preferences: ^Top	Descriptions
	 Color button displays the Trade Chart Colors section of the Preference Menu
 ^Top	This button will quickly display the Trade Details section of Trading Blox Preference Editor. When this section is displayed, the fields displayed in the Trade Information Section of Trade Results chart can be changed.

Chart Preferences:[^Top](#)**Descriptions**

The Trade Details button displays the Trade Detail Options of the Preference Menu. After a test has completed, the Main Screen's Chart area will provide a lot of information about each trade and how its effect on equity.

Show:

- Data Window
- Chart's Selected Trade Information
- Trade Record Listing

of the options control the trade detail section of the trade chart, lower right green area.

You can also right click on that area to bring up these preferences in Trading Blox.

You can right click on the cross hair price detail area, upper right purple area, to bring up those options. Show close price, hide close price, etc.

You can right click on the indicators, middle area, for all sorts of other options like edit and plot disable.

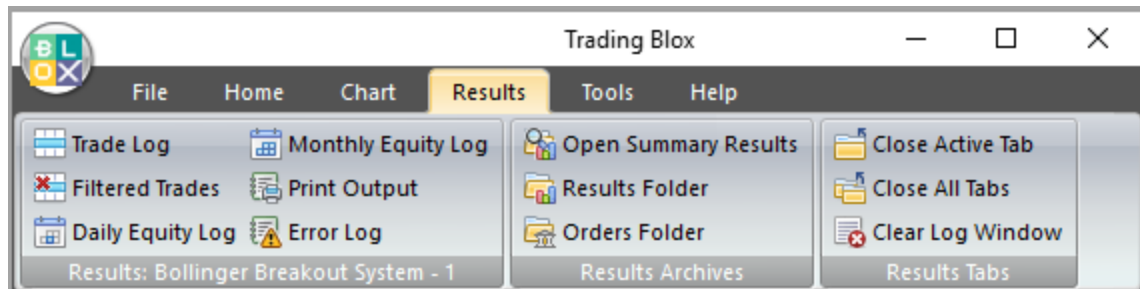
In the chart's lower right trade information window area is a list of the data information fields the position selected by the cross-hair cursor reveals. Data in this trade information area can be changed

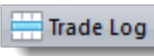
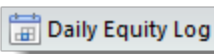
Chart Preferences: ^Top	Descriptions
	by changing the selection of fields enabled on the following Preference Trade Chart's Trade Detail Section of Trading Blox Preferences.  Chart Trade Information Window When a chart is displayed, any Right-Click of the mouse will display the Trade Detail Section of the Preference Editor so the enabled fields listed can be changed (see below).

Edit Time: 9/12/2020 9:50:00 AM

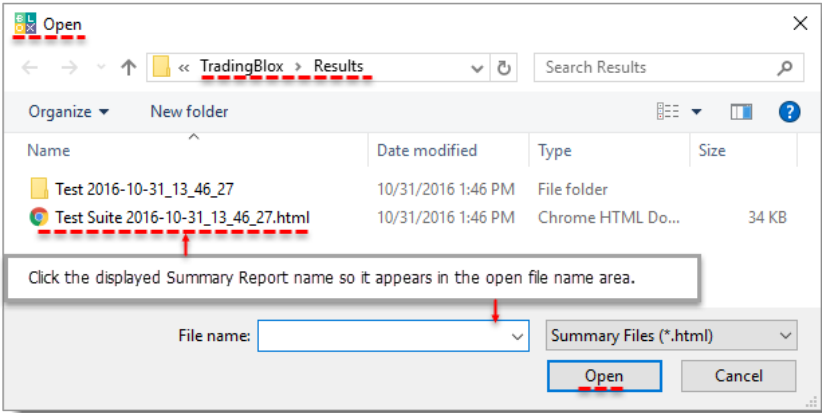
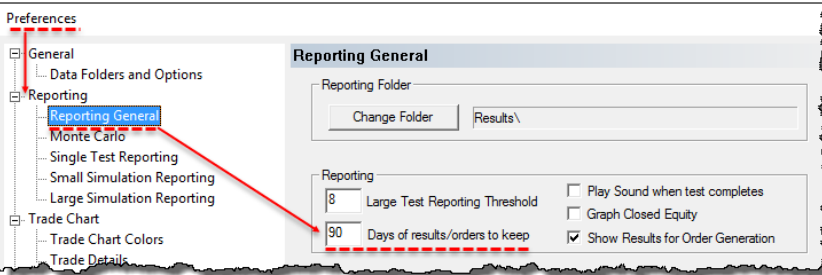
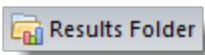
Topic ID#: 121

1.4 Results Menu Group



Results Logs: ^Top	Descriptions																																																																																					
 Trade Log ^Top	Folder Location: <drive-letter>:\TradingBlox\Results File Name: Trade Log.csv The results from a suite test will create a log of the trades that created the results of the "Test Suite Summary Results" report that appears when a test is finished. <table><tr><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th><th>G</th><th>H</th><th>I</th><th>J</th><th>K</th><th>L</th><th>M</th><th>N</th><th>O</th><th>P</th></tr><tr><td>1</td><td>Test</td><td>System Number</td><td>System Name</td><td>Symbol</td><td>Unit</td><td>Position</td><td>Entry Date</td><td>Entry Time</td><td>Entry Order Price</td><td>Stop Price</td><td>Risk%</td><td>Quantity</td><td>Entry Fill</td><td>Entry DPP</td><td>Entry Stock Split Ratio</td><td>Exit Date</td></tr><tr><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <table><tr><th></th><th>Q</th><th>R</th><th>S</th><th>T</th><th>U</th><th>V</th><th>W</th><th>X</th><th>Y</th><th>Z</th><th>AA</th><th>AB</th><th>AC</th><th>AD</th><th>AE</th><th>AF</th></tr><tr><td></td><td>Exit Time</td><td>Exit Order Price</td><td>Exit Fill</td><td>Exit DPP</td><td>Dividends</td><td>Commission</td><td>Forex Carry</td><td>Profit</td><td>Profit%</td><td>Closed Equity</td><td>Total Equity</td><td>Drawdown</td><td>Days in Trade</td><td>R Multiple</td><td>Custom</td><td>Rule</td></tr></table> Simulation or Orders Report of Trade Log Data Columns		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	1	Test	System Number	System Name	Symbol	Unit	Position	Entry Date	Entry Time	Entry Order Price	Stop Price	Risk%	Quantity	Entry Fill	Entry DPP	Entry Stock Split Ratio	Exit Date	2																		Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF		Exit Time	Exit Order Price	Exit Fill	Exit DPP	Dividends	Commission	Forex Carry	Profit	Profit%	Closed Equity	Total Equity	Drawdown	Days in Trade	R Multiple	Custom	Rule
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P																																																																						
1	Test	System Number	System Name	Symbol	Unit	Position	Entry Date	Entry Time	Entry Order Price	Stop Price	Risk%	Quantity	Entry Fill	Entry DPP	Entry Stock Split Ratio	Exit Date																																																																						
2																																																																																						
	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF																																																																						
	Exit Time	Exit Order Price	Exit Fill	Exit DPP	Dividends	Commission	Forex Carry	Profit	Profit%	Closed Equity	Total Equity	Drawdown	Days in Trade	R Multiple	Custom	Rule																																																																						
 Filtered Trades ^Top	Folder Location: <drive-letter>:\TradingBlox\Results File Name: Filter Log.csv The Trading Blox Filtered Trades Log is a comma separated file that will load into the computer's spreadsheet that is designed to open ".CSV" files. This is an important log to review when you finish a test because it reports rejected that might have happened. Each trade rejected will show these details: <table><tr><th></th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th><th>G</th><th>H</th><th>I</th><th>J</th><th>K</th></tr><tr><td>1</td><td>Test</td><td>Type</td><td>System Name</td><td>Symbol</td><td>Unit</td><td>Order Type</td><td>Execution Type</td><td>Quantity</td><td>Price</td><td>Date</td><td>Reason</td></tr><tr><td>2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> Filtered Trades Report Data Columns		A	B	C	D	E	F	G	H	I	J	K	1	Test	Type	System Name	Symbol	Unit	Order Type	Execution Type	Quantity	Price	Date	Reason	2																																																												
	A	B	C	D	E	F	G	H	I	J	K																																																																											
1	Test	Type	System Name	Symbol	Unit	Order Type	Execution Type	Quantity	Price	Date	Reason																																																																											
2																																																																																						
 Daily Equity Log ^Top	Folder Location: <drive-letter>:\TradingBlox\Results File Name: Daily Equity Log.csv The Trading Blox Daily Equity Log is a comma separated file that will load into the computer's spreadsheet that is designed to open ".CSV" files. This is an important log to review when you finish a test When you want to see how the daily equity changed over the period of the test.																																																																																					

Results Logs: ^Top	Descriptions
 Monthly Equity Log ^Top	Folder Location: <drive-letter>:\TradingBlox\Results File Name: Monthly Equity Log.csv The Trading Blox Monthly Equity Log is a comma separated file that will load into the computer's spreadsheet that is designed to open ".CSV" files. This is an important log to review when you finish a test When you want to see how the monthly equity changed over the period of the test.
 Print Output ^Top	Folder Location: <drive-letter>:\TradingBlox\Results File Name: Print Output.csv The Trading Blox Print Output is a comma separated file that will load into the computer's spreadsheet that is designed to open ".CSV" files. When Trading Blox finds something that should be changed it will be reported in this report. It is important to review this area when the main Log Window area is closed. This reporting method is also where custom reporting is displayed most often. This report is also were custom Print Output Statements that are active in any of the blox in are generating information.
 Error Log ^Top	Folder Location: <drive-letter>:\TradingBlox\Log Files File Name: Errors.txt The Trading Blox Error Log is a text file that will load into the computer's default text editor that Windows uses to open ".txt" files. This is an important log to review when you finish a test because it reports errors that might have happened.
Results Archives: ^Top	Descriptions
 Open Summary Results ^Top	Folder Location: <drive-letter>:\TradingBlox\Results This folder is the location where Trading Blox stores all of its summary performance results. When this control is clicked, it will open a Windows open file dialog with the most recent performance summary file name displayed in the open file dialog area. Clicking the file name and then on the Open button to open the summary performance report into a main screen tab.

Results Archives: ^Top	Descriptions
	 Open Performance Results Report Dialog In the Preference Editor's general Reporting section is a setting to control the number of days to preserve summary performance results and orders:  Preferences General Reporting Orders/Results Days Limit
	Folder Location: <drive-letter>:\TradingBlox\Results This button opens the Trading Blox folder with all the files in the folder displayed. Double-clicking any of the files listed will open them into Windows default program for each of the types listed. Files can also be copied or deleted. ^Top

Results Archives: [^Top](#)

Descriptions

Trading Blox Results Folder All Files Display

When any simulation results tabs are still displayed on the main screen, right-clicking on the results tab to display popup menu.

Simulation Test Results Tab Results Folder File Access Menu

[^Top](#)

Folder Location: <drive-letter>:\TradingBlox\Results

This control will display Trading Blox specified orders folder dialog so that current and earlier order files can be accessed.

Order Generation Created File Names & Types

Results Archives: [^Top](#)

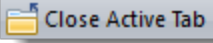
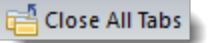
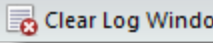
File Number:	Description:
1	".CSV" file is Comma Separated Values File primarily designed to open in a Spreadsheet program. This file is an "Orders.csv, file item #5" archive file that will be available until the Preference setting in Report General Days to Keep limit is reached.
2	".html" file is a browser file that will open with all common browsers. File contains a list of all active positions and all current orders for the date specified in the file.
3	".mht" file is a complete browser file that contains all the web page contents. This type of file makes sending of web pages easier because the web page's container folder that holds the files of images and resources are not needed because they are already stored in the ".mht" file. See file size differences between file #3 & file #4. File contains a list of all active positions and all current orders for the date specified in the file.
4	".pdf" file is a standard Adobe Systems Acrobat compatible PDF that will open with most common PDF reader programs. File contains a list of all active positions and all current orders for the date specified in the file.
5	"Orders.csv" is a list of all the orders that are active. File doesn't not contain the list of open positions. This file is replaced every the order generation process (F7) is executed.

All order files contain these columns of information for Active Positions and Current Orders:

	A	B	C	D	E	F	G	H	I
1	Order Type	Quantity	Symbol	Broker Symbol	Month	Unit	Order Price	Roll Info	Exec Type
2									
3									

	A	J	K	L	M	N	O	P	Q	R
1	Order Type	Stop Price	Market	Exchange	Position	Last Price	Distance	Last Date	Rules	Message
2										
3										

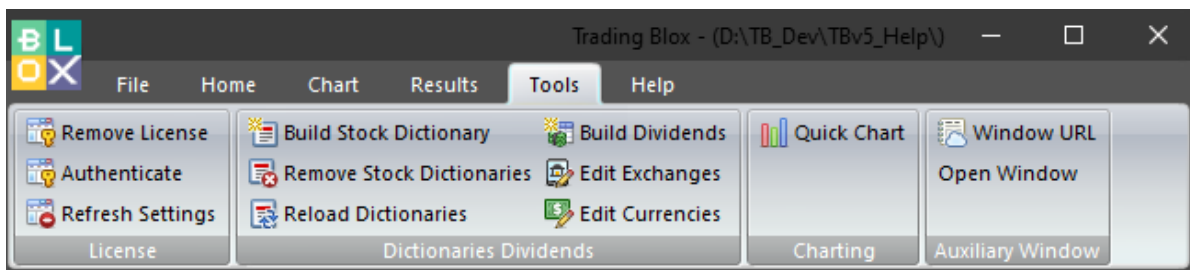
Positions & Orders Reporting Column Tiles

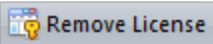
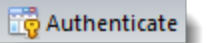
Results Tabs: ^Top	Descriptions
 ^Top	Click this button to close the current tab that is displaying the displayed " Test Suite Summary Results " report.
 ^Top	When this control is clicked all of the tabs that are displaying " Test Suite Summary Results " report will be closed.
 ^Top	Use this button to clear all the information displayed in the main screen's Log Window.

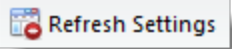
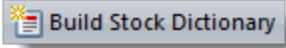
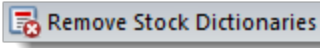
Edit Time: 9/12/2020 9:50:00 AM

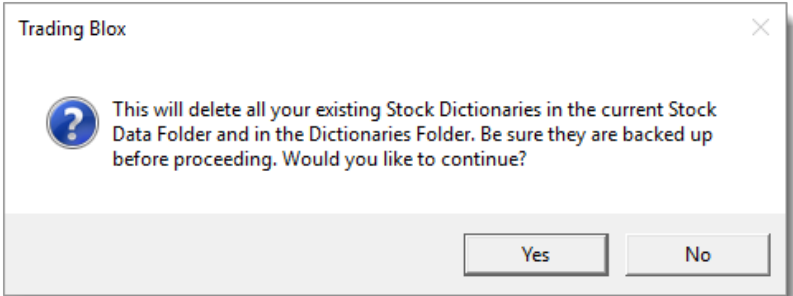
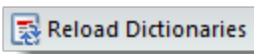
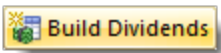
Topic ID#: 195

1.5 Tools Menu Group



License: ^Top	Descriptions
	Trading Blox standard license allows a user to install the software on two different computers, but there is no licensing limit to the number of times or the folder locations on the same computer where Trading Blox can be installed. If you get an additional license, you can remove a license from one of the computers and apply it to the added computer. You can also request a change in the number of computers that can have a licensed version of the Trading Blox.
 Trading Blox License Authenticate	A Trading Blox installation automatically authenticates the user's license code. That action uses the purchase date so it can remind the user of their initial Support Expiration Date. A month before the support expiration, Trading Blox will display a reminder notice to remind the user. Verification will also enable access to Futures Data, beta versions, thread count, and IB access when they are changed. Earlier versions of Trading Blox used the internet to perform an authentication each time Trading Blox started. That process

License: ^Top	Descriptions
	doesn't happen any longer. Instead, the authentication server assumes nothing has changed, allowing faster Trading Blox starts. A downside of this is when the license features have changed, they will need to be enabled by a new Authentication. When you upgrade the number of threads, or maintenance support, the license attributes, or other options, you should use this Authentication to refresh the license attributes.
 Refresh Trading Blox to its installed settings. ^Top	Purpose of this option is to RESET the settings to how they were when the program was First Installed. It should be used when you want the original settings, or when something appears to be affecting the operation, or the display of windows in Trading Blox. This button restores the software default settings in the computer's registry area. This means that any changes made to any settings in all of the Trading Blox installations since they were installed will be just like they were when Trading Blox first installed. For example, the program screen size and location, editor column sizes, and Quick Access buttons, information about the the last suite and system block used, will all be restored to their default values. Your user Preference settings are not changed. After this feature has been used, Trading Blox will require your license name and registration key information before it will complete a start-up and be available for use. Once the default settings have been restored, Trading Blox will shut down, and will restart itself with its default settings.
Dictionaries Dividends: ^Top	Descriptions
	When a Stock Dictionary is needed, or when there are stock additions or removals, this control will create a new Stock Dictionary. Stock dictionaries use the CSI market data file (sdbfacts.csv) to build the stock dictionary. See the Stock Dictionary in the Historical Data Setup section for more detailed information.
	This button will remove the entire Stock Dictionary. When the button is clicked, this dialog will appear asking for confirmation

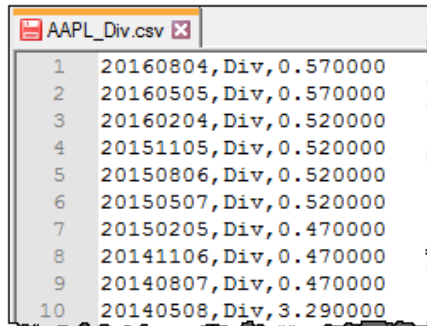
Dictionaries Dividends: ^Top	Descriptions
	before deleting the dictionary:  Delete All the Stock Dictionaries Confirmation Dialog
 ^Top	Trading Blox loads all dictionaries when it is started. Any changes to a dictionary requires the dictionary to be reloaded. A change to a dictionary happens when location of the dictionary is changed using the settings in Preferences setting section Data Folders and Data Option Preferences, or when any of the symbols in a dictionary are added, modified, or removed. Use this option to reload Trading Blox dictionaries.
 ^Top	This option uses CSI's Unfair Advantage Dividend subscription to build dividend files in the folder location specified in the Preferences section Data Folders and Data Option Preferences. When this button is clicked it will start Unfair Advantage so that it can read the dividend files. Dividend File Requirements: CSI's stock dividend data is delivered as encrypted files and are not useable by Trading Blox until they are unencrypted. They will deliver decrypted files if you contact CSI and make their added cost arrangements to apply to your account subscription. Alternately you can build your own dividend files manually from another source for each stock and place the files in the Dividend Folder. The format is Date,Dividend and you can see example files in the sample data that ships with the product. Dividend sample data files install when Trading Blox is installed. Dividend columns must follow this data layout sequence with each field be separated by a comma: Date, Dividend, Dividend Value

Dictionaries Dividends:

[^Top](#)

Descriptions

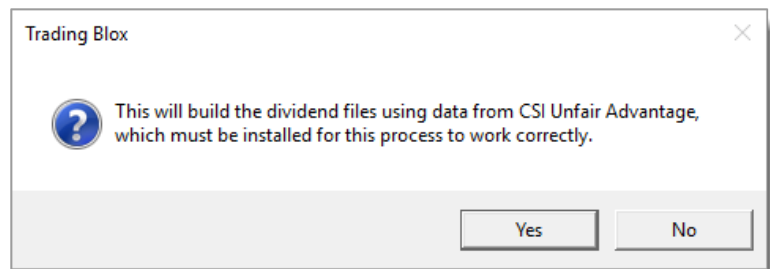
Dividend File Requirements:



	Dividend Date	Dividend Amount
1	20160804	Div, 0.570000
2	20160505	Div, 0.570000
3	20160204	Div, 0.520000
4	20151105	Div, 0.520000
5	20150806	Div, 0.520000
6	20150507	Div, 0.520000
7	20150205	Div, 0.470000
8	20141106	Div, 0.470000
9	20140807	Div, 0.470000
10	20140508	Div, 3.290000

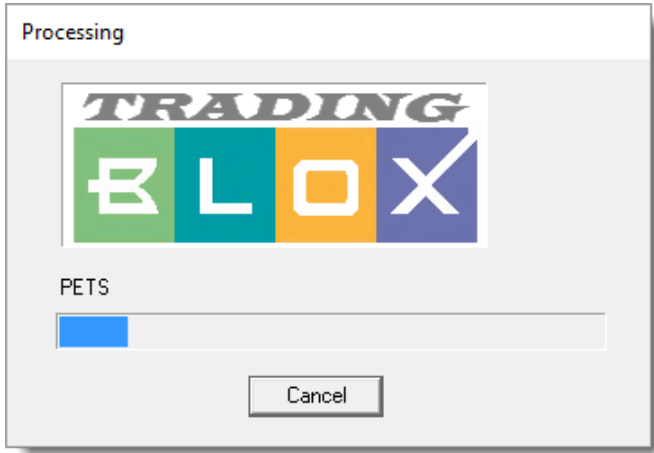
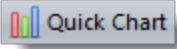
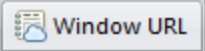
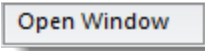
Example Dividend File Data Column Sequence

When this button is clicked Trading Blox will display this confirmation request to proceed:



Update Dividend Files Confirmation Dialog

Clicking on "**NO**", will halt the dividend data updates. A "**YES**" response will trigger Unfair Advantage to execute. When Unfair Advantage is ready to accept Trading Blox connection, this next dialog will appear where each dividend file symbol will be displayed until the blue progress bar reaches the end. When all the dividend files are updated the Processing dialog will disappear.

Dictionaries Dividends: ^Top	Descriptions
	 The image shows a dialog box titled "Processing". Inside, there is a logo for "TRADING BLOX" with the word "BLOX" in large, colorful letters (green, blue, orange, purple). Below the logo, the text "PETS" is displayed above a progress bar that is partially filled with blue. At the bottom right of the dialog is a "Cancel" button. Dividend File Process Dialog
Quick Chart: ^Top	Descriptions
	Quick Charts Option is a "Builder Licensed" version of a simple to use charting feature that will display any of Futures, Forex or Stocks stored in the Trading Blox folder paths set in the Data Folders and Data Option Preferences section of the Preferences setting editor.
Auxiliary Windows	Descriptions
  Open Auxiliary Window	Description pending. There is an optional Auxiliary Window in Trading Blox that can display text during a test. When Trading Blox is first installed, the Log Window and the Auxiliary Window are not displayed. The Main Menu "Tools' Auxiliary Window" section will display the Auxiliary Window. To close an open Auxiliary Window, click on the "x" at the top of the of the Auxiliary Window's display. If an Auxiliary Window is open when you exit Trading Blox, the Auxiliary Window will be displayed the next time you run Trading Blox. This is also true for the Log Window being open when you exit Trading Blox. Trading Blox allows the Log Window and Auxiliary Window to be open at the same time. Both windows can be

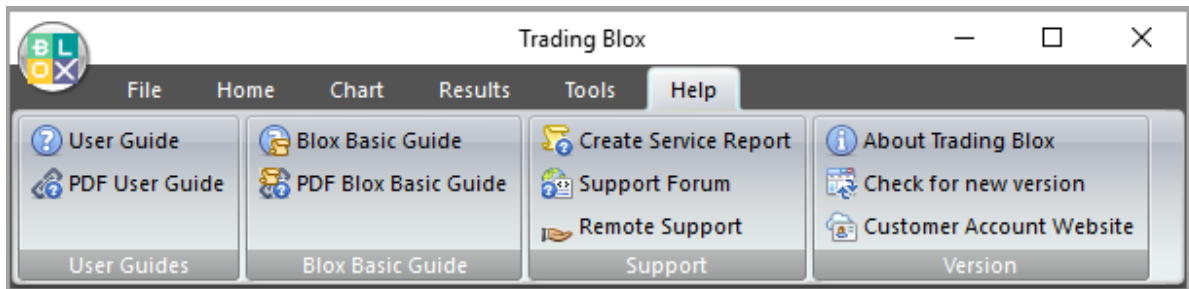
Quick Chart: ^Top	Descriptions
	Docked so they display as part of the Trading Blox main window. Both windows can be Undocked by holding the mouse button down on each windows title area, and as a handle that the mouse can use to move the window to another location on the computer's screen. There is a brief video in the Trading Blox Round Table Forum that demonstrates how to Access, Undock, Dock and move both the Log Window and the Auxiliary Window: TradingBlox-Docked-Floating-Docked.mp4 Text Display: Each window area is different in how they get and display text information. The Log Window displays text during a test using the Print function. Text in the Log Window is also stored in the "Print Output.csv" file that is in the Trading Blox Test-Results folder. The Auxiliary Window does not respond to Print statements. Instead, it depends upon a data collection that converts numerical values and other details into a String of information. Those Strings are saved in a "BPV Text-Series-Array" element. This series must be created by the user as part of the text display process. In the Trading Blox Builder's Help file, there is an example that shows how to handle create a process of sending text to the Auxiliary Window in the function: SetAuxiliaryWindowText(textSeriesName). More details about these window options are available at these Trading Blox Round Table links: TB-User's Guide Help File: Log Window => https://tinyurl.com/yxvgpgks TB-User's Guide Help File: Auxiliary Window => https://tinyurl.com/yxkkq5fz TB-Builder's Help File: SetAuxiliaryWindowText => https://tinyurl.com/y5qp3553

Quick Chart: ^Top	Descriptions
	Suite with Blox Used in the Window Demonstration Video Below: Export Any-Idea Test-05.zip Attached Video: Docked-Un-Docked-Floating Windows TradingBlox-Docked-Floating-Docked.mp4
Open Window	

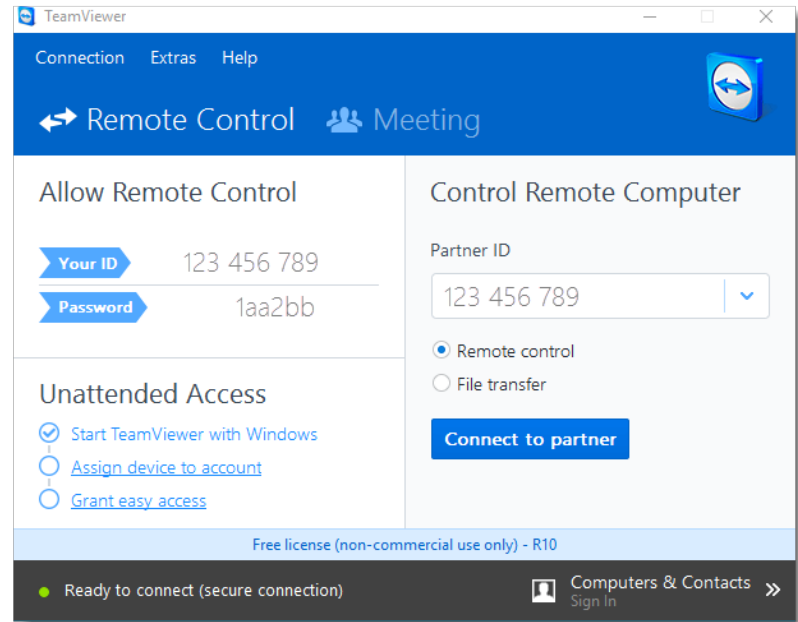
Edit Time: 8/24/2021 10:32:05 AM

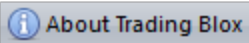
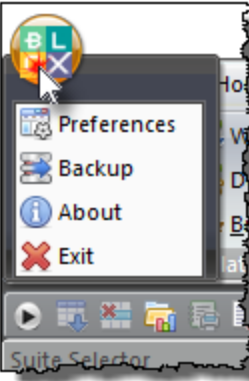
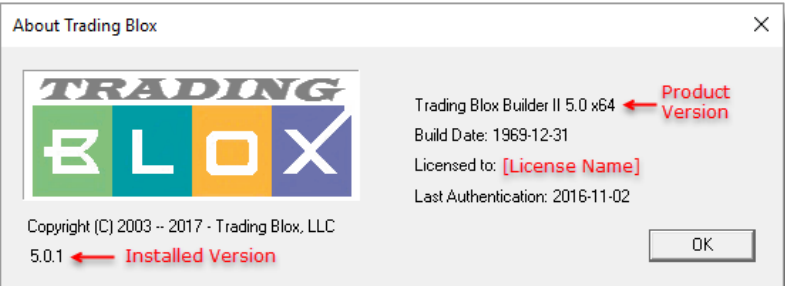
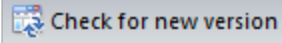
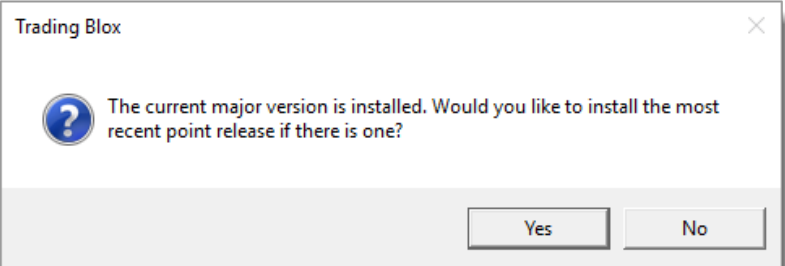
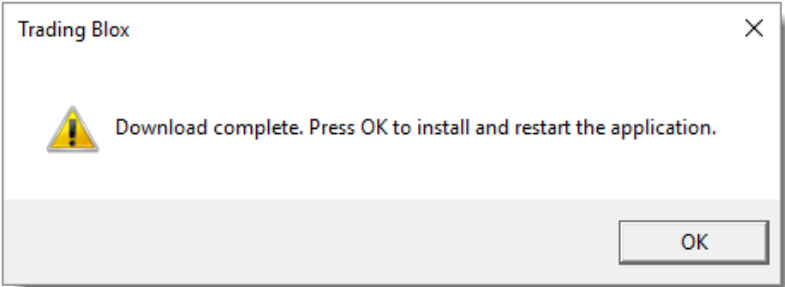
Topic ID#: 225

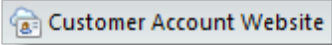
1.6 Help Menu Group



User Guides: ^Top	Description:
User Guide ^Top	This menu option will display the Help file you are reading here.
PDF User Guide ^Top	Clicking this link will display a PDF version the above Trading Blox User's Guide.
Blox Basic Guides: ^Top	Description:
Blox Basic Guide ^Top	Click on this button to display Trading Blox Blox Basic Builders Guide Help file.
PDF Blox Basic Guide ^Top	Clicking this button will display a PDF version of the above help file.
Support: ^Top	Description:
Create Service Report ^Top	If a problem surfaces with how Trading Blox is performing, technical support might request you generate a service report so they can see the settings and installed files. When this is needed, clicking on this button will create a zip file that can be sent to technical support for analysis.
Support Forum ^Top	Trading Blox has a forum for mechanical system traders called: Traders' Roundtable . To access the forum, send a note to Trading Blox Sales or Trading Blox Support . There is a "Chat" feature on the Trading Blox Home Page .
Remote Support ^Top	This option is for the times when you are having an issue and are in need of some technical assistance that can't be provided by sharing an installation Service Report , emails or phone conversations. To start a remote support connection, click on this button and one of two types of screens will appear. Most likely screen to appear will be this small dialog screen where a User ID and Password are displayed:

Support: ^Top	Description:
	 Remote Support TeamViewer Connection Dialog When TeamViewer is already running this type of dialog will appear:  Remote Support TeamViewer Installed Connection Dialog Regardless of which type of dialog appears, the user ID and Password will need to be exchanged between the location needing support and the technical support location.

Support: ^Top	Description:
Version: ^Top	Description:
 ^Top 	When this menu item is clicked it will display the Trading Blox license name and information about the software version installed:  About Trading Blox A quick way to access the About dialog is to lick on the Trading Blox icon at the top of the program.
 ^Top	This option allows Trading Blox to check for the latest version. When this option is clicked, it will first display this dialog:  Check For Current Version Confirmation If you click the No button, the dialog message will go away and nothing else will happen. If you click the Yes button, Trading Blox will download the upgrade files and install them. When the installation is complete, this next message will appear:  Download Complete Message Update When you click the OK button, Trading Blox will exit from the screen while the upgrade files are being installed. After the

Version: ^Top	Description:
	upgrade completes, Trading Blox will appear on the screen. In most cases this next dialog message will appear requesting you put the license name and license key into the two blank fields, if this is the first time, or just click on the register button if fields are already filled out:  Trading Blox License Name & Key Registration Dialog If there is Registration message, Trading Blox will look like it always does when it first starts up.
 ^Top	This button provides a quick and easy access to your account page section where your user profile is listed showing the license information, access to upgrade or full version installation packages and support renewal:  Trading Blox Customer Account Web Page

1.7 Quick Access Tool Bar

Quick Access Tool Bar can be customized to have more, or less quick access controls. This Tool Bar can also be moved from its default position below the main menu ribbon and tab controls, to Trading Blox Title Bar area. It also has a control arrow that will hide the main ribbon and provide access to the [Quick Access Customization](#) settings"

Quick Access Tool Bar Position Control:

Trading Blox Quick Access Toolbar location is easily changed. Access to changes is available by Right-Clicking on the small Quick Access Toolbar Control and then clicking on the "[Customize Quick Access Toolbar Menu's](#) "Show Above/Below the Ribbon" menu option.

Customizing which menu controls are available on in the Quick Access Tool Bar made available Right-Clicking on a menu control button and selecting Add To Quick Access Control or selecting or deselecting the listed controls listed in the [Built-In Control List](#). To customize the Quick Access Toolbar area use the features available the [Tool Bar Control Editor](#).

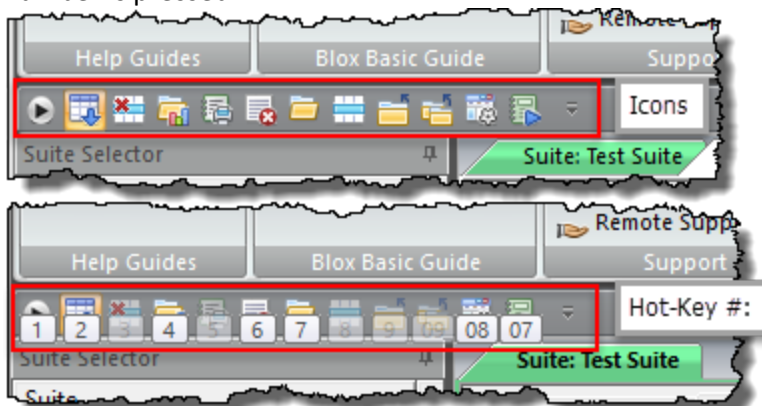
In the [Tool Bar Control Editor](#) Trading Blox provides a [Keyboard Shortcuts Editor](#) where you can assigning keyboard commands to the various menu options. Quick Access toolbar will also respond to its displayed number when the keyboard's "Alt" button is pressed.

Default position is below the Main Menu Tab/Ribbon Controls. [^Top](#)



Quick Access Custom Tool Bar Menu Below Main Menu Ribbon

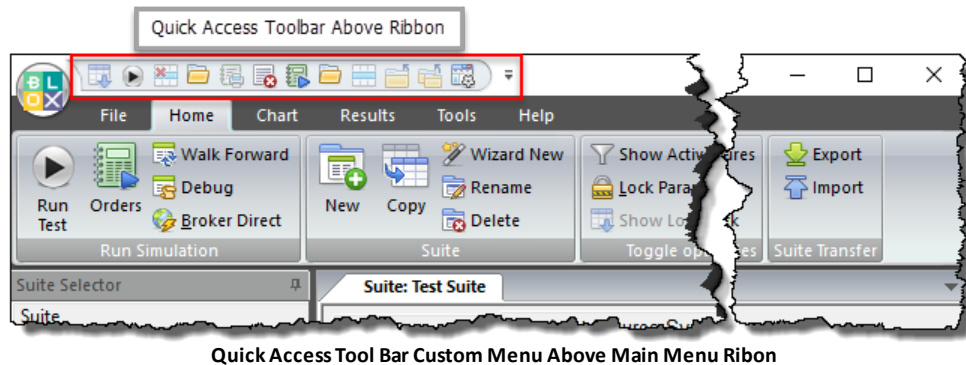
Quick Access keyboard control is available after the "Alt" key is pressed and the displayed bright number is pressed:



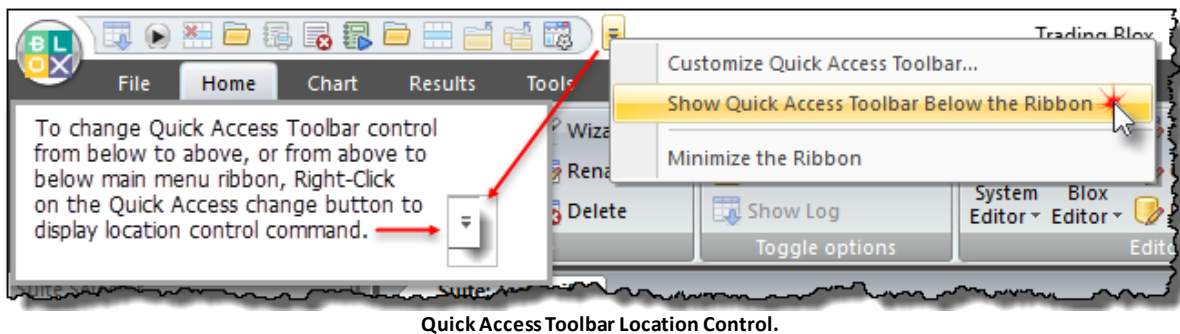
Keyboard Hot-Key numbers appear after "Alt" key is pressed. Pressing any of the bright numbers will execute that Quick Access command action.

Optional position location can be placed in the Title Bar above the Main Menu Tab/Ribbon Control to increase vertical space. [^Top](#)

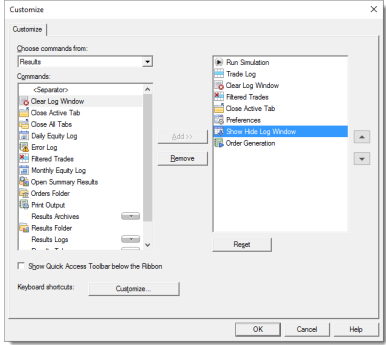
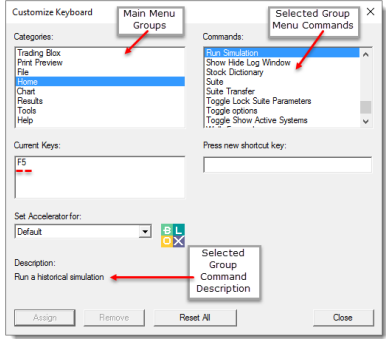
Quick Access Tool Bar Position Control:



Changing the location of the Quick Access Toolbar is a simple process of right-clicking the Quick Access Control button and then selecting the location where you wish to display the Quick Access Toolbar.



Tool Bar Options	Description:	
Quick Access Add: ^Top	Quick Access Toolbar can add an additional item by Right-Clicking on the control button and then selecting the "Add to Quick Access Toolbar."	Quick Access - Quick Add Control Shortcut.
Built-In Control List: ^Top	The number of Quick Access icons shown on the toolbar can be expanded or diminished. When the Customize Quick Access Toolbar menu appears, click on the "More Commands..." selection to change which icons are displayed, and in which left to right order icons are displayed.	Quick Access Built-In Control Options.

Tool Bar Options	Description:
Tool Bar Control Editor: ^Top	Quick Access Control Editor allows the user to add, remove and rearrange where the Quick Access control icons are displayed.  Quick Access Tool Bar Custom Menu Command Options
Keyboard Shortcuts: ^Top	Menu control can be assigned a Shortcut key combination using this editor. Process is easy and fast. Select the control you wish to assigned a shortcut key combination from the menu group, and then from the available commands. then add a combination key sequence and give the description of what that key combination will perform.  Menu Command Short-Cut Key Assignment Editor

Section 2 – Suite Selector List

A Trading Blox Simulation Suite embodies a complete group of settings for a simulation. These settings include global suite parameters, selected systems. Suite Suite Parameters contain the settings for slippage, commissions, and other transaction needs plus many others. System parameters determine the settings specific to each system.

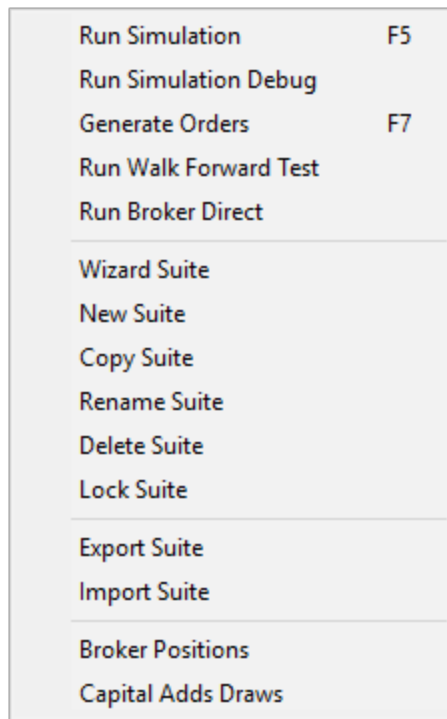
Suites can contain more than one system. Account equity allocations for each systems can be different or set to be the same. The Equity Manager's account amount can use leveraged factor or 1 or greater. Each system in a suite can use a different portfolio, and each portfolio can be any one of the three different instrument classes supported. Any system in a suite can be operated in virtual mode so the information of a virtual system can be made available to other systems in the suite.

All suites will have one active "Global Suite Parameters" tab so that the settings of the suite will apply to all the systems contained within the suite.

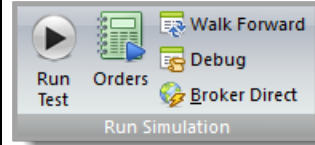
All Simulation Suites let you work on different system setting for the same system or with different logic modules at the same time. For example, you might have one suite that contains the settings for the systems you are currently trading, and several other systems of the same name that contain settings for ideas so as to provide information for ongoing research. Suite are open to a wide range of testing and information collection ideas.

Suite Control Options: ^Top
When a Suite item is selected, right-click the mouse button to display the quick reference display of Home menu commands.

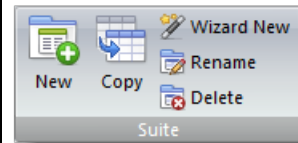
Suite Control Options: [^Top](#)



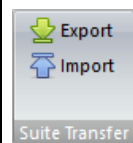
Suite Control Options



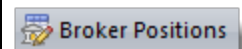
Home Menu Tab Run Simulation Controls



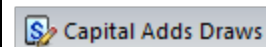
Main Menu Suite Controls



Home Menu Suite Transfer Controls



Home Menu Broker Position Editor



Home Menu Capital Adds & Draws

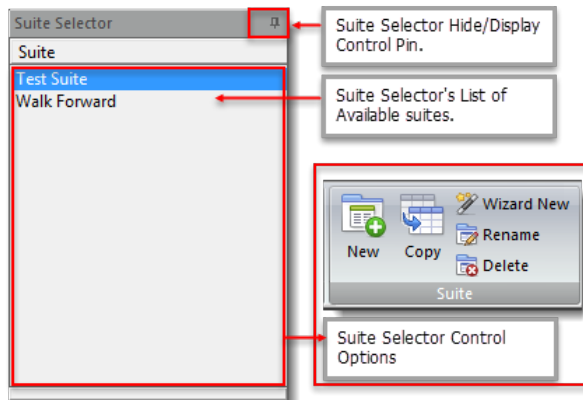
Naming Rules:

Suite, System and Blox module names can include the following keyboard characters:

- All letters letters "A - Z" and "a - z".
- All numbers from "0 to 9".
- The keyboard space " ", hyphen "-", and underscore "_" symbols.

Suite Selector Information & Suite Selector Display Modes:

Always Visible Mode: [^Top](#)



Suite Selector List of Available Suites

[Suite Selector Controls](#) enable the creation of a new suite, changing the name of an existing suite, or permanently deleting a displayed suite name. In the image on the left, the Suite image in the lower right red box area will display the Suite controls in the main screen Home menu tab. Image on the right will appear when any of the suites in the Suite Selector are selected and the mouse right-button is clicked. When the Pop-Up Suite control appears, clicking on any of the options will follow the details shown here.

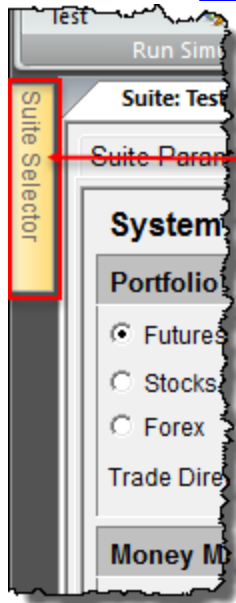
Suite Selector Information & Suite Selector Display Modes:

The image shown displays the Suite Selector List is in "Always Visible," (Docking), mode.

In this display mode the controls for creating a new suite, changing an existing suite name, or removing a suite is easy to manage.

Any Simulation Suite in the [Suite List Area](#) can be select to execute a test and generate orders. Selecting another Simulation Suite will replace the contents of the [System List Area](#) and [Edit Area](#) with the editors corresponding to that Simulation Suite.

Auto Hide Mode: [^Top](#)



Suite & System Selectors are hidden.

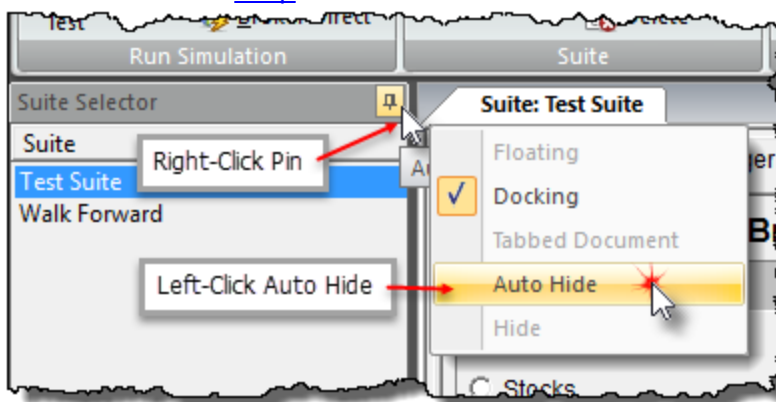
To display list again, click on the Suite Selector tab area.

Suite Selector and System Selector List displays will hide automatically when the Suite Selector control pin is changed from Docking to Auto Hide mode, and the mouse is clicked on an area that isn't in the Suite or System Selector area.

This next image shows how the Suite Selector display can be hidden when other areas in the main screen are being used. For example, if the Suite Selector Hide/Display Control pin is toggle so that the Suite Selector is allowed to hide when there is no Suite Selector or System Selector activity is happening, the Suite and System selector group will fold into the left side of the main screen.

When both list are hidden, clicking on the Suite Selector tab will immediately display the Suite and System Selector list. List will stay visible until the mouse is clicked elsewhere, or an editor is accessed.

Auto Hide Control: [^Top](#)



This next image shows how the Suite Selector display can be hidden when other areas in the main screen are being used. For example, if the Suite Selector Hide/Display Control pin is toggle so that the Suite Selector is allowed to hide when there is no Suite Selector or System Selector activity is happening, the Suite and System selector group will fold into the left side of the main screen.

group will fold into the left side of the main screen.

When both list are hidden, clicking on the Suite Selector tab will immediately display the Suite and System Selector list. List will stay visible until the mouse is clicked elsewhere, or an editor is accessed.

Suite Selector Information & Suite Selector Display Modes:

Button is a toggle that will either condense the list to only show the selected suite's systems, or it will show all the system names available for testing. Click button to show all the available systems. Click to reduce the system list to only the systems in the selected suite.

Selecting another Simulation Suite can change the display contents of assigned systems in the [System List Area](#) and [Edit Area](#) with the editors corresponding to that Simulation Suite.

You can [Create](#) a New Simulation Suite by selecting a Simulation Suite and then pressing the [New](#) button in Home menu's [Suite](#) group area. Clicking on the [Copy](#) button in the same group will display a dialog the name of a new suite can be entered.

[Locking Suites](#)

Simulation Suites can be locked which will disable the editing of all the parameter control settings. This is helpful in preventing inadvertent changes to a system. To [Lock or Unlock](#) a Simulation Suite's parameters, simply press the desired action in the [Home Menu's Toggle Group](#) group area.

When a Simulation Suite's parameters have been locked, the button changes to an **Unlock Parameters** button which can be used to unlock the Simulation Suite. Similarly, the **Lock Parameters** menu item changed to **Unlock Parameters** when the current suite is a locked Simulation Suite.

Note:

This toggle control ability does not prevent the System or the Blox editor from making changes to a system or any of its blox modules. This securing options is just limiting changes to the selected suite's active parameters settings so as to prevent unintended changes when a suite generating active orders that could be disastrous to a trading account.

See Also:

[Show All / Active Systems](#), [Lock Parameters](#)

Section 3 – System Selector List

The System List Area on the lower left side of the screen contains a list the systems are available for testing and editing. When a system is created, it will appear in this list, and be available. Users can activate a particular systems for testing by clicking on the check box next to the system in the system list. Any system name displayed can be added to a suite, and Multiple systems can be in a suite. When multiple systems are in a suite, they don't need to be the same type of system, or be designed to work the same type instruments. All the systems can use the same or different portfolios as needed.

All suites will have one active "Global Suite Parameters" tab so that the settings of the suite will apply to all the systems contained within the suite.

The Trading Blox System Editor can create and change the design of any system. All systems in Trading Blox are a structured list of the blox module perform the task that help to support the design idea of a trading approach. When a system is selected to perform a task it is usually designed to support a specific type instruments (Forex, Futures or Stocks) class.

Almost all the system names in a system list are standard trading systems that operate mostly within the scope of the systems logic. There is a Global type of instruments that is available to suite so that it can take actions a specific stage of a simulation. Global systems most often don't attempt to create orders for entry and exits. Instead, they are usually designed to be manage different types of tasks like data collecting, reporting and suite operational control of the standard trading systems.

System Selector Information:

All Systems Display List: [^Top](#)

System	%
☒ Suite Parameters	
☐ ADX System	0%
☐ ATR Channel Breakout	0%
☒ Bollinger Breakout	100%
☐ Bollinger Counter Trend	0%
☐ Chart	0%
☐ Donchian w Rule Labels	0%
☐ Donchian	0%
☐ Dual Moving Average	0%
☐ Examine_Data	0%
☐ MACD System	0%
☐ RSI Trend Catcher	0%
☐ Stochastic System	0%
☐ Top Ranked Stocks	0%
☐ Triple Moving Average	0%
☐ Turtle	0%
☐ Walk Forward	

System Selector All Systems
Displayed List

Trading Blox default System Selector List display shows all the available system names.

All suites will automatically be assigned a [Suite Parameters](#). These parameters are required because every suite needs the parameter setting information establish in the attached parameters so it will know how to handle the results of system test.

[Suite Parameters](#) are global settings that pertain to each suite and all the systems in the suite. Other suites can have the exact same settings as another suite, if a suite is copied and renamed and the settings are not change. A users can duplicate the suite parameter settings of a different suite when it is created using the [New Suite](#) command when a suite is selected. All the settings established in a suite only pertain to the system(s) attached to that suite.

System Selector Information:

[Walk Forward](#) is a Global System. Global Systems are displayed at the bottom of the [Suite Parameters](#) display just below the [Equity Manager](#) parameters. All blox attached to a global system use different execution timing than blox assigned to a standard testing system. Information about this specific blox is available in [Walk Forward](#) and its control parameters are available in [Walk Forward Parameters](#).

Active System Displayed List: [^Top](#)

System	%
☒ Suite Parameters	
☒ Bollinger Breakout	100%

System Selector Active System List

Trading Blox provides an option to limit the system name list so it only displays the system names assigned to the selected suite.

This feature is enabled in this image. It is controlled by a toggling action of the [Home Menu Show All/Active](#) button. Clicking will change it from displaying all available systems. Clicking it again will limit the display to only show the systems connected to the selected suite.

When All systems are displayed, finding a system is easy because the list is sorted in an ascending alphabetical order that can be scrolled. When it is displaying on the suite's assigned systems, it is easier to understand what systems are assigned to a selected suite.

Global Systems: [^Top](#)

System	%
☒ Suite Parameters	
☒ ATR Channel Breakout	25%
☒ Bollinger Breakout	10%
☒ Donchian	15%
☒ MACD System	35%
☒ MultipleSystemSuite	
☒ Triple Moving Average	5%
☒ Turtle	10%

Active Suite

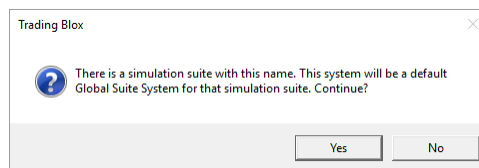
Suite's Global Parameters

Suite's Global System

When a suite has more than one system, each system is allowed to have a portion of the [Equity Manager's Test Starting Equity](#) amount. However, a GSS system is not assigned any equity.

[Allocation percentage distribution](#) to systems is determined by the parameter setting showing the name of a system.

This image's "GlobalSystemSuite" is automatically determined when a system name is exactly the same as a suite name. When that happens this dialog will appear as a reminder that the system being created will be treated as a Global System (GSS).



Global System Creation Reminder Dialog

Section 4 – Simulation Parameter Editor

The Trading Blox application includes separate System Parameter Editors for each of the systems where users can change test values like the number of days in a moving average, the trading portfolio, stop size, etc.

Since the rules for each system are different, the associated parameter values for each system are likewise different. So each system has its own separate parameter editor for editing those system specific values. For information on the parameter values specific to each system, see the [Built-in Systems](#) section.

Suite simulations are at the heart of how Trading Blox develops test results. Understanding the areas that affect test results will be helpful in understanding of what might be helpful to improve testing performance.

Simulation results can change when any of these areas are changed:

- [Global Suite Parameters](#).
- [Systems and their included blox selections](#).
- [Some Global Suit Systems \(GSS\) settings](#).
- [Instrument changes in a portfolio](#).
- [Instrument data updates, or global date ranges](#).
- [Any system parameter setting](#).

System simulation parameters affecting transactions accounting are contained in the [Global Suite Parameters](#) sections. This is where the cost of trading, slippage and account equity is handled. This area is also where the testing period and the type of testing is controlled and adjusted.

System Editing Tabs:

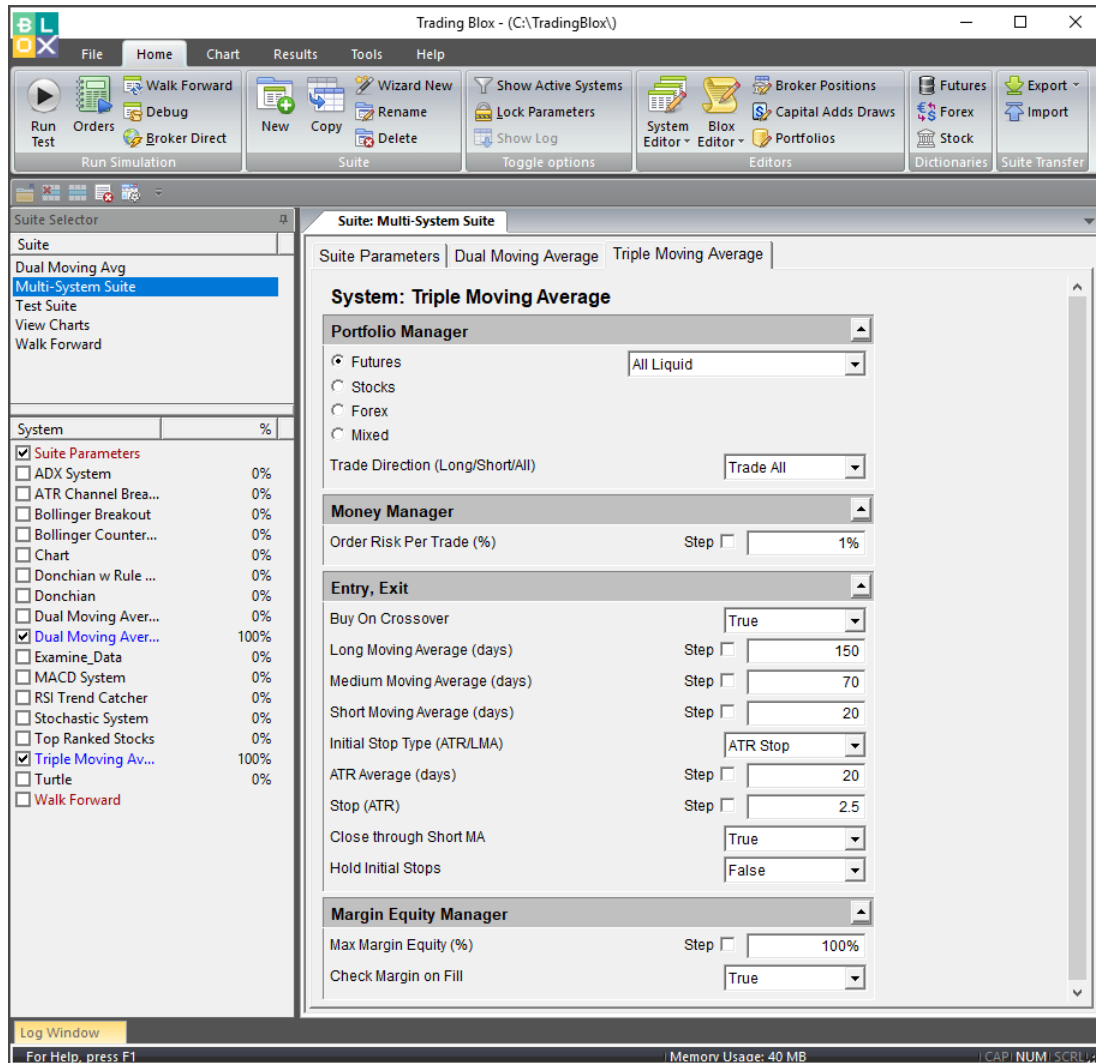
All active systems assigned to a suite have an associated editor tab. Clicking a tab brings up the System Parameter Editor for that system. When system's parameter settings are displayed, they can be changed. Parameters displayed in the main screen area and provide access to all the user adjustable settings of the displayed system. When more than one system is included in a suite, each system name tab can be selected so its settings can be changed.

The majority of the changes that will happen during a system development cycle will be made when this screen is display. Almost all stepped test simulations are created by enabling a parameter's Step option and then entering the starting, ending and step-size values for a stepped test.

When any of the blox in a system are changed, that change will be displayed when the changed system tab is selected so the system blox selections are displayed. In some cases, when a blox doesn't have any parameters, it might not be visible. When a system in suite is removed and replaced by another system, the name on a suite's system tab will change to the name of the replacement system.

When a system is first added to a suite, the parameter values it shows are the default values set during the development of the system.

Click an image Red-Outlined sections for more information:

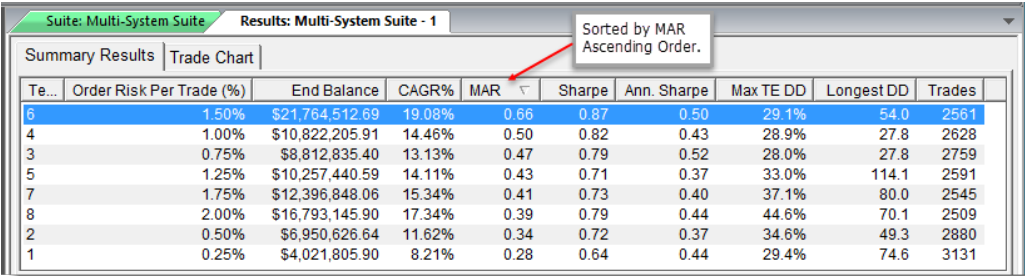
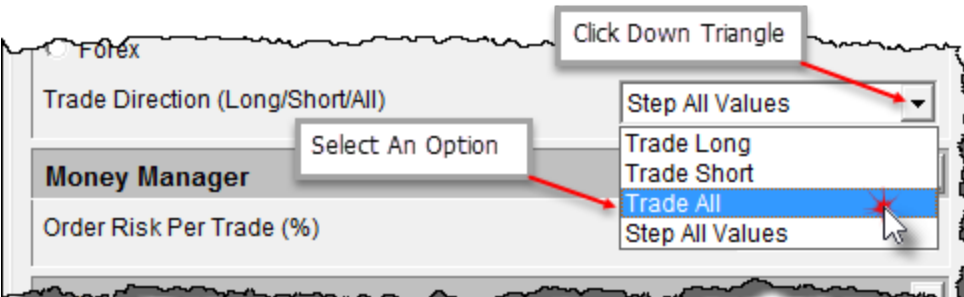


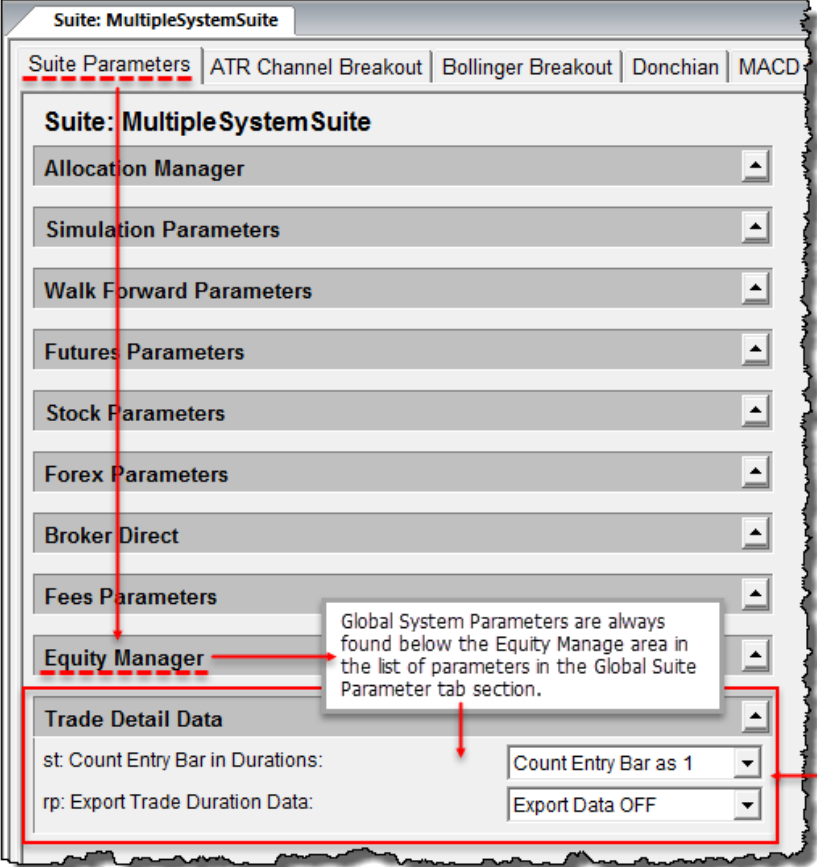
Trading Blox Parameter Simulation EditorScreen

System Selection Tab:	Description:
	Changing the value of a system parameter is a simple edit when it is a number or a text value. When it is drop down list of options, clicking on the down pointed triangle and then selecting the desired setting System number is dependent upon the system's name location is an ascending sorted list of all the names selected for the system. The first system in the sorted list of system names is assigned the system scripting ID of 1. The second system in the sorted list of systems selected for the suite is assigned a system ID of
Dual Moving	Each system tab brings the selected The system's tab name location is dependent upon it system ID number.

System Selection Tab:	Description:
Average: ^Top	System number is dependent upon the system's name location is an ascending sorted list of all the names selected for the system. The first system in the sorted list of system names is assigned the system scripting ID of 1.
Triple Moving Average: ^Top	Suite System. System number is dependent upon the system's name location is an ascending sorted list of all the names selected for the system. The first system in the sorted list of system names is assigned the system scripting ID of 1.
Portfolio Manager :	Description:
Each system in a Suite requires instruments to test and generate signals. A portfolio is designed to have many instruments. When a portfolio doesn't have any instruments selected, or when the files for an instruments symbol in the portfolio are missing, the system will complain must have a portfolio of at least one instrument. When the instruments in the portfolio have a matching file, but the start and end dates of the portfolio don't align with the Suite's Start & End Date setting, this error will be display: "No instruments in the portfolio could be loaded. Check if the instruments' price data is not within the start and end dates of the test, or adjust the test start and end dates to include a wider range."	
Futures: ^Top	Any Futures contract symbol displayed in the Futures Dictionary and also in the Futures Folder of symbol files can be added to a Futures Portfolio name.
Stocks: ^Top	Any Stock share displayed in the Stock's folder files will appear in the Stock Portfolio Editor can be included in a Stock Portfolio name.
Forex: ^Top	Any Forex files displayed in the Forex Dictionary and its file folder can be added to a Forex Portfolio name.
Mixed: ^Top	A Mixed portfolio name can contain any of the three instrument class options. Any instrument in the Futures, Stock or Forex symbol can be assigned in a Mixed portfolio name. Mixed Portfolios are important when it is better to have instruments of various classes contained within the same portfolio. In some systems, this is easier than having a system for each instrument class. However, there are some areas in a system that that only work with on instrument class. For example, the money management methods of Futures only work with Futures. This is also true for Stocks. There are other computations that need to know what the asset class is being processed.

System Parameter Editing:	Description:																																																																																										
Numeric Parameters: ^Top	Changing the value of a numeric system parameter is a simple keyboard edit where the current value is replaced by a new value.																																																																																										
Numeric Parameter Stepping: ^Top	Parameter stepping tests are simple. To initiate a stepped numeric value test, the first step is to determine the starting value, then its ending value, and finally the value that is added to each step. Numeric Step Settings Information In the example above, the test results will report the performance of eight different test steps. Test step counts are determined by taking the ending value minus the starting value and dividing the difference by the step size value. In this case, the difference of 1.75 divided by 0.25 is 7. However, to test all the values a one must be added to get to eight test steps. This table shows the summary test results of how the Multi-system performed when its risk rate was test across the range of risk values of each test step. Initial table of testing results is always sorted in test step incrementing order. <table><tr><th></th><th>Order Risk Per Trade (%)</th><th>End Balance</th><th>CAGR%</th><th>MAR</th><th>Sharpe</th><th>Ann. Sharpe</th><th>Max TE DD</th><th>Longest DD</th><th>Trades</th></tr><tr><td>1</td><td>0.25%</td><td>\$4,021,805.90</td><td>8.21%</td><td>0.28</td><td>0.64</td><td>0.44</td><td>29.4%</td><td>74.6</td><td>3131</td></tr><tr><td>2</td><td>0.50%</td><td>\$6,950,626.64</td><td>11.62%</td><td>0.34</td><td>0.72</td><td>0.37</td><td>34.6%</td><td>49.3</td><td>2880</td></tr><tr><td>3</td><td>0.75%</td><td>\$8,812,835.40</td><td>13.13%</td><td>0.47</td><td>0.79</td><td>0.52</td><td>28.0%</td><td>27.8</td><td>2759</td></tr><tr><td>4</td><td>1.00%</td><td>\$10,822,205.91</td><td>14.46%</td><td>0.50</td><td>0.82</td><td>0.43</td><td>28.9%</td><td>27.8</td><td>2628</td></tr><tr><td>5</td><td>1.25%</td><td>\$10,257,440.59</td><td>14.11%</td><td>0.43</td><td>0.71</td><td>0.37</td><td>33.0%</td><td>114.1</td><td>2591</td></tr><tr><td>6</td><td>1.50%</td><td>\$21,764,512.69</td><td>19.08%</td><td>0.66</td><td>0.87</td><td>0.50</td><td>29.1%</td><td>54.0</td><td>2561</td></tr><tr><td>7</td><td>1.75%</td><td>\$12,396,848.06</td><td>15.34%</td><td>0.41</td><td>0.73</td><td>0.40</td><td>37.1%</td><td>80.0</td><td>2545</td></tr><tr><td>8</td><td>2.00%</td><td>\$16,793,145.90</td><td>17.34%</td><td>0.39</td><td>0.79</td><td>0.44</td><td>44.6%</td><td>70.1</td><td>2509</td></tr></table> Sorted Step # Test Results Table		Order Risk Per Trade (%)	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE DD	Longest DD	Trades	1	0.25%	\$4,021,805.90	8.21%	0.28	0.64	0.44	29.4%	74.6	3131	2	0.50%	\$6,950,626.64	11.62%	0.34	0.72	0.37	34.6%	49.3	2880	3	0.75%	\$8,812,835.40	13.13%	0.47	0.79	0.52	28.0%	27.8	2759	4	1.00%	\$10,822,205.91	14.46%	0.50	0.82	0.43	28.9%	27.8	2628	5	1.25%	\$10,257,440.59	14.11%	0.43	0.71	0.37	33.0%	114.1	2591	6	1.50%	\$21,764,512.69	19.08%	0.66	0.87	0.50	29.1%	54.0	2561	7	1.75%	\$12,396,848.06	15.34%	0.41	0.73	0.40	37.1%	80.0	2545	8	2.00%	\$16,793,145.90	17.34%	0.39	0.79	0.44	44.6%	70.1	2509
	Order Risk Per Trade (%)	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE DD	Longest DD	Trades																																																																																		
1	0.25%	\$4,021,805.90	8.21%	0.28	0.64	0.44	29.4%	74.6	3131																																																																																		
2	0.50%	\$6,950,626.64	11.62%	0.34	0.72	0.37	34.6%	49.3	2880																																																																																		
3	0.75%	\$8,812,835.40	13.13%	0.47	0.79	0.52	28.0%	27.8	2759																																																																																		
4	1.00%	\$10,822,205.91	14.46%	0.50	0.82	0.43	28.9%	27.8	2628																																																																																		
5	1.25%	\$10,257,440.59	14.11%	0.43	0.71	0.37	33.0%	114.1	2591																																																																																		
6	1.50%	\$21,764,512.69	19.08%	0.66	0.87	0.50	29.1%	54.0	2561																																																																																		
7	1.75%	\$12,396,848.06	15.34%	0.41	0.73	0.40	37.1%	80.0	2545																																																																																		
8	2.00%	\$16,793,145.90	17.34%	0.39	0.79	0.44	44.6%	70.1	2509																																																																																		

System Parameter Editing:	Description:
	To see which of the test steps provided the best MAR Ratio performance statistic, click on the column title where "MAR" is displayed until the list is sorted in descending "MAR" values.  Sorted MAR Statistic Test Results Table Reporting stepped performance results changes based upon the number of steps in a test. Information about how to change the reporting is available in the Preference's General Reporting section.
Text Parameters: ^Top	It isn't practical to step a text field, so text entries like those used for a file, or symbol name are don't have a stepping option. To replace the value of a text field, remove the old value and enter a new text value.
Drop Down List Options: ^Top	Drop down list where a two or more options are provided can be stepped by selecting the "Step All Values" option when the drop down list was designed to be stepped. In this next image of the installed portfolio direction manager, the options "Trade Long", "Trade Short", "Trade All", and "Step All Values" are available for controlling in which direction trades will be allowed. For example, if the selection is to "Trade Long" only orders intended as a "Buy" order will be allowed. To limit positions to "Sell" orders, "Trade Short" is the selection. To trade in both the long and short direction, "Trade All" would be selected. To see a performance summary in the same summary table for each of the options, select "Step All Values" for the test.  Selecting Option from Drop Down List Selecting an option from a drop down list is simple. Click on the down triangle on the right of the information field and then click on the desired option.

System Parameter Editing:	Description:
Global System Parameters: ^Top	Global Parameter Editor in Trading Blox allows users to edit global parameters like starting balances, slippage and commission amounts, etc. using the Global Parameter Editor. Global Suite Parameters (GSS) are accessible in the Suite Parameters section just below the Equity Manager section:  Global Parameter Section - GSS Parameters Location

Edit Time: 10/10/2020 10:44:28 AM

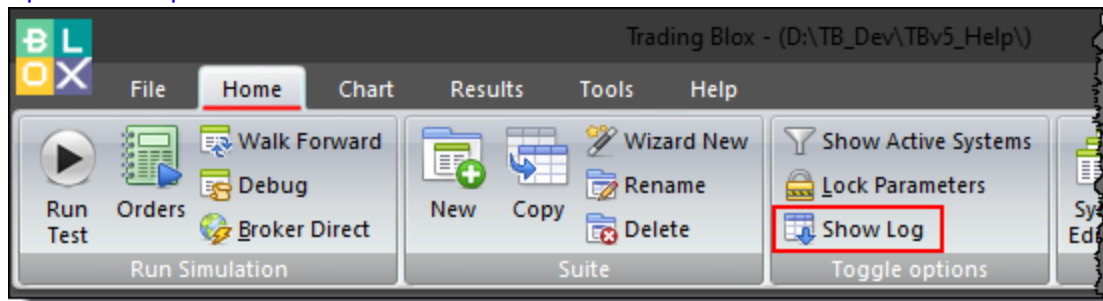
Topic ID#: 202

Section 5 – Log Window

When the Blox Log window is open, any test any messages intended to inform a can appear in the Log Window. If the Log Window is not active, no information will be recorded. When the Log Window is active, and hidden text will be recorded.

Open Log Window:

By default, the Trading Blox **Log Window** is not active or displayed right after a new installation. To open the **Log Window** so it is active, click on the **Log Window** button **Show Log** in the **Toggle Options Group** of the **Home Menu** tab shown next:



To Open the Log Window when it is inactive, click the Show Log button.

The **OpenLogWindow** command can also open the **Log Window** when it is scripted into a system in blox. A system test can also close the **Log Window**, and **ClearLogWindow** when those commands are also executed from a blox.

Auxiliary Window Active:

When the Log Window is active and the **Auto-Hide** option is not enabled (Pin points down), messages will be displayed.



Log Window in Docked-State Display

Text messages can show details of where a system area needs attention. As a diagnostic messages source it is especially good when blox, script section and **LineNumber** references are included. It can also report symbols dates that are not current, or are not accessible to the system.

When the **Log Window** is active and opened, a simulation test will show when a test time at start. During the test, if an error occurs, or the system being tested is using **PRINT** statements to inform the user, that information will appear. At the end of the simulation, the time the test ended, and test duration time will be displayed.

Normal Test Message Items:

Starting Simulation: *start-time*

Simulation: *simulation-duration-time*

Errors: *<Only when errors happen>*

Ending Test Run: *test-end-time*

Overall: *overall-test-duration*

Before each new test is started, the **Log Window** text will be automatically cleared. Text in the **Log Window** can be cleared using the "**Clear Log Window**" menu button, or the text can be cleared using the Trading Blox Basic script function: **ClearLogWindow**.

Log Window Auto Hide:

Trading Blox provides an option to allow the **Log Window** to be active when it is needed, and to hide itself automatically when it isn't needed. When a **Log Window** is **hidden**, it becomes **inactive**, but it increases the main window's available space.

To enable or disable **Auto-Hide**, click on the **Auto-Hide** control **Pin** icon in the right upper side of the **Log Window** shown in the previous image needs to point left to Auto-Hide, and point down to stay open.



The Auto-Hide Log Window in hidden mode increases main screen space when hidden.

The the Log Window is collapsed into its hidden state, this **Log Window** in the image above will open the text area when clicked. When an open **Log Window** appears, it will stay open when the **Auto-Hide** Pin in the **Log Window's Title Bar** point is down.

Log Window InActive:

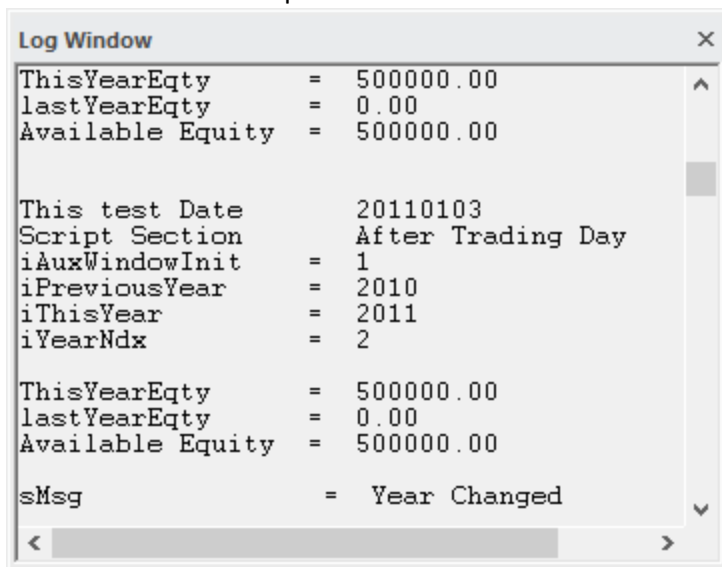
In order for a test **PRINT** statement to send text to **Log Window's**, the text area must be visible. When **Log Window** text area is visible, the **Log Window** is active and **PRINT** information will appear.

When the **Log Window** is **not active**, **PRINT** information sent to the Log Window will be lost, but it will appear in the "**Print Output.csv**" file in the Trading Blox **Results** folder. To obtain the lost text, open the **Log Window**, and then run the test again to see if there are any reported errors. An alternative is to use the **Main Menu's Results Tab** section where a click on the **Print Output** button to open that text file. When the button is used to open the file, it will attempt to open in a spreadsheet program. The file is a text that can be opened by any Text-Editor Window's **Notepad** editor.

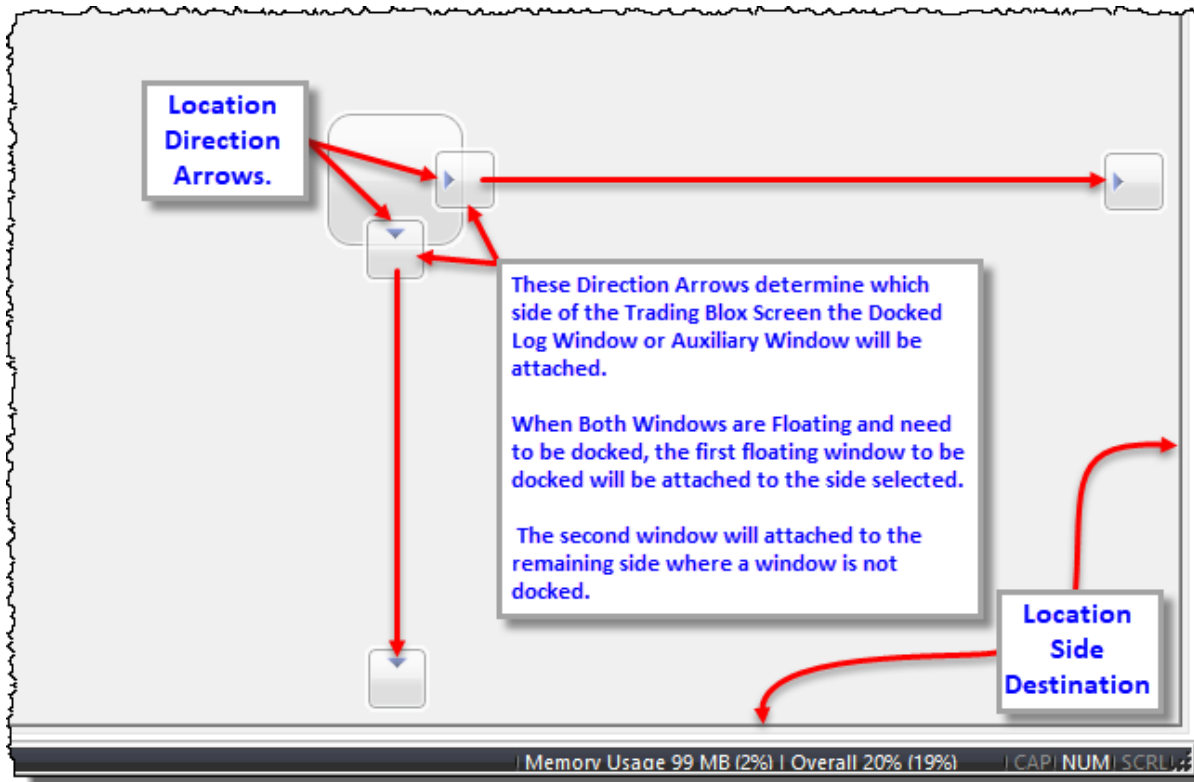
Log Window Floating & Docking Mode:

To change the **Log Window** from **Docked** to **Floating**, the **Auto-Hide** option must be disabled (**Pin** is pointing down). To **float** the **Log Window**, click any place on the **Log Window Title** area and hold the mouse button down. With the mouse button pressed down, you can move the **Log Window**. As soon as it begins to move, it will change size as it pops out of Docked mode. With your mouse pointer still pressed down, you will be able to place the **Log Window** anywhere on

the screen. Once the **Log Window** is placed, you can release the mouse button and change **Log Window's** size and shape.



Log Window in Floating Mode



Log Window Docking Controls

The **Log Window** width and length can be anything you want within the limits of computer's the display screen. To move the **Log Window** to another location, just click on the **Log Window's** Title area again and hold the mouse button down until you move the **Log Window** to a new location.

Docking the Log Window again is simple. Grab the **Log Window Title** area and hold the mouse-down button while you move it of the Trading Blox Main Screen area. As soon as the software detects the mouse location is over its main screen, it will display the **Docking** destination locator arrows. These locating arrows are used to dock the floating window to the right side, or the bottom side of the main screen. When the main screen color changes to a translucent blue shade, release the mouse button and the window will dock to the side of the button that you crossed.

A video in the Trading Blox Forum: [Blox MarketPlace](#), shows how these options work :
[Docked-Undocked-Floating Windows](#)

Edit Time: 2/17/2022 12:19:56 PM

Topic ID#: 164

Section 6 – Auxiliary Window

Trading Blox has an optional **Auxiliary Window** text area that is built-in and available. By default, the **Auxiliary Window** is closed when the software is installed. It will remain closed unless the user opens it, or has it open when Trading Blox is closed.

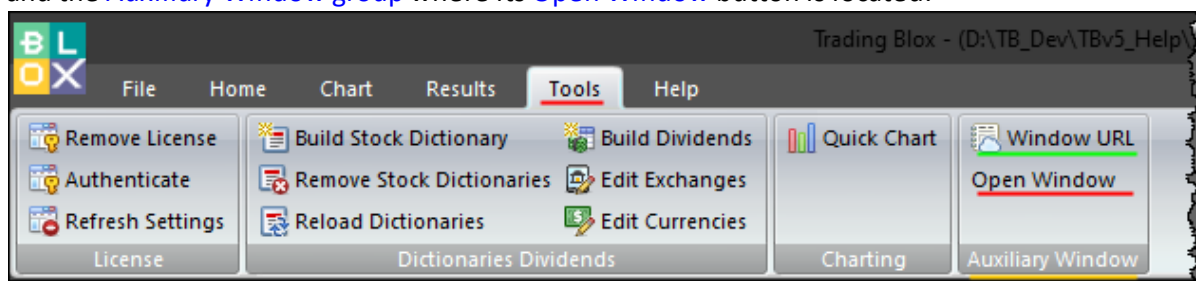
When the Auxiliary Window is opened from a closed state, it will appear in the last location where it was placed. This text window only needs to be active for it to receive text information during a simulation test.

When this window is active and open, it can be a fixed text display, a floating text display, or it can be docked to the right side of the Trading Blox main screen.

Open Auxiliary Window:

By default, Trading Blox **Auxiliary Window** is not active or displayed after a new Trading Blox installation. If the user opens and then later closes the Auxiliary Window prior to exiting Trading Blox, it will be closed when Trading Blox is started again.

To open the Auxiliary Window, click on the **Auxiliary Window** button in the Auxiliary Window group located in the Main Menu's Tools tab. This next image shows the [Home Menu's Tools](#) tab, and the [Auxiliary Window group](#) where its [Open Window](#) button is located:



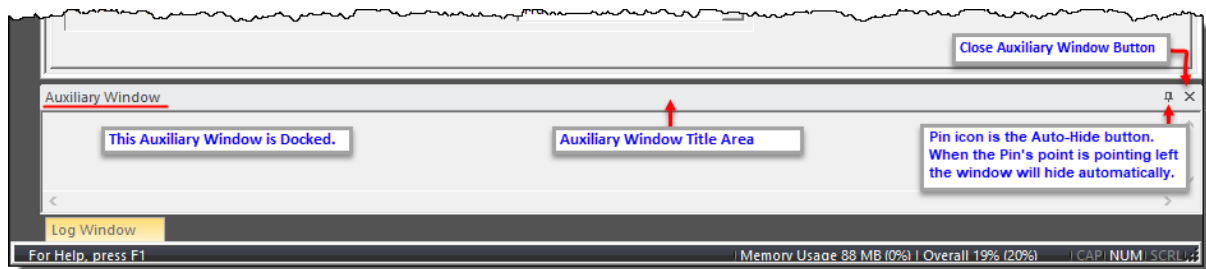
Tools Open Auxiliary Window Menu

The **Auxiliary Window** can also be displayed by the Trading Blox script function [OpenAuxiliaryWindow](#). As long as there is an active presence that the **Auxiliary Window** is on or near Trading Blox, text sent to it will be received. A hidden **Auxiliary Window** is considered to be active.

Auxiliary Window Active:

When the **Auxiliary Window** is not open, it can be opened by clicking on the **Main Menu's Auxiliary Window** button named, **Open Window**. It can also be displayed by a system blox that is designed to execute the [OpenAuxiliaryWindow](#) function.

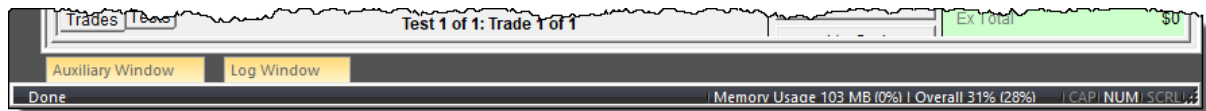
When the **Auxiliary Window** appears on the **Main Screen**, it will display as a **Docked** area (see next image.)



Auxiliary Window Docked-State Display

The above docked **Auxiliary Window** shows the text area open. It also shows it is above an **inactive Docked Log Window**.

The option that collapses the text area and adds a **Open Log** button is the **Auto-Hide Button** that looks like a **Pin**. The **Auxiliary Window** shows the **Auto-Hide Pin Button** shows the Pin's Point is down, which is the **Auto-Hide Off** position. When an Auxiliary Window is active, it will receive text information regardless of its Hidden or Open state.



Auxiliary & Log Windows are docked on the main screen and are hidden by their Auto-Hide option being active.

To open an Auxiliary Window when its **Auto-Hide** option shows the text area hidden, click the **Auxiliary Window** button to open the text area. To keep the text area open, change the direction of the Auto-Hide Pin to point down.

An **Auxiliary Window** can be closed and made inactive by clicking on the **"X"** button at the far right side of the window's **Title Bar** area. It can also be closed by a blox script that calls the function: [closeAuxiliaryWindow](#) script.

Auxiliary Window Messages:

Text information can be sent to the **Log Window** by adding the **PRINT** function to **"text character statement"**.

Text information is created for the **Auxiliary Window** by the person who captures information from a simulation that is converts into a line of text information. When the scripting process finishes the message, it sends the information to an indexed row in a **STRING** series.

When the simulation reaches a stage where it is ready to send the completed message, it knows it must be first sent to a text type row in a **STRING** series. Each text **STRING** series has many rows where it can save many messages. The number of the messages it can handled, is determined by the person who is creating the messages.

At some point the logic in the blox will decide it is time to send the stored messages to the Auxiliary Window. Sending all the stored messages happens in one step when the function [SetAuxiliaryWindowText\(stringSeries\)](#) is executed.

Auxiliary Window Floating & Docking Mode:

To change the **Auxiliary Window** from **Docked** to **Floating**, the **Auto-Hide** option must be disabled (**Pin** pointing down).

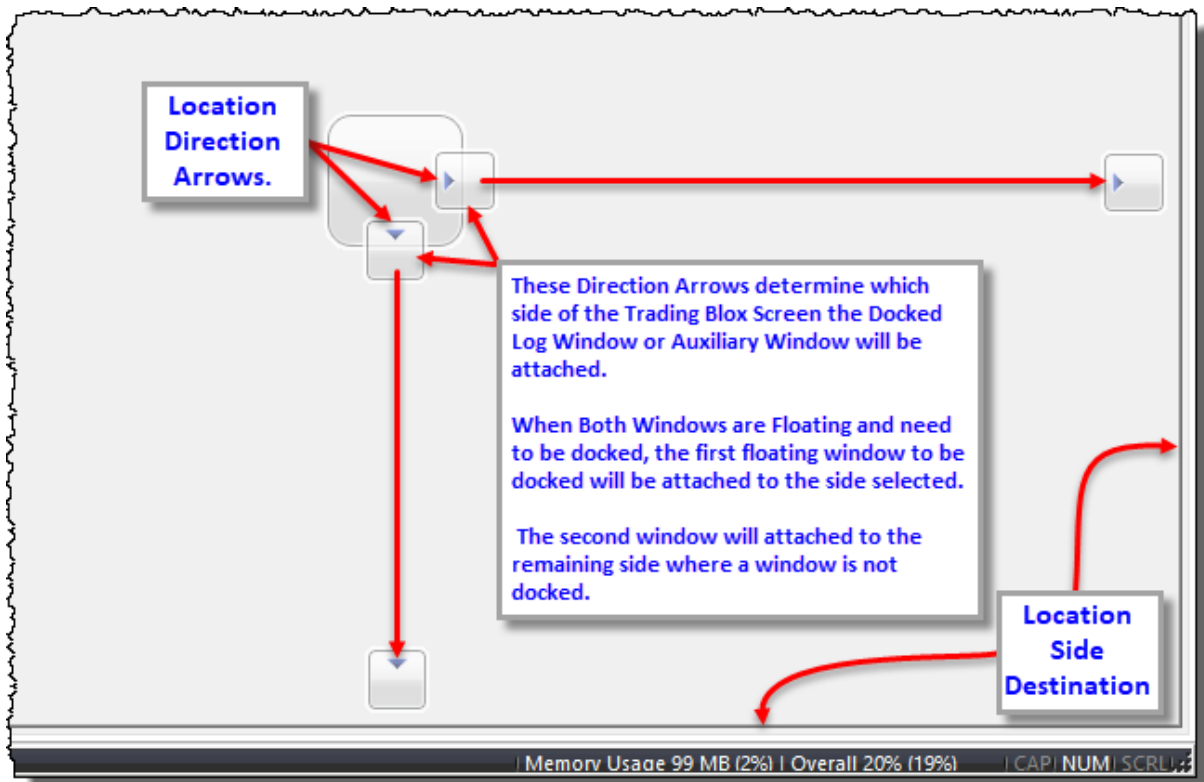
To undock the **Auxiliary Window** so it can be moved, click any place on the Title Bar area and hold the mouse button down. With the mouse held down, move the **Auxiliary Window** a little. As soon as it is moved, it will change size as it pops out of docked mode.

Continue to move the **Auxiliary Window** with your mouse pressed down until you have it where you would like it. If you change your mind about its location, click on the Title Bar again and move it to another location.

Once the **Auxiliary Window** is where you want it, you can change its size and shape just like you change other dialogs or software displays.

Auxiliary Window			
For Year-End	2010,	Total Equity	= 0.00
For Year-End	2011,	Total Equity	= 502088.38
For Year-End	2012,	Total Equity	= -17721.94
For Year-End	2013,	Total Equity	= 516393.06
For Year-End	2014,	Total Equity	= 25075.32
For Year-End	2015,	Total Equity	= 532434.68
For Year-End	2016,	Total Equity	= 29047.47
For Year-End	2017,	Total Equity	= 522996.54
For Year-End	2018,	Total Equity	= 38983.50
For Year-End	2019,	Total Equity	= 504870.58
For Year-End	2020,	Total Equity	= 99397.44

Auxiliary Window in Floating Mode



Log Window Docking Controls

Docking the Auxiliary Window again is simple. Grab the **Auxiliary Window Title Bar** area with the mouse and hold it down. With the mouse held down, move the window until the mouse pointer is over the main screen area. As soon as Trading Blox recognizes the mouse is over the main screen, it will display the Docking direction and destination locators. As the mouse nears a Docking Location Direction boxed arrow, the screen color will show a translucent blue shade. Release the mouse button and the window will dock to the location the boxed arrow button that your mouse triggered.

These locating arrows are used to dock the floating window to the right side, or the bottom side of the main screen. The first window to be docked, will be the window that is closer to the edge where it is docked. The second window to dock, will be above or to the left of the window that is

docked on that side. Each of the two text area windows can be docked on the same side, or each can be docked on different sides. Both can be floating at the same time.

A video in the Trading Blox Forum: [Blox MarketPlace](#), shows how these options work :
[Docked-Undocked-Floating Windows](#)

Edit Time: 10/21/2020 11:43:18 AM

Topic ID#: 253

Section 7 – System Information

A system in Trading Blox is a collection of blox modules saved as a list of blox file names. Most often the system name is a recognizable name that informs the type of system being selected. Each blox module in a system will have specific design and perform specific tasks. Some modules can have more abilities, and some can have similar logic that does some of the same task but has different rules or logic. When a system is assembled and saved, its name is displayed in the system list on left side the Editor's screen. The name of all the blox modules in the system are displayed in the center column where the various types of blox modules are listed. In the right most list of items are the names all the blox modules known to that specific installation of Trading Blox. Access to the scripted logic in any of the blox is available when a blox is opened in the [Blox Basic Editor](#).

Suites:

Suites must have at least one system in order for anything to execute. However, Suites can have more than one systems. Each system in a suite can perform different types of testing using different classes of instruments, and each with the same or a different portfolio. When a system is selected for a suite, its name is saved as a system reference.

System Numbers:

When more than one system is in a suite, those additional systems are added to the suite in alphabetical order, which is also the sequence in which system numbers are assigned. For example, when more than one system is added to a suite, the system names in the suite file are sorted in ascending alphabetical order. The system name that is assigned the left most tab is the first system in the list of suite sorted system names. In simple terms, if the system names are ASystem, BSystem, CSystem, etc. ASystem will be assigned system number one, BSystem will be system number two, CSystem will be system number 3, and any other names will get the next integer number until all the systems are assigned a identification number.

System Testing:

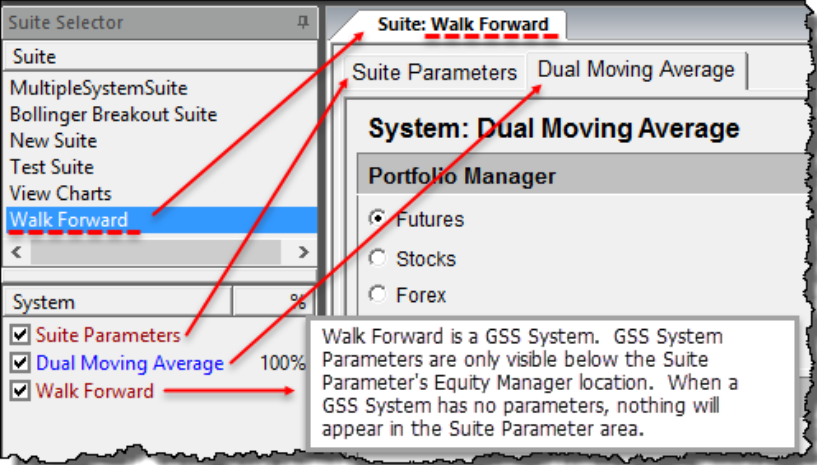
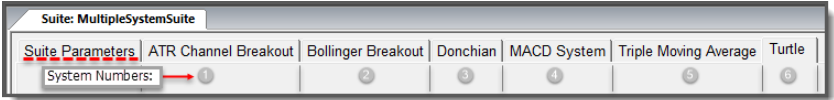
Suites test systems in a looping process that begins with testing the lowest system number up to the highest system number. Each system is exercised for each test date determined by using the earliest date to the latest date. By testing each system on each date before the next test date is tested creates a horizontal date test record for each test date. It is also how we trade the markets.

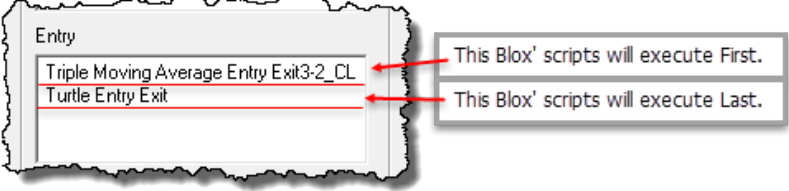
System Type Information:

Suites can have three different type of files listed as a system.

System	%
☒ Suite Parameters	
☒ Dual Moving Average	100%
☒ Walk Forward	

Different System List Records

Suite Name Types:	Descriptions:
Global Suite Parameters: ^Top	All suites must have a global Suite Parameter assignment. Global Suite Parameters control all testing and trade result operations in its assigned suite. Access to each of the Global Suite Parameters always displayed in the tab area of the main screen where system parameters, and test result tabs are displayed after a simulation test:  GSS System Parameter Display Details When more than one system is in a suite, the settings in the Global Suite Parameter section will control those other systems as well. Multiple system suites will display their system name in a tab where each system name is listed in alphabetical order from left to right.  Suite's Alphabetical Ascending System Name Order When a Global System is also included in the list of names, access to the Global System blox parameters will be available below the Equity Manager section at the bottom of Global Suite Parameter section.
Global Suite System (GSS): ^Top	When the system name that has the same name as the selected suite, the system is automatically treated as a Global Suite System (GSS). Global Systems do not have a display tab. Their parameters are available at the bottom of the Global Suite Parameter section, and their settings will apply to all the systems in the suite. A Global Suite System is colored red in the System Selector List. All blox modules use the same scripting section names for standard systems and for GSS systems. However, there is a difference in the timing of when a standard system blox section executes its code for some of the script sections. In a GSS assigned blox module some of

Suite Name Types:	Descriptions:
	the script section will execute ahead of the standard script section, and some will execute after the standard system script sections. Some script section in a GSS assigned blox module won't execute at all. A table showing when standard system blox and GSS assigned blox script section execute is available in Blox Builder Help information. To find the table, enter the topic name in the Help file's Search field: Global Script Timing.
Standard System: ^Top	All standard systems will execute all the scripts in the blox where code is shown in a script section. Blox script timing execution information shows which script execute, and the conditions that influence enable or disable their execution. To learn which scripts have conditional executions, review the Blox Script Timing topic in the the Blox Builder Help file. When a system contains more than one blox of the same type entered into a system, the blox in each of the sections will execute in the order they are displayed. Note: When adding more than one type of blox to a system, i.e Entry Exit, or Auxiliary, the names of the blox will sorted alphabetically. Sorting happens after the blox is added to the system list and the OK button in the System Editor is clicked to close and save the work. This means that the first blox listed when the system is opened again may change after it is opened. However, the first blox listed above the other blox in a system section will execute the scripts with the same name first. Scripts in the next blox name listed will execute their same name scripts second.  In the above image each blox will have an Entry Orders and an Exit Orders script sections. The Entry Orders script in the first blox will execute first, and then Trading Blox will execute the Entry Orders section in the next blox right after the first Entry Orders script section reaches the end of the code statements in the Entry Orders script section of the first blox.

Suite Name Types:	Descriptions:
	This is also how Exit Orders, and Auxiliary blox script section with the same name will be timed for execution.

Edit Time: 10/8/2020 4:06:00 PM

Topic ID#: 213

Section 8 – Trade Chart

When the Trading Blox Preference [Log Trades](#) option is enabled, the Trade Chart will display the trades of a test. Specific details of selected trade will be displayed the [Trade Details](#) panel. Indicators designed to appear will be displayed and their values are visible in the Indicator panel. Information for any location in the chart area is displayed in the [Price Data](#) panel.

When some of the information in [Trade Details](#) is unselected, the Trade Details display window will show less information.

When the [Log Trades](#) option is unselected, there will be no trade information to display and the Trade Chart tab will be missing from all selected test results that were performed when [Log Trades](#) option was disabled.

When "[Log Trades](#)" in Preferences are enabled, the Chart Area shows these areas:



Click Test Trade Results Information Area

A complete list of each trade from each test is shown below the chart. When you click on a trade in the list, Trading Blox will select that trade and display it on the chart. You can sort the list by clicking on any of the column headers. For example, you can easily determine the largest percentage losing trades by clicking on the Profit % column. Or, you could sort the symbol column so all the trades for a symbol are grouped together. When a test has multiple steps from incrementing one or more parameters, the trades each step can be selected in the Test section of the trade record area.

Trade Chart Area: [^Top](#)

Entry and Exit Dates

The entry and exit dates for the selected trade are shown below the chart date legend. These show the exact date and time of entry (Open, Day, or Close).

Trade Chart area displays the selected system trade record price data, indicators plots and Entry--Exit locations. How Entry and Exit are displayed on the chart can be changed by the controls in the main menu [Chart Menu the Chart Trades Group](#). This set of controls can change whether all the Entry and Exit locations or a just the selected trade Entry and Exit location is displayed.

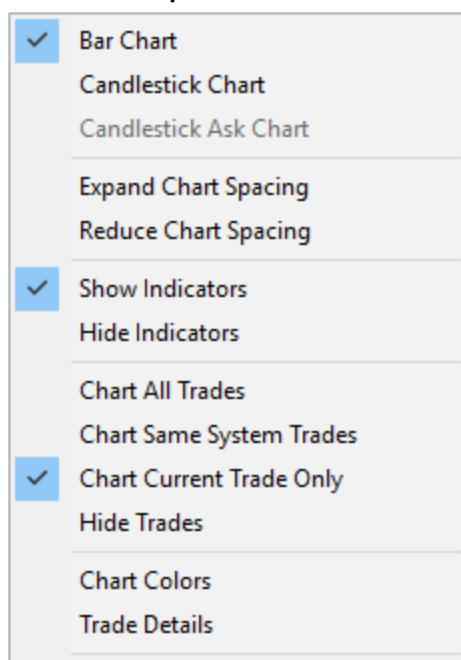
Entry and Exit Prices

The trade Entry, Exit, and initial Stop are plotted so you can visually see the exact trade prices.

Price and Indicator Information

The selected trade is plotted on a daily bar chart graph of the prices for the particular market or stock issue. The indicators for the system that generated the trade are also plotted on the graph so you can see how the system indicators control trade entry and exit. The Chart menu and associated buttons on the tool bar control which trades are displayed on the graph.

Chart Control Options



Trade Chart Display Controls

Chart information controls are available by clicking the right-mouse button so the control options shown on the left side appear. These options are text versions of the main menu Chart tab. Clicking on any of the controls will create a change when the control panel closes.

Roll-Forward and Dividend Distributions

Roll-Forward and Dividend Distributions

Stock and Futures will display the date markers for

Futures roll-forward  events, and Stock Dividend distribution  events in the area just above the chart's horizontal date labels.

Multiple Suit Steps

Trade Chart Area: [^Top](#)

When a suite creates multiple steps each step can have different results and show different trades. This ability is controlled by the settings [Reporting General Test Reporting](#) sections where [Small](#) and [Large](#) simulation test determine when the [Log Trades](#) option is enabled.

#	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE DD	Longest ...	Trades
1	\$11,160,550	14.66%	0.3...	0.813...	0.41963	38.1%	45.9	2618
2	\$152,516	-10.11%	-0....	-0.71...	-0.74019	85.4%	211.5	544
3	\$10,822,206	14.46%	0.5...	0.816...	0.42931	28.9%	27.8	2628

Trades Tests **Test 1 of 3: Trade 2632 of 2632**

Suite Stepped Test Chart Display Selection Table

Test results where the Log Trades option was enabled during its multiple steps will have a record in the Test Table tab area. Each record listed will have [Trade Records](#) that can be selected and displayed in the chart area.

NOTE:

Before creating a parameter stepping simulation, the user should review the settings information established in the Preference section where Trading Blox [Reporting General](#) Trade Log decisions for [Small](#) and [Large](#) Simulation settings determine the information made available for Trade Chart viewing are established.

Chart Display Controls: [^Top](#)

Controls to change how information on a chart display are the Main menu's Chart tab section:

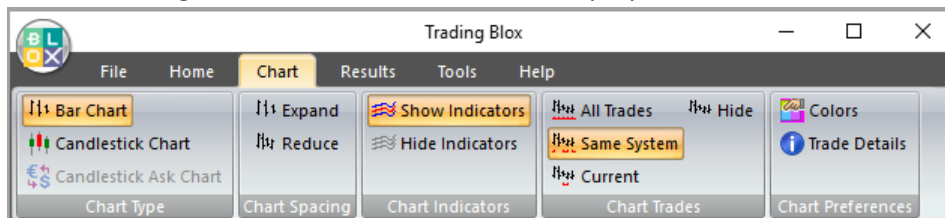


Chart Display Controls – Click to see more information.

Above display controls allow single-click access to common chart display controls that include: expanding and collapsing the width of a price bar; showing and hiding the trade indicators; and controlling how many trades are displayed at the same time; and others.

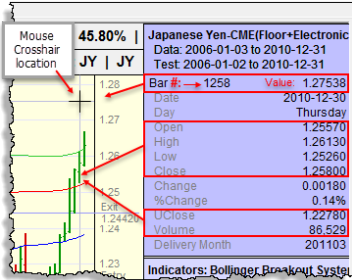
Trade Records:[^Top](#)**Description:**

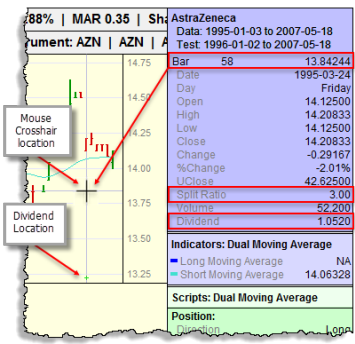
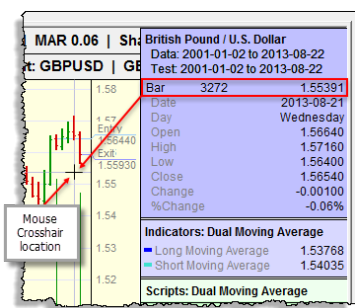
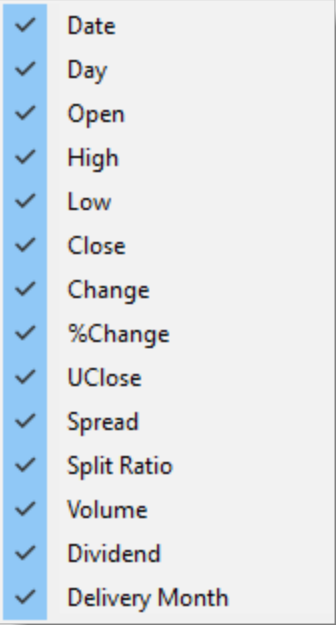
System	Symbol	B Sym	Desc	Pos	Unit	Qty	Entry Date	Entry Fill	Profit	%	Exit Date	Exit Fill	Rule
ETF_Relative...	CURE	CURE	CURE	L	1	3,552	2014-02-03 Open	17.65005	\$78,651	24.4%	2015-05-29 Open	39.79280	Le@On_Open,
ETF_Relative...	BIB	BIB	BIB	L	1	668	2014-03-03 Open	48.37070	\$-13,762	-4.0%	2014-03-31 Open	38.07000	Le@On_Open,
ETF_Relative...	XLI	XLI	XLI	L	1	1,190	2014-04-01 Open	52.55640	\$202	0.1%	2014-07-31 Open	52.47620	Le@On_Open,
ETF_Relative...	DSI	DSI	DSI	L	1	856	2014-04-14 Open	68.70340	\$946	0.3%	2014-04-30 Open	69.80880	Le@On_Open,
ETF_Relative...	DRN	DRN	DRN	L	1	1,102	2014-07-01 Open	14.99225	\$-6,091	-1.8%	2014-09-30 Open	13.61055	Le@On_Open,
ETF_Relative...	XLV	XLV	XLV	L	1	1,075	2014-08-01 Open	60.78000	\$10,495	3.1%	2015-01-30 Open	70.05200	Le@On_Open,
ETF_Relative...	BIB	BIB	BIB	L	1	1,318	2014-10-01 Open	51.83040	\$48,465	13.7%	2015-05-29 Open	88.60220	Le@On_Open,

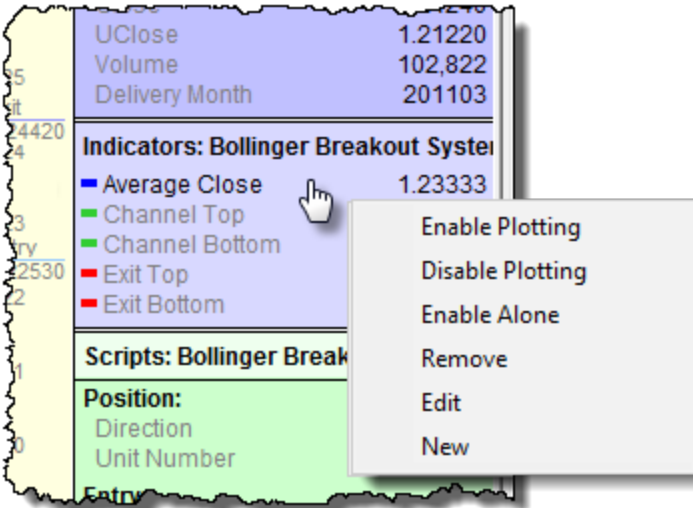
Trades Tests **Test 1 of 1: Trade 195 of 217**

Example Trade Record Table.

Trade Records: ^Top	Description:
System	Unit's System Name.
Symbol	Unit's Symbol.
B Sym	Unit's Broker Symbol.
Desc	Symbol's market description.
Pos	Trade's position direction.
Unit	Trade's Unit number.
Qty	Unit's trade contract or share quantity. Right clicking on the trade's quantity value will open the Trading Blox UNIT SIZE SCRIPT section where the scripted logic is displayed.
Entry Date	Unit's Entry Date. Right clicking on the trade's Entry Date will open will open the Trading Blox ENTRY ORDERS SCRIPT section where the scripted logic is displayed.
Entry Fill	Unit's Entry Fill price. Right clicking on the trade's Entry Date will open will open the Trading Blox ENTRY ORDERS SCRIPT section where the scripted logic is displayed.
Profit %	Unit's profit percentage rate.
Exit Date	Unit's Exit Date. Right clicking on the trade's Entry Date will open will open the Trading Blox EXIT ORDERS SCRIPT section where the scripted logic is displayed.
Exit Fill	Unit's Exit Fill price. Right clicking on the trade's Entry Date will open will open the Trading Blox EXIT ORDERS SCRIPT section where the scripted logic is displayed.
Rule	Unit's order information. An <u>Exit Rule</u> is added to positions that are forced to exit by the application: • Broker Position Exit is caused by a new broker position that must exit the exiting open position, or a broker position of OUT. • End of Data is an exit on the last bar of data when Running a Simulation. These positions are not exited when Generating Orders.

Trade Records: ^Top	Description:
	• Test End Exit is an exit at the end date of the test (based on the test start and end dates), when the last bar of data is not reached.
Trade Record Controls: ^Top	Description:
 Record Buttons	Each of the six buttons provide make moving through the trade records simple. Each button's name has an active letter key that makes allows the user to focus on the display of trades simple.
trade <u>E</u>ntry	Press the <u>E</u> key, or click the button to display the entry location.
trade <u>e</u>Xit	Press the <u>X</u> key, or click the button to display the exit location.
<u>N</u>ext by symbol	Press the <u>N</u> key, or click the button to display the symbol's next trade.
<u>P</u>rev by symbol	Press the <u>P</u> key, or click the button to display the symbol's previous trade.
next by <u>S</u>ystem	Press the <u>S</u> key, or click the button to display the next trade.
prev by <u>s</u>Ystem	Press the <u>Y</u> key, or click the button to display the next trade.
Price Data Information: ^Top	Description:
Trade record selected in the Trade Record List will display information about the trade and the market where the trade was positioned.	
Cross-hair Details: ^Top  Future's Price Information at Mouse Location.	Information displayed on the chart can be viewed in detail in the price data window on the right side of the chart. Moving the mouse over the chart details will be updated as the mouse moves. Moving the mouse from sideways will show date changes, and moving the mouse up or down will display the price scale value changes. As long as the mouse is above the price area of the chart the chart's data window will be displayed. Information displayed in the window can be held by stopping the mouse from moving. To lock the mouse location, double-click the left mouse button. Double-click the mouse again to regain mouse movement.

Price Data Information: ^Top	Description:
 Stock's Price Information at Mouse Location.  Forex' Price Information at Mouse Location.	Data displayed in the price data window is dependent upon instrument type and the the price window settings. To see the available fields of information and those enabled, right-click on the price window and this list of data fields will appear:  Data Window Price Information Data Selector.
	To enable or disable the display of data field item, left-click on the item to enable or to disable. After an item has been clicked, the list of data fields will close. Right-click in the data window area to display the settings again. Which fields that can be displayed is control by the instrument displayed in the chart area. This means that Futures and Forex instruments cannot display information the Split Ratio, Dividend information. Stocks cannot display the Delivery Month information. Information not available for an instrument will be hidden from the data window regardless of whether that information is enabled. When a data item is disabled from being displayed, the price data window vertical size will be reduced. As the price data window shrinks, information below its location can increase when the windows in that area are not fully displayed.

Indicator Information ^Top	Description:										
Trading Blox Indicator Detail window default setting shows system indicator values aligned with the bar last used to calculate the value shown for current mouse position (See Offset Plot Ahead One Bar). Custom indicators can be created to show an indicator's calculated value from an earlier period so that it appears at a later price location.											
Indicator Display Information:  The screenshot shows a window titled 'Indicator Display Control Example.' with a list of indicators. A right-click context menu is open over the 'Average Close' indicator, showing options: 'Enable Plotting', 'Disable Plotting', 'Enable Alone', 'Remove', 'Edit', and 'New'.	When a suite is selected and then tested, the chart display (see Log Trades) it will display indicators designed to appear in the chart area. Indicator appearance changed using the chart's indicator plotting control. Indicator plotting control will appear as a pop-up display when you right-click on the indicator detail window. To change an indicators, click on the indicator name and then right-click and select a change action item.										
Indicator Display Control: <table border="1"> <thead> <tr> <th>Change Item:</th><th>Description:</th></tr> </thead> <tbody> <tr> <td>Enable Plotting</td><td>When an indicator is not displayed on the chart, but appears in the indicator window, selecting this change item will enable the plotting information to appear on the chart.</td></tr> <tr> <td>Disable Plotting</td><td>Select this change item when an indicator is listed in the indicator window and it is not plotting on in the chart area.</td></tr> <tr> <td>Enable Alone</td><td>This change item will hide the all the other indicators on the chart except for the indicator that is selected when this change item is enabled.</td></tr> <tr> <td>Remove</td><td>Use this change item remove an indicator's plotting from the chart</td></tr> </tbody> </table>		Change Item:	Description:	Enable Plotting	When an indicator is not displayed on the chart, but appears in the indicator window, selecting this change item will enable the plotting information to appear on the chart.	Disable Plotting	Select this change item when an indicator is listed in the indicator window and it is not plotting on in the chart area.	Enable Alone	This change item will hide the all the other indicators on the chart except for the indicator that is selected when this change item is enabled.	Remove	Use this change item remove an indicator's plotting from the chart
Change Item:	Description:										
Enable Plotting	When an indicator is not displayed on the chart, but appears in the indicator window, selecting this change item will enable the plotting information to appear on the chart.										
Disable Plotting	Select this change item when an indicator is listed in the indicator window and it is not plotting on in the chart area.										
Enable Alone	This change item will hide the all the other indicators on the chart except for the indicator that is selected when this change item is enabled.										
Remove	Use this change item remove an indicator's plotting from the chart										

Indicator Information ^Top	Description:	
		and from the indicator details window. When an indicator is removed from test result display it won't appear again until another test is performed.
	Edit	This change item will bring up the indicator's creation dialog where changes can be performed.
	New	This control will open a indicator dialog where a new indicator can be added to a system. Indicator will appear when the system is then tested after the new indicator is created.
	Offset Plot Ahead One Bar: Indicators can be offset on bar ahead to enable the value of the indicator calculated today to display its plot in the market's next price location. This can be changed by using the Trading Blox Blox Basic Editor, or by using the Indicator Display Edit option.	

Trade Details: ^Top	Description:
Scripts: Bollinger Breakout Position: Direction Short Unit Number 1 Entry: Date 2016-10-17 Open Quantity 28 Order 126.25 Fill 126.25 Stop 128.03 Risk 1.17% GBP 1,218.1000 Exit: Date 2016-10-18 Close Quantity 28 Order 126.81 Fill 126.81 GBP 1,229.4000 Profit: Points -0.56 Commission \$0.0000 Profit \$-688.4640 Position \$-19,276.99 Percent % -0.37% R-Multiple™ -0.32 Equity: En Trading \$5,184,487 Ex Closed \$5,159,573 Ex Total \$5,291,615 Miscellaneous: Drawdown % 25.34% Days 1 Bars 1 Selected Trade Information Window.	Each selected trade will show all the information enabled in the Trade Records table. This trade information is always displayed in the Trade Details window area. This display area shows the details like: - entry and exit dates and prices, - profit, - profit %, - entry risk %, trade duration, etc. When the displayed screen is too small to display all the information, move the mouse cross-hair from the price area of the display so the Price Data and the Indicator Information panels

Trade Details: ^Top	Description:
	disappear. Removing these two panels allows more lines of trade information to appear when those two panels are not displayed. Moving the mouse back over the price display area will enable the Price Data and the Indicator Information panels to appear again.

Trade Details: ^Top	Description:
Trade Details Position ☒ Long or Short ☒ Unit Number ☒ Quantity Entry ☒ Date ☒ Order Price ☒ Fill Price ☒ Stop Price ☒ Risk % Exit ☒ Date ☒ Order Price ☒ Fill Price Profit ☒ Profit Points ☒ Profit ☒ Position ☒ Profit Percent ☒ R-Multiple Equity ☒ Entry Trading Equity ☒ Exit Closed Equity ☒ Exit Total Equity Miscellaneous ☒ Drawdown % ☒ Days Check each item that you want to appear in the Trade Details Section . Trade Detail Preference Setting Options	A click of the right-mouse button anywhere on the Trade Details panel will display Trading Blox Preference's Trade Details information options. Default is to enable all the available trade information fields. When a trade is displayed on the chart and the mouse right-click displays the Preference's Trade Details information options, any selection changes made to those options will immediately be displayed in the Trade Detail panel after the Preference's Trade Details display is closed.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 4101

Global Suite Parameters

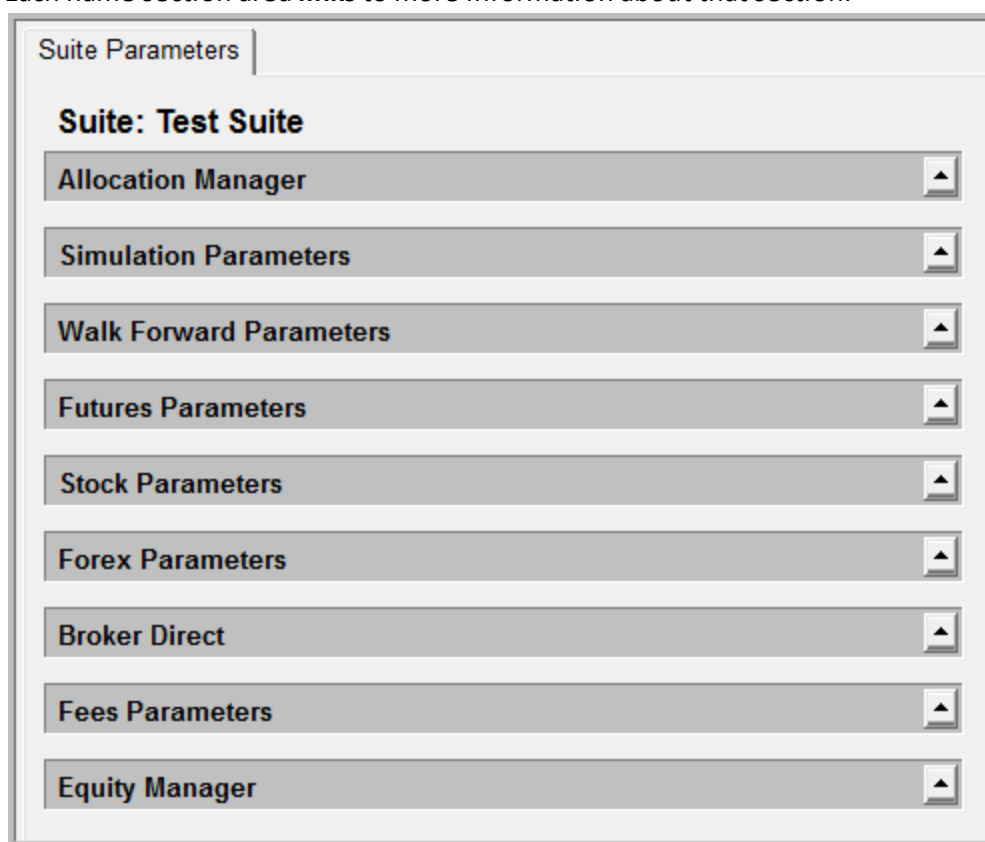
Part

IV

Part 4 – Global Suite Parameters

Suite Parameters are Global setting parameters for all the systems selected for a suite. These settings control the test dates, allocation rates, initial equity, trade expenses and slippage and others test settings in a suite. In Trading Blox version 5 there are some differences in how the Global Parameters, which are now Global Suite Parameters can affect the operation of a simulation and an order generation. On this [Upgrade Information](#) topic is a list the changes made to this major Trading Blox release version.

Each name section area **links** to more information about that section.



Link to more information about each Global Parameter section.

Links:
See Also:

Section 1 – Allocation Manager

Allocation Manager parameters are only visible when a selected suite is selected to have one or more systems for testing or order generation.



Allocation Manager example of a two systems suite.

- The Allocation Manager area will show the names of all the system selected in the suite.
- A minimum of one system is required for a test and for the Allocation Manager to display any system name.

Allocation Percentages:

Each system in a suite is allocated a starting equity amount equal to [Equity Manager](#)'s Test Starting Equity multiplied by the Leverage value and multiplied by allocation percentage number entered for a system.

During a test, any systems in the suite that uses the `system.tradingEquity` property for its account equity, will find the equity allocation at the start of each test date show the allocation adjustments for that date and time. These sequential adjustments allow the systems in a suite to keep the same allocation distributions throughout the test.

Trading Blox will allow allocations that total more than 100%.

At first glance, it might seem that this would violate common sense and should not be permitted. However, since futures contracts have high leverage most of the time, a single trading system uses a small percentage of the available equity for margin. Consider two trading systems each of which uses a maximum of 25% of trading equity for margin. A single account could easily trade both systems at the same time.

System Equity (typically used by the money managers, but not always) is the test equity times the allocation times leverage adjusted by the draw down threshold.

The trading equity amount in the Fixed Fractional Money Manager is to determine trade size. It value is the test equity times this allocation slider times the leverage adjusted by the draw-down threshold.

Editing allocation value of a system determines the percentage adjustment of the current allocation amount.

Minimum Allocation Amounts:

Each system in the suite must show a System Allocation percentage value greater than zero. When set to zero, Trading Blox will remove that system from the suite so that it doesn't have any

Minimum Allocation Amounts:

effect on the test results.

Instead of reducing a system's allocation to zero, or a tiny allocation percentage, consider the following options:

- Set the system to operate in Virtual Mode by adding the **Virtual System** installed blox in the **System Editor**. By default, the System's Virtual Mode test setting is set to **FALSE**. This can easily be changed by using the [Virtual System](#) blox and then setting its parameter value to **TRUE**. Systems tested with their [Virtual Mode](#) enabled, will generate test results, but those results will not be included in the Summary Performance test result calculations. When the system allocation is greater than **0%**, and the Virtual Control is set to **TRUE**, the [Summary Results Report's System Parameter Settings](#) will show all the systems selected systems in the suite and their parameter settings. When the Virtual System blox is attached to a system its parameter setting will also be displayed.

Links:**See Also:**

Section 2 – Simulation Parameters

The following is a list of the Trading Blox global Simulation Parameters. Each parameter option is a link to information about of how each is used. The parameters in the Suite Parameter section are applied to all of the systems in the suite being tested.

A description of how each parameter affects the systems in the suite is a link to an explanation section in the following table. Trading Blox default values are displayed in this **Suite Parameters** image:

The screenshot shows the 'Simulation Parameters' dialog box. It contains various settings for simulation. Red arrows point from specific parameters to callout boxes:

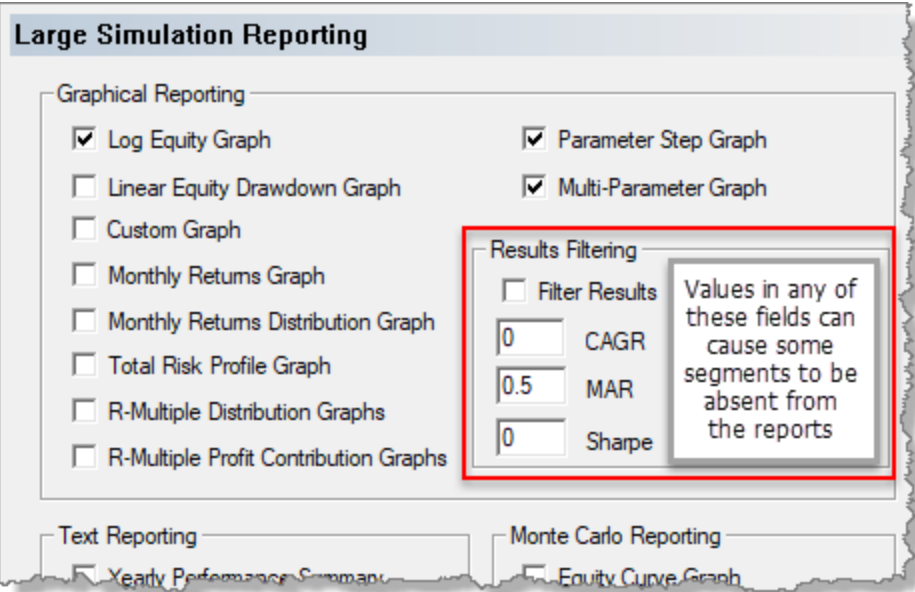
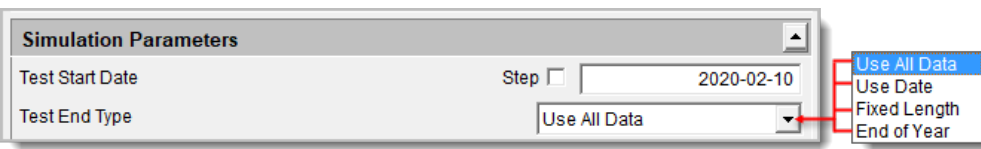
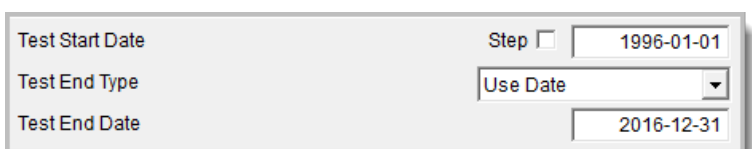
- Start-Date Stepping:** Points to the 'Test Start Date' field, which is set to '1996-01-02'.
- Use All Data:** Points to the 'Test End Type' dropdown menu, which is currently set to 'Use All Data'. Other options visible are 'Use Date', 'Fixed Length', and 'End of Year'.
- Step True to False:** Points to the 'Use Broker Positions' dropdown menu, which is currently set to 'False'. Other options visible are 'True' and 'Step True to False'.

Other parameters visible in the dialog include: Earn Interest (True), Slippage (%) (5%), Max Margin/Equity to trade (%) (100%), Trade Always on Tick (True), Smart Fill Exit (True), Entry Day Retracement (%) (25%), Max Position Units (1), Thread Count (1), and Multi-Parameter Goodness (MAR, Sharpe).

Simulation Default Parameter Settings

Global Parameter:	Description:
Test Start Date: ^Top	Date value entered will be the date when a Suite test can begin.
Start Date Stepping: ^Top	Trading Blox provides a progressive set of tested segments using Start-Date stepping feature. This provision will test all the dates set in the Simulation Parameters Date settings. Period lengths of each tested segment are dependent up the Test-End date settings. Each stepped-date segment will create a Suite tested segment specified by the period days specified for the testing length between each start-date incremented test. Each of the segments in the series of tested Suite segments will display graph tested segment information for each tested segment.

Global Parameter:	Description:																																																																																																																																																																																																																																																																																																																																
	To create a Stepped-Date test, enable Stepping of the Start Date Simulation Parameter. Then enter the the test segment days the Start-Date will increment after each test segment is finished: Simulation Parameters Test Start Date 1996-01-02 to 2018-01-02 by 365 Step ☒ from Test End Type Use All Data Use All Data Use Date Fixed Length Start-Date Stepping Example When the above test is performed, it will generate a table at the top of the Summary Performance Report that looks similar to the following table: <table><tr><th colspan="10">Summary Results</th></tr><tr><th>/</th><th>Test Start Date</th><th>End Balance</th><th>CAGR%</th><th>MAR</th><th>Sharpe</th><th>Ann. Sharpe</th><th>Max TE ...</th><th>Longest ...</th><th>Trades</th></tr><tr><td>1</td><td>19900101</td><td>\$62,376,166.09</td><td>20.92%</td><td>0.70</td><td>0.94</td><td>0.63</td><td>29.7%</td><td>47.7</td><td>3295</td></tr><tr><td>2</td><td>19910101</td><td>\$51,374,856.22</td><td>20.90%</td><td>0.63</td><td>0.92</td><td>0.64</td><td>32.9%</td><td>47.7</td><td>3215</td></tr><tr><td>3</td><td>19920101</td><td>\$42,741,546.26</td><td>20.93%</td><td>0.64</td><td>0.92</td><td>0.63</td><td>32.9%</td><td>47.7</td><td>3110</td></tr><tr><td>4</td><td>19921231</td><td>\$35,821,554.08</td><td>20.99%</td><td>0.61</td><td>0.92</td><td>0.67</td><td>34.7%</td><td>47.7</td><td>2993</td></tr><tr><td>5</td><td>19931231</td><td>\$24,980,460.68</td><td>20.17%</td><td>0.56</td><td>0.87</td><td>0.63</td><td>36.0%</td><td>47.7</td><td>2891</td></tr><tr><td>6</td><td>19941231</td><td>\$19,288,979.19</td><td>19.80%</td><td>0.49</td><td>0.87</td><td>0.61</td><td>40.6%</td><td>47.9</td><td>2788</td></tr><tr><td>7</td><td>19951231</td><td>\$18,537,679.89</td><td>20.54%</td><td>0.50</td><td>0.90</td><td>0.61</td><td>41.1%</td><td>47.9</td><td>2677</td></tr><tr><td>8</td><td>19961230</td><td>\$15,159,104.43</td><td>20.45%</td><td>0.53</td><td>0.88</td><td>0.59</td><td>38.4%</td><td>36.1</td><td>2557</td></tr><tr><td>9</td><td>19971230</td><td>\$12,992,001.31</td><td>20.63%</td><td>0.53</td><td>0.87</td><td>0.56</td><td>38.7%</td><td>36.1</td><td>2459</td></tr><tr><td>10</td><td>19981230</td><td>\$11,494,110.30</td><td>21.03%</td><td>0.55</td><td>0.88</td><td>0.60</td><td>38.0%</td><td>36.0</td><td>2348</td></tr><tr><td>11</td><td>19991230</td><td>\$12,547,639.35</td><td>22.81%</td><td>0.63</td><td>0.95</td><td>0.60</td><td>35.9%</td><td>36.1</td><td>2202</td></tr><tr><td>12</td><td>20001229</td><td>\$9,458,086.63</td><td>22.28%</td><td>0.61</td><td>0.93</td><td>0.58</td><td>36.4%</td><td>35.5</td><td>2094</td></tr><tr><td>13</td><td>20011229</td><td>\$7,839,443.15</td><td>22.38%</td><td>0.64</td><td>0.94</td><td>0.58</td><td>35.1%</td><td>35.5</td><td>1933</td></tr><tr><td>14</td><td>20021229</td><td>\$4,845,261.55</td><td>20.27%</td><td>0.60</td><td>0.86</td><td>0.52</td><td>34.0%</td><td>42.4</td><td>1852</td></tr><tr><td>15</td><td>20031229</td><td>\$3,437,854.79</td><td>19.01%</td><td>0.58</td><td>0.85</td><td>0.50</td><td>32.9%</td><td>34.5</td><td>1734</td></tr><tr><td>16</td><td>20041228</td><td>\$3,022,625.96</td><td>19.39%</td><td>0.59</td><td>0.85</td><td>0.51</td><td>32.8%</td><td>34.5</td><td>1619</td></tr><tr><td>17</td><td>20051228</td><td>\$3,023,241.40</td><td>21.02%</td><td>0.65</td><td>0.91</td><td>0.52</td><td>32.5%</td><td>34.4</td><td>1471</td></tr><tr><td>18</td><td>20061228</td><td>\$2,685,986.64</td><td>21.75%</td><td>0.67</td><td>0.93</td><td>0.51</td><td>32.5%</td><td>34.3</td><td>1345</td></tr><tr><td>19</td><td>20071228</td><td>\$1,428,265.09</td><td>17.06%</td><td>0.55</td><td>0.83</td><td>0.51</td><td>31.0%</td><td>33.4</td><td>1213</td></tr><tr><td>20</td><td>20081227</td><td>\$815,313.57</td><td>12.47%</td><td>0.43</td><td>0.73</td><td>0.49</td><td>29.1%</td><td>33.3</td><td>1138</td></tr><tr><td>21</td><td>20091227</td><td>\$593,841.07</td><td>10.01%</td><td>0.34</td><td>0.61</td><td>0.35</td><td>29.1%</td><td>33.2</td><td>1070</td></tr><tr><td>22</td><td>20101227</td><td>\$419,222.83</td><td>6.62%</td><td>0.20</td><td>0.44</td><td>0.23</td><td>33.8%</td><td>29.5</td><td>914</td></tr><tr><td>23</td><td>20111227</td><td>\$454,721.23</td><td>8.83%</td><td>0.26</td><td>0.55</td><td>0.29</td><td>34.6%</td><td>29.3</td><td>839</td></tr><tr><td>24</td><td>20121226</td><td>\$492,834.94</td><td>11.83%</td><td>0.44</td><td>0.68</td><td>0.39</td><td>27.0%</td><td>22.7</td><td>761</td></tr><tr><td>25</td><td>20131226</td><td>\$471,584.69</td><td>13.33%</td><td>0.49</td><td>0.75</td><td>0.62</td><td>27.4%</td><td>22.7</td><td>643</td></tr><tr><td>26</td><td>20141226</td><td>\$278,069.24</td><td>2.65%</td><td>0.13</td><td>0.23</td><td>0.35</td><td>20.1%</td><td>22.2</td><td>519</td></tr><tr><td>27</td><td>20151226</td><td>\$264,322.09</td><td>1.83%</td><td>0.10</td><td>0.19</td><td>0.30</td><td>18.8%</td><td>10.0</td><td>400</td></tr><tr><td>28</td><td>20161225</td><td>\$258,168.09</td><td>1.57%</td><td>0.08</td><td>0.17</td><td>0.26</td><td>19.1%</td><td>4.8</td><td>274</td></tr><tr><td>29</td><td>20171225</td><td>\$216,003.26</td><td>-12.73%</td><td>-0.70</td><td>-0.74</td><td>-4.91</td><td>18.2%</td><td>8.6</td><td>153</td></tr><tr><td>30</td><td>20181225</td><td>\$250,578.03</td><td>3.17%</td><td>0.70</td><td>0.23</td><td>1.54</td><td>4.5%</td><td>0.6</td><td>19</td></tr></table> Start-Date Test Use-All-Data Segment Table In this example, the Suite testing range was set to use all the data available. The test end-date in the above example has the Use All Data option selected. The other two test-end date options can be used when a test range is to be less than all the available data for an end-date. A Stepped Starting Date test often has enough steps to qualify as a Large Scale Report. These large reports can be influenced by any of the three filtering controls displayed in the Preferences' Large Reports Section. These filters will remove one or more of the stepped period results from the list of possible results. When that happens, it can cause some stepped reports to go	Summary Results										/	Test Start Date	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades	1	19900101	\$62,376,166.09	20.92%	0.70	0.94	0.63	29.7%	47.7	3295	2	19910101	\$51,374,856.22	20.90%	0.63	0.92	0.64	32.9%	47.7	3215	3	19920101	\$42,741,546.26	20.93%	0.64	0.92	0.63	32.9%	47.7	3110	4	19921231	\$35,821,554.08	20.99%	0.61	0.92	0.67	34.7%	47.7	2993	5	19931231	\$24,980,460.68	20.17%	0.56	0.87	0.63	36.0%	47.7	2891	6	19941231	\$19,288,979.19	19.80%	0.49	0.87	0.61	40.6%	47.9	2788	7	19951231	\$18,537,679.89	20.54%	0.50	0.90	0.61	41.1%	47.9	2677	8	19961230	\$15,159,104.43	20.45%	0.53	0.88	0.59	38.4%	36.1	2557	9	19971230	\$12,992,001.31	20.63%	0.53	0.87	0.56	38.7%	36.1	2459	10	19981230	\$11,494,110.30	21.03%	0.55	0.88	0.60	38.0%	36.0	2348	11	19991230	\$12,547,639.35	22.81%	0.63	0.95	0.60	35.9%	36.1	2202	12	20001229	\$9,458,086.63	22.28%	0.61	0.93	0.58	36.4%	35.5	2094	13	20011229	\$7,839,443.15	22.38%	0.64	0.94	0.58	35.1%	35.5	1933	14	20021229	\$4,845,261.55	20.27%	0.60	0.86	0.52	34.0%	42.4	1852	15	20031229	\$3,437,854.79	19.01%	0.58	0.85	0.50	32.9%	34.5	1734	16	20041228	\$3,022,625.96	19.39%	0.59	0.85	0.51	32.8%	34.5	1619	17	20051228	\$3,023,241.40	21.02%	0.65	0.91	0.52	32.5%	34.4	1471	18	20061228	\$2,685,986.64	21.75%	0.67	0.93	0.51	32.5%	34.3	1345	19	20071228	\$1,428,265.09	17.06%	0.55	0.83	0.51	31.0%	33.4	1213	20	20081227	\$815,313.57	12.47%	0.43	0.73	0.49	29.1%	33.3	1138	21	20091227	\$593,841.07	10.01%	0.34	0.61	0.35	29.1%	33.2	1070	22	20101227	\$419,222.83	6.62%	0.20	0.44	0.23	33.8%	29.5	914	23	20111227	\$454,721.23	8.83%	0.26	0.55	0.29	34.6%	29.3	839	24	20121226	\$492,834.94	11.83%	0.44	0.68	0.39	27.0%	22.7	761	25	20131226	\$471,584.69	13.33%	0.49	0.75	0.62	27.4%	22.7	643	26	20141226	\$278,069.24	2.65%	0.13	0.23	0.35	20.1%	22.2	519	27	20151226	\$264,322.09	1.83%	0.10	0.19	0.30	18.8%	10.0	400	28	20161225	\$258,168.09	1.57%	0.08	0.17	0.26	19.1%	4.8	274	29	20171225	\$216,003.26	-12.73%	-0.70	-0.74	-4.91	18.2%	8.6	153	30	20181225	\$250,578.03	3.17%	0.70	0.23	1.54	4.5%	0.6	19
Summary Results																																																																																																																																																																																																																																																																																																																																	
/	Test Start Date	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades																																																																																																																																																																																																																																																																																																																								
1	19900101	\$62,376,166.09	20.92%	0.70	0.94	0.63	29.7%	47.7	3295																																																																																																																																																																																																																																																																																																																								
2	19910101	\$51,374,856.22	20.90%	0.63	0.92	0.64	32.9%	47.7	3215																																																																																																																																																																																																																																																																																																																								
3	19920101	\$42,741,546.26	20.93%	0.64	0.92	0.63	32.9%	47.7	3110																																																																																																																																																																																																																																																																																																																								
4	19921231	\$35,821,554.08	20.99%	0.61	0.92	0.67	34.7%	47.7	2993																																																																																																																																																																																																																																																																																																																								
5	19931231	\$24,980,460.68	20.17%	0.56	0.87	0.63	36.0%	47.7	2891																																																																																																																																																																																																																																																																																																																								
6	19941231	\$19,288,979.19	19.80%	0.49	0.87	0.61	40.6%	47.9	2788																																																																																																																																																																																																																																																																																																																								
7	19951231	\$18,537,679.89	20.54%	0.50	0.90	0.61	41.1%	47.9	2677																																																																																																																																																																																																																																																																																																																								
8	19961230	\$15,159,104.43	20.45%	0.53	0.88	0.59	38.4%	36.1	2557																																																																																																																																																																																																																																																																																																																								
9	19971230	\$12,992,001.31	20.63%	0.53	0.87	0.56	38.7%	36.1	2459																																																																																																																																																																																																																																																																																																																								
10	19981230	\$11,494,110.30	21.03%	0.55	0.88	0.60	38.0%	36.0	2348																																																																																																																																																																																																																																																																																																																								
11	19991230	\$12,547,639.35	22.81%	0.63	0.95	0.60	35.9%	36.1	2202																																																																																																																																																																																																																																																																																																																								
12	20001229	\$9,458,086.63	22.28%	0.61	0.93	0.58	36.4%	35.5	2094																																																																																																																																																																																																																																																																																																																								
13	20011229	\$7,839,443.15	22.38%	0.64	0.94	0.58	35.1%	35.5	1933																																																																																																																																																																																																																																																																																																																								
14	20021229	\$4,845,261.55	20.27%	0.60	0.86	0.52	34.0%	42.4	1852																																																																																																																																																																																																																																																																																																																								
15	20031229	\$3,437,854.79	19.01%	0.58	0.85	0.50	32.9%	34.5	1734																																																																																																																																																																																																																																																																																																																								
16	20041228	\$3,022,625.96	19.39%	0.59	0.85	0.51	32.8%	34.5	1619																																																																																																																																																																																																																																																																																																																								
17	20051228	\$3,023,241.40	21.02%	0.65	0.91	0.52	32.5%	34.4	1471																																																																																																																																																																																																																																																																																																																								
18	20061228	\$2,685,986.64	21.75%	0.67	0.93	0.51	32.5%	34.3	1345																																																																																																																																																																																																																																																																																																																								
19	20071228	\$1,428,265.09	17.06%	0.55	0.83	0.51	31.0%	33.4	1213																																																																																																																																																																																																																																																																																																																								
20	20081227	\$815,313.57	12.47%	0.43	0.73	0.49	29.1%	33.3	1138																																																																																																																																																																																																																																																																																																																								
21	20091227	\$593,841.07	10.01%	0.34	0.61	0.35	29.1%	33.2	1070																																																																																																																																																																																																																																																																																																																								
22	20101227	\$419,222.83	6.62%	0.20	0.44	0.23	33.8%	29.5	914																																																																																																																																																																																																																																																																																																																								
23	20111227	\$454,721.23	8.83%	0.26	0.55	0.29	34.6%	29.3	839																																																																																																																																																																																																																																																																																																																								
24	20121226	\$492,834.94	11.83%	0.44	0.68	0.39	27.0%	22.7	761																																																																																																																																																																																																																																																																																																																								
25	20131226	\$471,584.69	13.33%	0.49	0.75	0.62	27.4%	22.7	643																																																																																																																																																																																																																																																																																																																								
26	20141226	\$278,069.24	2.65%	0.13	0.23	0.35	20.1%	22.2	519																																																																																																																																																																																																																																																																																																																								
27	20151226	\$264,322.09	1.83%	0.10	0.19	0.30	18.8%	10.0	400																																																																																																																																																																																																																																																																																																																								
28	20161225	\$258,168.09	1.57%	0.08	0.17	0.26	19.1%	4.8	274																																																																																																																																																																																																																																																																																																																								
29	20171225	\$216,003.26	-12.73%	-0.70	-0.74	-4.91	18.2%	8.6	153																																																																																																																																																																																																																																																																																																																								
30	20181225	\$250,578.03	3.17%	0.70	0.23	1.54	4.5%	0.6	19																																																																																																																																																																																																																																																																																																																								

Global Parameter:	Description:
	missing. If you find some reports missing, check to see the filtering control is active. Removing the values in any of three filtering categories will disable the filter:  Large Scale Reporting Filters Can Limit Stepped Start-Date Performance Summary Segment Reporting
Test End Type: ^Top	Testing can end in three different ways:  Test Start & End Date Timing Options Use All Data: ^Top When this option is selected, a test that starts on the Test Start Date will continue to run until there are no files with data. Even if the last date in some instruments are later than those without data. "Use All Data" is always the better option when a system is generating trading orders. It is better because the orders won't be stuck on at earlier date than the current date needed for trading when the data being updated is later than the "Use Date" value. Use Date: ^Top 

Global Parameter:	Description:																																																																																																																									
		This option allows the testing to be bounded to results that were available between the dates listed. Stepped Start-Date with a Fixed End-Date Testing Segment Example: Simulation Parameters Test Start Date1990-01-01to1999-01-01by365 Test End TypeUse Date Test End Date1999-12-31 Start-Date Stepped Use-End-Date Segment Parameters Setting Example. Summary Results <table><tr><th></th><th>Test Start Date</th><th>End Balance</th><th>CAGR%</th><th>MAR</th><th>Sharpe</th><th>Ann. Sharpe</th><th>Max TE ...</th><th>Longest ...</th><th>Trades</th></tr><tr><td>1</td><td>19900101</td><td>\$4,319,294.51</td><td>32.98%</td><td>1.30</td><td>1.17</td><td>0.85</td><td>25.4%</td><td>6.7</td><td>1179</td></tr><tr><td>2</td><td>19910101</td><td>\$2,815,659.62</td><td>30.89%</td><td>1.26</td><td>1.09</td><td>0.91</td><td>24.5%</td><td>6.2</td><td>1095</td></tr><tr><td>3</td><td>19920101</td><td>\$2,040,011.04</td><td>30.02%</td><td>1.30</td><td>1.09</td><td>0.92</td><td>23.0%</td><td>6.4</td><td>972</td></tr><tr><td>4</td><td>19921231</td><td>\$1,689,816.05</td><td>31.40%</td><td>1.42</td><td>1.13</td><td>1.30</td><td>22.1%</td><td>6.2</td><td>850</td></tr><tr><td>5</td><td>19931231</td><td>\$969,716.60</td><td>25.35%</td><td>1.18</td><td>0.98</td><td>1.17</td><td>21.6%</td><td>7.9</td><td>731</td></tr><tr><td>6</td><td>19941231</td><td>\$787,354.80</td><td>25.83%</td><td>1.28</td><td>1.06</td><td>1.60</td><td>20.2%</td><td>6.2</td><td>609</td></tr><tr><td>7</td><td>19951231</td><td>\$615,877.18</td><td>25.30%</td><td>1.25</td><td>1.10</td><td>1.41</td><td>20.2%</td><td>5.7</td><td>503</td></tr><tr><td>8</td><td>19961230</td><td>\$435,049.07</td><td>20.28%</td><td>0.93</td><td>0.87</td><td>0.98</td><td>21.8%</td><td>7.1</td><td>375</td></tr><tr><td>9</td><td>19971230</td><td>\$283,453.27</td><td>6.48%</td><td>0.20</td><td>0.36</td><td>0.62</td><td>32.4%</td><td>7.9</td><td>264</td></tr><tr><td>10</td><td>19981230</td><td>\$255,199.54</td><td>2.08%</td><td>0.09</td><td>0.20</td><td>2.12</td><td>22.1%</td><td>5.7</td><td>142</td></tr><tr><td>11</td><td>19991230</td><td>\$251,092.21</td><td>391.48...</td><td>NA</td><td>NA</td><td>NA</td><td>0.0%</td><td>0.0</td><td>2</td></tr></table> Start-Date Stepped Test Segment Fix End Date Table		Test Start Date	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades	1	19900101	\$4,319,294.51	32.98%	1.30	1.17	0.85	25.4%	6.7	1179	2	19910101	\$2,815,659.62	30.89%	1.26	1.09	0.91	24.5%	6.2	1095	3	19920101	\$2,040,011.04	30.02%	1.30	1.09	0.92	23.0%	6.4	972	4	19921231	\$1,689,816.05	31.40%	1.42	1.13	1.30	22.1%	6.2	850	5	19931231	\$969,716.60	25.35%	1.18	0.98	1.17	21.6%	7.9	731	6	19941231	\$787,354.80	25.83%	1.28	1.06	1.60	20.2%	6.2	609	7	19951231	\$615,877.18	25.30%	1.25	1.10	1.41	20.2%	5.7	503	8	19961230	\$435,049.07	20.28%	0.93	0.87	0.98	21.8%	7.1	375	9	19971230	\$283,453.27	6.48%	0.20	0.36	0.62	32.4%	7.9	264	10	19981230	\$255,199.54	2.08%	0.09	0.20	2.12	22.1%	5.7	142	11	19991230	\$251,092.21	391.48...	NA	NA	NA	0.0%	0.0	2
	Test Start Date	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades																																																																																																																	
1	19900101	\$4,319,294.51	32.98%	1.30	1.17	0.85	25.4%	6.7	1179																																																																																																																	
2	19910101	\$2,815,659.62	30.89%	1.26	1.09	0.91	24.5%	6.2	1095																																																																																																																	
3	19920101	\$2,040,011.04	30.02%	1.30	1.09	0.92	23.0%	6.4	972																																																																																																																	
4	19921231	\$1,689,816.05	31.40%	1.42	1.13	1.30	22.1%	6.2	850																																																																																																																	
5	19931231	\$969,716.60	25.35%	1.18	0.98	1.17	21.6%	7.9	731																																																																																																																	
6	19941231	\$787,354.80	25.83%	1.28	1.06	1.60	20.2%	6.2	609																																																																																																																	
7	19951231	\$615,877.18	25.30%	1.25	1.10	1.41	20.2%	5.7	503																																																																																																																	
8	19961230	\$435,049.07	20.28%	0.93	0.87	0.98	21.8%	7.1	375																																																																																																																	
9	19971230	\$283,453.27	6.48%	0.20	0.36	0.62	32.4%	7.9	264																																																																																																																	
10	19981230	\$255,199.54	2.08%	0.09	0.20	2.12	22.1%	5.7	142																																																																																																																	
11	19991230	\$251,092.21	391.48...	NA	NA	NA	0.0%	0.0	2																																																																																																																	
	Fixed Length: Test Fixed Length (days):^Top Set Test Duration (days)^Top	Test Start Date1996-01-01 Test End TypeFixed Length Test Fixed Length (days)365 This option enables Trading Blox to generate a test of segmented test period by enabling a process that increment and applied the incrementing the Start date by the number of days in the Test Fixed length (days) setting. Previously, this was the "Use Start Date Stepping" option that created a performance table of progressive start dates from the first date all the way to the specified number of Test days.																																																																																																																								

Global Parameter:	Description:																																																																																																																																																																																																																																																																																																																																
		Summary Results <table><tr><th>#</th><th>Test Start Date</th><th>End Balance</th><th>CAGR%</th><th>MAR</th><th>Sharpe</th><th>Ann. Sharpe</th><th>Max TE ...</th><th>Longest ...</th><th>Trades</th></tr><tr><td>1</td><td>19900101</td><td>\$550,544.41</td><td>120.34%</td><td>16.35</td><td>4.12</td><td>2.00</td><td>7.4%</td><td>1.4</td><td>104</td></tr><tr><td>2</td><td>19910101</td><td>\$423,510.55</td><td>69.47%</td><td>4.06</td><td>1.71</td><td>2.00</td><td>17.1%</td><td>4.4</td><td>128</td></tr><tr><td>3</td><td>19920101</td><td>\$295,814.00</td><td>18.34%</td><td>1.11</td><td>0.84</td><td>NA</td><td>16.5%</td><td>4.8</td><td>125</td></tr><tr><td>4</td><td>19921231</td><td>\$445,993.97</td><td>78.47%</td><td>4.65</td><td>2.40</td><td>2.00</td><td>16.9%</td><td>4.3</td><td>136</td></tr><tr><td>5</td><td>19931231</td><td>\$400,789.01</td><td>60.58%</td><td>5.91</td><td>1.92</td><td>2.01</td><td>10.2%</td><td>2.2</td><td>128</td></tr><tr><td>6</td><td>19941231</td><td>\$352,272.67</td><td>41.48%</td><td>2.34</td><td>1.55</td><td>NA</td><td>17.8%</td><td>5.3</td><td>125</td></tr><tr><td>7</td><td>19951231</td><td>\$323,971.10</td><td>29.70%</td><td>2.28</td><td>1.47</td><td>NA</td><td>13.1%</td><td>5.3</td><td>132</td></tr><tr><td>8</td><td>19961230</td><td>\$274,646.82</td><td>9.87%</td><td>0.45</td><td>0.48</td><td>1.89</td><td>21.8%</td><td>7.1</td><td>134</td></tr><tr><td>9</td><td>19971230</td><td>\$299,662.47</td><td>19.88%</td><td>0.61</td><td>0.71</td><td>2.08</td><td>32.4%</td><td>4.7</td><td>129</td></tr><tr><td>10</td><td>19981230</td><td>\$253,023.12</td><td>1.21%</td><td>0.05</td><td>0.16</td><td>2.23</td><td>22.1%</td><td>5.7</td><td>142</td></tr><tr><td>11</td><td>19991230</td><td>\$318,128.38</td><td>27.27%</td><td>1.29</td><td>1.34</td><td>2.08</td><td>21.2%</td><td>5.5</td><td>138</td></tr><tr><td>12</td><td>20001229</td><td>\$340,932.21</td><td>36.52%</td><td>1.61</td><td>1.18</td><td>2.01</td><td>22.7%</td><td>6.3</td><td>140</td></tr><tr><td>13</td><td>20011229</td><td>\$400,216.57</td><td>60.98%</td><td>7.34</td><td>2.12</td><td>2.03</td><td>8.3%</td><td>2.9</td><td>106</td></tr><tr><td>14</td><td>20021229</td><td>\$348,883.86</td><td>39.71%</td><td>2.22</td><td>1.25</td><td>2.01</td><td>17.9%</td><td>7.1</td><td>135</td></tr><tr><td>15</td><td>20031229</td><td>\$270,716.28</td><td>8.29%</td><td>0.36</td><td>0.45</td><td>2.23</td><td>22.9%</td><td>9.2</td><td>126</td></tr><tr><td>16</td><td>20041228</td><td>\$269,211.60</td><td>7.69%</td><td>0.36</td><td>0.44</td><td>2.01</td><td>21.6%</td><td>9.7</td><td>155</td></tr><tr><td>17</td><td>20051228</td><td>\$308,963.74</td><td>23.60%</td><td>1.46</td><td>1.22</td><td>1.90</td><td>16.1%</td><td>7.6</td><td>137</td></tr><tr><td>18</td><td>20061228</td><td>\$435,288.90</td><td>74.18%</td><td>4.68</td><td>2.38</td><td>1.98</td><td>15.8%</td><td>2.2</td><td>136</td></tr><tr><td>19</td><td>20071228</td><td>\$457,022.44</td><td>83.19%</td><td>4.40</td><td>1.83</td><td>1.99</td><td>18.9%</td><td>7.1</td><td>76</td></tr><tr><td>20</td><td>20081227</td><td>\$304,432.96</td><td>22.06%</td><td>2.10</td><td>1.40</td><td>2.03</td><td>10.5%</td><td>2.8</td><td>66</td></tr><tr><td>21</td><td>20091227</td><td>\$321,922.04</td><td>28.88%</td><td>1.82</td><td>1.63</td><td>2.04</td><td>15.8%</td><td>4.2</td><td>105</td></tr><tr><td>22</td><td>20101227</td><td>\$245,439.93</td><td>-1.83%</td><td>-0.12</td><td>-0.03</td><td>-2.75</td><td>15.2%</td><td>6.7</td><td>84</td></tr><tr><td>23</td><td>20111227</td><td>\$229,861.73</td><td>-8.06%</td><td>-0.43</td><td>-0.48</td><td>-2.01</td><td>18.5%</td><td>9.3</td><td>98</td></tr><tr><td>24</td><td>20121226</td><td>\$214,111.50</td><td>-14.36%</td><td>-0.53</td><td>-1.25</td><td>-1.98</td><td>27.0%</td><td>10.2</td><td>130</td></tr><tr><td>25</td><td>20131226</td><td>\$387,892.87</td><td>55.20%</td><td>5.39</td><td>2.44</td><td>1.89</td><td>10.2%</td><td>5.7</td><td>119</td></tr><tr><td>26</td><td>20141226</td><td>\$243,107.89</td><td>-2.77%</td><td>-0.14</td><td>-0.11</td><td>-2.06</td><td>19.4%</td><td>11.0</td><td>130</td></tr><tr><td>27</td><td>20151226</td><td>\$273,991.58</td><td>9.71%</td><td>0.55</td><td>0.53</td><td>1.77</td><td>17.5%</td><td>10.0</td><td>125</td></tr><tr><td>28</td><td>20161225</td><td>\$288,595.39</td><td>15.50%</td><td>1.35</td><td>0.87</td><td>1.93</td><td>11.5%</td><td>4.8</td><td>130</td></tr><tr><td>29</td><td>20171225</td><td>\$223,114.66</td><td>-10.76%</td><td>-0.62</td><td>-0.59</td><td>-2.93</td><td>17.5%</td><td>7.7</td><td>145</td></tr><tr><td>30</td><td>20181225</td><td>\$250,578.03</td><td>3.17%</td><td>0.70</td><td>0.23</td><td>1.54</td><td>4.5%</td><td>0.6</td><td>19</td></tr></table>										#	Test Start Date	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades	1	19900101	\$550,544.41	120.34%	16.35	4.12	2.00	7.4%	1.4	104	2	19910101	\$423,510.55	69.47%	4.06	1.71	2.00	17.1%	4.4	128	3	19920101	\$295,814.00	18.34%	1.11	0.84	NA	16.5%	4.8	125	4	19921231	\$445,993.97	78.47%	4.65	2.40	2.00	16.9%	4.3	136	5	19931231	\$400,789.01	60.58%	5.91	1.92	2.01	10.2%	2.2	128	6	19941231	\$352,272.67	41.48%	2.34	1.55	NA	17.8%	5.3	125	7	19951231	\$323,971.10	29.70%	2.28	1.47	NA	13.1%	5.3	132	8	19961230	\$274,646.82	9.87%	0.45	0.48	1.89	21.8%	7.1	134	9	19971230	\$299,662.47	19.88%	0.61	0.71	2.08	32.4%	4.7	129	10	19981230	\$253,023.12	1.21%	0.05	0.16	2.23	22.1%	5.7	142	11	19991230	\$318,128.38	27.27%	1.29	1.34	2.08	21.2%	5.5	138	12	20001229	\$340,932.21	36.52%	1.61	1.18	2.01	22.7%	6.3	140	13	20011229	\$400,216.57	60.98%	7.34	2.12	2.03	8.3%	2.9	106	14	20021229	\$348,883.86	39.71%	2.22	1.25	2.01	17.9%	7.1	135	15	20031229	\$270,716.28	8.29%	0.36	0.45	2.23	22.9%	9.2	126	16	20041228	\$269,211.60	7.69%	0.36	0.44	2.01	21.6%	9.7	155	17	20051228	\$308,963.74	23.60%	1.46	1.22	1.90	16.1%	7.6	137	18	20061228	\$435,288.90	74.18%	4.68	2.38	1.98	15.8%	2.2	136	19	20071228	\$457,022.44	83.19%	4.40	1.83	1.99	18.9%	7.1	76	20	20081227	\$304,432.96	22.06%	2.10	1.40	2.03	10.5%	2.8	66	21	20091227	\$321,922.04	28.88%	1.82	1.63	2.04	15.8%	4.2	105	22	20101227	\$245,439.93	-1.83%	-0.12	-0.03	-2.75	15.2%	6.7	84	23	20111227	\$229,861.73	-8.06%	-0.43	-0.48	-2.01	18.5%	9.3	98	24	20121226	\$214,111.50	-14.36%	-0.53	-1.25	-1.98	27.0%	10.2	130	25	20131226	\$387,892.87	55.20%	5.39	2.44	1.89	10.2%	5.7	119	26	20141226	\$243,107.89	-2.77%	-0.14	-0.11	-2.06	19.4%	11.0	130	27	20151226	\$273,991.58	9.71%	0.55	0.53	1.77	17.5%	10.0	125	28	20161225	\$288,595.39	15.50%	1.35	0.87	1.93	11.5%	4.8	130	29	20171225	\$223,114.66	-10.76%	-0.62	-0.59	-2.93	17.5%	7.7	145	30	20181225	\$250,578.03	3.17%	0.70	0.23	1.54	4.5%	0.6	19
	#	Test Start Date	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades																																																																																																																																																																																																																																																																																																																							
	1	19900101	\$550,544.41	120.34%	16.35	4.12	2.00	7.4%	1.4	104																																																																																																																																																																																																																																																																																																																							
2	19910101	\$423,510.55	69.47%	4.06	1.71	2.00	17.1%	4.4	128																																																																																																																																																																																																																																																																																																																								
3	19920101	\$295,814.00	18.34%	1.11	0.84	NA	16.5%	4.8	125																																																																																																																																																																																																																																																																																																																								
4	19921231	\$445,993.97	78.47%	4.65	2.40	2.00	16.9%	4.3	136																																																																																																																																																																																																																																																																																																																								
5	19931231	\$400,789.01	60.58%	5.91	1.92	2.01	10.2%	2.2	128																																																																																																																																																																																																																																																																																																																								
6	19941231	\$352,272.67	41.48%	2.34	1.55	NA	17.8%	5.3	125																																																																																																																																																																																																																																																																																																																								
7	19951231	\$323,971.10	29.70%	2.28	1.47	NA	13.1%	5.3	132																																																																																																																																																																																																																																																																																																																								
8	19961230	\$274,646.82	9.87%	0.45	0.48	1.89	21.8%	7.1	134																																																																																																																																																																																																																																																																																																																								
9	19971230	\$299,662.47	19.88%	0.61	0.71	2.08	32.4%	4.7	129																																																																																																																																																																																																																																																																																																																								
10	19981230	\$253,023.12	1.21%	0.05	0.16	2.23	22.1%	5.7	142																																																																																																																																																																																																																																																																																																																								
11	19991230	\$318,128.38	27.27%	1.29	1.34	2.08	21.2%	5.5	138																																																																																																																																																																																																																																																																																																																								
12	20001229	\$340,932.21	36.52%	1.61	1.18	2.01	22.7%	6.3	140																																																																																																																																																																																																																																																																																																																								
13	20011229	\$400,216.57	60.98%	7.34	2.12	2.03	8.3%	2.9	106																																																																																																																																																																																																																																																																																																																								
14	20021229	\$348,883.86	39.71%	2.22	1.25	2.01	17.9%	7.1	135																																																																																																																																																																																																																																																																																																																								
15	20031229	\$270,716.28	8.29%	0.36	0.45	2.23	22.9%	9.2	126																																																																																																																																																																																																																																																																																																																								
16	20041228	\$269,211.60	7.69%	0.36	0.44	2.01	21.6%	9.7	155																																																																																																																																																																																																																																																																																																																								
17	20051228	\$308,963.74	23.60%	1.46	1.22	1.90	16.1%	7.6	137																																																																																																																																																																																																																																																																																																																								
18	20061228	\$435,288.90	74.18%	4.68	2.38	1.98	15.8%	2.2	136																																																																																																																																																																																																																																																																																																																								
19	20071228	\$457,022.44	83.19%	4.40	1.83	1.99	18.9%	7.1	76																																																																																																																																																																																																																																																																																																																								
20	20081227	\$304,432.96	22.06%	2.10	1.40	2.03	10.5%	2.8	66																																																																																																																																																																																																																																																																																																																								
21	20091227	\$321,922.04	28.88%	1.82	1.63	2.04	15.8%	4.2	105																																																																																																																																																																																																																																																																																																																								
22	20101227	\$245,439.93	-1.83%	-0.12	-0.03	-2.75	15.2%	6.7	84																																																																																																																																																																																																																																																																																																																								
23	20111227	\$229,861.73	-8.06%	-0.43	-0.48	-2.01	18.5%	9.3	98																																																																																																																																																																																																																																																																																																																								
24	20121226	\$214,111.50	-14.36%	-0.53	-1.25	-1.98	27.0%	10.2	130																																																																																																																																																																																																																																																																																																																								
25	20131226	\$387,892.87	55.20%	5.39	2.44	1.89	10.2%	5.7	119																																																																																																																																																																																																																																																																																																																								
26	20141226	\$243,107.89	-2.77%	-0.14	-0.11	-2.06	19.4%	11.0	130																																																																																																																																																																																																																																																																																																																								
27	20151226	\$273,991.58	9.71%	0.55	0.53	1.77	17.5%	10.0	125																																																																																																																																																																																																																																																																																																																								
28	20161225	\$288,595.39	15.50%	1.35	0.87	1.93	11.5%	4.8	130																																																																																																																																																																																																																																																																																																																								
29	20171225	\$223,114.66	-10.76%	-0.62	-0.59	-2.93	17.5%	7.7	145																																																																																																																																																																																																																																																																																																																								
30	20181225	\$250,578.03	3.17%	0.70	0.23	1.54	4.5%	0.6	19																																																																																																																																																																																																																																																																																																																								
	End of Year: ^Top	This End of Year option creates a test period runs until 12/31 of each year. When there is more data available, it will snap to 01/January of the next year, or the earliest date available in the instrument files.																																																																																																																																																																																																																																																																																																																															
Earn Interest: ^Top	Earned Interest default setting is TRUE and it will add interest from available cash to the account when set to True . A False setting disables the interest earning option. Interest rates used during a simulation are determined by the currency rate files provided in the Forex Rate file folder for your system base currency:																																																																																																																																																																																																																																																																																																																																

Global Parameter:	Description:
	Forex Rate File Folder Disk Location These interest rate files must be maintained when there is a change in interest rates. Preference section of Trading Blox determines which Currency Rate file to use in a test: Forex Rate Setting Preference Section Above image example shows USD as the base currency. This USD setting will use the USD_Rates.txt file in the Forex Rates folder lending rate to determine the earned interest. Other currency designators will use the interest rate files that match the setting in the field. Only the cash amount earns interest:

Global Parameter:	Description:
	• Cash for Futures and Forex is the total equity. Money used as margin for Futures trading is not deducted from the cash balance since many brokers will allow T-Bills for margin requirements allowing traders to earn interest on the money that is used for margin. • Cash for Stock class instruments is closed equity minus purchase equity. This means money used to purchase stocks is deducted from the cash balance.
Slippage Percent: ^Top	The frictional cost of trading has two components: commissions, and slippage (sometimes also known as "skid"). In actual trading, slippage is the difference between a trade's entry or exit order price, and the price at which the trade is actually filled. In order to accurately reflect the conditions of real trading, the impact of slippage must be simulated during back testing. Slippage: The frictional cost of trading has two components: commissions, and slippage (sometimes also known as "skid"). In actual trading, slippage is the difference between a trade's entry or exit order price, and the price at which the trade is actually filled. In order to accurately reflect the conditions of real trading, the impact of slippage must be simulated during back testing. Since slippage can vary dramatically from trade to trade, depending on market conditions at the time an order is executed, Trading Blox employs a slippage assessment technique that is based on market volatility. The simulated fill price is obtained by calculating a slippage factor, which is added to (or subtracted from), the theoretical entry price. For a long entry, the slippage factor is calculated by measuring the range from the theoretical entry price to the day's highest price, and multiplying that amount by the Slippage Percent. (For short entries, the slippage factor is calculated by measuring the range from the theoretical entry price to the low). The slippage factor is then added to, or subtracted from the theoretical entry price, to obtain the simulated fill price. Here is how it works for a buy trade:

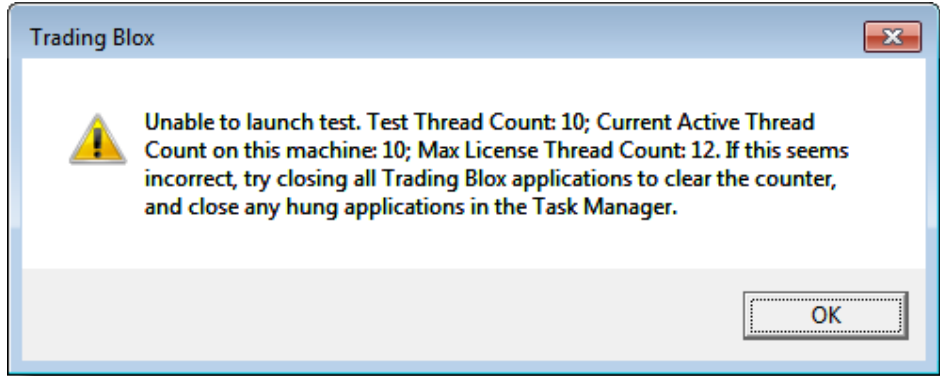
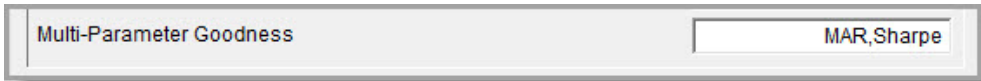
Global Parameter:	Description:
	Order Fill - 105 Order Price - 100 120 - Daily High 90 - Daily Low Slippage percent: 25% Theoretical buy order price: 100 High Price (for the day): 120 Slippage Factor: $(120 - 100) \times 0.25 = (20 \times 0.25) = 5$ Simulated fill price: Order Price + Slippage Factor = $(100 + 5) = 105$ The distance between the high price and the order price is multiplied by the slippage factor. In this example, the difference between the high price and the order price is 20 points. The 20 points are multiplied by the 25% slippage to get an estimated slippage of 5 points. The fill price for the order will be 5 points worse than the stop order price of 100 simulating a fill at 105. Slippage for sell orders is computed using a similar calculation using the distance between the order price and the low of the day. In historical back testing, failing to accurately estimate slippage can lead to two types of mistakes: Underestimating frictional costs may lead you to trade a system that produces spectacular hypothetical results, but does not hold up well under real trading. Conversely, overestimating frictional costs may dissuade you from trading an otherwise good system. It is also worth noting that the more frequently a system trades, the more profound the impact of frictional costs will be.
Minimum Slippage: ^Top	The Global Parameter property "Minimum Slippage" is a Futures only parameter. In testing and order generation, this parameter control is only active for portfolios with Futures instruments. Stock results are not affected by this Global Property setting. When a Suite is exported and or imported between version 4 version 5, the setting of this property will be set to zero. Slippage works in conjunction with Slippage Percent parameter.

Global Parameter:	Description:
	When this parameter is set to a non-zero value, results will have some slippage cost is assessed against every trade. Trading Blox will impose Minimum Slippage only if the currency value resulting from the Slippage calculation (based on Slippage Percent, above) is less than the currency value of slippage as calculated by the Minimum Slippage parameter. Where entry occurs at or near the high or the low of the day, the potential adverse range is practically nonexistent, so the Slippage Percent calculation would be at or near zero. In this case, Minimum Slippage can ensure that some slippage is assessed on the trade. If Slippage Percent is set to zero, then the slippage for all trades will be the value specified by the Minimum Slippage parameter value. Note: Minimum Slippage can cause the fill price to be outside of daily high low range.
<i>Max Percent Market Volume</i>	This option is no longer available as a built-in Global Parameter method. A blox module with this ability can easily be added when this area of control is needed during system development.
Max Margin/Equity to Trade: ^Top	The default is 100%, in that if a requested trade would require more than 100% of available equity in margin or cash, then the trade will be filtered. This parameter allows you to modify this behavior. If you enter 50 here, the system will filter trades if the new total margin required would be greater than 50% of the total available equity. To obtain these dynamically updated values as orders are being processed, use the Test Object property: <code>test.marketOrderValue</code>. This property contains the net margin required for tomorrow's market on open orders. Plus the margin for entries, minus margin for exits. The remainder value is added to the normal current outstanding margin process.
Trade Always on Tick: ^Top	When this parameter is set to false, the system will trade with maximum precision. So if the system buys Gold on a stop at the moving average, which is 365.44789, then the system will buy at that exact value. Conversely it will exit the trade at an exact value as well. The trade price as listed in the trade details will only show the digits of precision of the instrument, as set in the Futures Dictionary, so the trade price would look like 36.45. But the trade profit would not match that value exactly. When this parameter is set to true, the system will always trade on the tick. It will round up to the nearest tick for buy orders, and down for sell orders. So in the above example, if you placed an order to buy on stop at 365.44789, the system would place the order to buy at 365.45.

Global Parameter:	Description:
	The fill prices would also be on tick, so a slippage of 10% could become more as the order price and then fill price gets pushed to a tick.
Smart Fill Exit: ^Top (Smart Exit Fills)	When set to true, this function will only fill the exit stop/limit order closest to the open for any particular unit. This can be helpful when using multiple exit blox that place stop/limit orders. If the open is at 10, (and the high is 12 and the low is 7), and you have an exit stop at 11 and a limit order at 8, the stop exit order will be filled.
Entry Day Retracement : ^Top	Entry Day Retracement is the percent of the bar range (High-Low) used to compute the trigger price, which determines whether to fill the protective stop on the entry bar. For a long entry order: The protective sell stop will be filled if the trigger price is less than or equal to the stop price. The trigger price is computed as: ' trigger-price cannot be higher than the close or lower than the low. $\text{trigger-price} = \text{entry-order-fill-price} - (\text{bar-range} * \text{entry-day-retracement})$ Therefore, using 100% as the entry day retracement is the most conservative approach where the trigger price will be the low of the day, and if the low is less than the protective stop price then the stop will be filled, and the position exited on the entry bar. Note that if on average the fill price is in the middle of the bar, a setting of 50% will on average also use the low of the day as the trigger price. Using 0% as the entry day retracement is the most aggressive approach where the trigger price will be the close of the day, and if the close is less than the protective stop price, then the stop will be filled, and the position exited on the entry bar. Using -1 as the entry day retracement will disable this feature, and the protective stop will be ignored for the entry bar. When orders with a protective price Trading Blox will place the entry-risk points, and order-entry protective price in the order and instrument unit properties, so they are available after the bar of entry. Entry Day Retracement description and rules information.
Start Date Stepping	This feature is now part of the Start and End date options shown at this link Start Date Stepping location. In Trading Blox version 5.x.x.x and later, the process is

Global Parameter:	Description:	
<i>Increment Test Start Set Test Duration</i>	significantly different from previous Trading Blox version.	
Use Broker Positions: ^Top	FALSE	Default value is FALSE . In this position state, any positions listed in the Broker Position Editor are ignored when the test simulation is run and test results will only be from positions generated by the systems in the suite.
	TRUE	Set to TRUE only when positions you want in test, or order generation results derived from the the broker positions. In order this option setting to have an affect there must be positions entered in the Broker Position Editor .
Ignore All Test Positions: ^Top Hidden when Use Broker Positions = False	FALSE	If you want a mix of actual positions and theoretical test positions be sure this option is set to FALSE .
	TRUE	This option is only available when the above Use Broker Positions is set to TRUE. Set to true to ignore all the test generated positions. Use this option only when you want positions in the test, or order generation results to come from the entered broker positions. Often this is used on the very first day of order generation so that you can start with a clean slate and no open positions. If you want a mix of actual positions and theoretical test positions, then leave this box unchecked.
Max Position Units: ^Top	A position's quantity is the number of shares or contracts are applied to an order. A position's Unit count represents the number of orders that have been applied during the the time current position has been active. Note: This is a new Global Setting Simulation Parameter. When a suite has multiple systems, and some of the systems create multiple units, ensure this value is set to allow a multiple units. When there is any doubt, check to see if any of the systems that enables multiple-units is limited by this parameter setting.	
Thread Count: ^Top	Global Parameter Thread Count represents the number of test in a stepped simulation that can run at the same time. Each step in a stepped test runs a complete copy of the Suite's construction at the same time. This means that a complete copy of the entire Suite's systems, blox	

Global Parameter:	Description:
	and data files needed to run a test-step is loaded into thread-space memory so it can run as if only one test was being tested. Trading Blox Builder default license allows two threads. Additional threads can be allowed by accessing your Customer Login section in the Trading Blox Website. Each computer's thread count can be between 1 and maximum number allowed by the Trading Blox license. Thread counts should not be more than twice the number of cores in your computer's CPU. Available memory is important when using a large number of steps in a stepped simulation or when large portfolios are being used. Reducing Optimization Time provides more detailed information about how to judge how many threads would be effective with your computer. Global parameter Thread Count's value can easily be changed and saved with each Suite as the needs of each Suite are known. Note: Each Trading Blox user license has two threads by default. When more threads are needed and licensed, the added threads are summed with the two default threads to determine how many threads can be allowed on the computer. For example, if four threads are purchased, then the total number of threads that can be active at the same time is six threads. The number of threads used for one application, or multiple applications on the same computer, are limited to the maximum number of licensed threads. When the Global Parameter setting is set to use more than the licensed threads, an error message will appear reporting there must be an adjustment to the Global Thread's parameter Thread Count setting. Another way for the insufficient thread message to appear is when more than one instance of Trading Blox is running a simulation and the total required active threads being tested are more than the total allowed threads.

Global Parameter:	Description:
	 Trading Blox Unable to launch test. Test Thread Count: 10; Current Active Thread Count on this machine: 10; Max License Thread Count: 12. If this seems incorrect, try closing all Trading Blox applications to clear the counter, and close any hung applications in the Task Manager. OK Click image to expand and reduce. Under normal circumstances, this means in parallel, so you can have several applications open and sitting idle without affecting the count. The thread count in use on the computer is determined by the threads in use actually running a simulation. - Trading Blox can enable up to 200 Threads. - Should one of the applications have a critical error and stops running, those threads in the Trading Blox that was halted by an error will still be reserved. They can be released by exiting all open Trading Blox application. When Trading Blox installations are restarted that will reset the threads in use count. Simulation Scoped Variables Each thread in a multiple thread stepped test contains an entire copy of the Suite's systems, blox, and data files. This also means that a copy of all the variables and scripts in a suite is contained in each thread.
Multi-Parameter Goodness: ^Top	 Multi-Parameter Goodness MAR, Sharpe Multi-Parameter Goodness Setting Field The Multi-Parameter Goodness settings used with multiple-parameter test stepping test: End Balance CAGR% MAR Sharpe Ann. Sharpe MAX TE DD Longest DD Trades

Global Parameter:	Description:
	When this field is left blank, the MAR statistic will be used. To enable this option, more than a single method test can be added. Default setting is to use the "MAR & Sharpe" calculations. Previously setting the measure of goodness a number entered into the Preferences, or in scripting. In the current versions, these are now in the Suite's Parameters tab. The multi-parameter goodness measures are entered as a setting in the suite parameters area. Any number of statistics names separated by a comma determines how the Goodness setting responds. For example, the statistics entered might be: "MAR, Sharpe". If you are stepping multiple parameters, this will then control the goodness measure used for the surface and 3D charts (heat maps) created for a multi-parameter stepped test. You can use custom statistics as well, so if you use <code>test.AddStatistic("MyStat", 30)</code>, you could then use that "MyStat" statistic as a goodness measure. When stepping a single parameter, the stepped graphs show a measure of goodness vs. the parameter being stepped. Only the first statistic entered is used in these graphs because only one output appears in goodness chart analysis. Likewise, in walk-forward testing (WF), only one statistic is used. WF testing sample optimization runs are looped over, and one set of parameters is used. The sample uses the best goodness measure. You can select the statistic used for this measure of goodness in the Walk Forward Suite Parameters section. Additional Charting: You can see the optimization run results for the measure of goodness in the Log Window and also PRINT the information in the After Test script. Trading Blox does not save the testing information for charting. However, access to all statistics for all optimization and OOS runs is available. To chart in a different way, you can save the testing information to a file and use it as desired.

2.1 Entry Day Retracement

This parameter is used to determine if a new position would have been stopped out on the same day as entry. It can also be used with systems that don't provide a protective exit price with their orders by using a special feature to disable entry-order bar protective order execution when this parameter is set to **"-1"**.

Entry Day Retracement Calculations:

An Entry Day Retracement percent is created by multiplying the price-bar's Range, that is the distance between the high to the low price of the price-bar, to obtain a point estimate.



This point-estimate is then added, or subtracted from the order fill price to determine if the percentage applied would execute the protective order price so that a retracement price can be determined. To trigger a protective price the retracement price must be beyond the protective entry-order protective price to enable a same-day exit from the position.

Example:

For long entries:

$$\text{ExitTriggerPrice} = \text{OrderFillPrice} - (\text{High} - \text{Low}) * \text{Retracement \%}$$

For short entries:

$$\text{ExitTriggerPrice} = \text{OrderFillPrice} + (\text{High} - \text{Low}) * \text{Retracement \%}$$

Retracement Percent (%) Rules: [^Top](#)

- An ExitTriggerPrice is never more than the **High**, or less than the **Low** prices of the entry-bar. An ExitTriggerPrice is never more than the close for **Long** entries, or less than the **Close** for **Short** entries.
- A **Retracement%** of **zero** is the default setting, and that behavior is used as outlined below for an aggressive approach description. A **Retracement%** of **100** will use the extreme of the day for a conservative approach.
- If the position was filled during the day (**Stop/Limit orders**), and the **Close** is less than the **Stop** for **Long** entries, or more than the **Stop** for **Short** entries, then the position is stopped **Out**.
- If the position was filled on the **Open** of the day, and the **Low** is less than the **Stop** for **Long** entries or the **High** is more than the **Stop** for **Short** entries, then the position is stopped **Out**.

- An Entry Day [Retracement](#) of less than **0**, such as **-1**, will disable the entry day [Stop](#). This can be useful when you want to use [Stops](#) for risk measurement purposes, but not actually place those [Stops](#) into the market on the entry day.

**Note:**

When Trading Blox triggers a protective exit price order using the Entry Day [Retracement](#) process on the bar of entry, the reference [Order-ID](#) for that order will be [Zero](#).

Last Edit: 10/5/2020

Edit Time: 10/5/2020 1:08:29 PM

Topic ID#: 132

2.2 Reducing Optimization Time

[Quicker Optimizations in Trading Blox 4](#) (Web Link)

Previous Trading Blox versions executed stepped optimizations one test step at a time. Versions 3.x and earlier executed a stepped optimization in a serial 1-step at a time sequence. Serial step testing requires each test step to complete before the next step begins testing. Processing optimization steps serially adds each steps test time to the amount of time the optimization consumes to complete all test steps.

Version 4 changes how optimizations are executed by putting each test step into a thread. Two threads are available when Trading Blox 4 is first installed, and more threads are available as an option. Each thread can test 1-step at a time, and multiple threads can test other steps in that same test period. With 2-threads always available in version 4 the time it will take to perform an optimization in version 3.x is about twice the amount of time it will take in version 4.x, assuming the same computer specifications and test suite.

Multiple threads can save time because each thread running a step test provides an independent area where calculations can be tested in parallel. By running 2-threads at the same time the amount of time 2-threads requires compared to how a single thread test is nearly half the time of a the single thread time.

To put this in more concrete terms, the following table shows the amount of time that four different optimized test needed to run using 1-thread. All four tests were executed on three different computers. One computer used both the 32-Bit versions of Trading Blox 3.8.4, and Trading Blox 4.2.4.5. Two computers supporting 64-Bit software tested both 32-Bit & 64-Bit version 3 and version 4.

Test duration details in this table represent how many steps were in the four tests and the number of seconds each test required. All tests in this table were limited to 1-thread. Versions prior to version 4.x were only able to use one thread tests, so that test limitation will be the baseline to show how additional threads help to reduce stepped test times.

CPU	CPU Cores	Computer Memory	Active Threads = 1							
			Trading Blox Version	Trading Blox Bit Size	Windows Version	Windows Bit Size	Test Steps 1	Test Steps 20	Test Steps 60	Test Steps 120
i7-920	4	2.5 GB	3.8.4.0	32	Win XP Pro	32	6	32	93	184
i7-920	4	2.5 GB	4.2.4.5	32	Win XP Pro	32	8	28	88	173
i7-2630QM	4	8 GB	4.2.4.5	32	Win 7 Pro	64	4	25	75	148
i7-2630QM	4	8 GB	3.8.4.0	32	Win 7 Pro	64	4	21	64	129
i7-2630QM	4	8 GB	4.2.4.5	64	Win 7 Pro	64	3	19	59	120
i7-2630QM	4	8 GB	3.8.4.0	64	Win 7 Pro	64	3	18	55	110
i7-3930K	6	32 GB	4.2.4.5	32	Win 7 Pro	64	3	19	54	109
i7-3930K	6	32 GB	3.8.4.0	32	Win 7 Pro	64	2	17	51	101
i7-3930K	6	32 GB	3.8.4.0	64	Win 7 Pro	64	2	14	40	81
i7-3930K	6	32 GB	4.2.4.5	64	Win 7 Pro	64	2	14	40	76

Trading Blox 3 & 4.x - 1-Thread Active Stepped Optimization Test Time Results

Active Threads: [^Top](#)

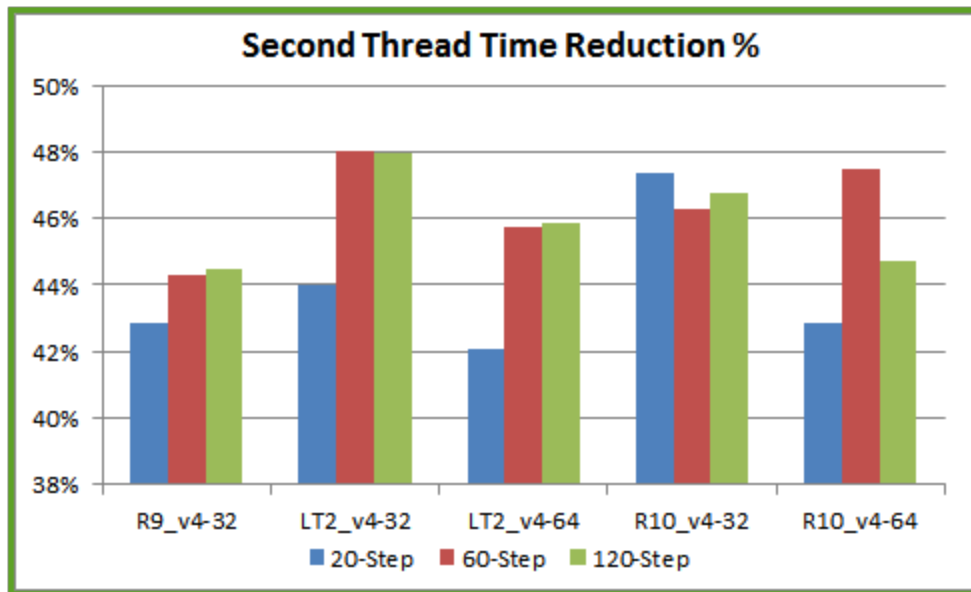
Being able to do an intensive processing required for good system development quickly is a significant enhancement in our trading platform. It is significant because it allows us to see the results of more ideas in less time, and having a second thread provides us with a significant time reduction from what was possible in version 3.

In this next table 2-threads are used to perform the same test as shown above, however only data for version 4 is available because version 3 cannot support a second thread to improve performance, so we need to accept the single thread times are the best older versions will be able to provide.

CPU	CPU Cores	Computer Memory	Active Threads = 2							
			Trading Blox Version	Trading Blox Bit Size	Windows Version	Windows Bit Size	Test Steps 1	Test Steps 20	Test Steps 60	Test Steps 120
i7-920	4	2.5 GB	3.8.4.0	32	Win XP Pro	32				
i7-920	4	2.5 GB	4.2.4.5	32	Win XP Pro	32	6	16	49	96
i7-2630QM	4	8 GB	4.2.4.5	32	Win 7 Pro	64	3	14	39	77
i7-2630QM	4	8 GB	3.8.4.0	32	Win 7 Pro	64				
i7-2630QM	4	8 GB	4.2.4.5	64	Win 7 Pro	64	3	11	32	65
i7-2630QM	4	8 GB	3.8.4.0	64	Win 7 Pro	64				
i7-3930K	6	32 GB	4.2.4.5	32	Win 7 Pro	64	3	10	29	58
i7-3930K	6	32 GB	3.8.4.0	32	Win 7 Pro	64				
i7-3930K	6	32 GB	3.8.4.0	64	Win 7 Pro	64				
i7-3930K	6	32 GB	4.2.4.5	64	Win 7 Pro	64	2	8	21	42

Trading Blox 4.x - Default 2-Thread Stepped Optimization Test Time Results

Time saved versus the time needed for a 2-thread test in Trading Blox 4 versus the time it takes with 1-threads creates a significant time reduction across all three optimizations: [^Top](#)



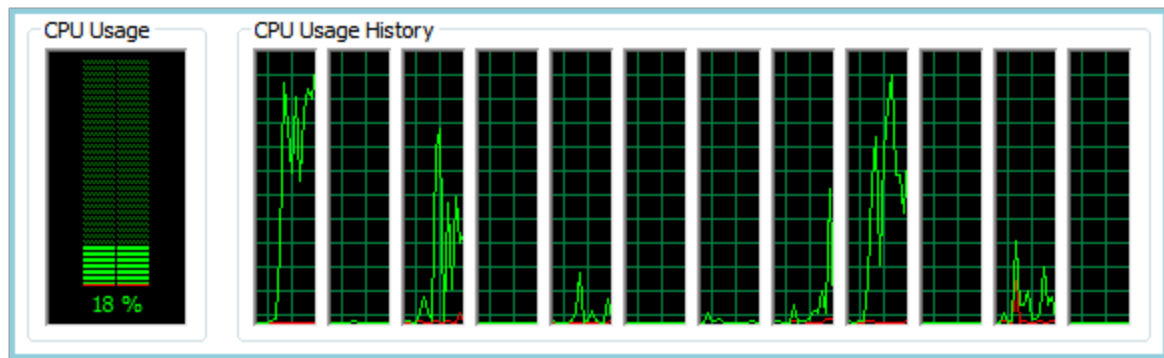
3-Stepped Optimization Time Reduction | 2-Threads versus 1-Thread Testing

A second thread run in parallel for optimization test is a noticeable improvement, and it also asks the question of whether two threads are enough. My answer is no, two threads are not enough. It is no because our test ranged from 20 to 120 steps is a fairly small optimization test, but it is a

practical size for careful data collection, and it shows how ineffectively we are using the computers capacity to get work done.

Computers capability should be a factor in making the decision about how many threads are enough; 2-threads on multi-core computers will only generate a low utilization. It is low because a single core limits 2-threads to process the entire test. This means all the other cores wont be helping to reduce test times.

This next image shows the six-core computer running our 120-step test. Windows Task Managers performance tab shows this thread limitation only being able to demonstrate an 18% utilization rate, leaving the other 82% of the computers capability at idle. [^Top](#)

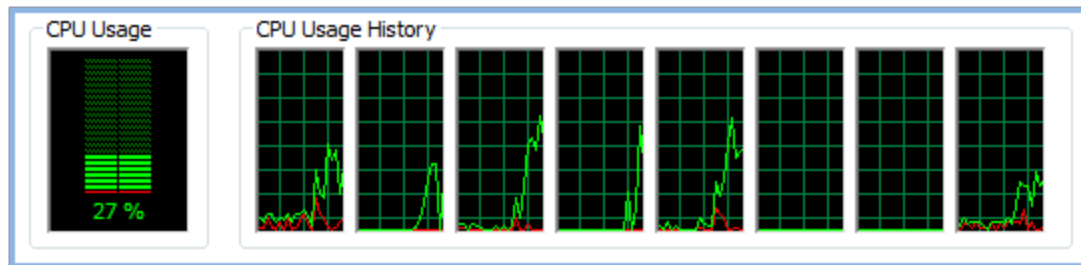


R10, TB 4, 120-Step Test, 2-Thread Setting, 6-Core CPU

For reference Windows Task Manager Performance Tab gives us a good estimation of the CPU Usage based on how much of the computers design capacity is available.

Each in these CPU Usage images the narrow vertical windows represents the number of threads the CPU was designed to handle at the same time. In this case there are 6-physical CPU cores in the chip, with each core capable of managing two threads, for a total of 12-threads available by design.

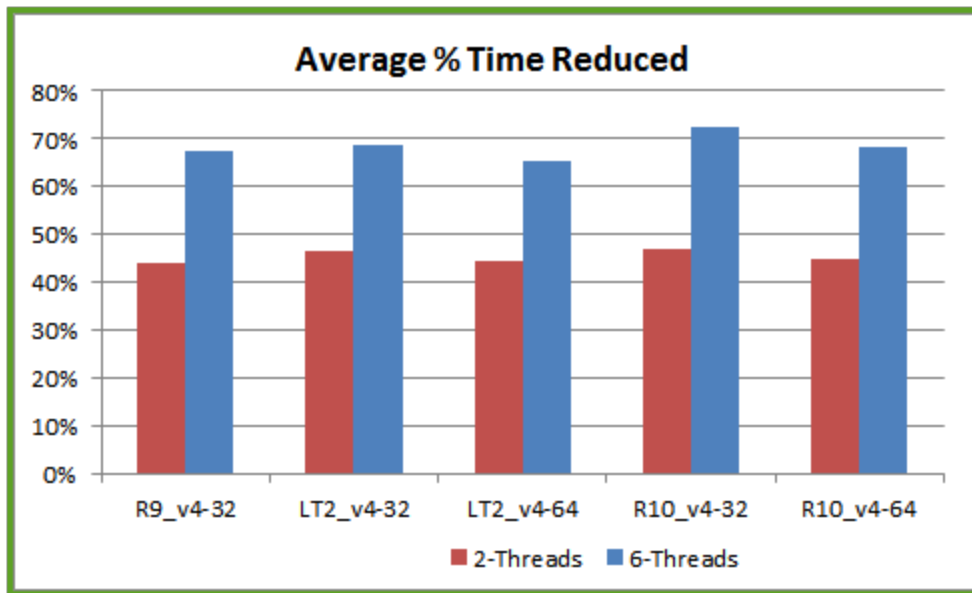
In this next image of a quad-core CPU the utilization running the same test creates an estimate of 27% utilization leaving 73% of the CPUs ability to sit at idle. [^Top](#)



LT2, TB 4, 120-Step Test, 2-Thread Setting, Quad-Core CPU

Being able to support 12-threads, or 8-threads for a quad-core CPU, doesn't mean all the cores should be loaded at 100% because that would interfere with the users need to do other things while the computer was running optimization test. More threads up to the design limit will

improve utilization as long as the operating system and the computers memory installation are capable of supporting all the threads. [^Top](#)



Ava. 3-Stepped Test Time Reduction Percentage | 2-Threads versus 6-Threads Optimizations

In this next thread count test results table image a full range the same our 120-Step test using 1-thread up to 12-threads on each of the computers was executed on all three computers. Test results details for the other steps lengths are available on request. [^Top](#)

Trading Blox 120-Step Optimization Speed Test						
Comp-ID:	R9	Lt-2	Lt-2	R10	R10	
Windows Ver:	WinXP PRO 32-Bit	Win-7 64-Bit	Win-7 64-Bit	Win-7 64-Bit	Win-7 64-Bit	
CPU Model:	i7-920	i7-2630QM	i7-2630QM	i7-3930K	i7-3930K	
CPU Cores:	4	4	4	6	6	
Memory Size:	2.5GB	8GB	8GB	32GB	32GB	
Trading Blox Bit Size 32 & 64:	32	32	64	32	64	
Trading Blox 4.2.4.5						
	Thread Count	Test Duration	Test Duration	Test Duration	Test Duration	Test Duration
	12	53	42	36	22	17
	11	53	42	36	22	18
	10	54	44	35	23	18
	9	54	44	36	25	19
	8	45	39	32	25	19
	7	48	42	34	27	21
	6	50	43	36	27	22
	5	54	45	38	30	24
	4	59	47	40	32	25
	3	71	57	47	41	29
Default:	2	96	77	65	58	42
	1	173	148	120	109	76
Trading Blox 3.8.4.0						
Trading Blox Bit Size 32 & 64:	32	32	64	32	64	
64 :	1			110		81
32 :	1	184	129	101		

Trading Blox 3 & 4 Available Thread Stepped Optimization Test Time Results

This table shows that 4-core computers performance improvement slows significantly when Trading Blox is sets to allow 6 of the possible 8 threads a quad-core CPU will support. On a 6-core computer the number of thread time reduction differences slows when there are 8 to 9 threads operating in memory. If these two thread to cores ratio is indicative, a Dual-Core CPU should easily handle 3-threads, and a single core CPU will be very busy with 2-threads.

Thread counts up to the thread-count design limit can give the best time, but if you try to do something else while a stepped simulation is executing the CPU will need to do some memory or thread swapping operations that will slow everything down. Memory swapping doesn't have a significant time reduction impact on performance, but disk swapping will reduce performance significantly.

Having enough memory to keep the all the available threads in memory throughout the test step testing time is critical to being able to keep the CPU from swapping threads to disk like the results of the R9 computer demonstrates because of Windows XP Pro max working memory size limitation. That memory limitations causes Windows XP Pro results to run slower than its less capable LT2 computer because XP is doing a lot of thread to memory swapping.

In the Trading Blox Roundtable Forum, there is a discussion that adds information to this topic:
[Multicore CPUs](#)

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

Stepped Optimization test Guidelines: [^Top](#)

- Short period test require less memory than long period test.
- Small portfolios require less memory than large portfolios.
- Multiple system optimization tests require more memory than a single system optimization test.
- In process custom data output disk activities require less time and memory than optimization that don't require custom data output operations.
- Small optimization steps create a larger processing need and a longer period of time to complete.
- Active threads that represent between 65 to 75% of CPU supported threads will allow for a high CPU utilization without making other processing needs like email and web browsing stall.

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

Thread Testing Caution: [^Top](#)

Keep in mind how a threads operate in parallel time. When a Stepped optimization is testing the First-Thread (Thread-1), the computer is also testing the Second-Thread (Thread-2) at nearly the same start time. This means if Thread-2, or any other thread test is dependent upon the completion of an earlier thread to get or report information, there will need to be a reference in your source code using the **test.threadIndex** property to ensure that the timing of the data that is need is passed when it is needed.

Here is what the test warning that appears in the main screen's Log Area reports. This warning will always appear when a **Simulation Scoped** variable is used. A Report for each of the Simulation scoped variables is given:

Using Simulation Scoped variable <your-variable name1> in multi-threaded test could have unintended consequences. Use with caution.

Using Simulation Scoped variable <your-variable name2> in multi-threaded test could have unintended consequences. Use with caution.

Speed Test Rules: [\(Web Link\)](#) [^Top](#)

Trading Blox original optimization [Original Speed Test](#) was created in January, 2008 to provide a fixed set of data that we could use with our Trading Blox installation. From that test result

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

comparison we would be able to see how our local performance fit into the overall range of results being reported.

When version 3.4.2 was released the [Second Speed Test](#) date range was increased to include more data, and test result times showed Trading Blox improved.

Now version 4.x has the ability to add multiple threads so it can do more in the same amount of time. With Trading Blox doing things even faster we needed to increase the number of testing steps and lengthen data test range again. Both changes in the new Speed Test requirements improve the resolution of time differences between threads. It also helps us get a better understanding of where the CPU utilization begins to make smaller time improvements as the thread count increases.

This table shows the actual test details used to generate the top posting: [^Top](#)

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

Optimization Speed Test Details			
Test Range			
Start Date:	1/1/1996		
End Date:	7/1/2013		
Portfolio:		Symbols	
All Liquid:		28	
Donchian /			
Turtle Single Unit			
Entry-Exit		Entry 80 Days	
Single Step Test:		22	
Optimization Test	Start Value	End Value	Step
1-Step:	22	22	0
20-Steps:	20	39	20
60-Steps:	20	79	60
120-Steps:	20	139	120

Setting Changes Used with the Speed Test Suites

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

Trader's RoundTable Blox Files: [Speed Test Modules.zip](#)

Modules in the zip files were exported by Trading Blox 4 using the Suites Export option. Each Suite will import in Trading Blox 4 using the Suites Import feature. Installing files into Trading Blox 3.x will require the files to be copied from their respective folders and placed into the same folders in your Trading Blox directory. Trading Blox 3.x should be closed while importing so the start-up logic finds them for display.

To import the SpeedTest's ZIP files contained in the above file, you should remove them from the above file so Trading Blox will find the folders it needs to move the modules.

When you unpack the above Zip file you should see this list of file names:

·All Liquid - Orig.set	- Futures Portfolio File List
·Export Speed Test Def Sim.zip	- Single Step Test
·Export Speed Test Step 020 Sim.zip	- 20-Step Test
·Export Speed Test Step 060 Sim.zip	- 60-Step Test
·Export Speed Test Step 120 Sim.zip	- 120-Step Test
·_Show Thread Portfolio Test-TB3.tbx	- Log Window Reporting Blox TB 3.x Only

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

This last Blox Module listed, "**_Show Thread Portfolio Test-TB3.tbx**", is a modified version of a similar module used with Trading Blox 4 included in the Suite packages shown above. All the modules in the Speed Test Packages will work with Trading Blox 3 except for the Auxiliary module that is used for Log Window Reporting.

For testing in Trading Blox 3, remove the module: "**_Show Thread Portfolio Test-TB4.tbx**" from all the System Lists and replace it with "**_Show Thread Portfolio Test-TB3.tbx**" Auxiliary Blox. Thread properties in the TB4 blox version are not available in TB3 and will cause TB3 to report an error if the Auxiliary module isnt replaced.

Before testing be sure to check the portfolio symbols in the All Liquid - Orig.set so that they show the correct Futures data file will be used.

Testing is best performed with Log Window on Trading Bloxs main screen open. Data from each will be generated it will be displayed log window where it can be copied. Log Window output will look like this:

[img] Sample_Test-Output_Log_Window_Example.png [/img] [^Top](#)

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

```
Starting Instrument Data Load at 15:20:08
Ending Instrument Data Load at 15:20:08

Finished Loading Instruments. Elapsed time: 0 seconds.

Using Simulation Scoped variable iDetailsSent in multi threaded
test could have unintended consequences. Use with caution.

Using Simulation Scoped variable sSpc2 in multi threaded test
could have unintended consequences. Use with caution.

Starting 20 Test Simulation
-----
Blox:  _Show Thread Portfolio Test-TB4                SIMULATION - START

Simulation Start Date   : 2013-07-16
Simulation Start Time   : 3:20:08 PM

Trading Blox Ver       : 4.2.4.5
Test Name              : Speed Test Step 020 Sim

System Name            : Donchian Speed Test
Portfolio Name         : All Liquid
Total Instruments      : 28

Parameter Tests Steps  : 20
```

Computer Memory Requirements: [^Top](#)

Stepped optimizations execute at their fastest rate when the entire optimization test is able to execute in memory without Windows needing to use disk space to support any part of the optimization test.

This means each computer's hardware configuration must have enough memory to handle the loading of all the instrument files into memory, and still have room to allocate dedicated memory to each of the threads active during the stepped simulation. It also means the large simulations using a large portfolio of stocks over a long period of time might not find enough memory to support the planned simulation without having to do a lot of memory to disk space swapping.

When Windows begins to swap memory space to disk so that it can execute other threads before finishing the current thread's processing, the amount of time for the simulation becomes longer than would happen had the computer provided more computer memory for the simulation.

If you find your simulation running slower than expected, check the Windows Task Manager Performance tab to see how much memory is being used. If the Task Manager shows a high percentage of memory is being used chances are Windows is doing a lot of memory to disk swapping that might be prevented if the computer can support a larger capacity of memory.

While there are no simple rules for determining how much memory is needed, most optimization test with Futures data can be easily be executed with 6 GB of memory. Optimization test on large Stock portfolios need more memory, but a capacity of 12 GB should be enough most of the time.

Test result generated on your computer should help you get a relative understanding of the time differences possible on your computer.

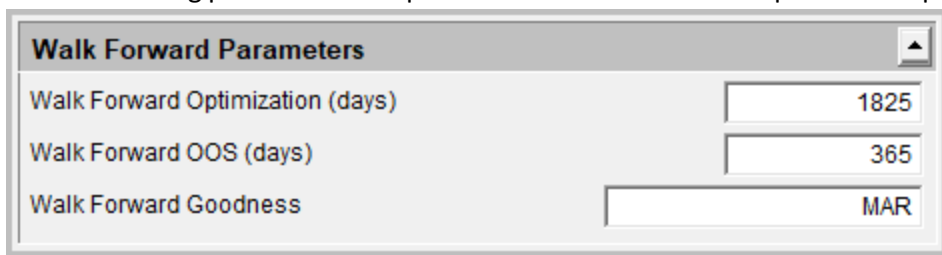
Section 3 – Walk Forward Parameters

Walk Forward testing can generate statistical information when the "**Walk Forward Stats**" blox module is included in a Walk Forward Suite test.

Walk Forward is a process that enables a series of test period segments to optimize parameters setup in a stepped test. If you are not familiar with Walk Forward Optimization testing, read: [Using Walk Forward](#).

Walk Forward Optimization Period Global Parameters:

To enable a Walk Forward System optimization test, Trading Blox first checks the **Global Parameter** to be sure the length of test and period of **Out of Sample (OOS)** days to include in a walk forwarding process are compatible for the walk forward optimization period specified.



The screenshot shows a dialog box titled "Walk Forward Parameters". It has three input fields with the following values:

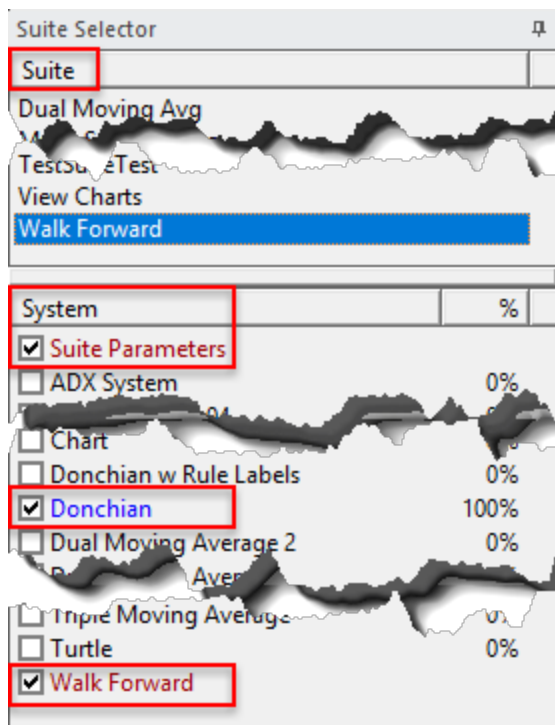
- Walk Forward Optimization (days): 1825
- Walk Forward OOS (days): 365
- Walk Forward Goodness: MAR

Global Parameter Walk Forward Method Control.

Global Parameter:	Description:
Walk Forward Optimization	Walk Forward parameter is a process for enabling a series of test period segments for the purposes of optimizing the value of one or more parameters.
Walk Forward OOS (days)	Sets the number of calendar days used for the Walk Forward out of sample process when running a Walk Forward test.
Walk Forward Goodness	Enter a single statistic to use as the measure of goodness for the optimization runs. Custom statistics added with test.AddStatistic can be used as well. Example: "MAR" or "Sharpe" or "MyStatistic".

Walk-Forward Suite Selection Requirements:

To perform a Walk-Forward Optimization Test, the suite needs a system where the one or more of the parameters are set to be stepped through a series of values. It also requires the Walk-Forward System be included in the Suite selection.



Walk-Forward Test Suite Selection Example

This image shows the Walk-Forward Suite Minimum selection items. In the System selection area, the Donchian System is selected and the Walk-Forward System is also selected. The [Walk-Forward System](#) is a [Global Suite Selected \(GSS\)](#) blox.

Links:

[Global Suite Parameters](#), [Preferences Editor](#), [Systems Editor](#)

3.1 Walk Forward Testing

Trading Blox version 5.x has integrated the Walk Forward Parameter rules so they are easier to execute. How that process works, and the steps a user can follow are listed in the linked areas of this table.

Walk Forward Information Sections:	
Walk Forward Testing Description	Statistical Measure Selections
Rolling Walk Forward Testing	Walk Forward Start and End Dates
Walk Forward Suite Selection	Walk Forward TestingThreads
System Selection	Walk Forward Statistics Reporter
Walk Forward Stepped Parameters	Walk Forward Simulation Test
Walk Forward Segment Settings	Stepped Parameter Test Summary Report
Walk Forward Goodness Setting	Walk Forward Stats 2 Graph Results
Walk Forward Stats 2 Testing Results	

Walk Forward Testing Description: [^Top](#)

The Walk Forward Test is a two-stage process that creates a series of data testing segments between the start and end dates selected in the Suite's Parameter date fields.

When a Walk Forward test runs, Trading Blox creates a groups of data segments. Each data segment is is determind by two of the three parameters displayed in the [Walk Forward Parameters](#) dialog. In the earlier data end of a Walk Forward segment, the size of the segment that is created by the [Walk Forward Optimization Days parameter](#). The data section created by the number of days listed in the [Walk Forward OOS](#) parameter.

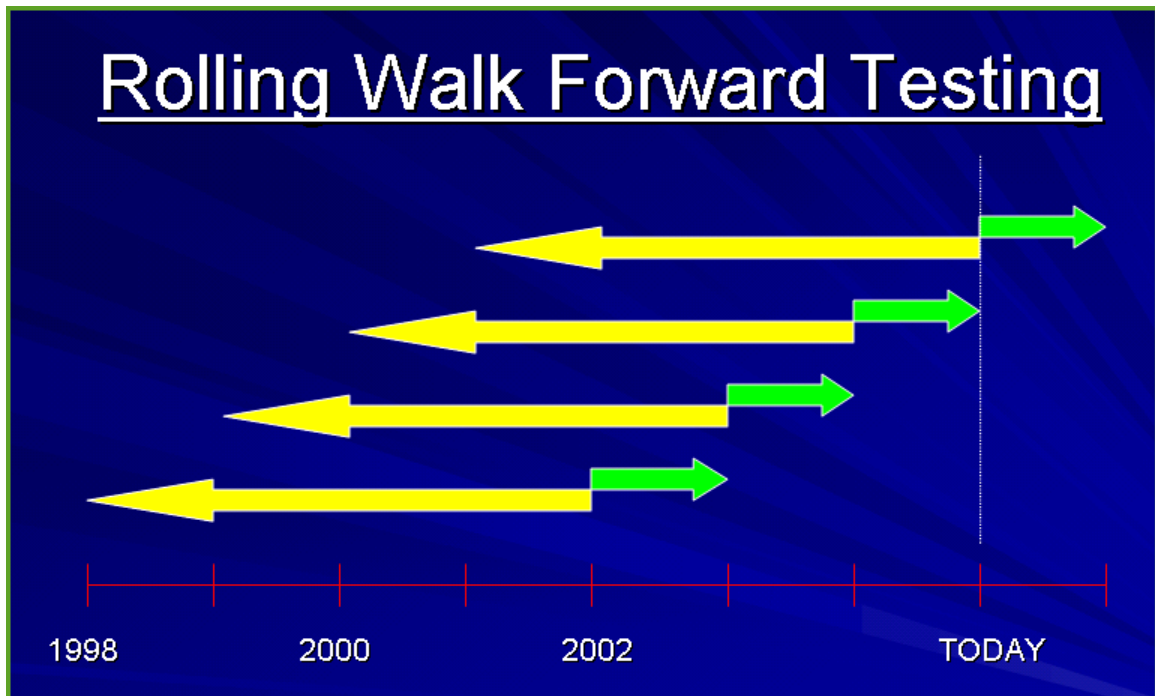
The optimal parameters values of the [Walk Forward Parameters](#) optimization process are determined by the statistical methods entered in the [Walk Forward Goodness](#) field. If the [Walk Forward Goodness](#) parameter field is left blank, the MAR statistic will be used to determine the optimal parameter values to test on the [OOS](#) section of a segment.

At the completion of the stepped parameter test in the optimizing segment, Trading Blox will apply the optimal parameter values using a normal single test to determine how the optimal parameters perform on the unseen [Out of Sample](#) data section of the tested segment.

Rolling Walk Forward Testing: [^Top](#)

This next chart shows arrows the make up the two sections of each data segment used by the Walk-Forward testing process. Yellow arrows point toward the left side of the image show the data range section used for parameter optimization. Green arrows pointing toward the right side of the image are the [Out of Sample \(OOS\)](#) data portion used to validate the optimal parameters.

The logic behind this type of testing is based on the concept that optimal parameters should be developed on historical data, but not accepted for trading until they prove themselves on unseen data.



Walk-Forward Step Process Example

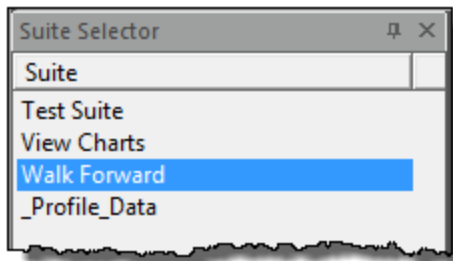
Unseen data can be any instrument's data that is large enough to provide a lot of record variations on which to develop a reliable measure of performance during the parameter optimization stage so it can perform reasonably close to the same performance on an additional large number of unseen record variations. Another way to test a system with OOS instrument records is to run the system after development off-line to see how it performs over the next year or more of time in paper-trading mode.

Walk Forward Simulation Example:

Walk Forward Suite Selection: [^Top](#)

Walk Forward testing can be performed with any suite name that has a system attached. The suite name doesn't need to be **Walk Forward** it can be a different suite name. When a suite is using a different name, there must also be a system with the same name in order for the **Walk Forward Stats 2** reporting blox to have a system in which it is enabled.

Our second step will be the same for all suite names. In this step we need to select a system. Any system that is test ready and has parameters that can step in value will work with Walk Forward optimization.



Any Suite name can be selected, but there must be a system name that matches the suite name when the Walk Forward Stats 2 blox statistical results are needed.

System Selection: [^Top](#)

Our example testing suite name is **Walk Forward**. We will included the **Walk Forward Stats 2** results in our testing example. This means we need a system that is named, Walk Forward so the statistical blox can report its result.



The default Donchian System will be this system we use to perform a Walk Forward Parameter Optimization test. Click the Donchian system that it shows that it is included in our Walk Forward Suite simulation. Our example test uses a Donchian system with the **Donchian SmartFills - EntryExit blox** as its entry exit order module.

Walk Forward Stepped Parameters: [^Top](#)

Our example test will step the breakout entry and exit bar length values as the parameters to optimize. If you are using the same system, click on the Step option for the Entry and the Exit Breakout parameters. Enter the values shown for each parameter when their fields expand:

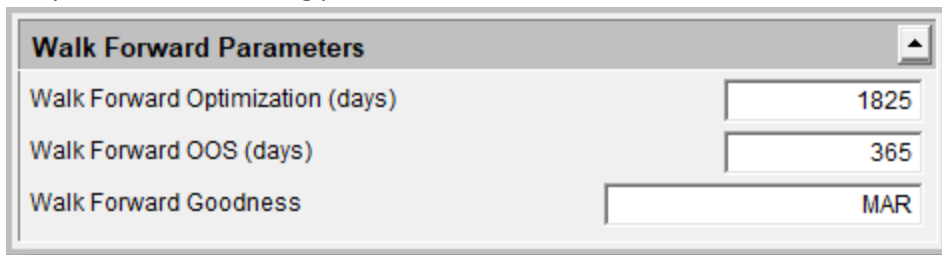
Entry, Exit	
Entry Breakout (bars)	Step ☒ from
22 to 25 by 1	
Entry Offset (atr)	Step ☐ -0.4
Stop (atr)	Step ☐ 1.8
Exit Breakout (bars)	Step ☒ from
10 to 40 by 5	
Exit Offset (atr)	Step ☐ 0
ATR (bars)	Step ☐ 39

Donchian Entry Exit with Smart Fills Stepped Parameters

If you are using a different system, Enable the Step option for the two parameters shown in your system and select a range you wish to test.

Walk Forward Segment Settings: [^Top](#)

First step is to establish the optimization and OOS data section lengths in the Suite's Walk Forward Parameter settings. This step is where the test in sample optimization data lengths and out of sample validation testing periods are established.



Walk Forward Parameters	
Walk Forward Optimization (days)	1825
Walk Forward OOS (days)	365
Walk Forward Goodness	MAR

Global Parameter Walk Forward Method Control.

In the image above the Suite's Walk Forward Parameter field the optimization days/bars needs the first 1,825 records will be the in sample optimization period, and the next 365 instrument records will be out of sample (OOS) will be the parameter validation period. Each additional segment will adjust its starting period forward by the number of OOS days and repeat the test again so as to generate a new OOS performance record.

Note:

Most instrument markets trade five days a week, and there are fifty-two weeks in a year in which there are often about eight holidays in each year that means there are about 252 trading days in a year. Using the above assumption, the optimization days value of 1,825 data records shown above will cover a time period of about 7.25 years. In the second parameter value of 365 days that creates an OOS sample Walk Forward time range of about 1.45 years. To have four independent test segments, the number of records required for each segment is 8.7 years for each test segment. To have enough data range between dates, using the same parameter values requires $(1,825 + (365 \times 3)) = 2,920$ records, or about 11.6 years of data.

Walk Forward Goodness Setting: [^Top](#)

Set the goodness measure in the Walk Forward Goodness parameter. This can be any built-in statistic or custom statistic added with [test.AddStatistic](#). The optimal run from the optimization runs is determined by the maximum goodness value.

In our Walk Forward test example will use the MAR statistic as the basis for selecting the best parameter values.

Statistical Measure Selections: [^Top](#)

If you wish to use a different statistical measure, then change the determined index number before clicking OK.

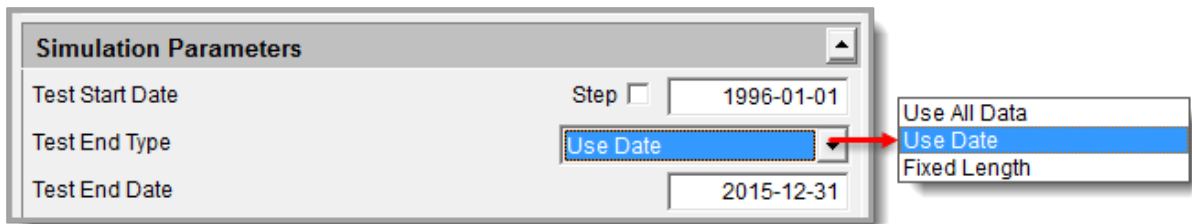
Statistical Measure Selections: [^Top](#)

The logic of the Walk Forward optimization process is designed to select a statistical measure's highest value. While that is how Walk Forward will return results, it is possible with some statistical measures to select the peak where the values on each side of a peak are significantly lower. Selecting the pinnacle value is a measure that shows a lot of volatility might not work out well because it is unlikely the selected parameters will perform as well during variations.

Graphs of the statistical measure show the measure's volatility when displayed. If the measure is volatile, consider a different statistical measure or consider changes to the logic so its performance can be from the top of a more flattened curve with gentle slopes.

Walk Forward Start and End Dates: [^Top](#)

Check the date range set for the Walk Forward simulation. Walk Forward will perform with all three of the date selection options, but if you wish to control how many segments, or want only one, then use the date information [Walk Forward Segment Settings](#) to help you establish how this Walk Forward test is performed.



Suite's Simulation Start & End Date Control Parameters

Our Walk Forward example will use the Walk Forward Stats 2 blox in the system name: Walk Forward.

Controlling Stops:

Saving of open positions from one OOS test to the next has limitation because any variables that might be tracking trailing stops in any manner are not brought over to the next testing segment.

Walk Forward Testing Threads: [^Top](#)

Each thread in a stepped simulation sets up an entire copy of the systems, blocks, objects, and variables, and runs the before and after simulation scripts for processing. You can access the parameter test number, and the thread index, to control the scripting as needed. The threads are run simultaneously, and in no particular order, so you can't share values between threads as you don't know which parameter test is running first or second within a thread group. Print out the parameter test number and thread index for a large test to see what I mean.

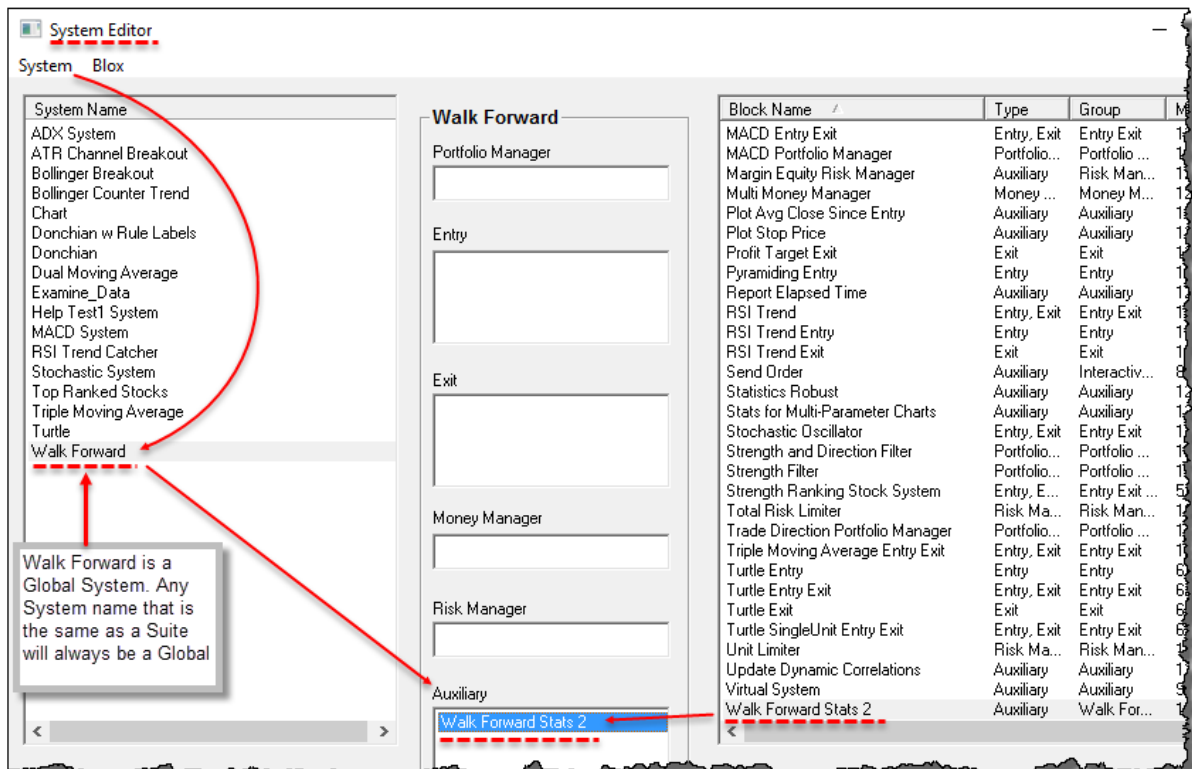
The way to save global values is either in the registry (for single values) or a file (for arrays). Take a look at the Walk Forward Stats 2 blox for an example of saving the stitched equity curve. In this case the out of sample runs (`test.walkForwardStatus=2`) do not execute at the same time, so we can output the equity values in order. If you are exporting values from regular test runs, keep in mind they can be out of order (test 4 running before test 1). You may run into conflicts attempting to write to the file from multiple threads at the same time, so the work

Walk Forward Testing Threads: [^Top](#)

around to that in the past has been to write to a different file for each thread, and then stitch them back together in the correct order in the After Test script section.

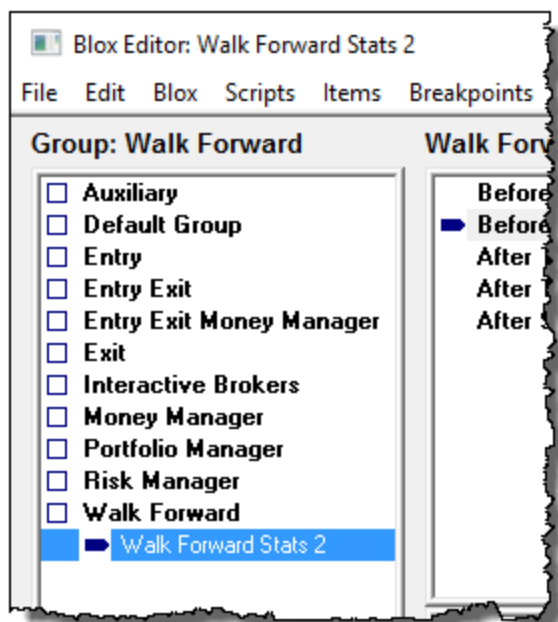
Walk Forward Statistics Reporter: [^Top](#)

Adding the Walk Forward Stats 2 blox statistical information to a Walk Forward test is similar to how others blox are added to a system, but there is one exception. To get started, enter the System Editor using the menu or by pressing F3:



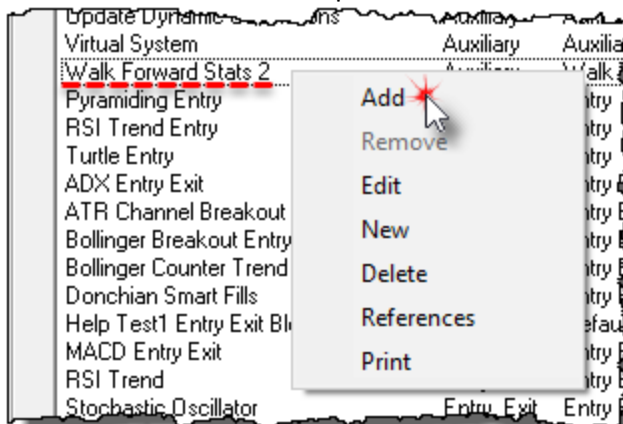
In the left column near the bottom is the **Global System** name, **Walk Forward**, created for this testing example. Click on that system name. If the **Walk Forward Stats 2** blox is not attached, follow the next steps to add it.

This Walk Forward Stats 2 blox is Installed when Trading Blox is installed. It is designed to report statistical and graphic results created during a Walk-Forward Parameter Optimization simulation.



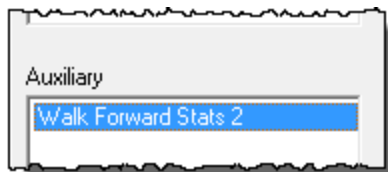
To include this additional statistical module it must be attached to a system with the same name as the Walk Forward testing suite. Any system that has the same name as a Suite, Trading Blox designates that system as a Global System. This statistical module will only work with a Global Suite. When it is included in a Walk Forward test the Walk Forward Stats 2 statistical results will be included in the [Stepped Parameter Test Summary Report](#), and the .

To find the Walk Forward Stats 2, open the System Editor and locate the blox name. Blox is designated as an Auxiliary blox. When the blox is found right click to bring up the pop-up menu, and then click on the Add option:



Add Walk Forward Stats 2 to Global Suite

When the add operation is complete the blox name will appear in the Auxiliary section of the System Editor.



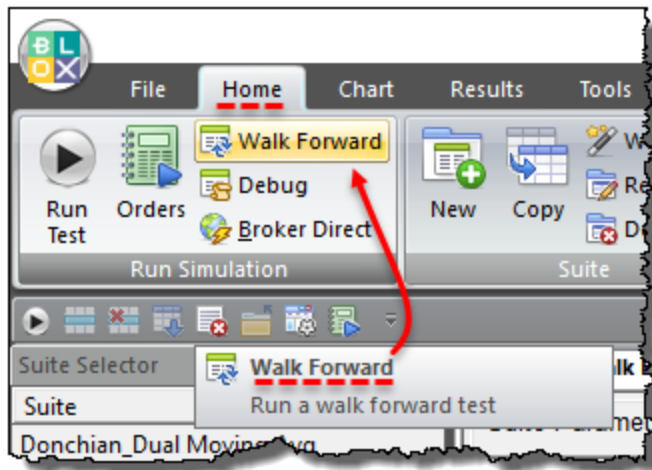
Walk Forward Stats 2 Blox is Added as an Auxiliary blox

Walk Forward Stats 2 Global System Only Blox: [^Top](#)

If you created a different Suite name to use with your system and want to see the Walk Forward Stats 2 blox results, Do Not allow the Walk Forward Stats 2 blox to be attached to a standard simulation system as the blox' internal error checking will cause Trading Blox to halt with an error and report the blox is attached in the wrong place. When Walk Forward Stats 2 is needed, attach it to a global system name that is the same as the selected test suite name.

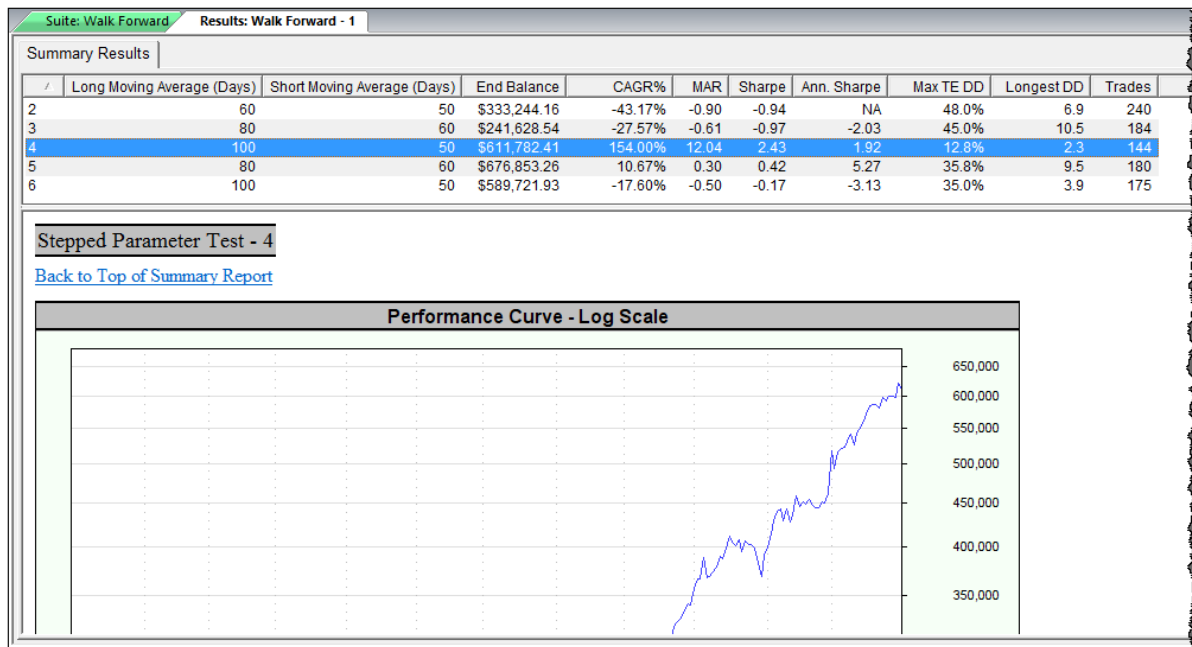
Walk Forward Simulation Test: [^Top](#)

Go to the Home main menu and in the Run Simulation section click on the Walk Forward button to begin the Walk Forward simulation test:



Stepped Parameter Test Summary Report: [^Top](#)

This table shows Summary Performance result from a simple Walk Forward Simulation example using the values shown here:

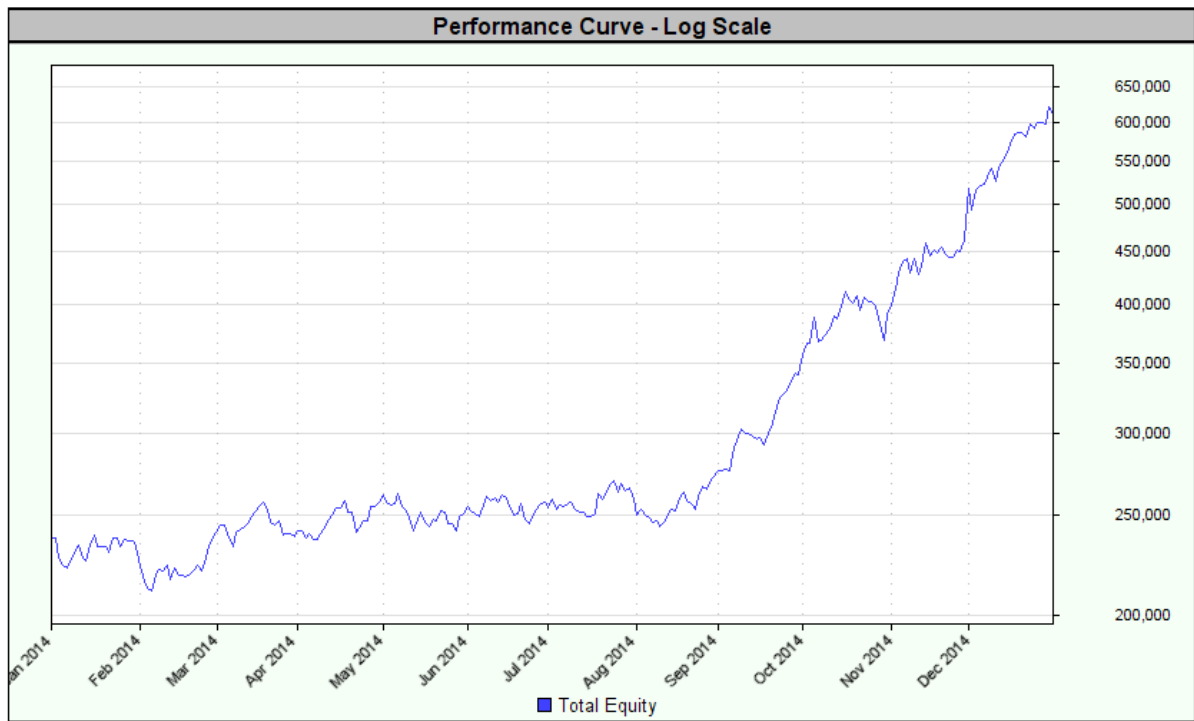


Walk-Forward Stats Performance Summary Example

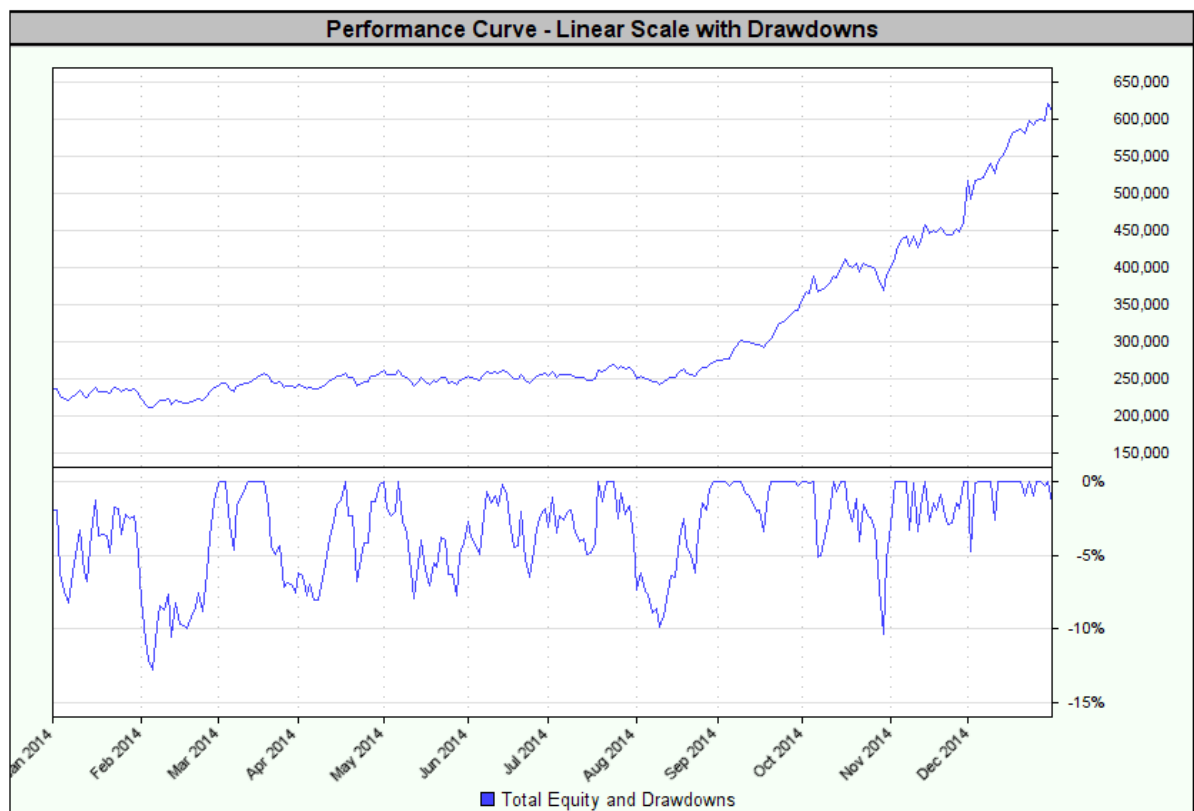
The above image performance table shows each parameter step section for each of the six simulation steps performed. Each step in the test will also show a Stepped Parameter Summary Test report for each of the stepped periods that will appear when any of the numbered test rows are clicked with the mouse.

Walk Forward Stats 2 Graph Results: [^Top](#)

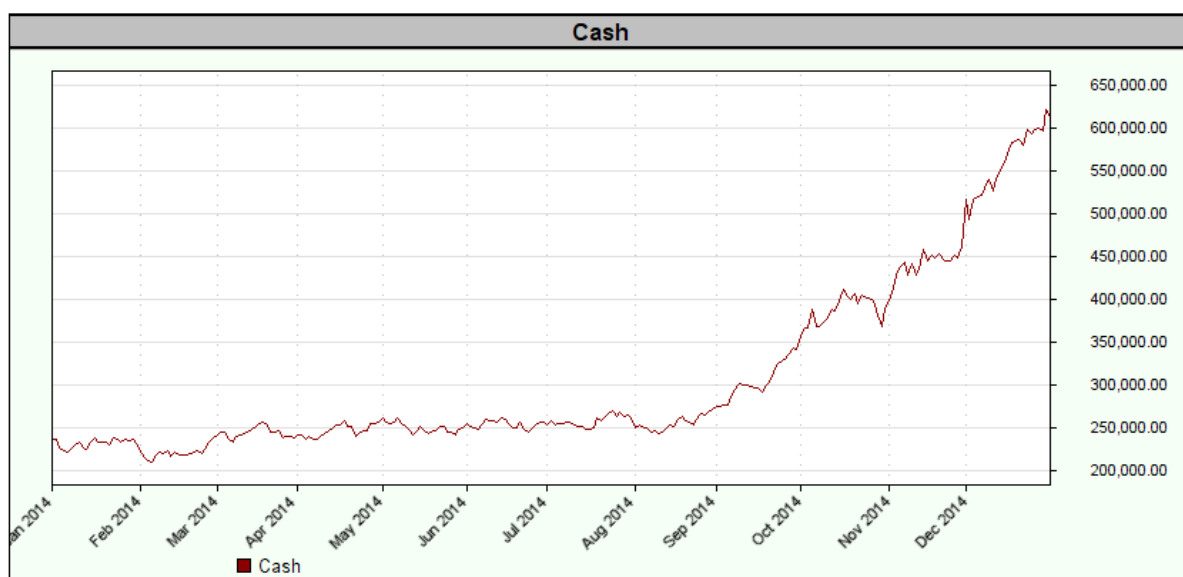
Walk Forward Stats 2 blox attached to the Global System does not have any parameters, how it will display three graphs of how each of the stepped test performed. Each of the three graph examples shown here will appear in each of the stepped performance section shown in the Stepped Performance Summary Report heading area shown above:



Walk Forward Stats 2 Performance Curve Log Scale



Walk Forward Stats 2 Performance Curve Linear Scalewith Drawdown



Walk Forward Stats 2 Cash

Walk Forward Stats 2 Testing Results: [^Top](#)

When the Trading Blox [Log Window](#) is active, this additional information will be available in the Log Window display.

```
Starting Instrument Data Load at 09:05:03
Ending Instrument Data Load at 09:05:04
Finished Loading Instruments. Elapsed time: 1 seconds.

Starting Simulation Setup at 09:05:04

28 Test Optimization for dates 20060102 to 20101231
Parameter test 17 will be used for out of sample test due to best goodness of 1.533831
Out of Sample test number 1 for dates 20110101 to 20111231
28 Test Optimization for dates 20070102 to 20111231
Parameter test 19 will be used for out of sample test due to best goodness of 0.953879
Out of Sample test number 2 for dates 20120101 to 20121230 with starting equity 420258.474522
28 Test Optimization for dates 20080102 to 20121230
Parameter test 14 will be used for out of sample test due to best goodness of 0.591911
Out of Sample test number 3 for dates 20121231 to 20131230 with starting equity 363317.638015
28 Test Optimization for dates 20090101 to 20131230
Parameter test 25 will be used for out of sample test due to best goodness of 0.083212
Out of Sample test number 4 for dates 20131231 to 20141230 with starting equity 280854.118846
28 Test Optimization for dates 20100101 to 20141230
Parameter test 20 will be used for out of sample test due to best goodness of 0.115529
Out of Sample test number 5 for dates 20141231 to 20151230 with starting equity 356956.081263
28 Test Optimization for dates 20110101 to 20151230
Parameter test 25 will be used for out of sample test due to best goodness of -0.098728
Out of Sample test number 6 for dates 20151231 to 20160916 with starting equity 317767.967484

Ending Test Run at 09:05:31
Finished Test Run. Elapsed time: 27 seconds.
```

Walk Forward Stats 2 Log Window Test Report

Section 4 – Futures Parameters

All Futures contract instrument order transaction cost are controlled by the settings in this section.

Futures Parameters	
Minimum Futures Volume (contracts)	Step ☐ 1,000
Commission per Contract	Step ☐ \$4.000
Minimum Futures Slippage	Step ☐ \$0.000
Trade Futures on Lock Day	False
Account for Contract Rolls	True
Roll Slippage (% of ATR)	Step ☐ 0%

Futures Parameters Global Setting Control

Global Parameter:	Description:
Minimum Futures Volume (contracts) ^Top	This parameter applies only to futures. It establishes the minimum daily trading volume, in contracts, required to enter a futures position. It is based on a 5-day exponential moving average of the volume. This value can be viewed by using the instrument.averageVolume property when using the Builder Edition.
Commission per Contract ^Top	This parameter indicates the round-trip charge for each futures contract traded. Assuming a <i>Commission per Contract</i> of \$12.50, buying and then subsequently selling 1,000 contracts of sugar would result in a total transaction cost of (\$12.50 per contract) x (1,000 contracts) = \$12,500.
Minimum Futures Slippage ^Top	
Trade Futures on Lock Day ^Top	When this parameter is set to false, the simulation will not fill entry or exit orders on days when the high = low, when the trade is in the direction of the lock. The idea is that in futures this could be a lock limit day, and it would be the most conservative assumption to assume you did not get filled. If you set this parameter to true, you will be filled on these days. This applies to futures only. Example: With this parameter set to false, and the high equals the low, so it is considered a lock day. If you want to enter long, or exit a short position, and the close of the lock day is less than the close of yesterday, then it will be allowed. But if the close of

Global Parameter:	Description:	
	the lock day is greater than or equal to the close of yesterday, then the fill will be denied.	
Account for Contract Rolls Roll Slippage (% of ATR) ^Top	True False Step True to False	

This parameter applies only to futures, and it controls whether or not Trading Blox should account for the increased commission and slippage that would have resulted when rolling contracts when a position is held for a long period of time.

When your data includes the [Delivery Month](#) for your futures contracts, Trading Blox roll will account for your roll transactions when the value in the [Delivery Month](#) column changes the expiration month assigned to data record.

When the [Delivery Month](#) column is missing from your futures contract data files, Trading Blox will roll every time the delivery month changes in relationship to the data record used in a position. Contract rolls will happen using the contract expiration month letters displayed is the Futures Dictionary's Trading Months column.

Symbol	Description	Format	File/Symbol	Exchange	Trading Months	Contract
AD	Australian Dollar CME	ASCII	AD_B T8D	CME	HMLCZ	250000
AEX	Euro Index (Amsterdam)	ASCII	AEX_B T8D	EDE	FHAKJMNQVWZ	200
BD	Soybean Oil CBT (P)	ASCII	BD_B T8D	CBT	FHAKJMNQVWZ	600000
BP	British Pound CME (P)	ASCII	BP_B T8D	CME	HMLCZ	625000
C	Commodities (Finnish)	ASCII	CP_B T8D	CBT	HMLCZ	50000

Futures Trading Contract Expiration Month Example

Rolls when Delivery Month Column is Missing:

If there are 4 trading months, then Trading Blox will calculate a contract roll every 3 months. If there are 12 trading months, Trading Blox will calculate a contract roll

Global Parameter:	Description:
	every month. The first simulated roll will occur in 1/2 the normal roll frequency because, on average, the first contract will be entered with 1/2 its trading life left. This process is based on calendar days and works for intraday, daily, weekly, or monthly data. Roll Slippage Calculation Basis: Each time a simulated roll occurs, Trading Blox accounts for the roll by deducting slippage and commissions for each contract in the position. The Open Equity is moved to Close Equity. If the futures is non-USD denominated, the currency conversion for the roll date will be used to move profit from open equity to closed. This adjustment locks the the profit in at the conversion rate of the roll. Slippage Percentage has more calculation details: When the Account for Contract Rolls option is set to True, all Futures will Account for Contract Rolls using the slippage rate entered into the Roll Slippage (% of ATR) field. The roll slippage percent is based upon the Wilder 20-day ATR (39 day non SMA primed Exponential Moving Average of the True Range). This is an un-primed value that can be accessed by the

Global Parameter:	Description:
	<code>instrument.defaultAverageTrueRange</code> property.

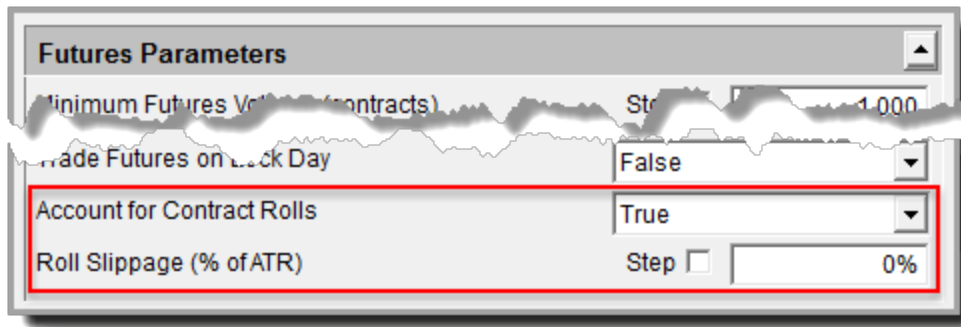
Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 146

4.1 Slippage Percentage

Roll Slippage Calculation Description:

Each time a simulated roll occurs, Trading Blox accounts for the roll by deducting slippage and commissions for each contract in the position. The Open Equity is moved to Close Equity. If the futures is non-USD denominated, the currency conversion for the roll date will be used to move profit from open equity to closed. This adjustment locks the the profit in at the conversion rate of the roll.



Global Simulation Futures Parameters Slippage Pct Parameters

The frictional cost of trading has two components: Commissions, and Slippage (sometimes also known as "skid").

In actual trading, slippage is the difference between a trade's entry or exit order price, and the price at which the trade is actually filled.

In order to accurately reflect the conditions of real trading, the impact of slippage must be simulated during back testing.

Since slippage can vary dramatically from trade to trade depending on market conditions at the time an order is executed, Trading Blox employs a slippage assessment technique that is based on market volatility.

This calculation creates a simulated fill price by applying a slippage factor value that is added or subtracted from the adjusted entry price. This difference between order price and fill-price is the simulated slippage.

Note:

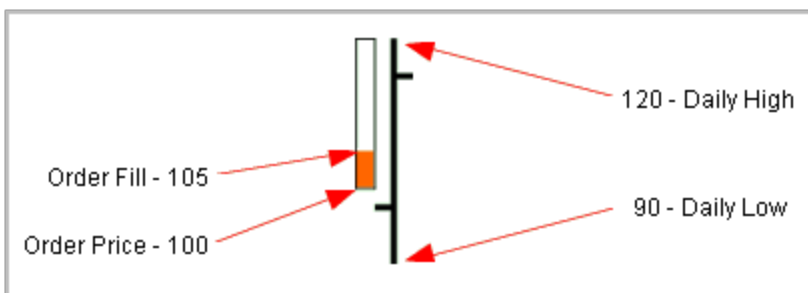
Orders that use On-Open, At-Stop price order executions have slippage applied; At-Limit and On_Close order executions do not have any slippage adjustments applied. On-Open, At-Stop price and At-Limit price order executions are also subject to gap Open price '**Open Slippage**' adjustments. On-Close execution orders do not experience gap opening price adjustments.

For a long entry the slippage factor is calculated by measuring the range from the theoretical entry price to the day's highest price. That value is then multiplied by the value entered into **Slippage %** field.

For short entries, the slippage factor is calculated by measuring the range from the theoretical entry price to the low. The slippage factor is then added to, or subtracted from the theoretical entry price, to obtain the simulated fill price.

Example: [^Top](#)

Here's is an example how slippage is handle with a Long Entry order:



Description:	Example Values:
Slippage percent	25%
Theoretical buy order price	100
High Price (for the day)	120
Slippage Factor	$(120 - 100) \times 0.25 = (20 \times 0.25) = 5$
Simulated fill price	Order Price + Slippage Factor = $(100 + 5) = 105$

The distance between the high price and the order price is multiplied by the slippage factor. In this example, the difference between the high price and the order price is 20 points. The 20 points are multiplied by the 25% slippage to get an estimated slippage of 5 points. The fill price for the order will be 5 points worse than the stop order price of 100 simulating a fill at 105.

Slippage for sell orders is computed using a similar calculation using the distance between the order price and the low of the day.

Note: [^Top](#)

In historical back testing an inability to accurately estimate slippage can lead to two types of mistakes:

- Underestimating frictional costs may lead you to trade a system that produces spectacular hypothetical results now, but does not hold up later on in real trading.
- Conversely, overestimating frictional costs may dissuade you from trading an otherwise good system.

It is also worth noting that the more frequently a system trades, the more profound the impact of frictional costs will be.

Minimum Slippage: [^Top](#)

This Global Parameter only applies to Futures trading.

Minimum Slippage is based on a fixed currency value and it applied in conjunction with **Slippage %** (above). This means when the **Slippage %** field is set to a value greater than Zero, this parameter ensures that some slippage cost is assessed against every trade. Trading Blox will impose **Minimum Slippage** only if the currency value resulting from the **Slippage %** calculation is less than the currency value of slippage, this parameter will ensure the **Minimum Slippage** parameter value is used to adjust the fill-price.

When an entry occurs at or near the price High or Low of the day, the potential adverse range is practically nonexistent, so the **Slippage %** calculation would be at or near zero. In this case the **Minimum Slippage** amount can ensure that some slippage is assessed to the trade.

If Slippage **Slippage %** is set to zero the slippage for all trades will be the value specified by the **Minimum Slippage** parameter amount.

Note: [^Top](#)

By using a Minimum Slippage value when the fill is at or near the High or Low price of the day, the fill-price can show a fill-value that is outside the range of the High or Low price of the fill-date.

Last Edit: 9/12/2020

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 204

Section 5 – Stock Parameters

All Stock class instrument order transaction cost are controlled by the settings in this section.

Stock Parameters		
Minimum Stock Volume (shares)	Step ☐	10,000
Commission per Stock Trade	Step ☐	\$0.000
Commission per Stock Share	Step ☐	\$0.010
Commission by Stock Value (%)	Step ☐	0%
Sell Stock Split Remainder		False
Use Stock Split Ratio		True
Earn Dividends		True
Pay Margin on Stocks		True

Stock Parameters Global Setting Control

Global Parameter:	Description:
Minimum Stock Volume (shares): ^Top	This parameter applies only to stocks. It establishes the minimum daily trading volume, in shares, required to enter a stock position. It is based on a 5-day exponential moving average of the volume. This value can be viewed by using the <code>instrument.averageVolume</code> property when using the Builder Edition.
Commission per Stock Trade : ^Top	This parameter indicates the commission charged on a per trade basis. This type of commission charge is used by some brokers for stock brokerage accounts. Assuming a theoretical <i>Commission per Trade</i> of \$5.00; Buying 100 shares of XYZ stock would count as one trade, and selling that same 100 shares would be a second trade, for a total transaction cost of \$10.00.
Commission per Stock Share : ^Top	This parameter indicates the round-trip charge for each stock share traded. This can be used in place of or in addition to the <i>Commission per Trade</i>. For example, assuming a <i>Commission per Contract</i> of \$0.02, buying and then selling 1,000 shares of IBM would result in a total transaction cost of (\$0.02 per share) x (1,000 contracts) = \$20.00. NOTE: Most stock brokerages charge commission for each one-way trade, once for the entry, and once for the exit. Trading Blox applies a <i>Commission per Stock Share</i> of double the per share commission charged by your broker for one-way trades.
Commission by Stock Value (%): ^Top	Uses a percent of the stock value as the commission amount.

Global Parameter:	Description:
Sell Stock Split Remainder: ^Top	Trading Blox recommends this option be set to off. A parameter's setting of TRUE, will enable Trading Blox to close any fractional shares that are less than the full price of a stock share. This result is not uncommon after a stocks price split/consolidation adjustment. It will happen to all fractional value shares that become an amount after a stock adjusted split that is less than the amount required for a single share price. A parameter setting of FALSE will show the number of shares that are all priced at the full share values. The remainder of share price adjustment transactions will be in the account value. A TRUE parameter setting will enable Trading Blox to remove the fractional share and value the open trade at the full share amount and place the value of terminated share value in the Closed amount to the account's total amount. Here is how the adjusted volume of share is processed: With stocks, when the data is back-adjusted for stock splits, the profits made/lost are respectively smaller as the prices get smaller. Take MSFT as an example. The close on Jan 2, 1987, was \$47.75, and the close on Feb 2 was \$73.00. But your back-adjusted data will probably show around \$.17 on Jan 2 and \$.25 on Feb 2. So if you bought 100 shares at the close on Jan 2 and sold at the close on Feb 3, you would have made \$25.25 per share, or \$2,525 total. But many simulations would show a profit of just \$.08 per share, or \$8 total. If you set this parameter to true, Trading Blox will determine the real profit for this trade was \$25.25 per share. Your simulation will then be accurate even though the data has as back-adjusted. Trading Blox will convert the profit on each trade by the stock split ratio between the trade entry and the trade exit. It will also convert the volume in the data file back to the real volume amount using the stock split ratio. The stock split ratio is the unadjusted close/adjusted close for any given day. This is the reason you should always have the unadjusted close included as a data series in the stocks data file. Here is another example: Create two MFST files in CSI, one without adjustments, and one with stock split adjustments. If you look at the volume on 2005-01-03 for MSFT using the unadjusted data, you will see that the volume presented is 39,545,600. (Note that there is a volume multiplier of 100 in play here as well, so the number you see displayed is 395456.) The

Global Parameter:	Description:
	stock split ratio on that day is 16. When the ratio amount is divided into the displayed value, the actual volume traded will show it was 2,471,600. If you are trading a max percent of the day's volume, then you would want to use the adjusted volume number rather than the unadjusted number that provided in the data file. Likewise, if you use the example above, you want to make sure the profit on a 100 share trade reflects the actual value of that trade in "today's" dollars. So Trading Blox converts the profit using the stock split ratio as of the trade entry. Also, Trading Blox needs to know how the stock split is applied to the dividend distribution. Let's use this example to see how a purchase of 100 shares of the stock with a 2x stock split and a \$.50/share dividend. Dividends are a different type of profit than if you purchased 100 shares and then the stock had a \$.50/share dividend, which then received a 2x stock split. This detailed information is why it's important to keep the stock splits and dividends separate. The stock splits alter the data in a geometric manner, whereas the dividends are simple unadjusted per share amounts. If you try to proportionally adjusted dividend data into a stock split adjusted series, you will get the wrong P&L every-time. Be sure to set your CSI UA setting to adjust the stock data by stock splits only, be sure to include the unadjusted close, and call CSI to get the unencrypted dividend data so Trading Blox can build those files for you. CSI enable unencrypted dividend data to all Trading Blox customers at no extract cost to a user's stock data subscription.
Use Stock Split Ratio: ^Top	Default value = True. This option when set to true, will adjust the profit of each trade based on the stock split ratio to bring the profit into current dollars (The current stock split ratio is by definition 1). A trade of 100 shares of MSFT from back in the day will yield a tiny profit due to the huge stock split ratio and super small prices. This amount is multiplied to convert back to today's dollar value. To look at it another way; it converts the way back prices to real unadjusted prices to compute the profit. When the stock splits during a trade, the quantity is adjusted. An example purchase of 100 shares that then has a 2x stock split will result in a closing trade of 200 shares. In this example, you will notice an entry quantity and exit quantity that can be different. This accounting

Global Parameter:	Description:
	necessary to enable this transaction to account for commission correctly both on the entry and exit and in the application of any potential dividends. Dividends are applied to the number of shares held on the date of the dividend was applied to the stock. Dividend prices provided by CSI are UNADJUSTED, so the actual quantity should always be used for trade sizes to make sure the dividend times the quantity is always correct. When set to False, this option accommodates the cases where the source of stock data is obtained from a source that doesn't provide the all the column values required in minimum stock data columns listed for Stocks. Trading Blox requires the data file creation include splits, and provide dividends for symbols with dividends in a separate dividend file. When data uses a nonstandard back adjust and adds dividends or capital changes to the symbol's data file this, the option can be set to False, so the test will know the data is nonstandard. Summary: When the "Use Stock Split Ratio" is set to True = ON, the data should be stock split adjusted and the profit will be adjusted based on this assumption. Any dividend files will be adjusted accordingly as well. When this parameter turned Off = FALSE, the data can be in any format, and the P&L will simply be the difference between entry and exit price.
Earn Dividends: ^Top	Set to true to account for dividends in the test. The unencrypted dividend files should be setup in the Dividends folder specified in the Data Folders and Data Options path location section.
Pay Margin on Stocks: ^Top	This parameter is for stocks only, and it defines whether the system will charge margin for excess cash used when buying stocks. The values used are from the system wide base currency rate files. In the case of USD, the file is the USD_Rates.txt file in the Forex Rates data directory. The Borrow Rate is used when accounting for margin. Margin Interest is applied as a charge every day the available cash falls below zero.

Section 6 – Forex Parameters

All Forex order transactions are controlled by the settings in this section.

Forex Parameters

Forex Trade Size (in base currency) Step ☐ 1,000

Account for Forex Carry True

Use Spread Slippage False

Forex Parameters Global Setting Control

Global Parameter:	Description:
Forex Trade Size:	The inter-bank forex market trades in units of 100,000 base currency. Many retail forex brokers allow trading units of smaller size. This parameter controls the unit size to allow for simulation of trading through retail forex brokers.
Account for Forex Carry:	This determines whether or not to account for Forex Carry charges (see Forex Carry Calculations for details).
Use Pip-Based Slippage:	Since many retail Forex brokers have fixed spreads per market and guarantee "zero slippage" fills, Trading Blox provides for this using a fixed per-market slippage which is defined by the spread in pips for each market. The spread for each market should be set to the figures your Forex broker uses or a larger number for more conservative testing. See Forex Dictionary for information on changing the Forex market spreads. When set to TRUE, the parameter <i>Use Pip-Based Slippage</i> overrides the normal Trading Bloxslippage calculations (based on a fixed amount or percentage) for the Forex markets only.

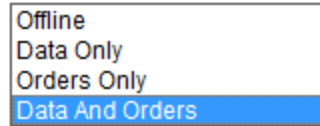
Links:

See Also:

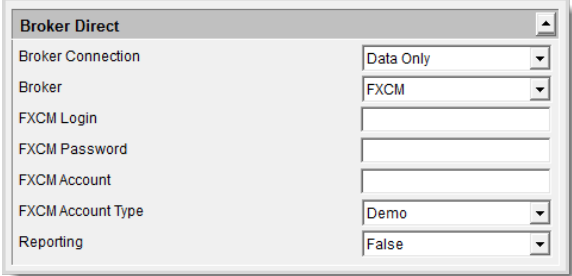
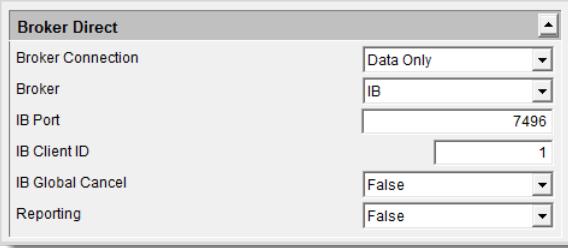
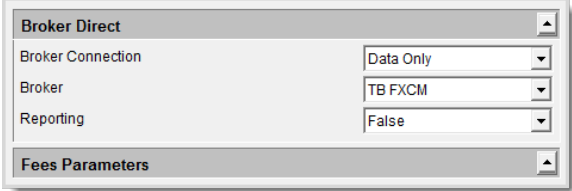
Section 7 – Broker Direct

Default setting for the Broker Direct interface is Offline.

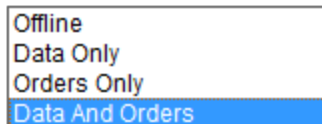
When it isn't Offline, its connection can be set to any one of these three options:

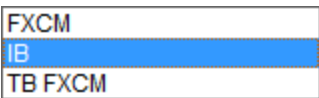


When the Broker Connection is not Offline, the Broker parameter option selection will be set to [Interactive Brokers \(IB\)](#), or one of the two [Forex Capital Markets \(FXCM\)](#) selection options. All three **Broker** selection options will show a different set of parameters. Each **Broker** selection shows the same set of parameters regardless of the selection of three connections that are not Offline. On the left side of this table set is the Offline parameter single parameter option, and below the Offline parameter image is the IB parameters required. On the right side of the table set are at the top is the **Broker** connection parameter options for **FXCM**, and just below that table area are the **TB_FXCM** parameters.

Broker Connection: Offline	Broker Connection: FXCM
	
Broker Connection: IB	Broker Connection: TB FXCM
	

Parameter:	Description:
Broker Connection: ^Top	Offline: The existing data will be used, no data from broker, no orders to broker. Data Only: TB will first update the data from the selected Broker, when possible. Note:



Parameter:	Description:	
		When using the Data option, the Dictionary entry for each instrument needs to be set for the Broker. When the data format is ASCII, then the data will not be updated from the broker. Orders Only: TB will use existing data only, synch positions, and send orders to broker. Data and Orders: TB will update data from broker, synch positions, and send orders.
Broker: ^Top 		FXCM: Data and Orders use the Forex Capital Markets (FXCM) brokerage. IB: Data and Orders use the Interactive Brokers (IB) brokerage. TB FXCM: This Broker selection option uses Forex Capital Markets (FXCM) free courtesy forex data. This option can only be used when the Broker Connection is set to Data Only . Orders cannot be sent to this data only account.
Reporting: ^Top		Broker Direct Log files are saved in Trading Blox Results Folder where the program is installed.
Interactive Brokers (IB):	Descriptions:	
IB Port: ^Top		
IB Client ID: ^Top		
IB Global Cancel: ^Top		
Forex Capital Markets (FXCM)	Descriptions:	
FXCM Login: ^Top		
FXCM Password: ^Top		

Forex Capital Markets (FXCM)	Descriptions:	
FXCM Account: ^Top		
FXCM Account Type: ^Top	Demo	
	Real	

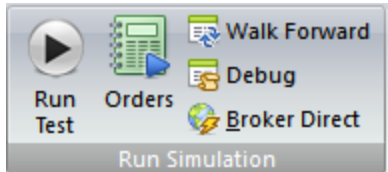
Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 115

7.1 Broker Selection Settings

Broker Direct Help Topic Information Background:

Trading Blox menu options:

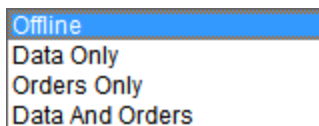


Use **Run Test** to run a normal simulation. If Broker Connection set to Data Only or Data and Orders, then the data will be updated before the test is run. If the data is already updated or static from an external source, then set the Broker Connection to Offline.

Use **Orders** to generate orders. If Broker Connection is set to Orders Only or Data and Orders, then the open positions will be synchronized and the orders will be sent to the broker.

Use **Broker Direct** to start the automated process that generates orders every bar. The first run is a plain simulation run, as it could be mid bar, but starting on the second run going forward, the Order Generation process will be run every bar as the new bar is created. Data is updated (if data connection type is selected), positions are synchronized between TB and the broker, and new orders are sent to the broker.

Broker Direct Broker Connection parameter options:



Broker Connection: Offline, Data Only, Orders Only, Data and Orders

- Offline: The existing data will be used, no data from broker, no orders to broker
- Data Only: TB will first update the data from the broker, if possible.
- Orders Only: TB will use existing data only, synch positions, and send orders to broker.
- Data and Orders: TB will update data from broker, synch positions, and send orders.

When using the Data option, the Dictionary entry for each instrument needs to be set for the Broker. If the data format is ASCII, then the data will not be updated from the broker.

Broker Direct Help Topic Information Background:**Broker Direct** Broker parameter options:

FXCM
IB
TB FXCM

FXCM: Use FXCM as the broker for data and orders.

IB: Use Interactive Brokers as the broker for data and orders.

TB FXCM: Use a built-in account to access free courtesy forex data from FXCM. This option can only be used with the Data Only Broker Connection. No orders can be sent to this data only account.

Broker Direct parameters for Interactive Brokers:

Broker Direct	
Broker Connection	Data And Orders
Broker	IB
IB Port	7496
IB Client ID	1
IB Global Cancel	False
Reporting	False

IB Port: Set to same port as TWS API global option

IB Client ID: Set to a unique number for each Trading Blox installation that is running concurrently against TWS. Useful for multiple installs that might be running different timeframes of different asset classes.

IB Global Cancel: Set to true for a full cancel and replace of all orders for each order generation run. This global cancel will cancel all orders placed in TWS or through any API connection.

Reporting: Set to true to create all normal simulation reports and order reports. Set to false to disable all the normal reports, charts, and graphs when running an intraday system that does not use or need that information.

Broker Direct parameters for FXCM:

Broker Direct Help Topic Information Background:

Broker Direct	
Broker Connection	Data And Orders
Broker	FXCM
FXCM Login	
FXCM Password	
FXCM Account	
FXCM Account Type	Demo
Reporting	False

FXCM Login: Enter the login information provided by FXCM.

FXCM Password: Enter the password provided by FXCM.

FXCM Account: Enter the account to use. Leave blank to use the default account for the login.

FXCM Account Type: Set to Demo or Real. Set based on account type.

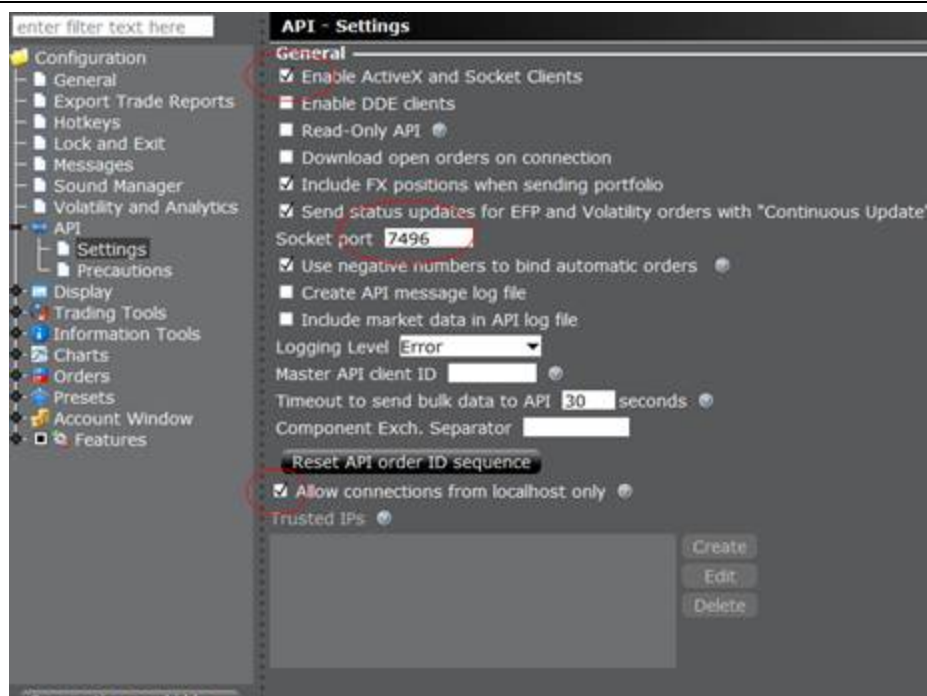
Broker Direct parameter for TB FXCM:

Broker Direct	
Broker Connection	Data Only
Broker	TB FXCM
Reporting	False

This account type can only be used for Data Only. Gets free courtesy data, cannot place orders.

TWS example. Check Enable, set port, check Allow Localhost.

Select Create API message log if you want to track IB messages and debug issues. Log will be located in the IB installation directory.

Broker Direct Help Topic Information Background:

Be sure to use a Paper Trading Account Only.

This is beta test functionality, not to be used for real trading. You are responsible for all trades sent from TB to IB, so be sure to monitor the process closely. Do not leave unattended.

Trading Blox is not responsible for your trading choices or actions.

Futures Dictionary Example:

- 1) Broker symbol needs to match the IB symbol
- 2) Exchange needs to match the IB exchange
- 3) Select IB as the Format
- 4) Select the Trading Month (YYYYMM) for the contract desired only if this is the first time getting data for this instrument. Once there is a text file with historic data available, the last record of data will be used for the futures month.
- 5) Set the bar size (m1, m5, m15, m30, H1, H2, H4, D1)
- 6) File name is used to save cached data loaded from IB. Each run will load more data and append to file. Delete file as needed if data becomes corrupted for any reason.

Broker Direct Help Topic Information Background:

Futures Dictionary Editor

Market Specifications

Symbol	GC	Contract Size	100 oz troy
Description	Gold	Big Point Value	100.00
Broker Symbol	GC	Broker BPV	0.00
Format	☐ ASCII Text ☐ Metastock ☒ IB	Margin	6,748.00
File Name	GCIB1Min.TXT	Display Digits	1
Folder	@	Tick Unit	0.1
Exchange	NYMEX - New York Mercar	Minimum Tick	1
Trading Months	201610	Round Lot	1
Currency	USD	Order Sort Value	820
Closely Correlated	CD:EC:SF:	Group 1	metal
Loosely Correlated	AD:BO:BP:CC:CL:EM:HG:F	Group 2	
		Custom 1	
		Custom 2	
		Bar Size	m1

Buttons: Examine, Edit...

Forex IB Dictionary Example:

- 1) Set Format to IB
- 2) Use **combo box** to select market (EURUSD) and bar size (m1). The symbol, broker symbol, description, and file name will be auto populated.
- 3) Set the tick unit, min tick, spread, and commission as needed.
- 4) File name is used to save cached data loaded from IB. Each run will load more data and append to file. Delete as needed if data corruption problems.

Broker Direct Help Topic Information Background:

Market Specifications

Format: **IB**

Symbol: EURUSD-m1 **EURUSD** **m1**

Broker: EURUSD

Description: EEU Euro / US Dollar - m1

File Name: EURUSD-m1.TXT Examine

Folder: @

Examine Base Rates Examine Quote Rates

Tick Unit: 0.00001

Min Tick: 5

Spread: 0.00

Commission: 0.00

Forex FXCM Dictionary example:

Market Specifications

Format: **FXCM**

Symbol: EURUSD-m1 **EURUSD** **m1**

Broker: EUR/USD

Description: EEU Euro / US Dollar - m1

File Name: EURUSD-m1.TXT Examine

Folder: @

Examine Base Rates Examine Quote Rates

Tick Unit: 0.00001

Min Tick: 1

Spread: 0.00

Commission: 0

Stock Dictionary Example:

- 1) Set the symbol as needed, with bar size or other descriptors.
- 2) Set the Broker Symbol to match what IB expects.
- 3) Set the Exchange to SMART
- 4) Set the format to IB
- 5) Set the bar size as needed
- 6) Cached data file is created as symbol.txt in the stock data folder set in preferences. Delete as needed if data corruption problems.

Broker Direct Help Topic Information Background:

Version	7											
Symbol	Description	Exchange	Sector	Industry	Country	Data Vendor ID	Broker Symbol	Active Status	Round Lot	Currency	Format	Bar Size
UPRO_m1	UPRO	SMART			USD		UPRO	A	1	USD	IB	m1
UPRO_m5	UPRO	SMART			USD		UPRO	A	1	USD	IB	m5
QQQ_m1	QQQ	SMART			USD		QQQ	A	1	USD	IB	m1
QQQ_m5	QQQ	SMART			USD		QQQ	A	1	USD	IB	m5

Note that when using daily data from an external source, the stock dictionary does not need to be changed. The IB format is only used for getting data from IB, not for sending orders. The bar size defaults to daily. A normal stock dictionary with ASCII as format and D1 as bar size can be used when using existing daily data and just sending daily orders to IB.

Section 8 – Equity Manager

The following is a comprehensive list of the Trading Blox global Equity Management parameters, and an explanation of how each is used. The Test Parameter values can be viewed and changed at the bottom of the Suite Parameters tab that is attached to every suite.

Allocation Equity Calculation:

Equity Manager area in the Global Parameters section sets basis for how much can be allocated to a system:

The screenshot shows a dialog box titled "Equity Manager". It contains the following parameters and their values:

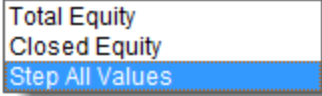
Parameter	Step	Value
Test Starting Equity	☐	\$500,000
Order Generation Equity	☐	\$0
Order Generation Equity High	☐	\$0
Leverage (fraction)	☐	1
Trading Equity Base		Total Equity (dropdown)
Drawdown Reduction Threshold (%)	☐	0%
Drawdown Reduction Amount (%)	☐	0%

Global Suite Equity Parameters

Property Name:	Description:
Test Starting Equity: ^Top	This parameter sets the currency value of the account balance at the start of the simulation (at this point Closed Equity = Total Equity). It is an initial value only. Trading Blox internally recalculates the various running account balances daily, for the duration of the simulation. When generating orders, Trading Blox uses the <i>Starting Balance</i> to calculate the number of shares or contracts for each order. In order to make sure that the order sizes are accurate, the Starting Balance value should be updated daily using actual trading account balances. The <i>Starting Balance</i> for orders should be set to the value corresponding to the <i>Trading Equity Base</i> parameter (see below). For Total Equity use the actual current daily account values. There are two special additional Options which affect order generation. These can be used in

Property Name:	Description:
	conjunction or separately from the Broker Positions .
Order Generation Equity: ^Top	When generating orders, the system first runs a test based on the Global Equity Settings. Let's say you start with \$100,000 and end with \$150,000 over the course of the test. However, to generate orders your actual account may have a different amount. Perhaps you added money to your account, or made a withdrawal. If you want your orders to reflect your account balance, enter it here. The Closed Equity is set to this amount minus any Open Equity in actual positions entered. So your total equity will equal the order generation equity when generating orders. Set to zero to disable this property and use the test simulation equity. Note: This is only used for Order Generation. It controls the amount of equity available for trading on the order generation date.
Order Generation Equity High: ^Top	Works in concert with <i>Starting Balance</i> (see previous). NOTE: This parameter was designed predominantly for use when generating orders, and is of consequence only if you are using the <i>Drawdown Reduction Threshold</i> (DRT) and <i>Drawdown Reduction Amount</i> (DRA) parameters (see Equity Management Parameters). If you are generating orders with DRT & DRA, Account High works in concert with Account Balance to replicate the current actual drawdown in an actual trading account. In order to mimic the way Trading Blox sizes positions during back testing using DRT and DRA, the value that should be entered in Account High is the highest equity value achieved to date. For instance, if the highest equity value achieved to date in a real trading account is \$500,000, and the Account Balance is \$400,000, then Trading Blox recognizes that the account is in a 20%

Property Name:	Description:
	drawdown, and passes this information along for use by DRT and DRA. Set to zero to disable this property and use the test simulation high. Order Generation High is disabled if Order Generation Equity is disabled.
Leverage: ^Top	While this parameter was designed primarily to allow simulations of stock margin accounts, it can also be used to adjust the leverage of futures trading. Since the leverage available in a futures account is already quite large, this parameter should generally be set to a value of one or less for futures testing. Lending institutions may lend funds that can be used to leverage your investments. If a lending institution agrees to lend you some amount for every amount of existing account equity, then in Trading Blox the <i>Trading Equity</i> (the total money available for trading) would be twice the amount of equity corresponding to the <i>Equity Base</i> parameter. Assuming <i>Equity Base</i> is set to "Closed Equity", this corresponds to a Leverage value of 2, with the actual trading equity equal to twice the Closed Equity at any point in time. If no money was available from lending institutions, then the trading equity would equal Closed Equity, and the value of <i>Leverage</i> would be 1. Trading Equity = Leverage x Account Equity For example, if <i>Leverage</i> is increased from a value of 1 to a value of 2, and position sizing parameters remain unchanged, then orders will be sized twice as large. <i>Leverage</i> can also be set to a fractional value. Setting Leverage to a value of 0.50, for instance, will simulate allocating 50% of your account to cash.

Property Name:	Description:
Trading Equity Base: ^Top	The Trading Equity Base parameter can have one of two separate values:  Closed Equity - Reflects changes in equity from closed out trades only Total Equity - Reflects Closed Equity plus the total Open Equity balance The value of the <i>Trading Equity Base</i> defines the available <i>Trading Equity</i>. <i>Trading Equity</i> is the equity figure used for all money management position sizing calculations and for determining the maximum amount of equity that can be used for margin or purchasing stocks. You can step the <i>Trading Equity Base</i> parameter by selecting the <i>Step All Values</i> item. This will result in two separate runs, one each for Total Equity and Closed Equity.
Drawdown Reduction Threshold: ^Top	This parameter and the one that follows allow simulation of the Turtle risk-management rules. They work in concert to reduce the likelihood of dramatic losses by cushioning equity drawdowns. While these parameters and the associated risk management algorithms were designed to support the Turtle System they can also be used with other systems <i>Drawdown Reduction Threshold</i> (DRT) defines the drawdown point at which the amount of

Property Name:	Description:						
	money available for trading is reduced. For example, if DRT is set to 10%, then when a 10% drawdown is incurred, trading equity will be reduced by the percentage specified for the following parameter, <i>Drawdown Reduction Amount</i> (DRA). If you wish to eliminate the effect of DRA entirely, then set the value of DRT to 100% and the DRA to 0%. The drawdown utilized by DRT is always measured from the most recent Closed Equity peak, and is calculated based on the Closed Equity.						
Drawdown Reduction Amount: ^Top	Works in concert with <i>Drawdown Reduction Threshold</i> (DRT), above. <i>Drawdown Reduction Amount</i> (DRA) specifies the percentage by which Closed Equity is reduced in the event of a drawdown of the magnitude specified by the DRT parameter. This parameter pair only affects the sizing of (and the capital available for) new trades. It does not modulate the size of existing positions. The table below is based on the default Turtles settings of 10% for DRT and 20% for DRA, and shows how DRT and DRA work together to reduce the equity available for investment during periods of drawdown. As is shown, Available Equity = Actual Closed Equity * (1 - DRA). In this example, the most recent Closed Equity high (the peak from which drawdown is measured), is \$1,000,000. In Trading Blox 2.0.10, DRA is calculated such that you will never get a zero or negative equity amount. In this way you can test a 90% DRA if you wish. <table data-bbox="906 1793 1385 1892"> <thead> <tr> <th>Actual Closed Equity</th><th>Actual DD</th></tr> </thead> <tbody> <tr> <td>1,000,000</td><td>0%</td></tr> <tr> <td>900,000</td><td>10%</td></tr> </tbody> </table>	Actual Closed Equity	Actual DD	1,000,000	0%	900,000	10%
Actual Closed Equity	Actual DD						
1,000,000	0%						
900,000	10%						

Property Name:	Description:								
	<table border="0"> <tr> <td>800,000</td><td>20%</td></tr> <tr> <td>700,000</td><td>30%</td></tr> <tr> <td>600,000</td><td>40%</td></tr> <tr> <td>500,000</td><td>50%</td></tr> </table> When actual closed equity drops by 10% (to \$900,000), the <i>Drawdown Reduction Threshold</i> is violated, and DRA kicks in, reducing the equity available for investment to \$720,000 (actual closed equity less 20%). Now, any new trades-instead of being sized to the actual closed equity figure of \$900,000-will be sized based on the available equity of \$720,000. DRT and DRA operate in a step-wise fashion: While equity fluctuates below \$900,000 but above \$800,000 positions will continue to be sized based on the available equity of \$720,000. The effect of DRT and DRA is cumulative (exponential not geometric): If the actual equity falls an additional 10%, to \$800,000 (a 20% drawdown as measured from the previous equity peak of \$1,000,000), then available equity is reduced further to \$512,000 or 64% of actual equity. The process continues as the drawdown steepens. DRT and DRA work symmetrically as the account begins to climb out of the drawdown, and have no further effect between the time the actual account equity exceeds the original <i>Drawdown Reduction Threshold</i> of \$900,000 and a new equity high is achieved. If the current drawdown should re-commence before a new equity high occurs, then DRA will kick in again at \$900,000. Notes: - o Whenever a new Closed Equity high is achieved, DRT and DRA sit idle, waiting for the next 10% drawdown (or other user-specified DRT value) to occur. - o When running historical simulation tests, the starting <i>Account Balance</i> is treated as an equity high when the simulation	800,000	20%	700,000	30%	600,000	40%	500,000	50%
800,000	20%								
700,000	30%								
600,000	40%								
500,000	50%								

Property Name:	Description:
	commences, unless the value of Account High has been set even higher (see Starting Balance and Account High, next). - O DRA is active only during drawdowns whose magnitude exceeds the percentage specified by DRT. So if DRT is set to a large value, it is possible that its threshold will never be violated. If DRT is set to 100%, or if DRA is set to 0%, then the effect of this parameter pair is completely disabled. - O If the values of DRT and DRA remain static, while <i>Percent Risk per Trade</i> is increased (i.e. as the percentage of equity risked on each trade is increased and drawdowns are magnified), then DRT will be triggered more frequently, and DRA's effect on performance will be more profound. - O During very large drawdowns (particularly those that occur near the start of a simulation), it is possible that available account equity-having been dramatically reduced by the effect of DRA-will be insufficient to allow new trades to be put on, particularly for futures. At this point, trading will cease. - O The drawdown utilized by DRT is always measured from the most recent Closed Equity peak, and is calculated based on the Closed Equity high.

Section 9 – Fees Parameters

When the [Use Global Adds Draws](#) option is set to **TRUE**, fee amounts will be recorded in the [Capital Adds Draws](#) Transaction file.

Global Parameter Fees will create a fee transactions based upon the [Charge Management Fee](#) selected setting. The amount of a fee event is applied at the end of a trade day off the current selected test date period. Fee transactions are applied as one of the last **After Trade Day** equity adjustments.

Each fee type can have a different accrual period. As fees accrue, they are booked as draws from the account equity.

Fees Parameters	
Management Fee (Annual % of Equity)	Step ☐ 0%
Charge Management Fee	Daily
Incentive Fee (% of Net New Profits)	Step ☐ 0%
Charge Incentive Fee	Daily
Use Capital Adds Draws	False

Global Parameters - Suite Fee Parameter Options

Capital Adds & Draws transactions can be included in the transaction basis.

Parameter:	Description:	
Management Fee (Annual % of Equity) ^Top	This fee is a percentage calculation. It is the cost of inclusion into the trading management structure.	
Charge Management Fee ^Top	Daily Monthly Quarterly Yearly	Management fees accrual timing is determined by this selected value.
Incentive Fee (% of Net New Profits) ^Top	Incentive fee is the reward fee available to management for the gain of the selected period.	
Charge Incentive Fee ^Top	Daily Monthly Quarterly Yearly	Incentive fees are determined by the user's selected accrual period.
Use Capital Adds Draws ^Top	True False Step True to False	When this parameter is set to FALSE . When FALSE , Capital Adds & Draws transaction records will not show fee transactions. When it is set to TRUE , the fee accrual transactions will become records in the the change in the account

Parameter:	Description:	
		equity value is included in the calculation basis percentages.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 138

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 138

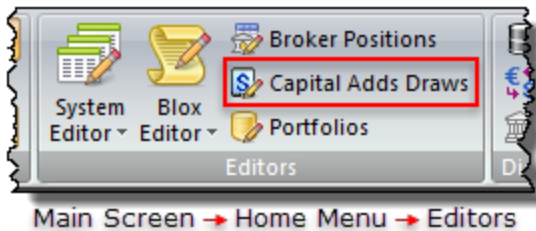
Section 10 – Capital Adds Draws

This Trading Blox feature provides a screen editing option to add or draw/remove capital from a trading account. This ability can be set to happen automatically, or entered earlier. To enter manually, the date the amount change is required so the simulation process will know when to implement the transaction.

As part of the automatic transaction features are some TB commands can be scripted to add or remove amounts as the scripted logic condition becomes satisfied.

Capital Additions and Draws Editor:

Access to the dialog where the information can be seen can be displayed from the main screen's **Editor** group item shown here.



This editor allows new capital amounts values to be added to existing amounts or removed. Capital transactions are accumulated in their own file and can be reported and associated with the [Global Suite Parameter Fees Control](#).

Values for an add or a draw amount can be created by keyboard entry using this dialog. Each record requires the user to enter a transaction date and an amount that is to be added or removed. Comments are not required, but often enable better understanding.

When transaction values are recorded through testing, they will always be on a test date that matches one of the simulation's test dates. All transaction are applied as one of the last transaction updates performed in the **After Trade Day** script section.

The **After Trade Day** script section is updated for every test date during the range of a simulation's test range.

How each Add or Draw value affects the trading account is determined by the **+/-** sign that is enter with each records value. A **+/-** a **+** sign adds value and a **-** sign removes value.

When the [Global Suite Parameter's Fees Parameters](#) section is enabled, it can be enabled when parameter option [Use Capital Adds Draws](#) is enabled so the Fees transactions appear the **Capital Adds Draws** transaction record file.

Capital Adds Draws

Date	Amount	Comments
20090105	-2500	Cost type 1
20100105	-2500	Cost type 1
20110105	3500	Account Transfer
20110105	-4500	Management Fee
20120515	-1500	Cost type 2

Records added earlier

Record being added

OK

Cancel

New Add Draw

Delete Add Draw

Accept edits, closes editor

Cancels edits, closes editor

Adds Record

Deletes Record

Date (YYYYMMDD) 20120515

Amount (+ add, - draw) -1500

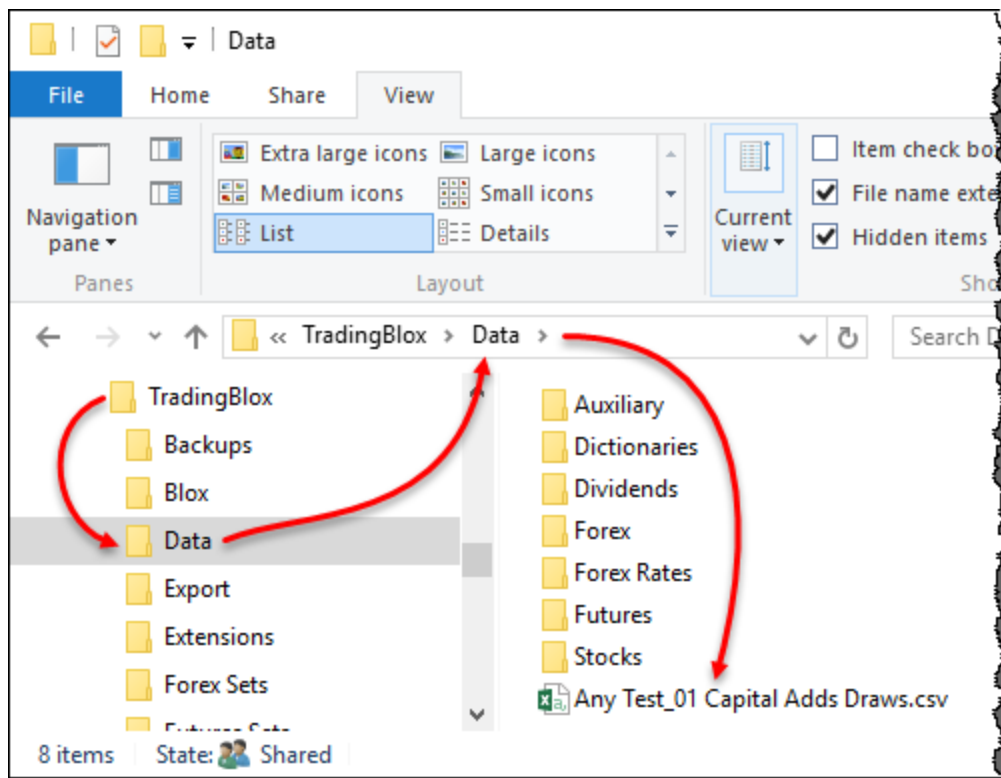
Comments Cost type 2

Tuesday

Capital Adds & Draws Item Information

Each time a capital value is added or removed, the Capital Adds Draws file is updated. As capital record transactions are recorded, and each transaction record date is on a matching test date, the transaction on that date is recognized and can change the equity amount.

Capital Add Draw records for each system is kept in the Trading Blox default Data folder created during program installation. However, it can be changed using the Data Section of the [Preferences Editor](#):



Capital Adds & Draws File Folder Location

The contents of the Capital Adds Draw file will displayed as shown for each program mentioned:

File display in Notepad: Spreadsheet display:

Version,7	Version	7	Comment
Date,Amount,Comment	Date	Amount	
20090105,-2500,			
20100105,-2500,			
20110105,2500,	20090105	-2500	
20120105,-2500,	20100105	-2500	
20130104,4500,	20110105	2500	
	20120105	-2500	
	20130104	4500	

The table shows a file example of how the information is contained in a Capital Adds Draw file.

Script Syntax:

Syntax:

value = test.capitalAddsDrawsTotal

Parameter:	Description:
<none>	See above property description above, and Return results below.

Example:

```
' Access current running total of Capital Adds & Draws  
value = test.capitalAddsDrawsTotal  
' Send capitalAddsDrawsTotal property value to Print Output.csv file.  
Print "Capital Adds & Draws: ", value
```

Return:

Capital Adds & Draws: , -500 (Using a script for the above returns this amount)

Links:

[Fees Parameters](#) [OtherExpenses](#), [UpdateOtherExpenses](#)

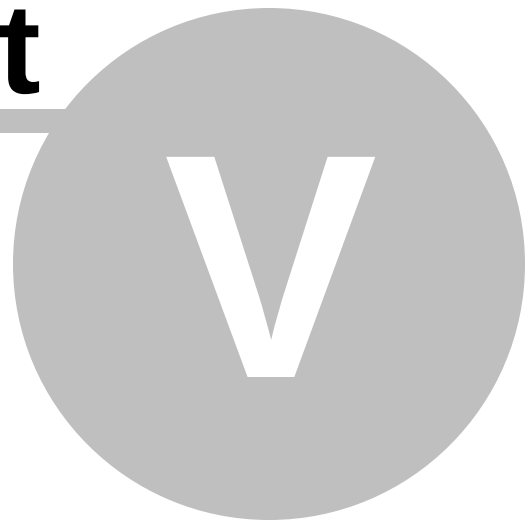
See Also:

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 119

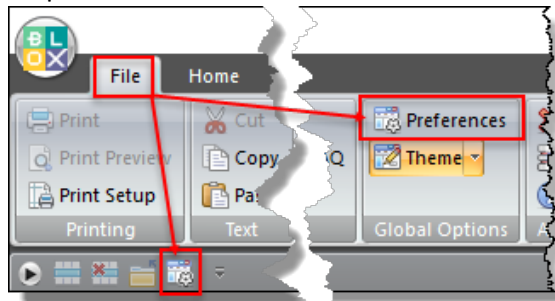
Preferences Editor Menu

Part



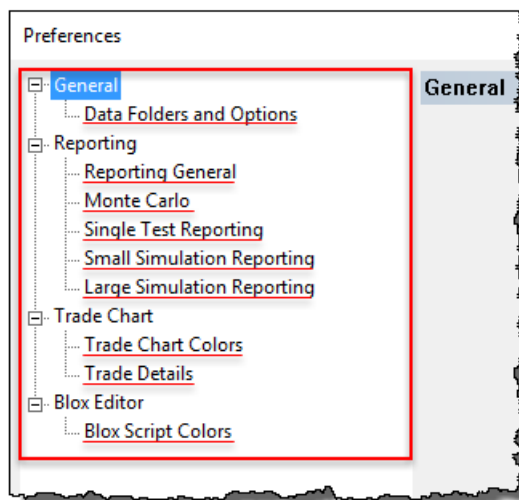
Part 5 – Preferences Editor Menu

Access to Preferences Editor Settings is available on the Main Menu File Tab's Global Options Preferences control and when the Preferences Editors icon appears in the Quick Access Toolbar strip.



Main Menu File Tab Preferences Access Control

Trading Blox's Preferences dialog let you customize your testing environment and reporting information.



Preferences Section Menu

Image items are links to each of the Preferences Editor Settings areas. Clicking on a link in the image above or in the table below will display information about each section.

Preference Sections:	Description:
Data and Order Folders	Sets the folders for data and the destination folder for orders.
Reporting General	Overall settings for reporting.
Monte Carlo	Setting for the Monte Carlo Simulations
Single Test Reporting	Sets the reports included for tests that have only one parameter combination.

Preference Sections:	Description:
Small Simulation Reporting	Sets the reports included for tests that have only a few parameter combinations. The threshold for determining the difference between a small test and a large test is set in the Reporting General section.
Large Simulation Reporting	Sets the reports included for tests that have many parameter combinations.
Trade Chart Colors	Sets the colors for the elements of the trade charts.
Trade Details	Setting for which trade detail items are shown on the chart.
Blox Basic Preferences	Settings and preferences for Blox Basic

Links:[Trading Blox Parameter Settings](#)**See Also:**

Edit Time: 3/6/2022 4:07:41 PM

Topic ID#: 5010

Section 1 – Data Folder and Data Options

Trading Blox will work with Futures, Stocks and Forex instrument data files that conform to the information available in the [Historical Data Setup](#) section. The data setup information describes how the data needs to be formatted for each of the three type of instrument types. It is important the data files follow the data format information so there aren't any issues with how Trading Blox recognizes the information in the files.

Data locations and their instrument formats for all three types of instruments are arranged during installation. Data updates from a data download service can use the default data folder locations, if the data service software uses the default data location as their file destination folder. If that isn't possible for any reason, the data folder locations will need to be changed.

Data files can be located anywhere on your disk. To change the data folder location, it should only be changed by the Windows folder access dialog. By using this file access dialog, shown in the next image below, the risk that the folder path information contains a typographical error is close to impossible. This is important because it will prevent Trading Blox from finding the data for testing. access to the data.

Change Current Settings:

To change an instrument type folder location click the button listed next to each market type to open the Windows folder so that the location where those files are located can be listed in each item's field area:

Data Folders and Data Options Preferences

[See Edit Settings Example below:](#)

Data Folder Name:	Default Folder Path Locations:
Futures Folder: ^Top	The default location folder used for storing the futures data is where the installed data is installed. Trading Blox will first look in the Futures Data folder for the Futures Dictionary, and then look in the Dictionaries folder. It is not recommended to have a Futures Dictionary in both locations. Within this folder can be other sub folders, or short cuts to other external folders. In addition, the Futures Dictionary has a folder location for each market that is actually used to find the market data. This default folder is where the Find Symbol button on the right side of the Futures Editor will search to locate the data file when that data can be elsewhere. When the Futures Data folder is changed, a message box will indicate whether there exists a Futures Dictionary in the new location and/or the Dictionaries folder. If a Futures Dictionary does exist it will be used. The message box will ask if the Folder Locations for all markets should be reset. This process will locate all the markets using the default folder and replace the current folder

Data Folder Name:	Default Folder Path Locations:
	location for each market with this new location. This is useful if the all the futures dictionary information is the same, but the folder location has changed. This is not recommended if custom or edited folder locations are used, as they will be replaced with the default location based on the new Futures Data folder location. If you do not have a Futures Dictionary (futuresinfo.txt file) in the Futures Data folder or Dictionaries folder, you have doesn't find a file, there will be warning messages for each file not found when Trading Blox is first started. Warning messages will be displayed in the main screen's Log Window display area when the Log Window is enabled. If the Log Window is not active, the messages will be lost. If the Log Window is active, but closed, the messages will be available. When a Futures Dictionary is missing, or a file is unavailable, you can use the Futures Dictionary Editor to create a new Futures Dictionary. If the file no longer exist, it can be removed, or edited if it is available under a different symbol reference. When dictionary file is not found or created, or has errors in the file name or location of the file, Trading Blox will not be able to read the specified data. New futures files must be added to the Futures Dictionary so that Trading Blox can find and load them for testing. Just add the markets for which you have data. See Note Reference: 2 Default: C:\TradingBlox\Data\Futures
Stock Data: ^Top	The location entered into this setting is the folder used for storing the stock data. (add details about how it works) Trading Blox Stock File Search Structure Process: Stocks are the only instruments that can have multiple dictionaries when it comes time to locate stock symbols. This means that when Trading Blox starts, it looks for files using the path settings established in the Preferences Data and Folder section. In each folder and each symbol it finds it will add it to a list of symbols that it will use to determine if it can be in a Stock Dictionary, or when a stock dictionary is not used, it will be displayed in the Portfolio Editor's Instrument Symbol list section. In this next set of images, the path is set to D:\Data\Stocks. That path has five sub-folders and a large number of symbol files. All the files in all the folders, plus all the symbols in the Stock Set's portfolio files will be record so they can be used in a Stock Dictionary, when that is the process being created, or as a list of symbols in the Portfolio Editor's Instrument Symbol list section.

Data Folder Name:	Default Folder Path Locations:
	Example Stock Folder and Files Structure Example Stock Set Default Stock Portfolio Example All the files and folders in all the places shown in the example above, which includes the symbols listed in the Stock Set Portfolio files, will be listed in the Instrument Symbol list section when a Stock Dictionary is not used. When a dictionary is being assembled, the same search structure will be used to locate symbols. If you do not have a Stock Dictionary (stockinfo.txt or stockinfo.csv file), it will look for a Metastock MASTER file with that name. If it finds the MASTER file, it will assume Metastock data and use that format process to read the data. If it does not find a MASTER file, it will assume ASCII Text files and look for all the .txt or .csv files in the specified folder, and all the folders contained with the top level Stock folder. Once all the folders have been searched, it will then search all the Stock set portfolio files and collect the symbols in those files. When there isn't a stock dictionary file, and the above symbol information is done its symbol search, Trading Blox will create a list of symbols from the file names and set files and display that list in the Stock option of the Portfolio Editor symbol listing. If a symbol is listed in a Stock file portfolio file, and there is no data file with the same name, that item in the list will be unable to load that symbol name information. Trading Blox flags portfolio symbols that don't have data files when it is first started and creates a warning message in the main screen's Log Window display area when the Log Window is enabled.

Data Folder Name:	Default Folder Path Locations:
	See Note Reference: 2 Default: C:\TradingBlox\Data\Stocks
Dividends: ^Top	The Dividend files are named using the symbol and the dividend file suffix, default suffix is "_Div.csv" but this can be changed in the TradingBlox.ini file. Notes: This folder supports the CSI dividend files that are available with a Stock subscription. Dividend files do not get updated when the stock file the dividend information applies. Instead, dividend files must be updated manually so the stock data will stay aligned to how the stock price will be effected when a dividend distribution is specified to be added to the account. Dividend files should be updated frequently unless they aren't used to build the dividend files for all stocks in your Stock Dictionary. . The stock dividend data from CSI is encrypted by default. To use this feature you must contact CSI and make arrangements to have this data unencrypted for your subscription account. This is an additional cost service from them. Alternately you can build your own dividend files manually from another source for each stock and place in the Dividend Folder. The format is Date,Dividend and you can see example files in the sample data that ships with the product. See Note Reference: 2 Default: C:\TradingBlox\Data\Dividends
Forex Data: ^Top	All Forex File data is stored in the Data Folder's Forex subfolder. See Note Reference: 2 Default: C:\TradingBlox\Data\Forex
Dictionaries: ^Top	All Data Dictionaries are stored in the Data folder's Dictionaries subfolder. See Note Reference: 2 Default: C:\TradingBlox\Data\Dictionaries
Orders: ^Top	This is the location of new and archived Order Files created by the Generating orders . See Note Reference: 2 Default: C:\TradingBlox\Data\Futures

Data Folder Name:	Default Folder Path Locations:
Change Folder: ^Top	When CSI Unfair Advantage installs its software , it will offer this data folder location: C:\UAI When Unfair Advantage is installed in a different folder location, this setting value must be changed so that Dictionaries have a reference information source.

Data Options:	Description:
Number of Extra Data Fields: ^Top	Extra Data Field values are 0 to the maximum value of 8. Extra Data Fields provide access to information when the data file has more columns than the regular data files are required to provide. When your data files have extra columns of data, enter the number, up to a value of eight, of the "Extra Data Cols/Fields" in your data. When your doesn't have extra data columns, enter a zero in this field to save memory space. All extra columns/field must be ASCII text formatted information: Futures D, 0, H, L, C, V, 0I, DM, UC, E1, E2, E3, E4, E5, E6, E7, E8 Stocks and Forex D, 0, H, L, C, V, UC, E1, E2, E3, E4, E5, E6, E7, E8. Understanding the format of the data is important. If you aren't familiar with the data requirement yet, please read the section on Historical Data Setup before making any changes from the default settings. See Note Reference: 1 Default: 0
Years of Priming Data: ^Top	Years of Priming Data Enter the number of years of priming data required. The default priming period is 5. This is the number of years of data that will be loaded, if available, prior to the user entered start date of the test. This data is generally used to prime indicators prior to the test start so the test can start right on the start date. Recommended to keep this at 5 years unless you are trading lots of stocks and need the memory. To conserve memory, set this as low as your indicator priming requires, or zero if you don't mind your trading starting later than your test start date. Instrument Priming information is available in the Builder's Help file. It is also available when the Builder's Help file is installed. To locate the topic in the Builder's Help, the information is available in the Instrument Object Accessing Instruments Topics.

Data Options:	Description:
	Default: 5 Years
Volume Multiplier: ^Top	This setting determines how Stock-type instrument Volume values are treated. It also determines how the selected data type methods are applied when the software starts. If your stock-class data reports its Volume values as 1,000 instead of 100,000 use a multiplier of 100. Futures volume, and intraday data volume information will not be changed by this setting. Default: 100
Load Volume: ^Top	Check to load the volume in your data file into memory. Un-check this option if you don't use volume for your testing, and would like to save memory when loading a large number stock data files. Be sure to turn off any volume filters if you don't load the volume. Default: Enabled
Load Unadjusted Close: ^Top	UnadjustedClose data is recommended to be included in the files of Futures and Stock data files. By default this setting is enabled. If you know that the files you will use are not back adjusted for Futures contract roll to a forward contract, and the Stock data files are not adjusted for stock splits or consolidations, it might not be necessary to have this data field. Under these conditions, this setting can be left unchecked, but the results should be audited to ensure the testing results are correct.
Process Daily Bars: ^Top	This option when checked will create daily bars from intraday bars. Use these properties in scripting for price field access: <pre>instrument.dayOpen[] instrument.dayHigh[] instrument.dayLow[] instrument.dayClose[]</pre> The property counts the number of historic days available at any time, and should be checked before doing a lookback on this series. <pre>instrument.dayIndex</pre> See Note Reference: 1 Default: Disabled
Process Weekly Bars: ^Top	This option when checked will create weekly bars from the daily bars. Use these properties in scripting for price field access: <pre>instrument.weekOpen[] instrument.weekHigh[] instrument.weekLow[] instrument.weekClose[]</pre>

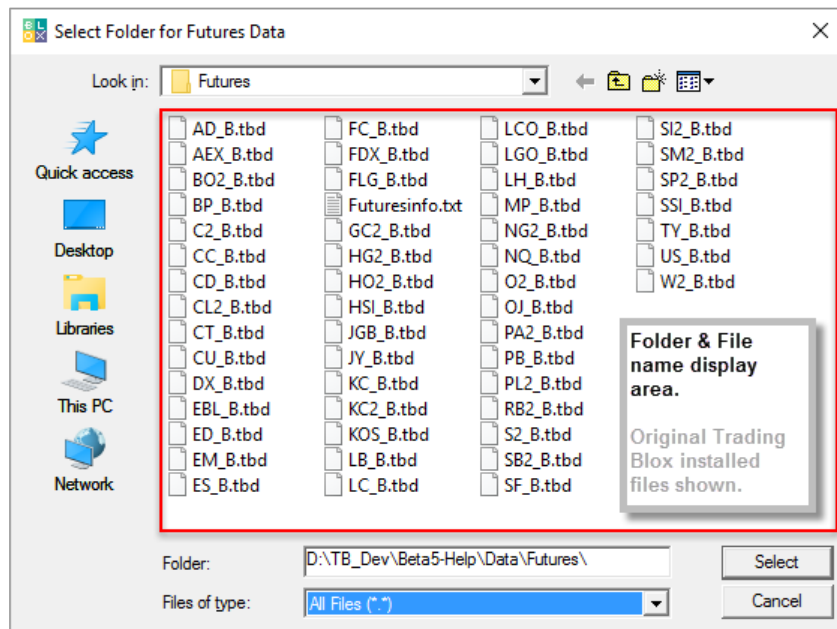
Data Options:	Description:
	This property counts the number of historic weeks available at any time, and should be checked before doing a lookback on this series. <code>instrument.weekIndex</code> See Note Reference: 1 Default: <code>Disabled</code>
Process Monthly Bars: ^Top	This option when checked will create monthly bars from the daily bars. Can be used in scripting using the Use these properties in scripting for price field access: <code>instrument.monthOpen[]</code> <code>instrument.monthHigh[]</code> <code>instrument.monthLow[]</code> <code>instrument.monthClose[]</code> This property counts the number of historic months available at any time, and should be checked before doing a lookback on this series. <code>instrument.monthIndex</code> See Note Reference: 1 Default: <code>Disabled</code>

Note References:	Information:
1 ^Top	To conserve memory when doing super large tests one or more of these option can be left unchecked so as to not use memory that isn't required. For example, the process daily, weekly, and monthly check-boxes should be unchecked if the test does not require the use of these properties. They additional data periods aren't required when the data in the file, which will always be available, is the only time period data required. For example, if the data file contains daily records that data will always be available. Meaning there is no need to enable Process Daily Bars. It also means that a suite test that doesn't need to know about weekly data when it will only reference the date and or time period data in the data file, there is no need to take up memory space with weekly or monthly information. Each data creation option will consume space This is also true for of the "Load Volume" option. When a system does not reference Volume information, leave this option unchecked when volume information is not needed by the system.
2 ^Top	Pressing a "Change Folder" button will bring up a window to select the location. To select a folder you must open and go "into" it, not just click and

Note References:	Information:
	highlight it. You can verify the folder that will be selected using the Folder Setting section of the dialog. If the data folder locations are changed, Trading Blox will use the folder location entered into each of the data type locations to access that instrument type. When updating data from a data subscription service that will create their files that match the instrument type formats, you can use the default data locations as the destination for the data files from that data service. If the data and dictionary locations are not changed from the default, there is no need to change the folder path settings from their default values. This preference section determine the location folders for the data types normally used for testing by Trading Blox.

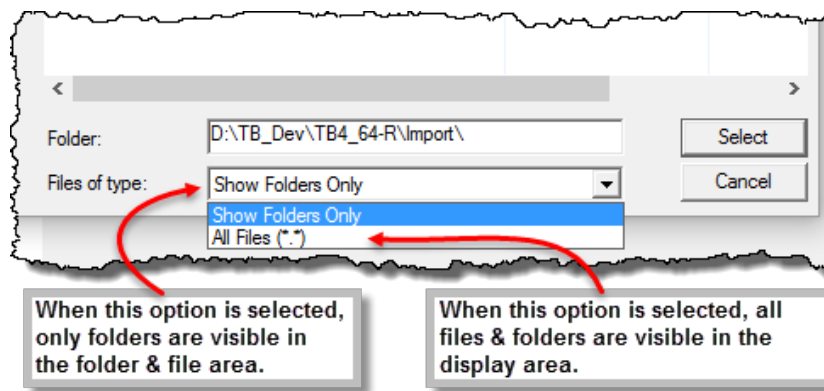
Edit Settings Example: [^Top](#)

By using a Window folder dialog, the accuracy of the data location is assured:



Windows Path & File Access Dialog

By default, the dialog will open with Windows "Show Folders Only" option displayed. If you are unsure if the files in the folder you are selecting is the correct location, change the dialog search method to "All Files (*.*)" option so you can view the files in the open folder location.



Windows File Type Display Selection Options

[^Top](#)**Links:****See Also:**

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 5011

Section 2 – General

General Reporting Information:

There are four section of information in this topic. Each section provides information that can change how reporting is performed.

[General Reporting Settings](#)

[Test Reporting Types](#)

[Test Reporting Sections](#)

[Reducing Optimization Time](#)

Report Settings:

Trading Blox preferences for all reporting allows these options for determining the type and information scope of a simulation test. Details of each setting are defined in the table below the image. To see a definition, click on the setting, or scroll down to the table to read them all:

General

Reporting Folder

Change Folder Results\

Reporting Options

8 Large Test Reporting Threshold ☐ Play Sound when test completes

90 Days of results/orders to keep ☒ Graph Closed Equity

1 Summary Chart Height Multiplier ☒ Add Non Trading Instruments to List

1 Summary Chart Width Multiplier

Application Options

USD System Wide Base Currency \$ Base Currency Symbol

30 Days of backups to keep ☒ Include Folder Path in Title Bar

☐ Use System Component Tabs

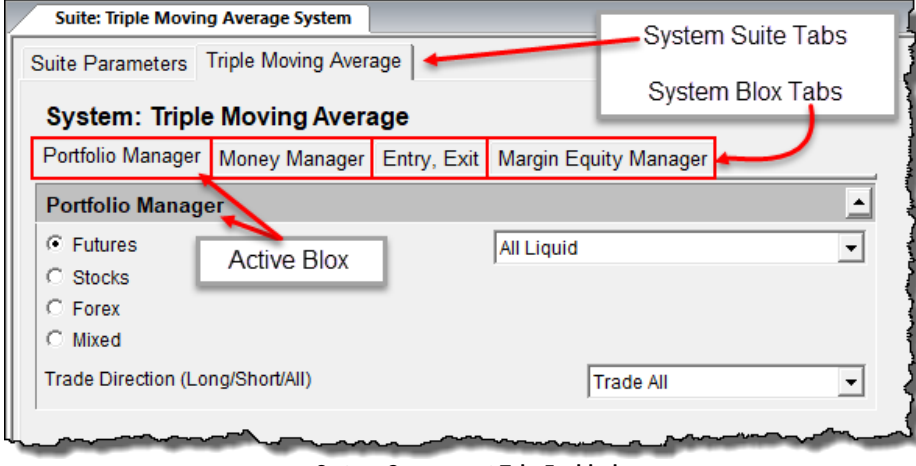
OK Cancel

Report settings for determining which report is generated, and other reporting options.

Reporting Folders: ^Top	Description:
Change Folder: ^Top	Determines the destination folder for the stored HTML reports generated when Trading Blox runs any of the three types of historical tests .

Reporting: ^Top	Description:
Large Test Reporting Threshold: ^Top	Sets the number of distinct parameter step combinations (or runs) which determines the cutoff between a Small Test and a Large Test (see the next section Report Content Specification for more details). In the above example, tests with 8 or less distinct parameter combinations will use the settings for Small Tests, test with 9 or more distinct parameter combinations will use the Large Test settings.
Days of Results And Orders To Keep: ^Top	Set the number of days of historical results and orders to keep in the Results and Orders folder. Defaults is to keep 90 days. If you set this value to 30, Trading Blox will delete any results files and order files that are more than 30 days old. Old report removal performed each time Trading Blox is started.
Play Sound When Test Completes: ^Top	Turns the test completion sound on or off.
Graph Closed Equity: ^Top	When checked, the closed equity line will be graphed along with the open equity in the summary results graph. If left unchecked, the closed equity will not be graphed.
Add Non-Trading Instruments to List: ^Top	When checked, all the instruments that did not trade for the duration of the test will have an entry in the trade log. The entry and exit dates and other information will be zero. This is useful so the chart is visible and all the indicators can be reviewed to understand why the market did not trade. When this is unchecked, only markets that have a trade will show in the trade list on the trade chart.
Summary Chart Height Multiplier: ^Top	When set to 1 the summary performance charts on the summary report will be the normal height. When set to 2 they will be twice as high.
Summary Chart Width Multiplier: ^Top	When set to 1 the summary performance charts on the summary report will be the normal width. When set to 2 they will be twice as wide.

Application Options: ^Top	Description:
Base: ^Top	Enter the symbol of the system wide base currency to use. The default is USD. This is the currency that all market results will be converted back into.

Application Options: ^Top	Description:
	You must have a forex pair in the Forex Data Folder to convert back from any foreign futures or stocks, back into the system wide base currency.
Base Currency Symbol: ^Top	Base Currency symbol supported are those listed in the Forex Dictionary. ???
Days of Backups To Keep: ^Top	Backups are created on first startup each day of selected important files such as blox and systems, zipped, and placed in the Backups folder. The backup process uses the user controlled backups.bat file which can be customized to include any data that should be backed up on a regular basis. This setting controls how many backups are retained in the Backups folder.
Include Folder Path in Title Bar: ^Top	When checked, the folder path of the installation folder will show in the Title bar. This is useful when running multiple installations of Trading Blox at the same time, so the installation location is apparent.  Trading Installed folder name displayed in Title Bar.
Use System Component Tabs: ^Top	When this option is enabled, the tabs for all the blox components in the displayed system will appear below the Suite's System tabs. When this option is disabled, it will display all the Suite's System Parameters in a vertical list that can scroll when necessary. Systems and System Tabs are sticky when changing suites and systems in the GUI When enabled the System Component Tabs will display as shown in this image:  System Component Tabs Enabled Tab and Section names can overlap to combine parameters into sections and tabs.

Reporting General Changes: ^Top	Description:
Show Results for Order Generation:	Show results for order generation is now fixed to always be available when orders are generated.
Measure Of Goodness Index: ^Top Moved to Simulation Parameters screen section.	The previous Preference setting for Measure Of Goodness Index is now named Multi-Parameter Goodness and it is located in the Simulation Parameters section of the Global Suite Parameters .

Test Reporting: [Top](#)

Generally, traders are interested in less detail when they run tests parameters are stepped. Simulation tests run with many different parameter combinations are focused on finding a general area of values in which to discover the best values in which to tune their system's settings. For the most part, they are generally concerned with summary information like CAGR%, MAR Ratio, Max Draw-down and a few other summary test values.

Trading Blox let's you specify which reports get included for different size tests. In this [Reporting General](#) preference section Trading Blox lets you determine what is included the various types of test results.

Reporting Types: ^Top	Description:
Trading Blox Preferences Reporting General section allow three different setting levels of reporting. Here are test reporting setting groups users find most useful:	
Single Test Reporting	Settings for a Single Test Report most often has all the available reporting preference settings enabled. It is a single test step simulation where none of parameters in the suite change value during the entire test run.
Small Simulation Reporting	Small Simulation Reporting tests only have a small number of parameters step test. Usually these numerical steps are limited to a small range of values. Small Simulation Reports need less information than a Single Test Report, but usually need more information than a Large Simulation Report. Small Simulation Reporting will automatically go into effect when a suite simulation parameter stepping exceeds the test count entered for Large Test Reporting Threshold setting shown in the General Reporting preference section. Default value limit for Small Simulation Reporting is

Reporting Types: ^Top	Description:
	"8" test steps, which can be changed.
Large Simulation Reporting	Tests that have many parameters stepping, or the parameters have a large range of values to test, will create a large number of performance test record results during a simulation. These large test simulation are more informative when they provide more overall summary information instead of a lot details.

Reporting Sections: [^Top](#)

In each three different types of simulation testing there are four sections where settings can be enabled or disabled. Single and Small step test usually will enable the report to contain more information than the Large step count report. Settings in each of the different reporting sections there are four groups of settings that control what is reported.

Note:

At startup of Trading Blox the default values assigned to the software's internal parameters can be changed. Changing them is a simple process that is explained once the [Trading Blox Parameter Settings](#) (link) topic is displayed.

ReportSettingTypes: ^Top	Description:
Graphs	Information and examples about the many graphs displayed on a test report.
Monte Carlo Simulations	Monte Carlo simulations provide a repetitive randomization method to determine the robustness of a system of a system and how that system would have performed had the many test sequences had been changed many thousands of times. In the link above, the seed value for the randomization is a default value that can be changed.
Text Tables	These text base reports show the summary information of the trading period where the instruments and system performance are summarized.
Trade and Equity Logs	This section shows the types and names of the logs that are part of a simulation test. These logs allow a user to validate and see in detail how the performance progressed through the testing period.
Filter Results	Filter Results is only displayed in the Large Simulation Reporting section. This section can use the three main statistical calculations to reduce the information displayed at the end of parameter test.

Links:

Links:**See Also:**

Edit Time: 10/29/2020 2:03:32 PM

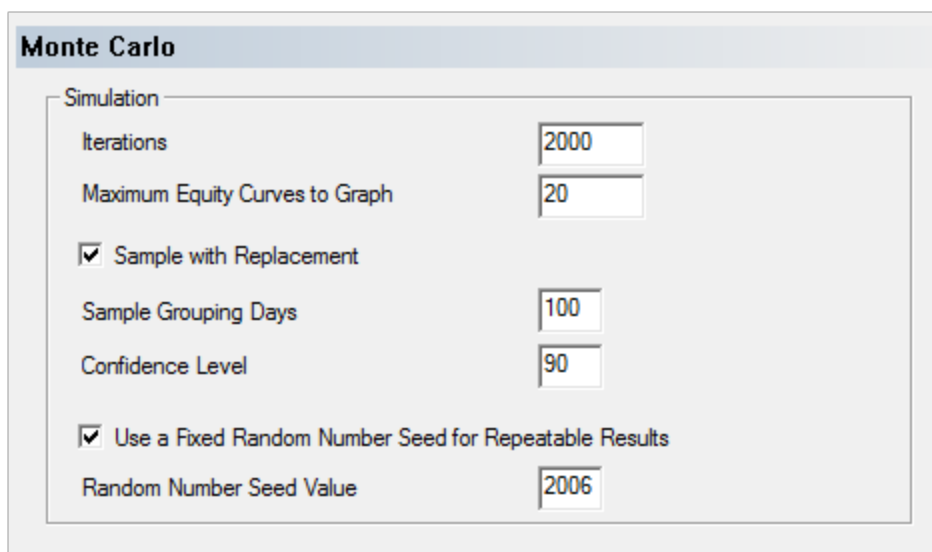
Topic ID#: 5012

Section 3 – Monte Carlo

Trading Blox [Monte Carlo Simulation Results](#) create a large group population samples on which to estimate the probability of the simulation test results.

The settings in this Monte Carlo preferences section determine how Trading Blox the Monte Carlo data randomizing loops generate statistical probabilities from the original population of suite simulation results. The primary test results are randomly re-sampled the number of times listed in the Monte Carlo's Iteration setting field. An optional Sample and Replacement option allows for additional variation results. Maximum Equity Curves to Graph provides the backdrop upon which the primary test result are overlaid to provide a visual reference of how the original results are positioned relative to the large Monte Carlo group of results.

There are seven Monte Carlo Simulation controls that influence Monte Carlo simulation results:



Monte Carlo

Simulation

Iterations: 2000

Maximum Equity Curves to Graph: 20

☒ Sample with Replacement

Sample Grouping Days: 100

Confidence Level: 90

☒ Use a Fixed Random Number Seed for Repeatable Results

Random Number Seed Value: 2006

Preference Setting:	Description:
Iterations: ^Top	This controls the total number of separate alternate equity curves which are used to generate the reports. Bigger is better but bigger takes longer. We recommend leaving these set to 2000 for a 10 year test.
Maximum Equity Curves to Graph: ^Top	This controls how many equity curves out of the total number of iterations will be graphed on the Monte Carlo Equity Curve Graph. You can set this to the same as iterations but this might result in it being impossible to see any distinct equity curves since they will all write over each other.
Sample with Replacement: ^Top	This controls how the alternate equity curves are generated. If you check this it means that we will randomly select a portion of the equity curve and the same portion might be selected more than once. This setting will result in differing end points for the equity curve.

Preference Setting:	Description:
	If this setting is not checked it will result in a simple reordering of the equity curve since each section of the initial equity curve will only be selected once and only once.
Sample Grouping Days: ^Top	This controls how many consecutive days at a time are selected from the initial equity curve. A setting greater than one will preserve some of the serial correlation in the initial equity curve.
Confidence Level: ^Top	This setting controls where the confidence lines are placed the graphs. It is a percentage which can range from 1 to 100. It also controls the value for the Confidence Level report. For example, if you set this to 85, we will report the MAR, CAGR%, Sharpe Ratio, R-Squared value, maximum drawdowns and length at the 85th percentile. In other words that value where 85% of the simulations had worse performance. This is fairly obvious from the graphs and is easier to understand when you look at it than by reading this.
Use a Fixed Random Number Seed for Repeatable Results: ^Top	When this is checked, then the seed value of the next field will be used to seed the random number generator. Effectively, this means that the sequence will repeat each time. When left unchecked a new random seed will start the random number sequence which means that each sequence will be different. Use this field setting to determine if the number of iterations is sufficient. For smaller numbers of iterations, less than a few thousand, you will get different graphs and confidence level values when there is a different set of random numbers. If this happens, it is a sign the number of iterations is not sufficient.
Random Number Seed Value: ^Top	The value to use to start the random number sequence. This setting is ignored if the above is not checked.

Links:**See Also:**

Section 4 – Single Test Reporting

Most Suite testing results will use the information selected in the setting you allow in this section of Preferences.

Single Test Reporting

Graphical Reporting

- ☒ Log Equity Graph
- ☒ Linear Equity Drawdown Graph
- ☒ Custom Graph
- ☒ Monthly Returns Graph
- ☒ Monthly Returns Distribution Graph
- ☒ Total Risk Profile Graph
- ☒ R-Multiple Distribution Graphs
- ☒ R-Multiple Profit Contribution Graphs

Text Reporting

- ☒ Yearly Performance Summary
- ☒ Trading Performance and Statistics
- ☒ Instrument Performance Summary
- ☒ System Parameter Settings

Trade and Equity Logs

- ☒ Log Trades
- ☒ Log Filtered Trades
- ☒ Log Daily Equity
- ☒ Log Monthly Equity and Performance

Monte Carlo Reporting

- ☒ Equity Curve Graph
- ☒ Returns Graph
- ☒ Sharpe Graph
- ☒ MAR Graph
- ☒ R Squared Graph
- ☒ Drawdown Graph
- ☒ Drawdown Length Graph
- ☒ Confidence Level Report

Single Test Reporting Options

In each three different types of simulation testing there are four sections where settings can be enabled or disabled. Single and Small step test usually will enable the report to contain more information than the Large step count report. Settings in each of the different reporting sections there are four groups of settings that control what is reported.

Note:

At startup of Trading Blox the default values assigned to the software's internal parameters can be changed. Changing them is a simple process that is explained once the [Trading Blox Parameter Settings](#) (link) topic is displayed.

ReportSettingTypes: ^Top	Description:
Graphs	Information and examples about the many graphs displayed on a test report.
Monte Carlo Simulations	Monte Carlo simulations provide a repetitive randomization method to determine the robustness of a system of a system and how that system would have performed had the many test sequences had been changed many thousands of times. In the link above, the seed value for the

ReportSettingTypes: ^Top	Description:
	randomization is a default value that can be changed.
Text Tables	These text base reports show the summary information of the trading period where the instruments and system performance are summarized.
Trade and Equity Logs	This section shows the types and names of the logs that are part of a simulation test. These logs allow a user to validate and see in detail how the performance progressed through the testing period.
Filter Results	Filter Results is only displayed in the Large Simulation Reporting section. This section can use the three main statistical calculations to reduce the information displayed at the end of parameter test.

Links:**See Also:**

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 203

Section 5 – Small Simulation Reporting

This Small Simulation Reporting section is provided to provide more information than [Large Simulation Reporting](#), but less information from [Single Test Reporting](#) so the other two test report settings can be left unchanged when the needs of a test are different than how those other test would provide.

Small Test Reporting Defaults:

Small Simulation Reporting

Graphical Reporting

- ☒ Log Equity Graph
- ☒ Linear Equity Drawdown Graph
- ☒ Custom Graph
- ☒ Monthly Returns Graph
- ☒ Monthly Returns Distribution Graph
- ☐ Total Risk Profile Graph
- ☐ R-Multiple Distribution Graphs
- ☐ R-Multiple Profit Contribution Graphs
- ☒ Parameter Step Graph
- ☒ Multi-Parameter Graph

Text Reporting

- ☒ Yearly Performance Summary
- ☒ Trading Performance and Statistics
- ☐ Instrument Performance Summary
- ☐ System Parameter Settings

Trade and Equity Logs

- ☒ Log Trades
- ☐ Log Filtered Trades
- ☐ Log Daily Equity
- ☐ Log Monthly Equity and Performance

Monte Carlo Reporting

- ☐ Equity Curve Graph
- ☐ Returns Graph
- ☐ Sharpe Graph
- ☐ MAR Graph
- ☐ R Squared Graph
- ☐ Drawdown Graph
- ☐ Drawdown Length Graph
- ☐ Confidence Level Report

Small Simulation Reporting Options

In each three different types of simulation testing there are four sections where settings can be enabled or disabled. Single and Small step test usually will enable the report to contain more information than the Large step count report. Settings in each of the different reporting sections there are four groups of settings that control what is reported.

Note:

At startup of Trading Blox the default values assigned to the software's internal parameters can be changed. Changing them is a simple process that is explained once the [Trading Blox Parameter Settings](#) (link) topic is displayed.

ReportSettingTypes: ^Top	Description:
Graphs	Information and examples about the many graphs displayed on a test report.

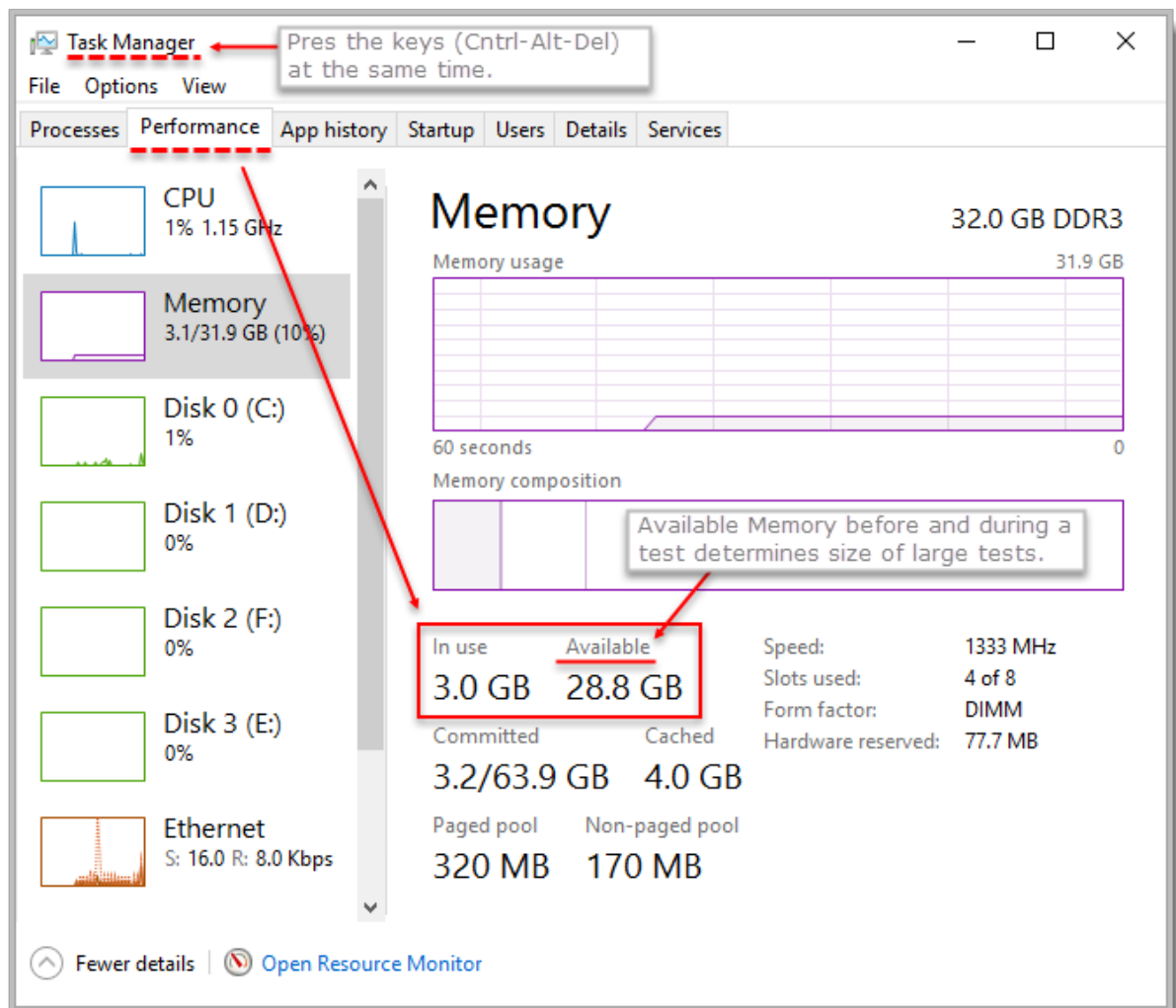
ReportSettingTypes: ^Top	Description:
Monte Carlo Simulations	Monte Carlo simulations provide a repetitive randomization method to determine the robustness of a system of a system and how that system would have performed had the many test sequences had been changed many thousands of times. In the link above, the seed value for the randomization is a default value that can be changed.
Text Tables	These text base reports show the summary information of the trading period where the instruments and system performance are summarized.
Trade and Equity Logs	This section shows the types and names of the logs that are part of a simulation test. These logs allow a user to validate and see in detail how the performance progressed through the testing period.
Filter Results	Filter Results is only displayed in the Large Simulation Reporting section. This section can use the three main statistical calculations to reduce the information displayed at the end of parameter test.

Links:**See Also:**

Section 6 – Large Simulation Reporting

Large Test Reporting is most often focused on how a simulation performs over a large range of parameter values. Often these large simulation will step many parameters in one or more systems. This scale of testing will generate a large number of test step increments where each step report will need more memory space. When a large number of symbols are included in a test, the loaded files will need a lot memory before the test starts.

Large simulation must consider if a large test will be able to run to completion in their computer's memory. To see how much memory is available, use the Windows Task Manager Performance information to see how much memory is available before testing, and during testing if there are any concerns:



Windows 10 Task Manager Performance Tab Information

Before executing a large simulation consider the above and check the settings in the Trading Blox [Data Options](#) group to ensure that the needed data options are enabled for a large test step series.

In some cases, where the size of a large simulation test will be near or beyond the limits of the available memory space, you might need to change some of the data option if memory is less than needed. When more memory is needed, a test won't be able to complete. Often experimenting with the step sizes of the parameters first can often be a solution to allow the test to complete, or at least run longer. Should the run longer, but not finish, reduce the steps more if that is possible, or reduce the amount or date range of the data and test again. All these work-around steps reduce the memory demands that usually allow a test to complete and generate a report.

[Reducing Optimization Time](#) explains how test where parameters stepped are enabled can take advantage of the built-in threads computers provide.

Default Large Simulation settings:

Large Simulation Reporting

Graphical Reporting

☒ Log Equity Graph ☒ Parameter Step Graph

☐ Linear Equity Drawdown Graph ☒ Multi-Parameter Graph

☐ Custom Graph

☐ Monthly Returns Graph

☐ Monthly Returns Distribution Graph

☐ Total Risk Profile Graph

☐ R-Multiple Distribution Graphs

☐ R-Multiple Profit Contribution Graphs

Results Filtering

☐ Filter Results

0 CAGR

0 MAR

0 Sharpe

Text Reporting

☐ Yearly Performance Summary

☐ Trading Performance and Statistics

☐ Instrument Performance Summary

☐ System Parameter Settings

Trade and Equity Logs

☐ Log Trades

☐ Log Filtered Trades

☐ Log Daily Equity

☐ Log Monthly Equity and Performance

Monte Carlo Reporting

☐ Equity Curve Graph

☐ Returns Graph

☐ Sharpe Graph

☐ MAR Graph

☐ R Squared Graph

☐ Drawdown Graph

☐ Drawdown Length Graph

☐ Confidence Level Report

Large Simulation Reporting Options

In each three different types of simulation testing there are four sections where settings can be enabled or disabled. Single and Small step test usually will enable the report to contain more information than the Large step count report. Settings in each of the different reporting sections there are four groups of settings that control what is reported.

Note:

At startup of Trading Blox the default values assigned to the software's internal parameters can be changed. Changing them is a simple process that is explained once the [Trading Blox Parameter Settings](#) (link) topic is displayed.

ReportSettingTypes: ^Top	Description:
Graphs	Information and examples about the many graphs displayed on a test report.
Monte Carlo Simulations	Monte Carlo simulations provide a repetitive randomization method to determine the robustness of a system of a system and how that system would have performed had the many test sequences had been changed many thousands of times. In the link above, the seed value for the randomization is a default value that can be changed.
Text Tables	These text base reports show the summary information of the trading period where the instruments and system performance are summarized.
Trade and Equity Logs	This section shows the types and names of the logs that are part of a simulation test. These logs allow a user to validate and see in detail how the performance progressed through the testing period.
Filter Results	Filter Results is only displayed in the Large Simulation Reporting section. This section can use the three main statistical calculations to reduce the information displayed at the end of parameter test.

Links:**See Also:**

Section 7 – Trade Chart Colors

Trading Blox lets you specify the colors for most of the elements on the trade chart using the Trade Chart Colors section of the Preferences Editor.

Default Chart Colors:

Trade Chart Colors

General Chart Colors

- Grid Color: [Color Swatch]
- Chart Background Color: [Color Swatch]
- Chart CrossHair Color: [Color Swatch]

Trade Marker Colors

- Long Trade Color: [Color Swatch]
- Short Trade Color: [Color Swatch]
- Entry Price Color: [Color Swatch]
- Stop Price Color: [Color Swatch]
- Exit Price Color: [Color Swatch]

Bar and Candlestick Colors

- Up Bar Color: [Color Swatch]
- Down Bar Color: [Color Swatch]
- Up Candle Color: [Color Swatch]
- Down Candle Color: [Color Swatch]

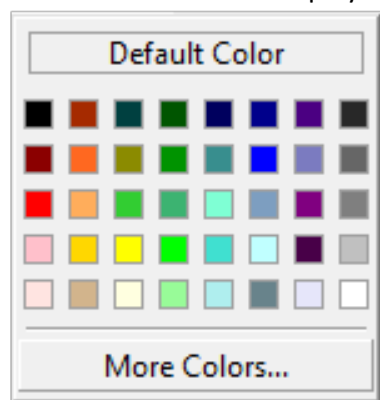
Custom Colors

	BGR Decimal	RGB Decimal
Custom Color 1 [Color Swatch]	0	0
Custom Color 2 [Color Swatch]	139	9109504
Custom Color 3 [Color Swatch]	16711680	255
Custom Color 4 [Color Swatch]	16777215	16777215

Color Coding Details

Editing Item Colors:

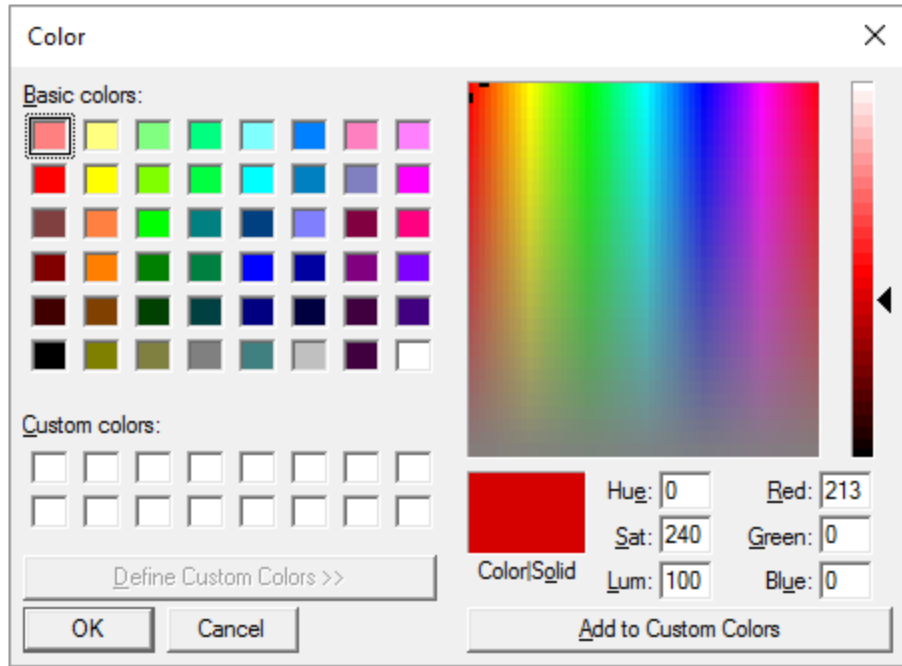
Click the chart item color display button so this image appears:



Color Palette

Selecting any of the colors shown and that color will be the item's color.

If a different color is needed, click the above image area "More Colors" section and this next color dialog will appear:



Color Palette Picker

This dialog allows a nearly infinite amount of colors to be used as a replacement color.

Links:
See Also:

Section 8 – Trade Details

After a test, the chart tab will display suite trade details in a column on the right side of the chart.

Scripts: Bollinger Breakout System	
Position:	
Direction	Long
Unit Number	0
Entry:	
Date	2010-09-30 Open
Quantity	16
Order	73.500
Fill	73.500
Stop	71.797
Risk	0.94%
USD	1,000.0000
Exit:	
Date	2010-11-18 Open
Quantity	16
Order	75.850
Fill	75.850
USD	1,000.0000
Profit:	
Points	2.350
Commission	\$0.0000
Profit	\$2,350.0000
Position	\$37,600.00
Percent %	1.30%
R-Multiple™	1.38
Equity:	
En Trading	\$2,889,915
Ex Closed	\$3,377,788
Ex Total	\$5,132,969
Miscellaneous:	
Drawdown %	30.40%
Days	49
Bars	35

Chart Trade Detail Window

Default Settings for Chart Trade Details:

This dialog is in the [Preferences](#) section.

Each item with a checkmark will display that item's test information in the above Trade Details column.

Trade Details

Position ☒ Long or Short ☒ Unit Number ☒ Quantity	Profit ☒ Profit Points ☒ Profit ☒ Position ☒ Profit Percent ☒ R-Multiple
Entry ☒ Date ☒ Order Price ☒ Fill Price ☒ Stop Price ☒ Risk %	Equity ☒ Entry Trading Equity ☒ Exit Closed Equity ☒ Exit Total Equity
Exit ☒ Date ☒ Order Price ☒ Fill Price	Miscellaneous ☒ Drawdown % ☒ Days

Check each item that you want to appear in the Trade Details Section .

Charting Trade Details Settings

Information displayed is for the trade detail selected:

Position Group:	Descriptions:
Long or Short	Trade's Long or Short order position.
Unit Number	Trade unit order number.
Quantity	Trade unit quantity.
Entry Group:	Descriptions:
Date	Position unit entry date.
Order Price	Unit number order price.
Fill Price	Unit number fill price.
Stop Price	Unit number stop price.
Risk %	Unit number risk percentage.
Exit Group:	Descriptions:
Date	Trade exit date.
Order Price	Trade exit order price.
Fill Price	Trade exit fill price.
Profit Group:	Descriptions:
Profit Points	Unit number profit points.
Profit	Unit number profit amount.

Profit Group:	Descriptions:
Position	Trade Long or Short position.
Profit Percent	Unit number profit percentage.
R-Multiple	R-Multiple size.
Equity Group:	Descriptions:
Entry Trading Equity	Trading equity at entry.
Exit Closed Equity	Closed equity at exit.
Exit Total Equity	Total equity at exit.
Miscellaneous Group:	Descriptions:
Drawdown %	Percentage of draw down.
Days	Unit number days in trade.

Links:**See Also:**

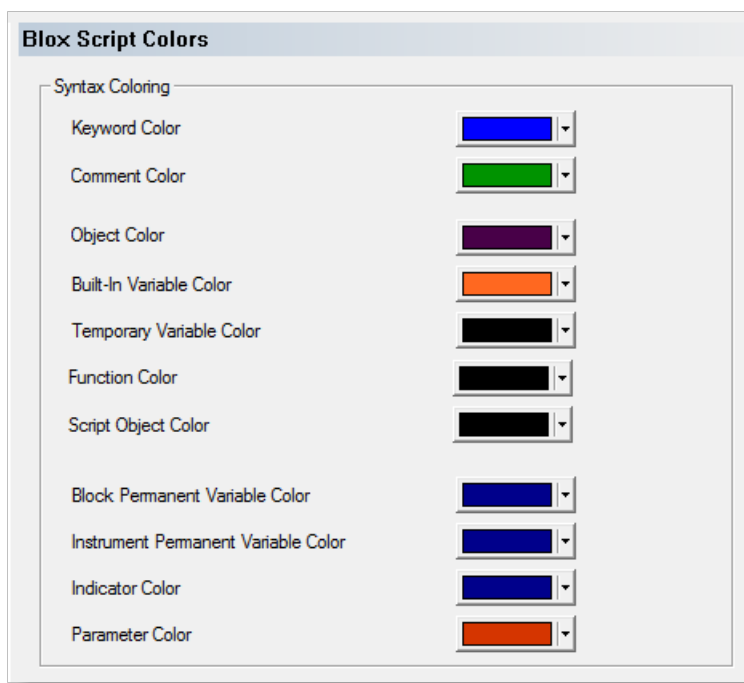
Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 5016

Section 9 – Blox Editor Colors

Trading Blox Builder edition can access these [Preference](#) for the Blox Editor.

Default Blox Editor Colors:



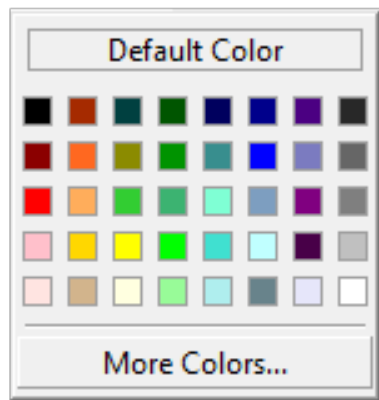
Blox Scripting Colors

Syntax Color Selection:	Syntax Description:
Keyword Color	Keywords. Example "If, Print, Integer"
Comment Color	Comments. Text following '.
Object Color	Objects. Examples: "Instrument, System"
Built-In Variable Color	Built in constants and preference properties. Examples: TRUE, NumberOfExtraDataFields, LicenseName
Temporary Variable Color	Local variables as defined in scripting. Example: "Variables: temp Type: Integer"
Function Color	Functions. Example: "Average(), IfThenElse()"
Script Object Color	"Script.Execute"
Block Permanent Variable Color	Block Permanent Variables as defined by user.
Instrument Permanent	Instrument Permanent Variables as defined by user

Syntax Color Selection:	Syntax Description:
Variable Color	
Indicator Color	Parameters as defined by user.
Parameter Color	Parameters as defined by user.

Editing Item Colors:

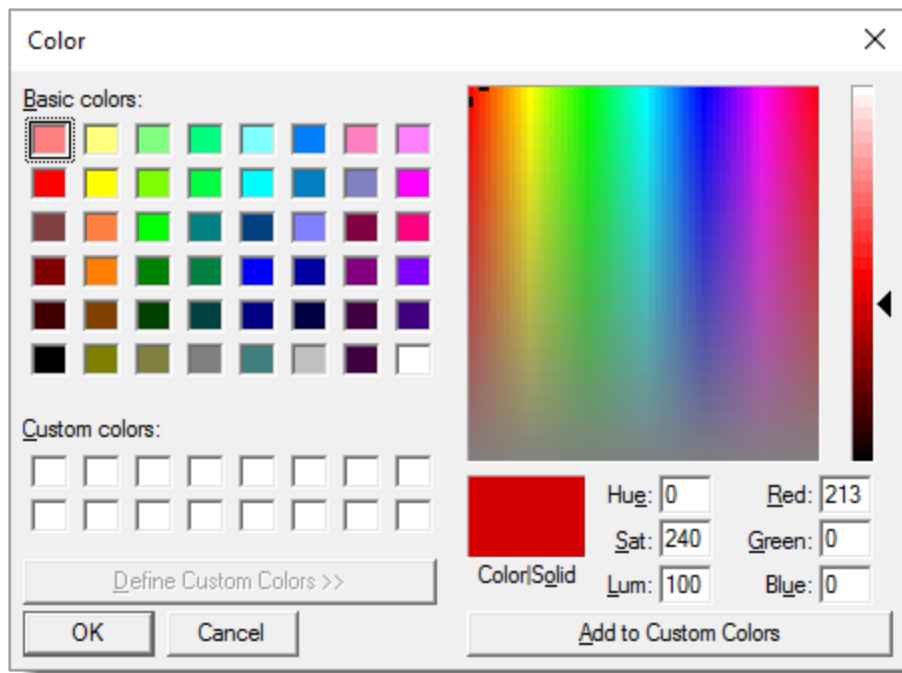
Click the chart item color display button so this image appears:



Color Palette

Selecting any of the colors shown and that color will be the item's color.

If a different color is needed, click the above image area "More Colors" section and this next color dialog will appear:



Color Palette Picker

This dialog allows a nearly infinite amount of colors to be used as a replacement color.

Links:**See Also:**

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 5017

Section 10 – Changing Default Settings

The following table of rarely used options can be accessed in the **tradingblox.ini** file. That file is stored in the same folder location as the main Trading Blox file.

To change one of more settings:

1. Exit from Trading Blox
2. Open the **TradingBlox.ini** in the Trading Blox file folder using a **text editor**.
3. Modify the setting
4. Save the file
5. Start Trading Blox again.

Setting	Description:
Dividend File Suffix	This setting is used as a naming convention for the dividend file suffix. Each dividend file uses the same symbol letters as the exchange symbol. An example is for the IBM stock data file where the dividend file for IBM will show: "IBM_Div.csv" The Dividend file names all include the market's symbol, plus the "_DIV" ahead of the file's format suffix: ".csv" which indicates the file is a Comma Delimited Text file that compatible with spreadsheets.
Graph File Suffix	This is ".png" suffix is the default suffix for all the summary graphs. Changing to ".gif" will produce a slightly larger but more portable file. Changing the graph suffix to ".jpg" produces a large file with more color accuracy with less data compression.
Process Weekends	This flag setting controls weekend processing. Normally if a weekend date is found anywhere in the data, then all weekends are included in the test loop. Setting Process Weekends to false will cause Trading Blox to skip any weekend dates from the data load that might exist in the data file. Indicators are computed based on data loaded, so the weekends would be excluded from the indicator calculations. Indicators are always based on available trading bars. If this flag is set to false, then weekends are excluded from the test loop.
Add Non Traded Instruments	This feature controls the process whereby instruments with no trades for a particular test are listed in the trade report with a zero trade. This zero trade is there so that you can access the chart and indicators, to explore why the instrument did not trade. In the case where there are 1000's of these in a large stock test, you might want to set this flag to false. This setting can also be controlled by a Blox that is added to a system.
Summary Report Test	This setting controls how many tests are allowed in the html summary output report before the results will be truncated.

Setting	Description:
Maximum	Too many test results in an html report file make the file so large that windows may not be able open the file. This value is usually set to 5,000 or less test results.
Risk Free Rate	This setting is used in some Sharpe statistical computations (i.e. Annual Sharpe but not Modified Sharpe).
Automatic Blox Script Parsing	This setting controls whether blox scripting will parse the script as each character is pressed. This is the default setting so that you can catch syntax errors immediately. If the blox script is so huge that this parsing process takes a few seconds, you can turn this automatic parsing off. The script is still parsed for syntax errors when the script is saved, on exit or by pressing control-s.
Blox Editor Font Size	The Trading Blox Base Editor the font size is controlled by this setting.
Blox Editor Font	This Blox Editor Font setting determines the font used in the Trading Blox Basic Editor .
Backup Archive Days	The option determines the number of days of backups to keep in the Trading Blox Backups folder.
Multi Parameter Chart	This option turns on/off the two multi parameter chart styles: contoured and 3D.
Global Heap Size	The Global Heap Size , sets the amount of memory the Trading Blox will allocate to a private memory heap on application startup. Setting to zero uses the default windows settings.
Use Jelly Fish Editor ??	The Use Jelly Fish Editor , tells the Blox Editor to save the scripts when opened into a .bas file in the Blox/JellyFish folder and open with Jelly Fish Pro. Script editing in the Blox Editor will be disabled.
Futures Data SSV ??	The Futures Data SSV , is not used. SSV is Space Separated Values. It was implemented to accommodate Tradestation data years ago, but is no longer used or enabled.
Integrity check	This feature can enable a strict rule feature during the loading of data and how it is checked. A more strict error setting will warn of as errors when the exchange or currency files are not available.
Use Population SD	This option controls the formula used by Trading Blox for Standard Deviation. By default this is TRUE , and it enables the Population Standard Deviation version for its calculations. If the Sample Calculation Version is required, set this option to FALSE .
Display Last Roll Date ??	The , controls the historic roll info on the positions section of the order generation report.

Setting	Description:
Memory Usage Threshold ??	The percentage of physical memory to use before raising an error is controlled by this setting. Default value is set to 90%. However, this can be set higher or lower percentage value as needed. A setting value of -1 will disable this ability.
Stock Margin Short Rate	The margin rate for short stock sales has a default value of 100% . Some brokers may require more or less margin for short sales.
Stock Margin Long Rate	The margin rate for long stock purchases has a default value of 100%

The **Forex Friday End Time** and **Forex Sunday Start Time** are used to set the start and end times of the forex trading week. Trading Blox will assume the market is closed during this time, and not load any new bars from **IB**, or generate any orders to send to **IB**.

The **Time Zone Adjust** setting is not currently used. Trading Blox now gets the time zone from the **IB** data, and also from the local computer, to determine if the market is open or closed based on the trading hours provided by **IB**. The hours provided by **IB** are local exchange time, and the assumption is that the TWS is displaying trading times and data, and providing to Trading Blox, in local time. When the market is determined to be closed, Trading Blox will not load data for bars, or generate orders.

The option to use all available trading hours, or only liquid trading hours, is another preference setting.

The following rarely used options are available in the tradingblox.ini file. To change, close Trading Blox, open the TradingBlox.ini file in Notepad, modify the setting, save, and then start Trading Blox.

The **Dividend File Suffix** is used as a naming convention for the dividend files, as the suffix to the symbol. The dividend file for IBM is IBM_Div.csv.

The **Graph File Suffix** is used as the suffix for all the summary graphs. Changing to .gif produces a slightly larger but more portable file, and changing to .jpg produces a large file with more defined colors, less compression.

The **Process Weekends** flag controls weekend processing. Normally if a weekend date is found anywhere in the data, then all weekends are included in the test loop. If this flag is set to false, then weekends are excluded from the test loop.

The **Add Non Traded Instruments** flag controls the process whereby instruments with no trades for a particular test are listed in the trade report with a zero trade. This zero trade is there so that you can access the chart and indicators, to explore why the instrument did not trade. But if you have 1000's of these in a large stock test, you might want to set this flag to false.

The **Summary Report Test Maximum** controls how many tests over which the html summary output will be truncated. Too many test results in an html file make the file so large that windows cannot open the file. So this number is usually set to 5000 or less.

The **Risk Free Rate** as used in some sharpe computations (Annual Sharpe but not Modified Sharpe).

The **Automatic Blox Script Parsing** flag controls whether blox scripting will parse the script as each character is pressed. This is the default setting so that you can catch syntax errors immediately. If the blox script is so huge that this parsing process takes a few seconds, you can turn this automatic parsing off. The script is still parsed for syntax errors when the script is saved, on exit or by pressing control-s.

The **Blox Editor Font Size** sets the font size in the Blox Editor

The **Blox Editor Font** sets the font used by the Blox Editor

The **Backup Archive Days** sets the number of days of backups to keep in the Backups folder.

The **Multi Parameter Chart** turns on/off the two multi parameter chart styles: contoured and 3D.

The **Global Heap Size** sets the amount of memory the Trading Blox will allocate to a private memory heap on application startup. Setting to zero uses the default windows settings.

The **Use Jelly Fish Editor** tells the Blox Editor to save the scripts when opened into a .bas file in the Blox/JellyFish folder and open with Jelly Fish Pro. Script editing in the Blox Editor will be disabled.

The **Futures Data SSV** is not used. SSV is Space Separated Values. It was implemented to accommodate Tradestation data years ago, but is no longer used or enabled.

The **Integrity check** enables more strict rules on data checking such as errors when the exchange or currency does not exist.

The **Use Population SD** sets the formula used by Trading Blox for Standard Deviation. By default this is true and uses Population Standard Deviation. If Sample is required, then set to false.

The **Display Last Roll Date** controls the historic roll info on the positions section of the order generation report.

The **Memory Usage Threshold** sets the percent of physical memory to use before raising an error. Default is 90, but this can be set higher or lower as needed. A setting of -1 will disable this check.

The **Stock Margin Short Rate** sets the margin rate for short stock sales. Default is 100%.

The **Stock Margin Long Rate** sets the margin rate for long stock purchases. Default is 100%, but some brokers require more or less margin for short sales.

Links:

[Trading Blox Default Settings](#), [Preferences Editor Menu](#), [Global Suite Parameters](#)

See Also:

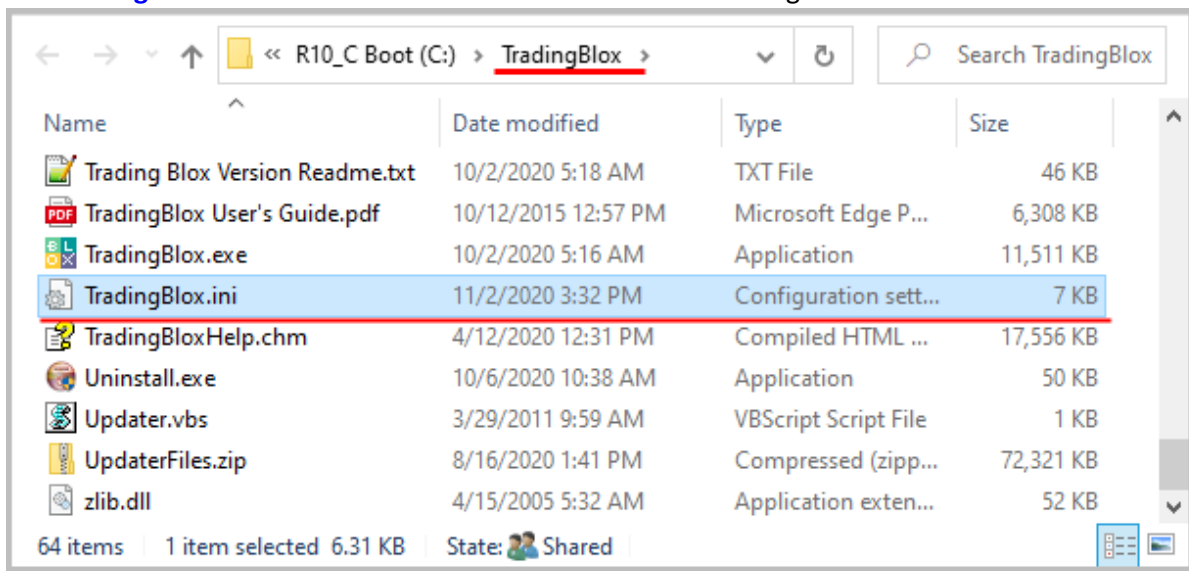
Edit Time: 3/6/2022 3:53:58 PM

Topic ID#: 256

Section 11 – Trading Blox Parameter Settings

Trading Blox Default Settings file: [TradingBlox.ini](#) is contained in the [Trading Blox Installation](#) folder.

The [TradingBlox.ini](#) file is located in the same location as Trading Blox:



Trading Blox.ini Default Setting File

Trading Blox version 5.4.5.3 contains fourteen groups in the [TradingBlox.ini](#) file. These setting values show the default values that are refreshed each time Trading Blox runs.

Changing the values in the "[TradingBlox.ini](#)" file is a simple process. To make a change to one or more parameter settings, for this list of statements:

1. Exit from Trading Blox
2. Open the [TradingBlox.ini](#) file in the Trading Blox installation folder location using a **text editor**.
3. Modify the parameter setting(s).
4. When the editing is done making changes, save the file to the Trading Blox installation folder.
5. Start Trading Blox again.

The [TradingBlox.ini](#) file can be loaded into any text editor. Microsoft Windows has a **NotePad.exe** application that is a text editor. There is a free text Editor named: [Notepad++](#) ([Link](#)). Both text editors provide a save feature as part of their application. Be sure to save your changes if you want to see the how the change will appear.

[TradingBlox Directories] ^Top
 Futures Data Directory=Data\Futures\
 Stock Data Directory=Data\Stocks\
 Forex Data Directory=Data\Forex\
 Dividend Data Directory=Data\Dividends\

Dictionary Directory=Data\Dictionaryes\
Orders Directory=Orders\
Results Directory=Results\
CSI UA Directory=C:\UA\

[Data] [^Top](#)

Integrity Check=**FALSE**
Num Extra Data Fields=0
Years Data=5
Load Volume=**TRUE**
Raise Negative Data Series=**FALSE**
Process Weekly Bars=**TRUE**
Process Monthly Bars=**FALSE**
Process Daily Bars=**FALSE**
Process Weekends=**TRUE**
Load Unadjusted Close=**TRUE**
Reverse Currency Conversion=**FALSE**
Base Currency ISO Code=USD
Base Currency **Sign**=\$
Stock Volume Multiplier=100
Global Heap Size=0
Post Data End Date Warnings=**TRUE**
Forex **Friday** End Time=2100
Forex **Sunday** Start Time=2200
Time Zone Adjust=0
IB Futures RTH=**FALSE**
IB Stocks RTH=**TRUE**
Dividend File Suffix=_Div.csv
Use IB Forex Bid Ask=**FALSE**
Find Stock Dictionary=**TRUE**

[Walk Forward] [^Top](#)

Optimization Size=5
Out Of Sample Size=1
Save Positions=**TRUE**

[General Reporting] [^Top](#)

Small **Test** Threshold=8
Goodness Index=3
Goodness **Min**=0.001
Goodness **Max**=5
Graph Closed Equity=**TRUE**
Play Report Sound=**FALSE**
Show **Order** Results=**TRUE**
Add Non Traded Instruments=**FALSE**
Summary Report **Test** Maximum=5000
Multi Parameter **Chart** 1=**TRUE**
Multi Parameter **Chart** 2=**TRUE**
Risk Free Rate=0.03
Use Population SD=**TRUE**
Display Last Roll Date=**TRUE**

Include Position Stop=**FALSE**
 Graph File Suffix=.png
Chart Width Multiplier=1
Chart Height Multiplier=1

[Monte Carlo] [^Top](#)

Iterations=2000
 Equity Graph Count=20
 Sample with Replacement=**TRUE**
 Grouping Days=100
 Confidence Level=90
 Use Fixed **Random** Seed=**TRUE**
Random Seed=2006

[Single Run Reporting] [^Top](#)

Log Trades=**TRUE**
Log Filtered Trades=**TRUE**
Log Daily Equity=**TRUE**
Log Monthly Equity=**TRUE**
 Graph **Log** Equity=**TRUE**
 Graph Linear Equity=**TRUE**
 Custom Graph=**TRUE**
 Graph Monthly Returns=**TRUE**
 Graph Monthly Distributions=**TRUE**
 Graph Total Risk Profile=**TRUE**
 Graph **System** R-Multiple Distribution=**TRUE**
 Graph **System** R-Multiple Contribution=**TRUE**
 Graph Multiple Parameters=**FALSE**
 Graph Parameter Steps=**FALSE**
 Report Yearly Performance=**TRUE**
 Report Trading Performance Statistics=**TRUE**
 Report **Instrument** Performance=**TRUE**
 Report **System** Parameter Settings=**TRUE**
 Graph Monte Carlo Equity Curve=**TRUE**
 Graph Monte Carlo Return=**TRUE**
 Graph Monte Carlo Sharpe=**TRUE**
 Graph Monte Carlo MAR=**TRUE**
 Graph Monte Carlo R Squared=**TRUE**
 Graph Monte Carlo Drawdowns=**TRUE**
 Graph Monte Carlo Drawdown Length=**TRUE**
 Report Monte Carlo Confidence Levels=**TRUE**
 Filter Results=**FALSE**
 Filter **CAGR** Threshold=0
 Filter MAR Threshold=0
 Filter Sharpe Threshold=0

[Small Test Reporting] [^Top](#)

Log Trades=**TRUE**
Log Filtered Trades=**FALSE**
Log Daily Equity=**FALSE**
Log Monthly Equity=**FALSE**

Graph **Log** Equity=**TRUE**
Graph Linear Equity=**TRUE**
Custom Graph=**TRUE**
Graph Monthly Returns=**TRUE**
Graph Monthly Distributions=**TRUE**
Graph Total Risk Profile=**FALSE**
Graph **System** R-Multiple Distribution=**FALSE**
Graph **System** R-Multiple Contribution=**FALSE**
Graph Multiple Parameters=**TRUE**
Graph Parameter Steps=**TRUE**
Report Yearly Performance=**TRUE**
Report Trading Performance Statistics=**TRUE**
Report **Instrument** Performance=**FALSE**
Report **System** Parameter Settings=**FALSE**
Graph Monte Carlo Equity Curve=**FALSE**
Graph Monte Carlo Return=**FALSE**
Graph Monte Carlo Sharpe=**FALSE**
Graph Monte Carlo MAR=**FALSE**
Graph Monte Carlo R Squared=**FALSE**
Graph Monte Carlo Drawdowns=**FALSE**
Graph Monte Carlo Drawdown Length=**FALSE**
Report Monte Carlo Confidence Levels=**FALSE**
Filter Results=**FALSE**
Filter **CAGR** Threshold=0
Filter MAR Threshold=0
Filter Sharpe Threshold=0

[Large Test Reporting] [^Top](#)

Log Trades=**FALSE**
Log Filtered Trades=**FALSE**
Log Daily Equity=**FALSE**
Log Monthly Equity=**FALSE**
Graph **Log** Equity=**TRUE**
Graph Linear Equity=**FALSE**
Custom Graph=**FALSE**
Graph Monthly Returns=**FALSE**
Graph Monthly Distributions=**FALSE**
Graph Total Risk Profile=**FALSE**
Graph **System** R-Multiple Distribution=**FALSE**
Graph **System** R-Multiple Contribution=**FALSE**
Graph Multiple Parameters=**TRUE**
Graph Parameter Steps=**TRUE**
Report Yearly Performance=**FALSE**
Report Trading Performance Statistics=**FALSE**
Report **Instrument** Performance=**FALSE**
Report **System** Parameter Settings=**FALSE**
Graph Monte Carlo Equity Curve=**FALSE**
Graph Monte Carlo Return=**FALSE**
Graph Monte Carlo Sharpe=**FALSE**
Graph Monte Carlo MAR=**FALSE**
Graph Monte Carlo R Squared=**FALSE**
Graph Monte Carlo Drawdowns=**FALSE**

Graph Monte Carlo Drawdown Length=**FALSE**
Report Monte Carlo Confidence Levels=**FALSE**
Filter Results=**TRUE**
Filter **CAGR** Threshold=0
Filter MAR Threshold=0
Filter Sharpe Threshold=0

[Trade Charts] [^Top](#)

Grid Color=16443110
Long Color=37632
Short Color=213
Entry Color=16764057
Stop Color=14527197
Exit Color=16751001
Up Bar Color=37632
Down Bar Color=213
Up Candle Color=37632
Down Candle Color=213
Chart Background Color=14745599
Chart CrossHair Color=12632256
Custom Color 1=0
Custom Color 2=139
Custom Color 3=16711680
Custom Color 4=16777215

[Trade Details] [^Top](#)

Entry Equity=**TRUE**
Closed Equity=**TRUE**
Core Equity=**FALSE**
Days=**TRUE**
Drawdown **Percent**=**TRUE**
Entry Date=**TRUE**
Entry Fill **Price**=**TRUE**
Entry **Order Price**=**TRUE**
Entry Stop **Price**=**TRUE**
Entry Risk=**TRUE**
Exit Date=**TRUE**
Exit Fill **Price**=**TRUE**
Exit **Order Price**=**TRUE**
Long or Short=**TRUE**
Profit Dollars=**TRUE**
Profit **Percent**=**TRUE**
Profit Points=**TRUE**
Profit Position Dollars=**TRUE**
Profit R-Multiple=**TRUE**
Quantity=**TRUE**
Total Equity=**TRUE**
Unit Number=**TRUE**

[Block Basic] [^Top](#)

Keyword Color=16711680

Function Color=0
Operator Color=0
Comment Color=37632
String Color=65535
Trading Object Color=4718664
Built-In Color=2124031
Temporary Color=0
Block Permanent Color=9109504
Instrument Permanent Color=9109504
Indicator Color=9109504
Parameter Color=13781
Include Default Scripts=**TRUE**
Automatic Blox **Script** Parsing=**TRUE**
Allow Priming Update Indicators=**TRUE**
Blox Editor Font Size=14
Blox Editor Font=Courier New
Use Jelly Fish Editor=**FALSE**
Blox Directory=
Parameter Tab Size=6

[Application Settings] [^Top](#)

Results Archive Days=20
Backup Archive Days=30
Memory Usage Threshold=90
Include Path In Title=**TRUE**
Listbox Color=15790320
Header Color=12632256
Title Color=0
Background Color=16187382
Broker Direct Trigger File=DataTrigger.txt
Run Silent=**FALSE**
System Edit Area Tabs=**FALSE**

[Backtesting] [^Top](#)

Stock Margin **Short** Rate=100
Stock Margin **Long** Rate=100
Use **Round** Lot=**TRUE**
Max Stepped Tests=0

[Interactive Brokers] [^Top](#)

Enable IB Emails=**FALSE**
Default IB Port=7496
Default IB Client=1
Synch IB Positions=**TRUE**
Weekly Restart Time=-1
Bypass Warning=**FALSE**
Position Synch **Percent**=0

[Data] [^Top](#)

Integrity Check=**FALSE**

Num Extra Data Fields=0
 Years Data=5
 Load Volume=TRUE
 Raise Negative Data Series=FALSE
 Process Weekly Bars=TRUE
 Process Monthly Bars=FALSE
 Process Daily Bars=FALSE
 Process Weekends=TRUE
 Load Unadjusted Close=TRUE
 Reverse Currency Conversion=FALSE
 Base Currency ISO Code=USD
 Base Currency Sign=\$
 Stock Volume Multiplier=100
 Global Heap Size=0
 Post Data End Date Warnings=TRUE
 Forex Friday End Time=2100
 Forex Sunday Start Time=2200
 Time Zone Adjust=0
 IB Futures RTH=FALSE
 IB Stocks RTH=TRUE
 Dividend File Suffix=_Div.csv
 Use IB Forex Bid Ask=FALSE
 Find Stock Dictionary=TRUE

[Walk Forward] [^Top](#)

Optimization Size=5
 Out Of Sample Size=1
 Save Positions=TRUE

[General Reporting] [^Top](#)

Small Test Threshold=8
 Goodness Index=3
 Goodness Min=0.001
 Goodness Max=5
 Graph Closed Equity=TRUE
 Play Report Sound=FALSE
 Show Order Results=TRUE
 Add Non Traded Instruments=FALSE
 Summary Report Test Maximum=5000
 Multi Parameter Chart 1=TRUE
 Multi Parameter Chart 2=TRUE
 Risk Free Rate=0.03
 Use Population SD=TRUE
 Display Last Roll Date=TRUE
 Include Position Stop=FALSE
 Graph File Suffix=.png
 Chart Width Multiplier=1
 Chart Height Multiplier=1

[Monte Carlo] [^Top](#)

Iterations=2000

Equity Graph Count=20
Sample with Replacement=TRUE
Grouping Days=100
Confidence Level=90
Use Fixed **Random** Seed=TRUE
Random Seed=2006

[Single Run Reporting] [^Top](#)

Log Trades=TRUE
Log Filtered Trades=TRUE
Log Daily Equity=TRUE
Log Monthly Equity=TRUE
Graph **Log** Equity=TRUE
Graph Linear Equity=TRUE
Custom Graph=TRUE
Graph Monthly Returns=TRUE
Graph Monthly Distributions=TRUE
Graph Total Risk Profile=TRUE
Graph **System** R-Multiple Distribution=TRUE
Graph **System** R-Multiple Contribution=TRUE
Graph Multiple Parameters=FALSE
Graph Parameter Steps=FALSE
Report Yearly Performance=TRUE
Report Trading Performance Statistics=TRUE
Report **Instrument** Performance=TRUE
Report **System** Parameter Settings=TRUE
Graph Monte Carlo Equity Curve=TRUE
Graph Monte Carlo Return=TRUE
Graph Monte Carlo Sharpe=TRUE
Graph Monte Carlo MAR=TRUE
Graph Monte Carlo R Squared=TRUE
Graph Monte Carlo Drawdowns=TRUE
Graph Monte Carlo Drawdown Length=TRUE
Report Monte Carlo Confidence Levels=TRUE
Filter Results=FALSE
Filter **CAGR** Threshold=0
Filter MAR Threshold=0
Filter Sharpe Threshold=0

[Small Test Reporting] [^Top](#)

Log Trades=TRUE
Log Filtered Trades=FALSE
Log Daily Equity=FALSE
Log Monthly Equity=FALSE
Graph **Log** Equity=TRUE
Graph Linear Equity=TRUE
Custom Graph=TRUE
Graph Monthly Returns=TRUE
Graph Monthly Distributions=TRUE
Graph Total Risk Profile=FALSE
Graph **System** R-Multiple Distribution=FALSE
Graph **System** R-Multiple Contribution=FALSE

Graph Multiple Parameters=**TRUE**
 Graph Parameter Steps=**TRUE**
 Report Yearly Performance=**TRUE**
 Report Trading Performance Statistics=**TRUE**
 Report **Instrument** Performance=**FALSE**
 Report **System** Parameter Settings=**FALSE**
 Graph Monte Carlo Equity Curve=**FALSE**
 Graph Monte Carlo Return=**FALSE**
 Graph Monte Carlo Sharpe=**FALSE**
 Graph Monte Carlo MAR=**FALSE**
 Graph Monte Carlo R Squared=**FALSE**
 Graph Monte Carlo Drawdowns=**FALSE**
 Graph Monte Carlo Drawdown Length=**FALSE**
 Report Monte Carlo Confidence Levels=**FALSE**
 Filter Results=**FALSE**
 Filter **CAGR** Threshold=0
 Filter MAR Threshold=0
 Filter Sharpe Threshold=0

[Large Test Reporting] [^Top](#)

Log Trades=**FALSE**
 Log Filtered Trades=**FALSE**
 Log Daily Equity=**FALSE**
 Log Monthly Equity=**FALSE**
 Graph **Log** Equity=**TRUE**
 Graph Linear Equity=**FALSE**
 Custom Graph=**FALSE**
 Graph Monthly Returns=**FALSE**
 Graph Monthly Distributions=**FALSE**
 Graph Total Risk Profile=**FALSE**
 Graph **System** R-Multiple Distribution=**FALSE**
 Graph **System** R-Multiple Contribution=**FALSE**
 Graph Multiple Parameters=**TRUE**
 Graph Parameter Steps=**TRUE**
 Report Yearly Performance=**FALSE**
 Report Trading Performance Statistics=**FALSE**
 Report **Instrument** Performance=**FALSE**
 Report **System** Parameter Settings=**FALSE**
 Graph Monte Carlo Equity Curve=**FALSE**
 Graph Monte Carlo Return=**FALSE**
 Graph Monte Carlo Sharpe=**FALSE**
 Graph Monte Carlo MAR=**FALSE**
 Graph Monte Carlo R Squared=**FALSE**
 Graph Monte Carlo Drawdowns=**FALSE**
 Graph Monte Carlo Drawdown Length=**FALSE**
 Report Monte Carlo Confidence Levels=**FALSE**
 Filter Results=**TRUE**
 Filter **CAGR** Threshold=0
 Filter MAR Threshold=0
 Filter Sharpe Threshold=0

[Trade Charts] [^Top](#)

Grid Color=16443110

Long Color=37632**Short** Color=213

Entry Color=16764057

Stop Color=14527197

Exit Color=16751001

Up Bar Color=37632

Down Bar Color=213

Up Candle Color=37632

Down Candle Color=213

Chart Background Color=14745599**Chart** CrossHair Color=12632256

Custom Color 1=0

Custom Color 2=139

Custom Color 3=16711680

Custom Color 4=16777215

[Trade Details] [^Top](#)Entry Equity=**TRUE**Closed Equity=**TRUE**Core Equity=**FALSE**Days=**TRUE**Drawdown **Percent**=**TRUE**Entry Date=**TRUE**Entry Fill **Price**=**TRUE**Entry **Order Price**=**TRUE**Entry Stop **Price**=**TRUE**Entry Risk=**TRUE**Exit Date=**TRUE**Exit Fill **Price**=**TRUE**Exit **Order Price**=**TRUE****Long or Short**=**TRUE**Profit Dollars=**TRUE**Profit **Percent**=**TRUE**Profit Points=**TRUE**Profit Position Dollars=**TRUE**Profit R-Multiple=**TRUE**Quantity=**TRUE**Total Equity=**TRUE**Unit Number=**TRUE**

[Block Basic] [^Top](#)

Keyword Color=16711680

Function Color=0

Operator Color=0

Comment Color=37632

String Color=65535

Trading Object Color=4718664

Built-In Color=2124031

Temporary Color=0

Block Permanent Color=9109504

Instrument Permanent Color=9109504
 Indicator Color=9109504
 Parameter Color=13781
 Include Default Scripts=**TRUE**
 Automatic Blox **Script** Parsing=**TRUE**
 Allow Priming Update Indicators=**TRUE**
 Blox Editor Font Size=14
 Blox Editor Font=Courier New
 Use Jelly Fish Editor=**FALSE**
 Blox Directory=
 Parameter Tab Size=6

[Application Settings] [^Top](#)

Results Archive Days=20
 Backup Archive Days=30
 Memory Usage Threshold=90
 Include Path In Title=**TRUE**
 Listbox Color=15790320
 Header Color=12632256
 Title Color=0
 Background Color=16187382
Broker Direct Trigger File=DataTrigger.txt
 Run Silent=**FALSE**
System Edit Area Tabs=**FALSE**

[Backtesting] [^Top](#)

Stock Margin **Short** Rate=100
 Stock Margin **Long** Rate=100
 Use **Round** Lot=**TRUE**
Max Stepped Tests=0

[Interactive Brokers] [^Top](#)

Enable IB Emails=**FALSE**
 Default IB Port=7496
 Default IB Client=1
 Synch IB Positions=**TRUE**
 Weekly Restart Time=-1
 Bypass Warning=**FALSE**
 Position Synch **Percent**=0

Links:

[Changing Default Settings](#), [Preferences Editor Menu](#), [Global Suite Parameters](#)

See Also:

Edit Time: 3/6/2022 4:06:53 PM

Topic ID#: 252

System Editor Information

Part

VI

Part 6 – System Editor Information

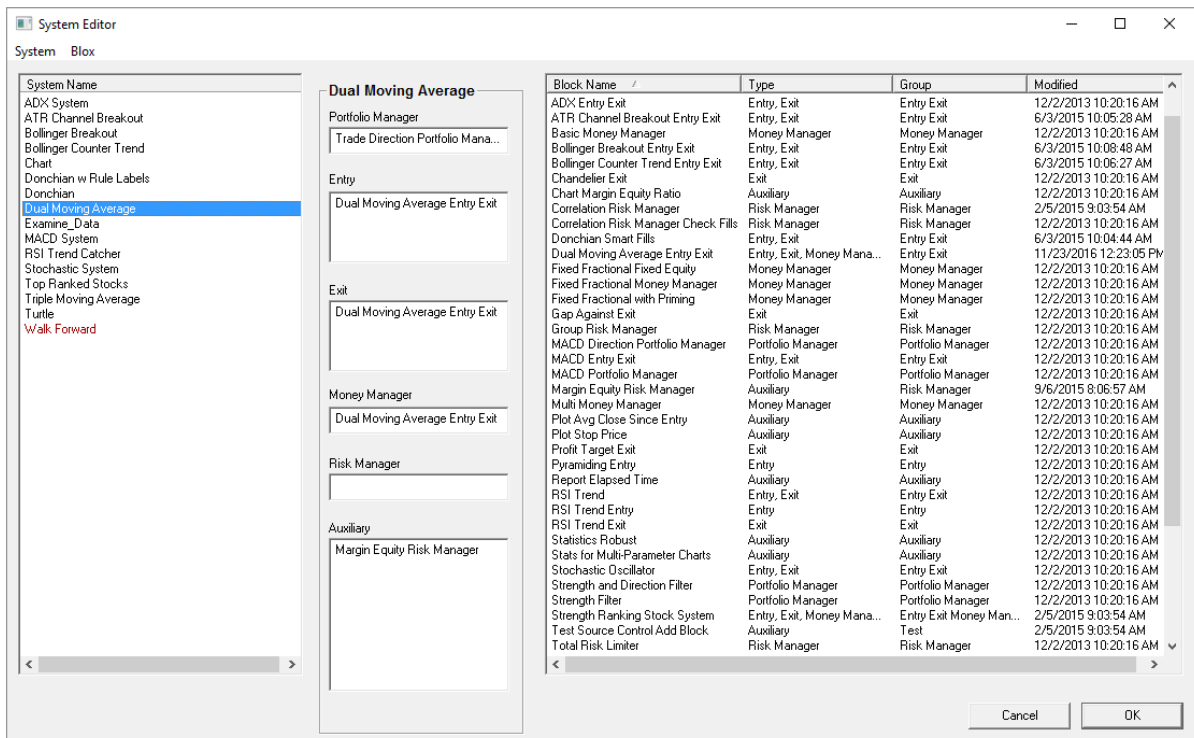
The Trading Blox System Editor is where the blox are selected for each listed system. Blox can be added, changed or removed from a system's using the System Editors capabilities. Image below shows the System Editor screen main sections. Each section is an active link to more information about that section. When a change is made in the System Editor, it will not be saved until the "OK" button is clicked and the editor is closed. To ignore a system change, click on the "Cancel" button and nothing will be changed.

NOTE:

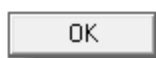
All Trading Blox Systems intended to generate signals must have a Entry Script, Exit Script and a Unit Sizing Script section. Entry Scripts are the primary location to generate orders. Exit Scripts are the primary location to remove positions. Unit Sizing Scripts are the required location for handling the quantity assigned to an Entry Order. Some orders can be generated in other scripts, and some positions can be removed from other scripts, but these are advanced methods that should not be considered while you are learning the capabilities of Trading Blox.

Creating or changing a system structure is made easy by accessing the menu options in the [System Editing](#) area. Adding, changing or removing blox from a system list is just as easy using the [System Blox Menu](#) options. Creating a new system is easy using the [System Editing](#) options.

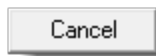
To access any of the three groups of the System Editor, click on any of the areas in this image to be linked to that ability area:



Main System Editor

System Updating:

All system changes are temporary until the "**OK**" button is pressed.



Pressing the "**Cancel**" button after making any changes to one or many systems will allow all the changes to be ignored when the System Editor is closed.

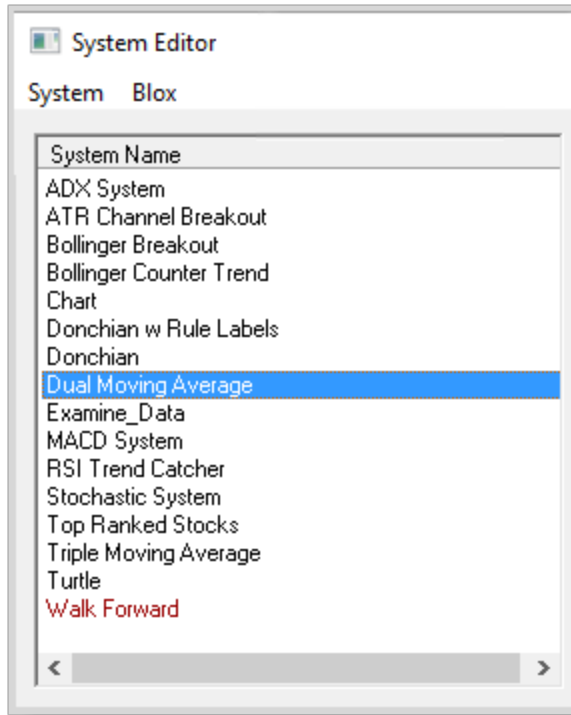
Links:**See Also:**

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 217

Section 1 – System Editing

%APPLICATION_NAME%> System Editor area is where systems are created, changed, or removed. A system name can be almost any name when the [Naming Rules](#) are understood and followed.



System Menus and Available System Names

Any system name in the list of systems can be changed, removed, exported, deleted or printed.

Access to the list of options is available when the [System Menu](#) option is clicked at the top of the System Editor, or when a system is selected and the system name list is right-clicked.

When a system name is selected its list of blox modules can be added, changed, or removed using the [System Blox Options](#) in the System Editor.

Naming Rules:

Suite, System and Blox module names can include the following keyboard characters:

- All letters letters "A - Z" and "a - z".
- All numbers from "0 to 9".
- The keyboard space " ", hyphen "-", and underscore "_" symbols.

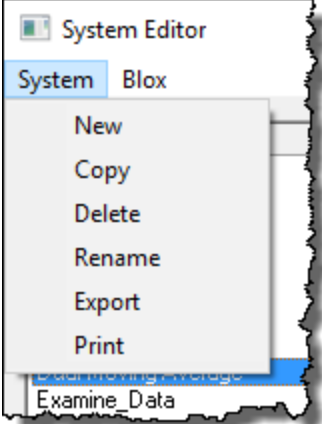
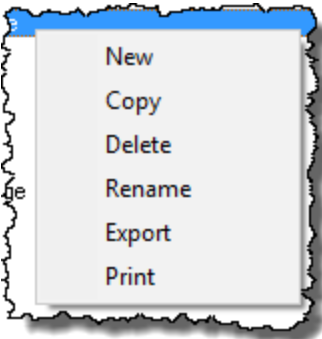
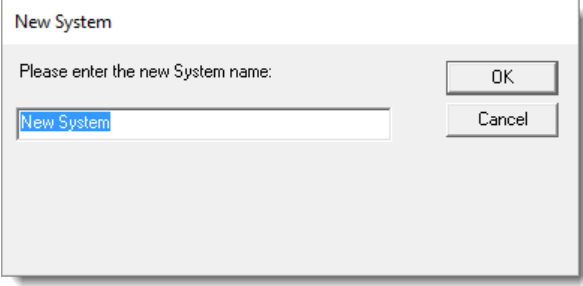
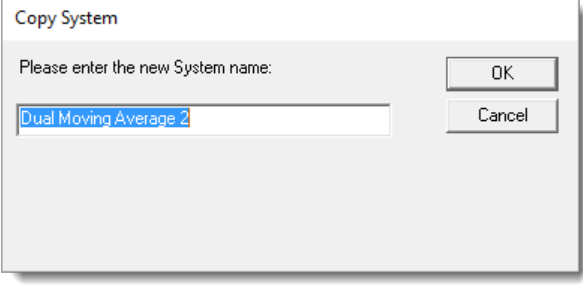
All system changes are temporary until the "OK" button is pressed. Pressing the "Cancel" button after making any changes to one or many systems will loose all the changes.

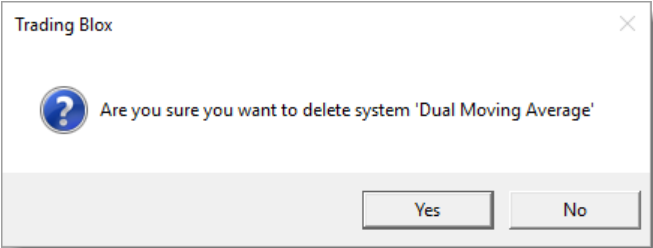
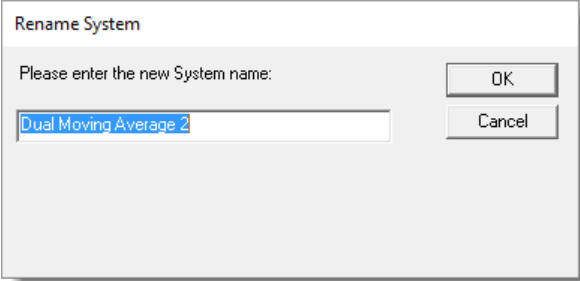
When a change is required, make the change and press the "OK" button. If more changes are needed to the same file or any other system file, open the System Editor again and make the changes needed.

System Types:	Description:
Standard System: ^Top	Standard systems are not a Global Suite System (GSS). A standard system is any system that does not have the same name as a Suite. Standard system are assigned testing sequence index value that reflects the order in which the system was added to a Suite. The first system added will have an index value of one. The second system added to the suite will have an index of two. As more systems are added the index will increment to the next higher integer. All the systems in a Suite are displayed in the tab order in which they were added to the Suite, and

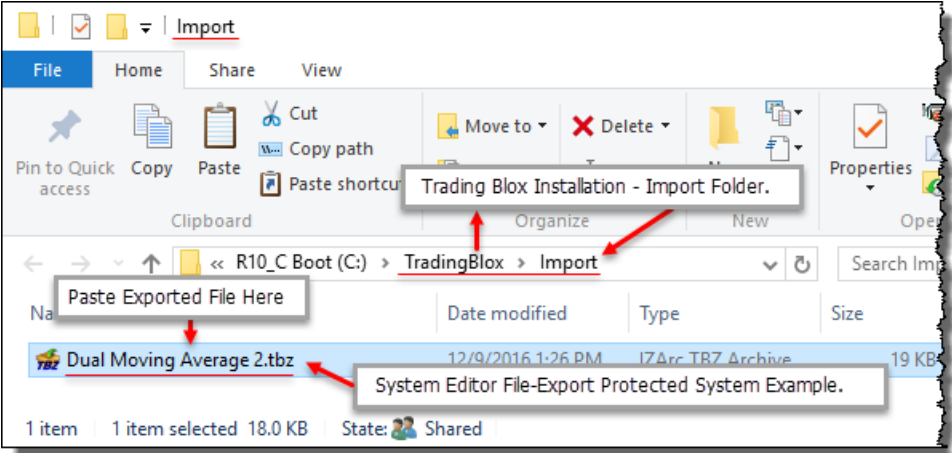
System Types:	Description:
	system name on a Tab selector shows a left to right tab order of when they were added to a Suite. All standard systems in a Suite will execute their blox same script name sections in the same order in as they were indexed (left to right index value of 1, 2, 3, etc.). Not all blox have the same script name section. Those without a script section that is being executed by the Suite, will be skipped over until the next script section name is selected for execution. Knowing the order in which scripts are executed can make a difference in how a system performs. Ideally, systems are created and adjusted as a single system in a Suite so the process of other systems are not influencing the test results.
Global System: ^Top	Any system name that matches a suite name in the list of available suite names will automatically be designated as a Global Suite System (GSS). These system get a system number designator of zero and have special timing for some of its script sections. GSS systems do not generate orders without special scripting methods.

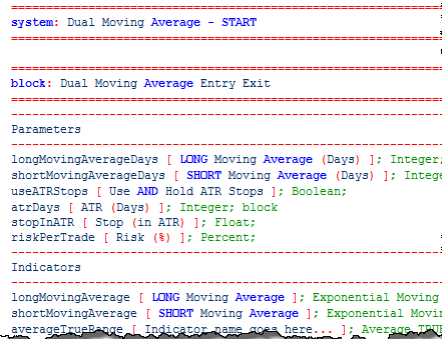
System Menu: ^Top	Descriptions:
	You can create new systems and delete systems, and change what is in a system using the system editor. You build systems by selecting, or creating a new system name on the left, and then add Blox from the available blox list on the right. Some of the lists accept multiple Blox and some lists can accept only one Block. The Portfolio Manager, the Risk Manager, and the Money Manager can accept only one Block per system. The Entry Block, Exit and Auxiliary block names can appear more than one time in a system. The reason for this is that you may want to have multiple entry/exit ideas executing at the same time. You might also want to have different risk, portfolio and reporting ideas operating at the same time. Trading Blox System Editor menu provides the controls for creating a new system name, making a copy of an existing system name, renaming a current system, exporting a system as an encrypted system, or displaying the system modules and their active script section is controlled by selecting one of the options in this menu. When a name is selected in the list of system names, and the System word above the list is selected, the menu that appears on the side of the display will enable changes listed. Right-Clicking the mouse on the selected system will display the same menu:

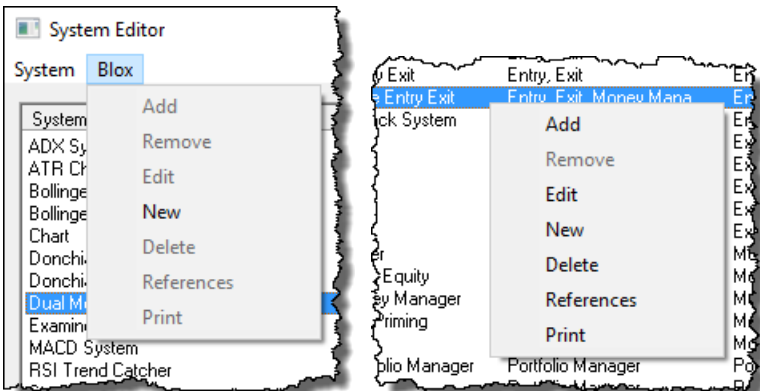
System Menu: ^Top	Descriptions:
 System Editor System Menu Options:	 System Menu Right-Click Access
New: ^Top	To create a new system name, enter the new name for the new system in the text area after clicking the System's New menu:  See: System Names
Copy: ^Top	To edit or copy a system name so it be the same system logic, but have a different system name, enter the name here:  Create a copy of a selected system name so it can be given a new system name, and or be modified to work differently by changing some of the blox ideas that will be different from the selected system name. See: System Types
Delete: ^Top	System Editor's System Delete menu will remove the system from the software and from the computer's disk. Deleting a system name does not

System Menu: ^Top	Descriptions:
	delete the blox assigned to the system because blox assignments are by name reference only. To delete a selected system name, click on the Delete menu and answer "Yes" or "No" to confirm or reject your action when this next dialog appears. When "Yes" is clicked, the selected system will be deleted from the disk and the software even if you click on the Cancel button to exit the System Editor. If you answer "No" to Not-Delete the system, the delete action will be aborted:  Notes: If you delete a systems by mistake it can be recreated again by using the techniques you would use to create a new system. If the system is a Trading Blox supplied system you can install a new copy of Trading Blox on your computer in a different directory location and the Suite Menu's Export and Import feature to bring back into the Trading Blox installation where it was accidentally deleted. This may also be one of those times when you remember how good a regular back up of your disk will feel. You can also use the automated Backup files to recover the deleted system if this day isn't the first day that the deleted system was in the list of system names.
Rename: ^Top	This menu selection will Rename a selected system name. Any name that is in accordance with the Naming Rules can be used. This can be handy when you are adding or exporting a system version so it can be added to a suite more than once, but tested with different parameter value or portfolio files, or any other idea. Edit or replace the system name to its new name: 

System Menu: ^Top	Descriptions:
	See: System Types
Export: ^Top	Creates a protected export file and opens the folder where the file is saved so it can be accessed. This option will Export and Encrypt the selected system name. Exported systems are exported and placed to Export folder as a special encrypted file. Imported encrypted system can be tested in a suite. Exported system parameters are visible and can be changed. Before exporting a any system ensure the system name you are exporting is unique. This is required because Trading Blox will not import a system name is the same as a system name in the Trading Blox installation where the file is to be imported. When the name is the same, this message will appear: Trading Blox Trading Blox encountered the following issues while loading the settings for the systems: <u>Error. Imported system Dual Moving Average already exists. System names must be unique.</u> OK Imported System Name Already Exists Error Message. When a system is exported it creates a protected system using encryption: Trading Blox Installation - Export Folder. Copy Exported File Dual Moving Average 2.tbz System Editor File-Export Protected System Example. System Editor's Export Menu Selection File & Folder Location Exporting Systems and Exporting Suites are different. In the Export folder shown above, it shows an exported system named: "Dual Moving Average.tbz." Exported Systems and Exported Suites share the same folder, but they are very different types of exported file structures.

System Menu: ^Top	Descriptions:
	When you export a System from the System Editor, only the system file and the blox files that are attached to the system are included in the exported file. Exported can be shared with others by placing a copy of the file in the Import folder of the intended Trading Blox installation. To add the exported file and its blox module to another Trading Blox installation, click on the main screen File Menu's Import icon in the Folder Group.  System Editor's Export File Goes in Import Folder Warning: Be sure that the name of the System and the name of all Blox in the system are unique. We recommend that you use your name, or some other unique identifier, when naming the blox and system modules. If the names used are found during importing they will not conflict with other blox or systems already be Trading Blox installation. When testing an export/import process into the same environment, make a renamed copy of the system and then export, delete the copy, and import. In this way the system name is not duplicated in the same environment. Never change the name of the exported encrypted ".tbz" file once it is created. It is linked to the system name and blox and changing the name will cause the system to not appear.
Print: ^Top	Creates a text document showing all the blox modules and the blox script sections with script statements. Text output is sent to the the default text editor. Clicking this button will generate a report that list all the Blox modules contained in the selected system. In that report each of the modules

System Menu: ^Top	Descriptions:
	variables, indicators, parameters and script sections where programming code has been entered will be displayed in a text report that will be displayed in the computer's text editor. Name of the report will be the name of the selected system prefixed with the "Preview". i.e. "Print Dual Moving Average Entry Exit.txt" will be the name if the Donchian w/Rule Labels System is the selected system.  When System Print option is selected a text file is created that shows all the blox and their script information displayed in text file like the example shown in this image. Print Dual Moving Average System

System Blox Options: ^Top	Menu Description:
System Editor's Blox menu option allows Blox Editor versions to get access to the Blox Editor. Selecting a Blox name in the list of available Blox names and right-clicking the mouse will display the same menu near the selected Blox: System Blox Editing:  System Editor Blox Menu Options: System Editor Right-Click Options	
Blox Add: ^Top	Selected blox name is added to the selected system name.
Blox Remove: ^Top	Selected blox name is removed from the selected system name.
Blox Edit: ^Top	Option: Trading Blox license required.

System Blox Options: ^Top	Menu Description:
	Select a block and click on this Edit menu option to open the selected blox name in the Trading Blox Builder's Blox Editor.
Blox New: ^Top	Option: Trading Blox license required. Creates a blox name in the Trading Blox Builder's Blox Editor.
Blox Delete: ^Top	Selecting the Delete menu open will opens a dialog of the selected blox name to be deleted. Select a block, and click on this button to delete the block from the system and delete the file as well. Blox cannot be delete if they are in a system. Use with caution as blox cannot be recovered once deleted.
Blox References: ^Top	Opens a window showing where the selected blox name is being used. Select a block, and click on this button to see a list of system references. All the systems the block is in will be listed.
Blox Print: ^Top	Prints the selected Blox Name scripts where statements are located. Prints the block to the printer. Output includes all scripts, variables, indicators, and parameters used the block.

Links:**See Also:**

1.1 Print System Scripts

This is an example printout of how the System Editor's Export menu option will create a text file of the system blox components and the script information contained in the printed system. This information will appear in the default text reader program, Notepad.exe, or other text reader that is assigned to open text files in Windows.

```
=====
=====
system: Dual Moving Average - START
=====
=====

=====
=====
block: Dual Moving Average Entry Exit
=====
=====

-----
-----
Parameters
-----
-----
longMovingAverageDays [ LONG Moving Average (Days) ]; Integer;
shortMovingAverageDays [ SHORT Moving Average (Days) ]; Integer;
useATRStops [ Use AND Hold ATR Stops ]; Boolean;
atrDays [ ATR (Days) ]; Integer; block
stopInATR [ Stop (in ATR) ]; Float;
riskPerTrade [ Risk (%) ]; Percent;
-----
-----
Indicators
-----
-----
longMovingAverage [ LONG Moving Average ]; Exponential Moving Average;
Close; block;
shortMovingAverage [ SHORT Moving Average ]; Exponential Moving Average;
Close; block;
averageTrueRange [ Indicator name goes here... ]; Average TRUE Range; ;
block;
EhlersNonLinearMA [ EhlersNonLinearMA ]; Ehlers NonLinear Ma; Close;
block;
-----
-----
script: Exit Orders
-----
-----
' Exit position on open if moving averages cross
If instrument.position = LONG AND shortMovingAverage < longMovingAverage
THEN
    broker.ExitAllUnitsOnOpen
ENDIF
```

```

If instrument.position = SHORT AND shortMovingAverage > longMovingAverage
THEN
    broker.ExitAllUnitsOnOpen
ENDIF
' -----
' Enter stop if "Use ATR Stop" is true
' -----
If useATRStops THEN
    broker.ExitAllUnitsOnStop( instrument.unitExitStop )
ENDIF

-----
script: Entry Orders
-----

VARIABLES: currentclose, stopPrice Type: Price
' Get the current close.
currentClose = instrument.close

' If we are not long and the faster moving averages are above the slower
one.
If instrument.position <> LONG AND
    shortMovingAverage > longMovingAverage THEN

    ' If we are using ATR stops...
    If useATRStops THEN
        ' Compute the stop price.
        stopPrice = currentClose - (stopInATR * averageTrueRange)

        ' Enter a long on the open using this stop.
        broker.EnterLongOnOpen( stopPrice )

    ' NOT using ATR stops...
    ELSE
        ' Enter a long on the open with no stop.
        broker.EnterLongOnOpen
    ENDIF
ENDIF

' If we are not short and the faster moving averages are below the slower
one.
If instrument.position <> SHORT AND
    shortMovingAverage < longMovingAverage THEN

    ' If we are using ATR stops...
    If useATRStops THEN
        ' Compute the stop price.
        stopPrice = currentClose + (stopInATR * averageTrueRange)

        ' Enter a short on the open using this stop.
        broker.EnterShortOnOpen( stopPrice )
    
```



```

    ' NOT using ATR stops...
ELSE
    ' Enter a short on the open with no stop.
    broker.EnterShortOnOpen
ENDIF
ENDIF

-----
script: Unit Size
-----

VARIABLES: riskEquity, dollarRisk Type: Money

' Compute the risk equity.
riskEquity = system.tradingEquity * riskPerTrade

' If we are using ATR stops...
If useATRStops THEN
    If instrument.IsStock THEN
        dollarRisk = order.entryRisk * instrument.stockSplitRatio *
instrument.conversionRate
    ELSE
        dollarRisk = order.entryRisk * instrument.bigPointValue
    ENDIF
ELSE
    If instrument.IsStock THEN
        dollarRisk = averageTrueRange * instrument.stockSplitRatio *
instrument.conversionRate
    ELSE
        dollarRisk = averageTrueRange * instrument.bigPointValue
    ENDIF
ENDIF

' Set the trade size
order.SetQuantity( riskEquity / dollarRisk )

If order.quantity < 1 THEN
    order.Reject( "Order Quantity less than 1" )
ENDIF

=====
block: Margin Equity Risk Manager
=====
-----

block Permanent VARIABLES
-----

newMarginValue [ ]; Floating; 0.000000; block

```

```
plotCash [ Cash ]; Series; 0.000000; test
```

```
-----
Parameters
-----
```

```
maxMarginEquity [ Max Margin Equity (%) ]; Percent; block
checkMarginOnFill [ Check Margin on Fill ]; Boolean; block
```

```
-----
script: Can Fill order
-----
```

```
If checkMarginOnFill AND order.isEntry THEN
```

```
    If instrument.IsFuture THEN
        newMarginValue = order.quantity * instrument.margin
        If test.totalMargin + newMarginValue > test.totalEquity *
maxMarginEquity THEN
            order.Reject( "Cannot add " + AsString( newMarginValue, 2 ) + "
additional margin to " + AsString( test.totalMargin, 2 ) + " existing with
total equity of " + AsString( test.totalEquity, 2 ) + " and max margin
equity threshold of " + AsString( maxMarginEquity, 2 ) )
        ENDIF
    ENDIF
```

```
    If instrument.IsStock THEN
        newMarginValue = order.quantity * order.fillPrice *
instrument.conversionRate * instrument.stockSplitRatio
        If newMarginValue > test.cash * maxMarginEquity THEN
            order.Reject( "Cannot add " + AsString( newMarginValue, 2 ) + "
additional purchase equity with cash of " + AsString( test.cash, 2 ) + "
and max threshold of " + AsString( maxMarginEquity, 2 ) )
        ENDIF
    ENDIF
ENDIF
```

```
-----
script: Can Add Unit
-----
```

```
If order.isEntry THEN
    If instrument.IsFuture THEN
        newMarginValue = order.quantity * instrument.margin
        If test.totalMargin + newMarginValue > test.totalEquity *
maxMarginEquity THEN
            order.Reject( "Cannot add " + AsString( newMarginValue, 2 ) + "
additional margin to " + AsString( test.totalMargin, 2 ) + " existing with
total equity of " + AsString( test.totalEquity, 2 ) + " and max margin
equity threshold of " + AsString( maxMarginEquity, 2 ) )
        ENDIF
    ENDIF
```

```

ENDIF

If instrument.IsStock THEN
    newMarginValue = order.quantity * order.orderPrice *
instrument.conversionRate * instrument.stockSplitRatio
    If newMarginValue > test.cash * maxMarginEquity THEN
        order.Reject( "Cannot add " + AsString( newMarginValue, 2 ) + "
additional purchase equity with cash of " + AsString( test.cash, 2 ) + "
and max threshold of " + AsString( maxMarginEquity, 2 ) )
    ENDIF
ENDIF
ENDIF
ENDIF

-----
-----
script: After Trading Day
-----
-----

=====
=====
block: Trade Direction Portfolio Manager
=====
=====

-----
-----
Parameters
-----
-----
tradeDirection [ Trade Direction (LONG/SHORT/All) ]; Selector;
-----
-----
script: Filter Portfolio
-----
-----
'Set to no trades
instrument.DenyAllTrades

'Set to trade TYPE: indicated by tradeDirection parameter
If tradeDirection = TRADE_ALL THEN instrument.AllowAllTrades
If tradeDirection = TRADE_LONG THEN instrument.AllowLongTrades
If tradeDirection = TRADE_SHORT THEN instrument.AllowShortTrades

=====
=====
system: Dual Moving Average - END
=====
=====

```


Section 2 – System Blox Modules

System Editor's center section is where the system's modules are displayed. This system component area is where you manage, change and create new or existing systems.

System names don't have any goal requirement, nor do they have a requirement that the blox added to the system be used for creating trades. It can be a systems that has one or more blox that analyze instrument files. For example, it might be necessary to examine each one or more data files for problems. There might be a need to profile Futures files to find the frequency and time of roll-forward timing. There might be a need to examine the timing of forward and reverse stock splits, timing and dividend payments. A systems can be a way to easily add reporting to a suite, or create a custom position and orders report. It can also means you can be creative and think beyond the limits of trading to using a system in a suite to handle some programming ideas.

All systems must have at least one blox to be functional. A system without blox that does something won't have anything to do and it will be ignored. Systems don't have any requirements for how many or what type of blox must always be in a system.

Any systems intended to generate trades will need a blox that can generate orders. All orders created will need a quantity of shares or contracts to generate results. An order's size can use any idea you have for determining a quantity, or you can use one of the supplied Money Manager made available when Trading Blox was installed.

Systems that generate orders must include the Unit Size sizing script in a blox. When order sizing is part of the blox, the name of that blox will be displayed in the Money Manager area of the system, and that name can also appear in other sections. Systems can have more than one type of module, but others can only have one of three specific types. See [System Section](#) rules.

When it comes to trading, systems can provide a basis for a lot of creativity, so don't be afraid to test new ideas, and test more ideas again, and more ideas until you find what works, and still works for a while afterward.

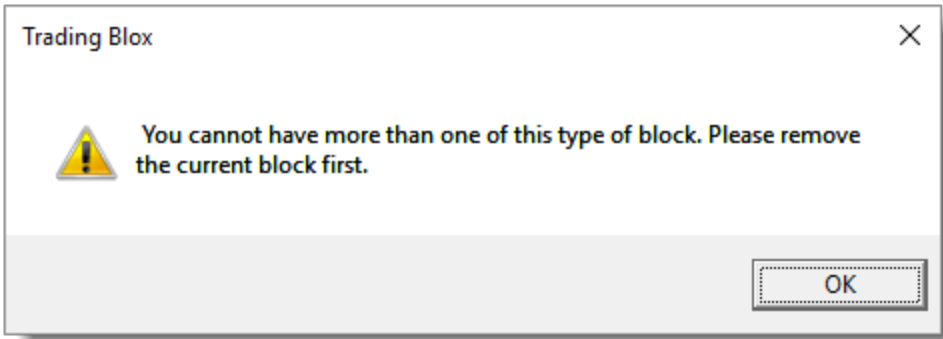
Note:

Multiple module names listed in any sections where more than one module can exist will execute scripts with the same name in the order in which the blox names are listed from top to the bottom. Top most blox name will execute its Entry Orders script first, the second blox will execute it Entry Orders next, the third blox will follow, etc. for all the blox with script section with the same name where there a programming statements .

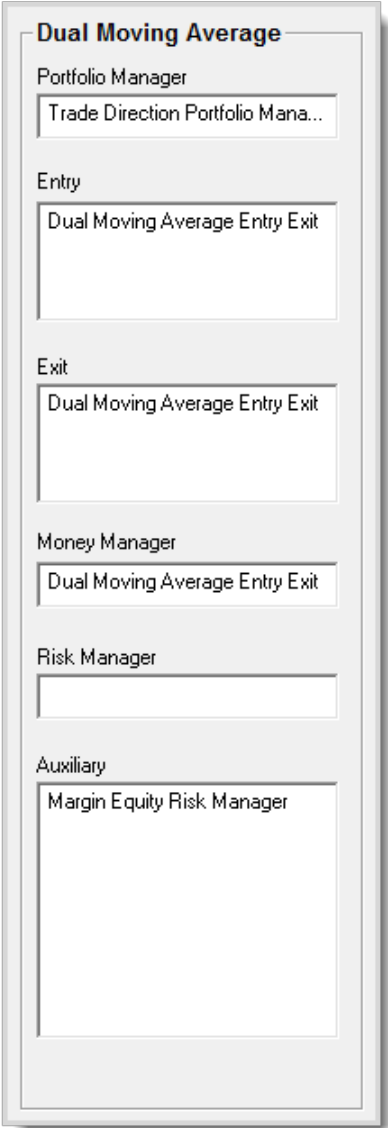
System Sections: [^Top](#)

**Portfolio
Manager
Money
Manager**

Only blox of these type can be assigned to each of these selected blox sections.

System Sections: ^Top	
Entry,Exit,Money Manager Risk Manager	 You Cannot Have More Than One of This Blox Error. When a blox is added to a system that is limited to only one blox with some of scripts in these three blox, this message will appear.
Entry Exit Auxiliary Custom Functions	All of these blox allow multiple blox to be added to any of these system area sections. When more than one blox is added to any of these script sections the name order of the blox will be sorted in alphabetical order. This is an important point to remember because the order in which each blox is listed in the system section area is the order in which script sections with the same name are executed. To change the order in which the script sections with similar names is executed, you must change the name of the blox so that it will be placed in a different position after it is sorted. Blox Script Sequencing: When more than one blox name is shown in a blox type area, the blox names will alphabetically be listed when the system editor is closed using the "OK" button, and then viewed again. Scripts in all the blox area use the sorted blox names to determine which script that is common to more than one blox is executed.

Selected Blox Types:	Blox Areas:	Area Descriptions:
	Portfolio Manager	Portfolio Manager type blocks appear in this area. Trying to add more than one Portfolio Manager is added to a system will show this Error Message. However, the Auxiliary modules can scripted to perform all of the task the typical Portfolio Manager performs. Using an Auxiliary blox to perform task that the currently selected portfolio manager was designed to perform.

Selected Blox Types:	Blox Areas:	Area Descriptions:
 System Selected Blox Type Areas	Entry	All Entry and Entry Exit type blox appear here. Entry and Exit sections do not need to be in the same blox. They can be, and often are in the same module. Systems can have multiple Entry, or multiple Exit, or multiple Entry Exit Blox in the same system. Any blox with an Entry Orders script section will automatically be displayed in this area. When the Entry module also have an Exit section, the same name will appear in the Exit section. See Blox Script Sequencing.
	Exit	All Exit type and Entry Exit type blox appear here. Entry and Exit sections do not need to be in the same blox. They can be, and often are in the same module. Systems can have multiple Entry or Entry Exit Blox. Any blox with an Exit Orders script section will automatically be displayed in this area. When the module has Entry and Exit section, the same name will appear in the Entry section. See Blox Script Sequencing.
	Money Manager	Money Managers and other blox contain a Unit Size section determine the size of an order. When a Money Manager is not included in a system, the system will not be able to create a quantity for an order that can become a position. There are other reasons why a system cannot generate trades. For example, the money needed to support a trade is too low. Risk size is greater than the equity available. Orders are being filtered or something in the Global Suite Parameters section is not set correctly. When no trades exist, review the Filtered Trades Log to see what it reports. Trying to add more than one blox that contains the Unit Size script to a system will show this Error Message.

Selected Blox Types:	Blox Areas:	Area Descriptions:
	Risk Manager	Risk Manager assignments appear in this section and can contain all or any of these script names: Initialize Risk Management Compute Instrument Risk Compute Risk Adjustments Adjust Instrument Risk Trying to add more than one blox that contains the Risk Manager scripts to a system will show this Error Message .
	Auxiliary	All Auxiliary blox assignments appear here. Auxiliary blox can be developed to perform many of the system task not typically provided in other blox. Multiple Auxiliary blox can be assigned to a system. When more than one Auxiliary blox appears in the module names will be sorted alphabetically. See Blox Script Sequencing .

Links:**See Also:**

Section 3 – Available Blox Modules

The Trading Blox System Editor's blox module list show all the blox names available in the active installation.

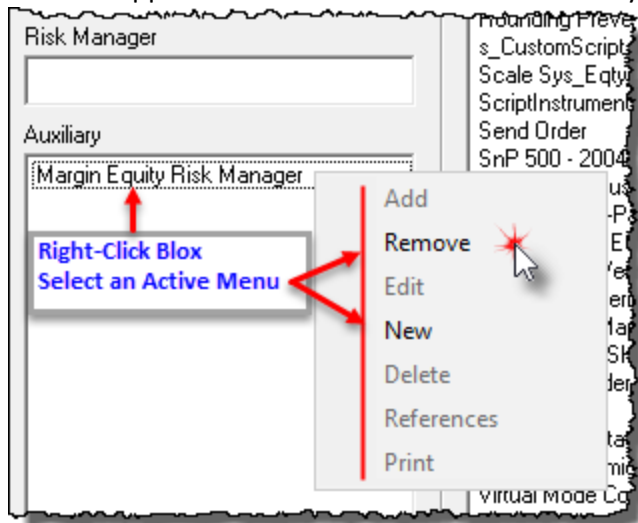
System Blox Limits:

Additional blox can be added or removed or assigned to a selected system, unless a system is limited to have [OnlyOneOfThese](#) type of blox can be in a system. Blox limited to contain only one blox types are Portfolio Managers, Unit Sizing, Entry-Exit-Money Manager, and Risk Manager. If you try to add an addition blox of the three listed in this paragraph, Trading Blox will not allow that blox to be added.

The Entry & Exit blox type without a built in Money Manager can be added to the Entry or Exit sections of the system construction display. See the Portfolio Manager, Money Manager, Entry-Exit-Money-Manager, Risk Manager. The restriction with the restriction with the Entry-Exit-Money-Manager is the script section Unit Sizing section happens because only one of these can be in any system.

Adding & Changing Blox in a System:

To change a system's blox in a system, the approach is to select the blox that is in a system and then click on the menu item that will perform the removal. To add a blox to a system, select the blox in the Blox Name list of blox, and right-click to display the menu. Click the Add item and the blox will appear in the list of blox in the selected system.



System Blox Add-Remove Method

Adding, removing or changing a blox assignment in a system starts by right-clicking on something and the selecting the menu item that will achieve the change.

the [Blox list of menu options](#). By selecting an editing option the blox assignment will change. An alternate is to click on a blox name and then click on the Blox menu option at to left side of the editor to access the [System Blox Options](#).

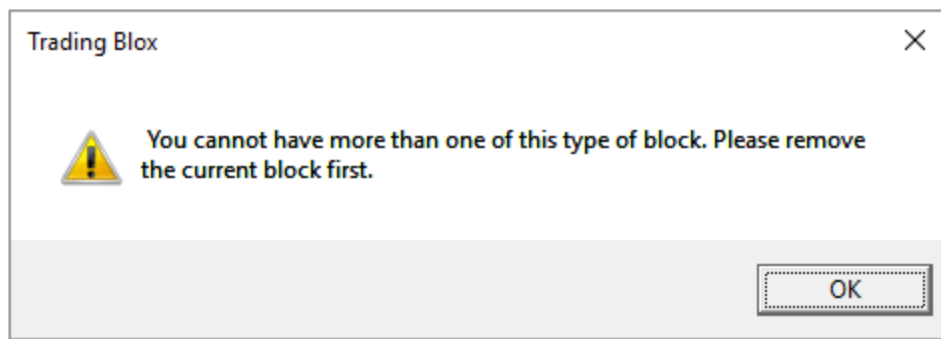
Block Name	Type	Group	Modified
ADX Entry Exit	Entry, Exit	Entry Exit	12/2/2013 10:20:16 AM
ATR Channel Breakout Entry Exit	Entry, Exit	Entry Exit	6/3/2015 10:05:28 AM
Basic Money Manager	Money Manager	Money Manager	12/2/2013 10:20:16 AM
Bollinger Breakout Entry Exit	Entry, Exit	Entry Exit	6/3/2015 10:08:48 AM
Bollinger Counter Trend Entry Exit	Entry, Exit	Entry Exit	6/3/2015 10:06:27 AM
Chandelier Exit	Exit	Exit	12/2/2013 10:20:16 AM
Chart Margin Equity Ratio	Auxiliary	Auxiliary	12/2/2013 10:20:16 AM
Correlation Risk Manager	Risk Manager	Risk Manager	2/5/2015 9:03:54 AM
Correlation Risk Manager Check Fills	Risk Manager	Risk Manager	12/2/2013 10:20:16 AM
Donchian Smart Fills	Entry, Exit	Entry Exit	6/3/2015 10:04:44 AM
Dual Moving Average Entry Exit	Entry, Exit, Money Mana...	Entry Exit	11/23/2016 12:23:05 PM
Fixed Fractional Fixed Equity	Money Manager	Money Manager	12/2/2013 10:20:16 AM
Fixed Fractional Money Manager	Money Manager	Money Manager	12/2/2013 10:20:16 AM
Fixed Fractional with Pining	Money Manager	Money Manager	12/2/2013 10:20:16 AM
Gap Against Exit	Exit	Exit	12/2/2013 10:20:16 AM
Group Risk Manager	Risk Manager	Risk Manager	12/2/2013 10:20:16 AM
MACD Direction Portfolio Manager	Portfolio Manager	Portfolio Manager	12/2/2013 10:20:16 AM
MACD Entry Exit	Entry, Exit	Entry Exit	12/2/2013 10:20:16 AM
MACD Portfolio Manager	Portfolio Manager	Portfolio Manager	12/2/2013 10:20:16 AM
Margin Equity Risk Manager	Auxiliary	Risk Manager	9/6/2015 8:06:57 AM
Multi Money Manager	Money Manager	Money Manager	12/2/2013 10:20:16 AM
Plot Avg Close Since Entry	Auxiliary	Auxiliary	12/2/2013 10:20:16 AM
Plot Stop Price	Auxiliary	Auxiliary	12/2/2013 10:20:16 AM
Profit Target Exit	Exit	Exit	12/2/2013 10:20:16 AM
Pyramiding Entry	Entry	Entry	12/2/2013 10:20:16 AM
Report Elapsed Time	Auxiliary	Auxiliary	12/2/2013 10:20:16 AM
RSI Trend	Entry, Exit	Entry Exit	12/2/2013 10:20:16 AM
RSI Trend Entry	Entry	Entry	12/2/2013 10:20:16 AM
RSI Trend Exit	Exit	Exit	12/2/2013 10:20:16 AM
Statistics Robust	Auxiliary	Auxiliary	12/2/2013 10:20:16 AM
Stats for Multi-Parameter Charts	Auxiliary	Auxiliary	12/2/2013 10:20:16 AM
Stochastic Oscillator	Entry, Exit	Entry Exit	12/2/2013 10:20:16 AM
Strength and Direction Filter	Portfolio Manager	Portfolio Manager	12/2/2013 10:20:16 AM
Strength Filter	Portfolio Manager	Portfolio Manager	12/2/2013 10:20:16 AM
Strength Ranking Stock System	Entry, Exit, Money Mana...	Entry Exit Money Man...	2/5/2015 9:03:54 AM
Test Source Control Add Block	Auxiliary	Test	2/5/2015 9:03:54 AM
Total Risk Limiter	Risk Manager	Risk Manager	12/2/2013 10:20:16 AM

System Available Blox Modules

System Sections: [^Top](#)

Portfolio Manager
Money Manager
Entry,Exit,Money Manager
Risk Manager

Only blox of these blox type can be assigned to each of these blox type sections.



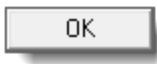
You Cannot Have More Than One of This Blox Error.

When a blox is added to a system that is limited to only one blox with some of scripts in these three blox, this message will appear.

Entry Exit
Auxiliary
Custom Functions

All of these blox allow multiple blox to be added to any of these system area sections. When more than one blox is added to any of these script sections the name order of the blox will be sorted in alphabetical order. This is an important point to remember because the order in which each blox is listed in the system section area is the order in which script sections with the same name are executed.

System Sections: ^Top	
	To change the order in which the script sections with similar names is executed, you must change the name of the blox so that it will be placed in a different position after it is sorted. Blox Script Sequencing: When more than one blox name is shown in a blox type area, the blox names will alphabetically be listed when the system editor is closed using the "OK" button, and then viewed again. Scripts in all the blox area use the sorted blox names to determine which script that is common to more than one blox is executed.

Blox Information: ^Top	Column Description:
Block Name	File name of blox. Name can only be edited with Trading Blox Editor included in the Builder edition.
Type	Primary Group names are listed in this column and must be one of the following: Portfolio Manager Entry Entry Exit Exit Money Manager Risk Manager Auxiliary Custom Group names will also appear in this column when they are used by any of the blox that are installed in the Trading Blox Blox folder.
Group	Primary Group names shown in the row above will be shown in this column when the blox item isn't using a custom group name. Custom Group names can be almost any name the blox creator wanted to identify and have that blox appear in a specific group section in the Blox Builder Editor.
Modified	Date listed in the date the file was created or modified.
	All changes made to any of the systems will be ignored and lost when the "Cancel" button is used to exit the System Editor.
	All changes made since entering the System Editor will be saved and available to the system when the System Editor is closed using this "OK" button.

Links:

Links:**See Also:**

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 108

Blox Basic Editor

Part



VII

Part 7 – Blox Basic Editor

This section of the manual about the primary editor for creating blox modules describes how the features in the editor can be used to create a new blox, change the scripting in a blox. It will also allow you to rename, copy the blox and modify the name, or remove the blox from the installation.

Before you try to modify any included system or any of its blox modules installed with Trading Blox, or obtained from another source, it is to your advantage to be familiar with this section before making and changes.

When you only want to change the blox modules included in a system, the [System Editor](#) section is where those changes must be made.

What is in a Trading Blox system?

Systems can be one Blox, or a collection of Blox modules. There are five Block types available for creating the logic that a system idea might need. Each type of blox module can be for a particular purpose in a trading idea. One blox module, Auxiliary, is a flexible use blox. This means it can perform the functions of another module like the Portfolio Manager's ranking and sorting operations. It can also be where custom functions that are needed to support a trading idea like trade on certain days, or when other ideas need to influence a system. Auxiliary blox are good module for creating custom reporting and file function ideas.

Supported trading system Blox types:

System Component:	Corresponding Block Type:
What to Trade	Portfolio Manager
When to Trade	Entry Blox
Whether to Trade	Risk Manager
How Much to Trade	Money Manager
When to get out	Exit Blox
Any New Idea Not Related to Above	Auxiliary

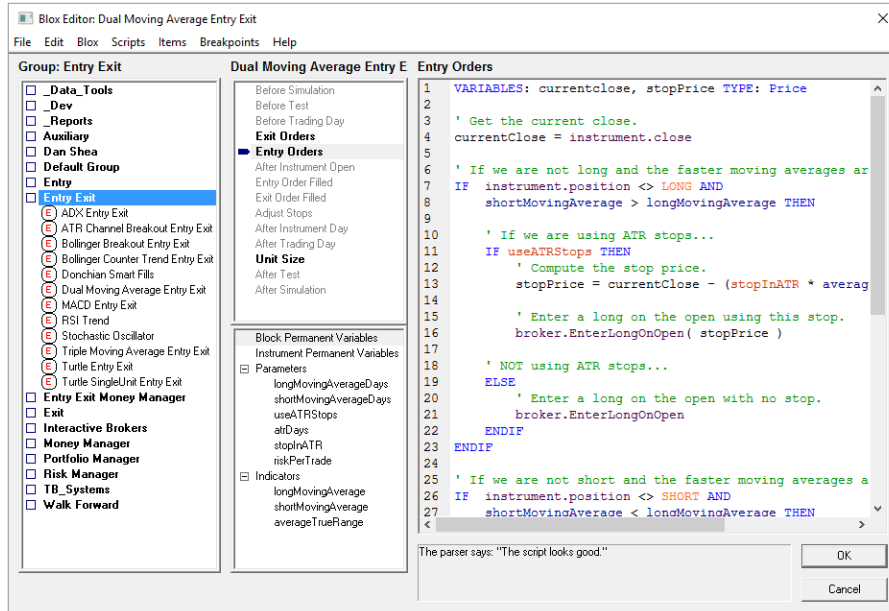
Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 111

Blox Basic Language System Editor:

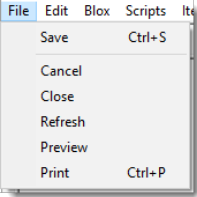
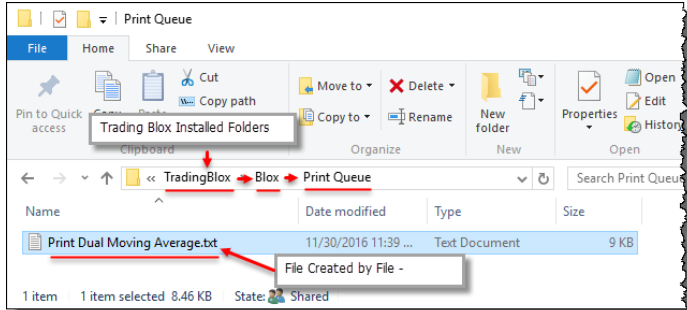
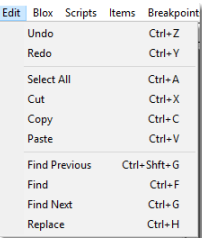
Blox Basic Editor is available with the Builder versions of Trading Blox. Only the Trading Blox Builder II Editor version will have access to all the possible blox creation capabilities made possible with the Blox Basic Editor environment.

System Editor allows you use the systems that come with Trading Blox as they are originally configured, or you can change them to test them with additional or changed logic. New systems can be created with any of the Blox listed in the Blox Catalog, or you can add them from a download or when you create a new Blox idea in the Blox Basic Editor.



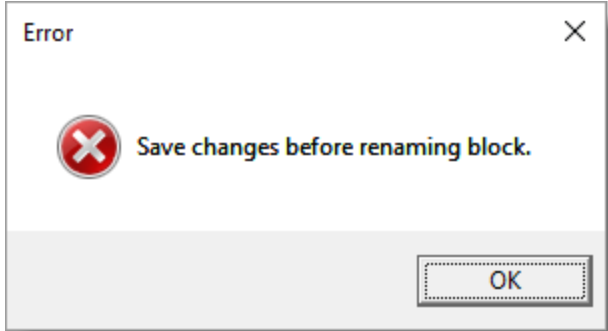
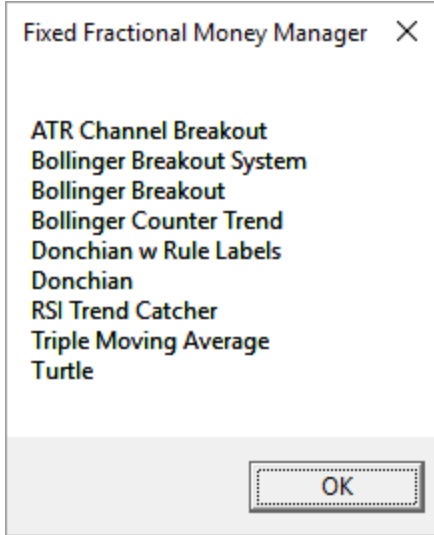
Trading Blox Basic Editor

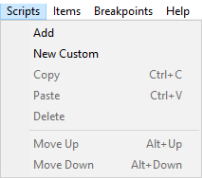
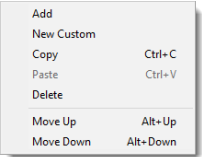
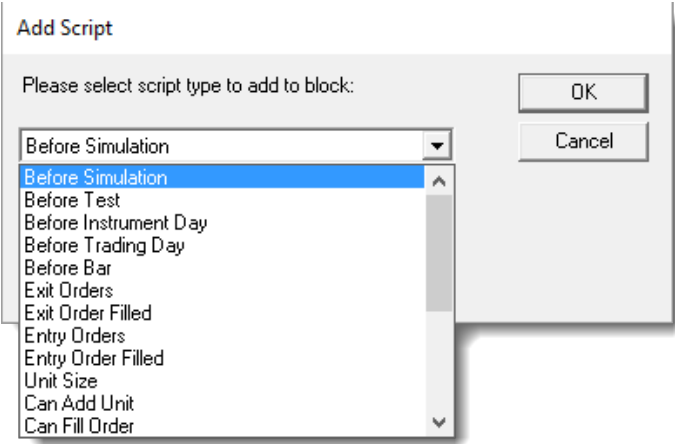
Close Editor Buttons:	Description:
OK: ^Back	Clicking the "OK" button to exit the editor will automatically save any changes made while the editor was opened.
CANCEL: ^Back	Clicking the "Cancel" button without using the " File - Save " menu option or " Ctrl+S " keyboard save command will generate this warning message: Editor Changes Not Saved Warning Message • Selecting the "Yes" button will save all changes and close the editor. • Selecting the "No" button will close the Editor without saving any changes made since the last "File - Save" menu option or "Ctrl+S" keyboard save command. • Selecting "Cancel" button will leave the editor open without saving any changes.

Menu Name:	Description:	
 Blox Basic Editor - File Menu ^Back	File Options:	Description:
	Save - Ctrl+S	Clicking the Save menu option or using the Ctrl+S hotkeys save all changes.
	Cancel	Using this menu option is the same as using the CANCEL Button .
	Close	This menu option is the same as using the OK Button .
	Refresh	Use this button when a blox module has been added to the blox folder and it doesn't appear in the list of available blox modules.
	Preview	Click on this menu item to capture all the script information into a text document that is displayed on the screen. Capture module script is saved in a file that is located here:
		 Editor File-Preview Menu File Saved Location
	Print - Ctrl+P	Use this option to quickly send a printout of the selected blox module to your printer.
 Blox Basic Editor - Edit Menu ^Back	Edit Options:	Description:
	Undo - Ctrl+Z	Use this to undo a recent edit.
	Redo - Ctrl+Y	Use this to repeat a recent change.
	Select All - Ctrl+A	Use this to select all the scripted text in the editing window area.
	Cut - Ctrl+X	Use this to cut the selected text from the editing window area.
	Copy - Ctrl+C	Use this to copy the selected text from the editing window area.
	Paste - Ctrl+V	Use this to paste previously copied text into the editing window area.

Menu Name:	Description:	
	Edit Options:	Description:
	Find Previous - Ctrl+Shft+G	Use this to find a previous term found in by the Find search option.
	Find - Ctrl+F	Find is a search function that looks for the text pasted into its search field.
	Find Next - Ctrl+G	Use this to find the next item that matches the Find's search term.
	Replace - Ctrl+H	Use this to replace a specific text with a replacement text.
Blox Scripts Items Breakpoints Help New Delete Copy Rename Systems Reload ☒ Sort Alphabetical ☐ Sort Modified Date ☐ Sort Type Blox Basic Editor - Blox Menu ^Back	Blox Options: Description: New Selecting this option will display this dialog: New Trading Block Name New Trading Block Enter Name Group Auxiliary New Blox Display Group Section [Section Name - Optional] Screen Name Block Type ☒ Entry and Exit ☐ Money Manager ☐ Portfolio Manager ☐ Risk Manager ☐ Auxiliary Select the Blox Design Type ☐ Include Default Scripts Unchecked includes minimal script section. Enable adds all scripts for the selected blox type. BloxMenu-New Blox Dialog Delete Selecting the Delete option will remove the blox from the disk even if the Blox Basic Editor is closed using the CANCEL Button. Delete Block ? Are you sure you want to delete the block 'New Test Delete Example'? Selected Blox Name OK Cancel Blox Menu-Delete Blox Warning Deleted can be recovered when they are deleted unless the deleted blox module is recovered from the	

Menu Name:	Description:	
	Blox Options:	Description:
		" Backups " that are automatically created each day that Trading Blox is run.
	Copy - Ctrl+C	Using this Blox menu option will create a copy of a selected blox that will first appear in this dialog where its blox name and display name can be changed:  Blox Menu-Copy Selected Blox Dialog Once a blox is copied and it appears in the list of available blox, it is not saved by the dialog until the Editor is closed with OK Button or Save - Ctrl+S function.
Rename		Select a blox module and click rename for this dialog to appear:  Blox Menu-Rename Selected Blox Dialog If the selected blox is not saved, this warning message will appear:

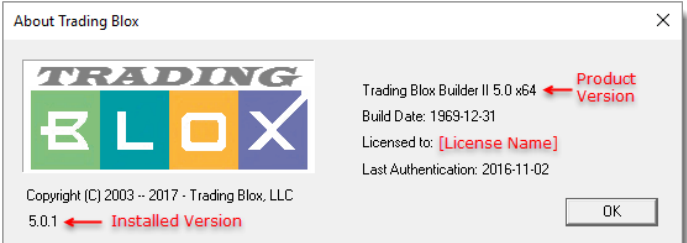
Menu Name:	Description:	
	Blox Options:	Description:
		 Blox Menu-Save Blox Changes Warning
	Systems	Systems is a search tool that reports which systems have the selected blox assigned their systems:  BloxMenu-Systems Where Used Search Results List This tool should be used prior to making any changes to a blox unless the changes planned are to be used by all the systems where this blox is assigned.
	Reload	If you add a block to the Blox folder while the block editor is open, this menu option will reload all the Blox in the Blox folder so they will all appear in the editor.
	Sort Alphabetical	
	Sort Modified Date	
	Sort Type	

Menu Name:	Description:	
 Script Menu Options ^Back  Script Menu - Select Script Section Name - Right-Click Menu	Script Options:	Description:
	Add	When a blox module doesn't have the script section needed, the missing script section can be added as long as it is allowed in the selected blox type where the script section is needed. When the Add menu option is selected this dialog will appear:  Script Menu - Add Script Section Option List When the above drop-down option list is displayed, select the script section needed and it will appear in the editor display of available script sections for the selected blox. When the script section that is needed doesn't appear, consider adding a new blox type where that script section is available.
	New Custom	Custom scripts create script names that can be called like a funtion using the Script Object functions and properties to call the custom script. A custom script section is created by selecting, or creating a new blox name, and then adding the a custom script section name:

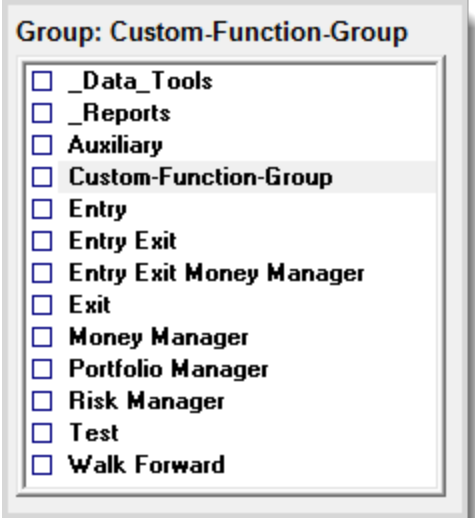
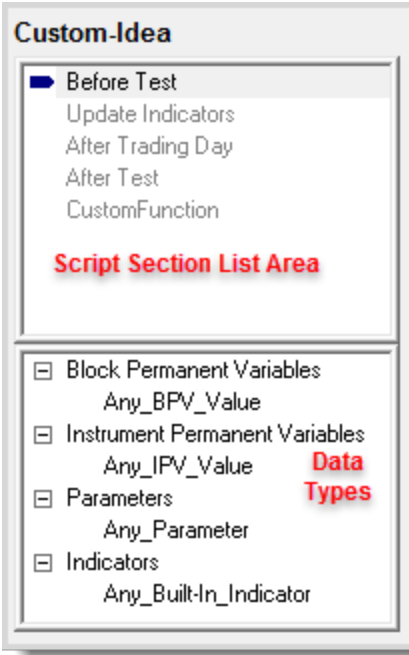
Menu Name:	Description:	
	Script Options:	Description: New Custom Script Please enter new custom script name: CustomScriptName Custom Script Name will become the name used to call the script section. Create the name without spaces or symbol characters. Script Menu - New Custom Script Section Creation Dialog When the script section is created, it will appear in the selected blox where it was added and the new custom script section will appear in the list of available script sections: Blox Editor: CustomFunctionExample File Edit Blox Scripts Items Breakpoints Help Group: Custom Functions CustomFunctionExample PortfolioRiskRate Script Menu - Added Custom Script Section
Copy - Ctrl+C	Script menu's Copy options will copy the name of a selected script and all of the scripted statement line contents of that script so that it can be pasted in another blox.	
Paste - Ctrl+V	Script menu's Paste will add a script section and it scripted contents to a blox where that script name does not exists.	
Delete	When a script section is selected and the Delete option is selected, the selected script section and its contents will be deleted.	
Move Up - Alt+Up	This script option will change the displayed row where a selected script is located when the "Move Up" option, or the Alt+Up-Arrow button is pressed.	
Move Down - Alt+Down	This script option will change the displayed row where a selected script is located when the "Move Down" option, or the Alt+Down-Arrow button is pressed.	
	This menu's options is for working with any of the for following blox data element types:	

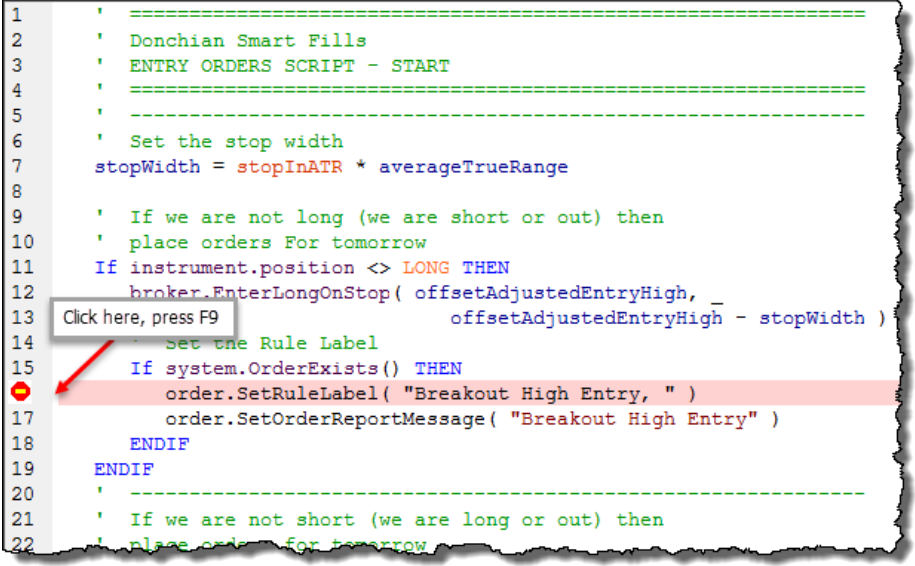
Menu Name:	Description:																
ItemsBreakpointsHelp New Edit CopyCtrl+C PasteCtrl+V Delete Move UpAlt+Up Move DownAlt+Down Items Menu Options ^Back	Block Permanent Variables (BPV) Instrument Permanent Variables (IPV) Parameters Built-in Indicators <table><tr><th>Editing Options:</th><th>Description:</th></tr><tr><td>New</td><td>When a data element is selected, click on this option will open a dialog where a new data element of the same type will be added to the group of that data type.</td></tr><tr><td>Edit</td><td>Clicking on this option when a data element is selected will open the selected data element's settings dialog.</td></tr><tr><td>Copy - Ctrl+C</td><td>Selecting a data element and then selecting this option will make a copy of the selected data element so that it can be pasted into another blox module.</td></tr><tr><td>Paste - Ctrl+V</td><td>Use this to paste a selected data element into blox where it doesn't already exists.</td></tr><tr><td>Delete</td><td>Use this option to delete a selected data element.</td></tr><tr><td>Move Up - Alt+Up</td><td>This Blox option moves a selected data element up a row in the list of data elements. The keyboard's arrow keys is a quick way to move data elements up with this hot-key combination: "Move Up = Alt+Up-Arrow keys. </td></tr><tr><td>Move Down - Alt+Down</td><td>This Blox option moves a selected data element down a row in the list of data elements. The keyboard's arrow keys is a quick way to move data elements down with this hot-key combination: "Move Down = Alt+Down-Arrow keys. </td></tr></table>	Editing Options:	Description:	New	When a data element is selected, click on this option will open a dialog where a new data element of the same type will be added to the group of that data type.	Edit	Clicking on this option when a data element is selected will open the selected data element's settings dialog.	Copy - Ctrl+C	Selecting a data element and then selecting this option will make a copy of the selected data element so that it can be pasted into another blox module.	Paste - Ctrl+V	Use this to paste a selected data element into blox where it doesn't already exists.	Delete	Use this option to delete a selected data element.	Move Up - Alt+Up	This Blox option moves a selected data element up a row in the list of data elements. The keyboard's arrow keys is a quick way to move data elements up with this hot-key combination: "Move Up = Alt+Up-Arrow keys.	Move Down - Alt+Down	This Blox option moves a selected data element down a row in the list of data elements. The keyboard's arrow keys is a quick way to move data elements down with this hot-key combination: "Move Down = Alt+Down-Arrow keys.
Editing Options:	Description:																
New	When a data element is selected, click on this option will open a dialog where a new data element of the same type will be added to the group of that data type.																
Edit	Clicking on this option when a data element is selected will open the selected data element's settings dialog.																
Copy - Ctrl+C	Selecting a data element and then selecting this option will make a copy of the selected data element so that it can be pasted into another blox module.																
Paste - Ctrl+V	Use this to paste a selected data element into blox where it doesn't already exists.																
Delete	Use this option to delete a selected data element.																
Move Up - Alt+Up	This Blox option moves a selected data element up a row in the list of data elements. The keyboard's arrow keys is a quick way to move data elements up with this hot-key combination: "Move Up = Alt+Up-Arrow keys.																
Move Down - Alt+Down	This Blox option moves a selected data element down a row in the list of data elements. The keyboard's arrow keys is a quick way to move data elements down with this hot-key combination: "Move Down = Alt+Down-Arrow keys.																
BreakpointsHelp Toggle BreakpointF9 Clear all BreakpointsShift+F9 Edit Breakpoint Blox Basic Editor - Breakpoints Menu ^Back	Breakpoints are temporary stops that are used to examine data values in the debugging environment. One a script execution is stopped, it can be stepped one line of script at a time, or allowed to jump to the next breakpoint. Script execution can also be allowed to resume automatic execution. Addition breakpoints can be added and removed during a debugging option by pressing F9, but those added or removed breakpoints are not permanent because debugging operation doesn't not save the changes. To add or remove a breakpoint location permanently, make edit in the editor when the script is not executing.																

Menu Name:	Description:						
	Breakpoints:	Description:					
	Toggle Breakpoint - F9	Breakpoints are added most often when the script section is not executing. They are added at line area just ahead of where a suspected problem might exists. They can be added during a debugging sessions when the script execution has stopped executing. Pressing F9 in the Blox Editor will add a red line overlay at the location where the mouse cursor is blinking. Pressing F9 on a breakpoint line will remove the breakpoint.					
	Clear all Breakpoints - Shift-F9	This option will remove all the breakpoint locations in the current script section when it is selected in the editor when the a test is not executing. It will remove all the breakpoints during a debugging operation, but the removal will not be permanent until it is performed in the editor when the execution has stopped.					
	Edit Breakpoint	Blox Editor Edit Breakpoint Settings					
Help User Guide PDF User Guide Blox Basic Guide PDF Blox Basic Guide Context Help Support Forum About Trading Blox Blox Basic Editor - Help Menu ^Back	<table><tr><th>Help Options:</th><th>Description:</th></tr><tr><td>User Guide</td><td>This option will open the installed Trading Blox User's Guide Help file. It is located in the Trading Blox installation folder as: TradingBloxHelp.chm</td></tr><tr><td>PDF User Guide</td><td>This is the PDF version of the User Guide Help information. When this option is clicked it will</td></tr></table>	Help Options:	Description:	User Guide	This option will open the installed Trading Blox User's Guide Help file. It is located in the Trading Blox installation folder as: TradingBloxHelp.chm	PDF User Guide	This is the PDF version of the User Guide Help information. When this option is clicked it will
Help Options:	Description:						
User Guide	This option will open the installed Trading Blox User's Guide Help file. It is located in the Trading Blox installation folder as: TradingBloxHelp.chm						
PDF User Guide	This is the PDF version of the User Guide Help information. When this option is clicked it will						

Menu Name:	Description:	
	Help Options:	Description:
		download the PDF version of the User Guide from the Trading Blox web site link: https://www.tradingblox.com/Manuals/TradingBlox%20User%27s%20Guide.pdf
	Blox Basic Guide	This option will open the installed Trading Blox Builder Guide Help file. It is located in the Trading Blox installation folder as: BloxBuilderHelp.chm
	PDF Blox Basic Guide	This is the PDF version of the Trading Blox Builder Guide Help information. When this option is clicked it will download the PDF version from the Trading Blox web site link: https://www.tradingblox.com/Manuals/Blox%20Builder%27s%20Guide.pdf
	Context Help	This option displays Trading Blox Builder Guide Help . Context specific topic display is not yet enabled.
	Support Forum	This menu item opens the Trader's Roundtable forum to the Trading Blox Support section located here: https://www.tradingblox.com/forum/viewforum.php?f=58
	About Trading Blox	 About Trading Blox

Editor Area:	Description:
Grouped Blox Modules: ^Back	

Editor Area:	Description:
	 Grouped List of Available Blox Modules
Script Sections: ^Back	 Blox Script Sections & Data Types All blox modules get their ability from the scripted statement written into the various script sections. There are many script sections available and some script sections determine the blox module type. To get a solid understanding of how many script sections there are, and how they used and executed, read through the Trading Blox Builder Guide Help file topic: Blox Module Reference.
Parameters & Variables: ^Back	Trading Blox Builder Guide Help file topic: Data Variables explains the different types of variables and how they are used. The list above shows there are two types of variables: • Block Permanent Variables • Instrument Permanent Variables. • Script sections can use Local Variables are constrained the script section in which they are declared and assigned values (no shown).

Editor Area:	Description:
	• Parameters are user accessible values that contain the settings and are available when a system tab that contains a blox with variables displays its parameters on the main screen.
Indicators: ^Back	Trading Blox Builder Guide Help file topic: Indicators provides a lot of information about this type of display and the calculations used in it sections: • Basic Indicators • Calculated Indicators • Creating Indicators • Custom Indicators • Indicator Access • Indicator Bar to Week Time Frame The details listed above focused on how to create and use indicators, but they will also provide some insight into the capabilities of how they can influence a system's operation.
Script Editing Area: ^Back	When a script section is selected, the script's section editing area will appear and the contents of that script section. when the script section is empty, the editing screen will be blank until something is added.  <pre> 1 ' ===== 2 ' Donchian Smart Fills 3 ' ENTRY ORDERS SCRIPT - START 4 ' ===== 5 ' ----- 6 ' Set the stop width 7 stopWidth = stopInATR * averageTrueRange 8 9 ' If we are not long (we are short or out) then 10 ' place orders For tomorrow 11 If instrument.position <> LONG THEN 12 broker.EnterLongOnStop(offsetAdjustedEntryHigh, _ 13 offsetAdjustedEntryHigh - stopWidth) 14 ' Set the Rule Label 15 If system.OrderExists() THEN 16 order.SetRuleLabel("Breakout High Entry, ") 17 order.SetOrderReportMessage("Breakout High Entry") 18 ENDIF 19 ENDIF 20 ' ----- 21 ' If we are not short (we are long or out) then 22 ' place orders for tomorrow </pre> Script Section Script Statement Editing Area F5 from Blox Editor saves changes and runs test.F5 from Blox Editor saves changes and runs test.
Parser Validation Area: ^Back	When the parser checks of a script section doesn't find any errors, it will report:

Editor Area:	Description:
	The parser says: "The script looks good." Parser Reports No Script Errors Found When a problem is found, a message in red text indicating there is a problem will be found: <pre> 14 ' Set the Rule Label 15 If system.OrderExists() THEN 16 order.SetRuleLabels("Breakout High Entry, ") 17 order.SetOrderReportMessage("Breakout High Entry") 18 ENDIF 19 END IF </pre> The parser says he didn't understand line 16 because of unknown identifier. PROPERTIES: FUNCTIONS: Script Parser Error Detected Message Example

Build or Editing a Systems: ^Top	Description:
Changing or Creating a New System	Systems can be built by selecting a system name on the side of the System Editor, or by creating a new system file name. When the system name is highlighted Blox can be added from the list of blox on the left. Blox already in a system can be removed from the items listed in the Static section in the center of the System Editor.
Adding and Removing Blox	There can only be one Blox of the following Type: - o Portfolio Manager - o Entry, Exit, Money Manager - o Money Manager - o Risk Manager There can be many Blox of the following Type: - o Entry - o Entry, Exit - o Exit - o Auxiliary To add a Blox to a system from the Blox listing and that Blox is not already in the system list, or more than Blox of the Blox being selected can be in the system, locate the Blox name and Right-Click with the mouse. Selected the Blox to add and then clicking the ADD button below the Static section will add the Blox to the system

Build or Editing a Systems: ^Top	Description:
	To remove a Blox from the Static section list of Blox, click on the Blox to remove and Right-Click the mouse. Clicking the Remove button when the Blox in the Static section is highlighted will remove it from the system. Note: System must have a way to provide size to an order. Money Managers perform that task. There are different Money Manager ideas available in the supplied list of Blox giving you some opportunity to see what works best. If you find your system executing and then reporting there are not any trades, that issue can be caused by various settings or when some parameter values are not compatible with how the system is configured. Before doing anything check the Filtered Log report that is available under the program's File -> Results -> Filtered Log menu path. If the system is a new system, be sure the system shows a name in the Money Manager section. If the system has items listed in the Filtered Log, check the settings that affect the reasons for those order rejections.
Global Suite Systems	When a system name is the same as the name of a suite it will be colored differently and be considered a Global Suite System (GSS). GSS designation allows Blox to be attached directly to the Suite structure. GSS Blox scripts are executed with different rules than the Blox scripts with the same name. Some GSS Blox scripts will execute ahead of system Blox scripts of the same name, and after the system Blox scripts with these other script names. GSS script are mostly used to access information in the suite's list of systems. GSS timing details are explained in the Blox Builder's Guide topic: Global Script Timing

Links:
See Also:

Historical Data Setup

Part



Part 8 – Historical Data Setup

Data Formats:

Trading Blox supports MetaStock and ASCII text formats. Sample data is ASCII since it is humanly readable and easier to debug.

Note:

MetaStock formatted files do not include the Unadjusted Close price, or the Delivery Month for Futures data. They do not allow the use of extra fields to be added, their values imported and accessible to Trading Blox.

Comprehensive data setup tutorials:

CSI Data: <https://www.tradingblox.com/tradingblox/howto-ua.htm>

[Data Provider Information](#)

Data File Types:**Metastock:**

With the Metastock format you don't need to deal with file names. If Trading Blox can read the Metastock Master file, it will locate all the files in the directory. For Futures, be sure that the symbol is correct in the Futures Dictionary.

Futures and Forex - (ASCII Text):

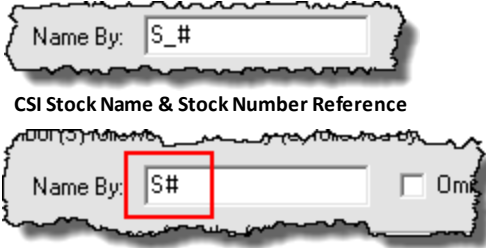
Futures or Forex Dictionaries require file names assigned to each symbol match the Dictionary symbol in its specified folder.

When the file name starts with the symbol letters, you can often find the file using the Futures Dictionary "**Find Symbol**" button.

Stocks - See [CSI Symbol Num](#) for Details:

CSI ASCII/EXCEL Setting:	File Name Example:
	AAPL.CSV or AAPL.TXT

CSI ASCII/EXCEL Setting:	File Name Example:
Name By: #	5902 . TXT

CSI ASCII/EXCEL Setting:	File Name Example:
 CSI Stock Name & Stock Number Reference Name By: S_# ☐ Omit New CSI's UA Version Auto-Add _ Character Older US version require the " _ " character. New versions of UA add it automatically. Leave the " _ " symbol out see if UA automatically adds it for you.	AAPL_5902.TXT ".csv" is OK It is best to use S# for stocks to avoid the wrong company data issue when a symbol is reused.

CSI ASCII/EXCEL Setting:	File Name Example:
Suffix: Both ".txt" or ".csv" suffix designators are allowed. "CSV" files open into a spreadsheet when double-clicked. Notes: ^Top Stock Dictionaries can be automatically updated using the main screen's Tool -> Build Stock Dictionary selection. When a Stock Dictionary is built, it will include all the stock symbol files in the selected "Stock" folder. It will also include all the stock symbol files that are contained in all the folders that are in the selected "Stock" folder. To select a stock folder destination that is different from where your stock files are being placed, the	

CSI ASCII/EXCEL Setting:	File Name Example:
When a Stock Dictionary is being built, it will first look to find that Symbol, or CSI #, in the Unfair Advantage "sdbfacts.csv" database file. During the building of the dictionary any symbols that have duplicates in the database file will not be included in the finished dictionary unless those symbols have a CSI # appended to its symbol, i.e. AAPL_5902.CSV. Symbol file names can have a ".CSV" or ".TXT" suffix identifier. When defining the field columns in the CSI Unfair Advantage software, be sure to use an upper case 'V' for the volume column. A Lower case 'v' is intended to Futures files that are intended to use the contract's volume. An upper case 'V' for volume in a Futures' file selects the total volume value created by adding the volume values from all of the active contracts on a record date.	

ASCII Data Files: [^Top](#)

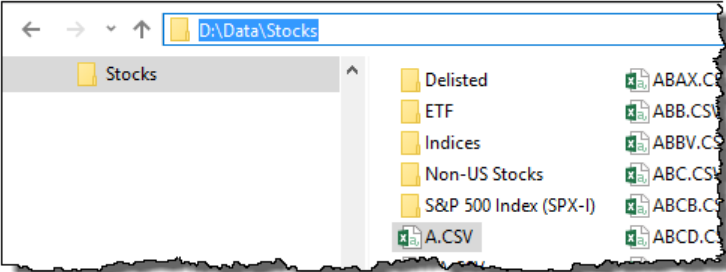
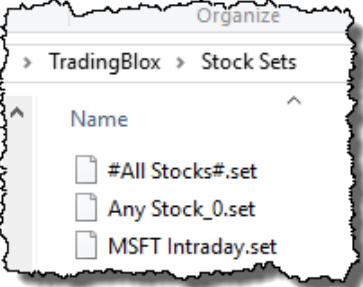
ASCII formatted files must be comma delimited. Dates must be in sequential order where the earliest date information is the first record, and the latest date is the last record in the file.

Daily Data Columns & Price Adjustments:

All fields must be set up as follows. A header record is optional for daily data.

Instrument Type: ^Top	Daily Data Columns
Futures: ^Top	YYYYMMDD, Open, High, Low, Close, Volume, Open Interest, Delivery Month (YYYYMM), Unadjusted Close, Extra Data 1 ... 8 It is recommended that Futures contracts be created using the Back-Adjusted process. Futures Back-Adjustments removes the price gap between expiration months on the date of when the expiring contract is added to the next forward contract. Roll calculations can use the Delivery Month information when that column is added after the Open Interest column. Price fields of Open, High, Low and Close are the prices that get changed during a price adjustment process. When prices are back-adjusted, the date where prices are adjusted will be different than they were on that date when the prices were originally active in the exchange. This means adjusted price values on previous dates after an adjustment will not create the same calculation result unless the price adjustment is removed when that date's adjustment price is applied to a calculation. This reason is why we recommend the data file contain the Unadjusted Close price column. This price column will have the actual price for each date available and it can be used to discover the actual price for each of the four other price fields.
Forex: ^Top	YYYYMMDD, Open, High, Low, Close, Volume, Unadjusted Close, Extra Data 1 ... 8 Forex files don't use price adjustments. All Forex records are assumed to be accurate transaction values for each record date. In addition, Forex files should be updated daily so the current exchange rate information is available during testing.
Stocks: ^Top	YYYYMMDD, Open, High, Low, Close, Volume, Unadjusted Close, Extra Data 1 ... 8 Stock Splits: Most stock files over time experience stock splits. Each Stock split changes the price and the number of outstanding shares. Stocks that have Dividends also affect the value of a share. When a stock is split it will show a gap in price that isn't suitable to computer based calculations. This failure happens because the change in number of shares that changes the price, isn't a change that was caused by an positive or adverse economic view of the company's stock.

Instrument Type: ^Top	Daily Data Columns
	In order for computer based calculations to be kept accurate, and provide transactions that reflect the economic conditions of how a stock's is viewed, stock split calculation should adjust the prices in the file so the change in shares is reflected in the price. This is important because if the change in price is not adjusted in the data, it is likely to cause testing to create an order transaction where it should not happen. False order events like this can be avoided by back-adjusting the change in price on the date the change shares takes place. When prices are back-adjusted, the date where prices are adjusted will be different than they were on that date when the prices were originally active in the exchange. This means adjusted price values on previous dates after an adjustment will not create the same calculation results unless the price adjustment is removed when that date's adjustment price is applied to a calculation. Price adjustments can be removed in testing by including the unadjusted close price in the data file. CSI data provides this price when the file creation settings in their software adds UnAdjustedClose price column to the data file (see the stock price fields recommended above). When a Stocks or Futures file contains the data column UnAdjustedClose, that field value can be used to remove the price adjustment so the calculation result will be the same as it was on the date the exchange created that price record. When it is available in the data, that price in a price adjusted data file can be used to calculate the split ratio and the split ratio direction. By knowing the split ration the basis for adjusting the price becomes available and the transactions that happen during a period where the price had to be adjusted, will be based upon how it would have at that time when there was no price adjustment. Dividends: When a dividend is added to a stock it increases the price of the stock. Trading Blox can use CSI dividend information to build dividend files for all stocks that have dividends. It builds dividend files that are visible in its Portfolio Editor's Instrument Symbol list section. Trading Blox Stock File Search Structure Process: Stocks are the only instruments that can have multiple dictionaries when it comes time to locate stock symbols. This means that when Trading Blox starts, it looks for files using the path settings established in the Preferences Data and Folder section. In each folder and each symbol it finds it will add it to a list of symbols that it will use to determine if it can be in a Stock Dictionary, or when a stock dictionary is not used, it will be displayed in the Portfolio Editor's Instrument Symbol list section.

Instrument Type: ^Top	Daily Data Columns
	In this next set of images, the path is set to D:\Data\Stocks. That path has five sub-folders and a large number of symbol files. All the files in all the folders, plus all the symbols in the Stock Set's portfolio files will be record so they can be used in a Stock Dictionary, when that is the process being created, or as a list of symbols in the Portfolio Editor's Instrument Symbol list section.  Example Stock Folder and Files Structure Example  Stock Set Default Stock Portfolio Example All the files and folders in all the places shown in the example above, which includes the symbols listed in the Stock Set Portfolio files, will be listed in the Instrument Symbol list section when a Stock Dictionary is not used. When a dictionary is being assembled, the same search structure will be used to locate symbols. If you do not have a Stock Dictionary (stockinfo.txt or stockinfo.csv file), it will look for a Metastock MASTER file with that name. If it finds the MASTER file, it will assume Metastock data and use that format process to read the data. If it does not find a MASTER file, it will assume ASCII Text files and look for all the .txt or .csv files in the specified folder, and all the folders contained with the top level Stock folder. Once all the folders have been searched, it will then search all the Stock set portfolio files and collect the symbols in those files. When there isn't a stock dictionary file, and the above symbol information is done its symbol search, Trading Blox will create a list of symbols from the file names and set files and display that list in the Stock option of the Portfolio Editor symbol listing.

Instrument Type: ^Top	Daily Data Columns	
	If a symbol is listed in a Stock file portfolio file, and there is no data file with the same name, that item in the list will be unable to load that symbol name information. Trading Blox flags portfolio symbols that don't have data files when it is first started and creates a warning message in the main screen's Log Window display area when the Log Window is enabled.	
	Portfolio Editor Symbols: ^Top	Description:
	Stock Dictionary Available	Stock dictionaries control which symbols are listed in the Portfolio Editor's Instrument list section. When new symbols are added, and the Stock Dictionary is rebuilt, the new dictionary file will include the newly added symbols. When files are removed, and the dictionary is rebuilt, the deleted symbols will no longer be included in the Instrument Symbol list section. In simple terms, when new symbols are added or removed from the computer, or the Stock folders list of folders and symbols, the dictionary should be rebuilt. Data Source Information: When you have data from multiple sources, or just separate folders for NASDAQ and NYSE, like the courtesy data, Trading Blox will read the stock data folder set in preferences, or in the Suite Parameters, and all sub folders looking for stock dictionaries. There can only be one stock dictionary per folder. Name needs to be StockInfo.csv.
	Stock Dictionary - Not Available	When a stock dictionary is not used, all the symbols in the structure of the Stock folder's list of folders and symbols will be displayed in the Instrument Symbol list section.
	Dividend Files	The stock dividend data from CSI is encrypted by default. To use this feature you must contact CSI and make arrangements to have this data unencrypted for your subscription account. This is an additional cost service from them. Alternately you can build your own dividend files manually from another source for each stock and place in the Dividend Folder. The format is Date,Dividend and you can see example files in the sample data that ships with the product.

Instrument Type: ^Top	Daily Data Columns
NOTE: ^Top All fields after the Date, Open, High, Low, Close (DOHLC) are optional. Files that only contains DOHLC are fine, but some features of Trading Blox will not be effective when the information, like "Delivery Month", or "UnAdjustedClose" are not in the file.	

Adding Data Columns: [^Top](#)

When a need to have more data columns for in a file is required, an additional 8-columns of information can be added to each date record after the file has been updated by your data service.

Trading Blox can provides access to additional data columns. However, access to additional data columns might require additional commas ahead of the extra data so that the information expected by Trading Blox isn't in the wrong place.

Futures information will only access the extra column information after the first 9-columns of specified information is found. Stock and Forex files will only access the extra column information after the first 7-columns of specified information is found. If you are going to add extra columns to a Futures file, but don't include columns for Open Interest and Delivery Month, the file structure will need to look like this:

YYYYMMDD, Open, High, Low, Close, Volume, Open Interest, , , Extra Data 1
... 8

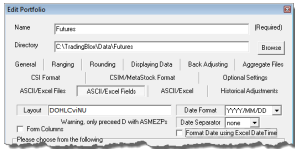
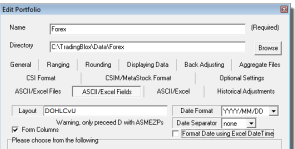
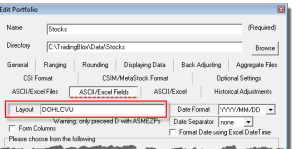
When this requirement is followed, Trading Blox will be able to access data in the extra columns because the Futures record found the first nine commas, and could then read the extra column information into the extra column properties for each date record in the file.

CSI Data Updating Service: [^Top](#)

CSI's UnFairAdvantage Updating service will support all the data type requirements shown.

When an UnFairAdvantage instrument portfolio of symbols is created for downloading each day, the following images of UnFairAdvantage's **ASCII/Excel Fields** will populate the data records with the above Trading Blox data requirements. Less columns can be used when Extra-Data columns are not going to be added after the file updating has completed.

CSI's UnFairAdvantage Portfolio Data Letter Requirements: [^Top](#)

Futures:	Forex:	Stocks:
 CSI's Futures data column letters.	 CSI's Forex data column letters.	 CSI's Stock-Type data column letters.

Intraday Data Columns:

For intraday data, a header must be used with at least the first two items "Date,Time", such as:

Date, Time, Open, High, Low, Close, Volume

Thereafter the format of the data is: [^Top](#)

Futures:

YYYYMMDD, HHMM, Open, High, Low, Close, Volume, Open Interest,
Delivery Month (YYYYMM), Unadjusted Close, Extra Data 1 ... 8

Stocks and Forex:

YYYYMMDD, HHMM, Open, High, Low, Close, Volume, Unadjusted
Close, Extra Data 1 ... 8

Trading Blox accepts the following optional date formats: YYYYMMDD, YYMMDD, MM/DD/YYYY,
MM/DD/YY

Trading Blox accepts the following optional time formats: HHMM, HH:MM

Data Dictionary File Names: [^Top](#)

Stocks: StockInfo.txt

Futures: FuturesInfo.txt

Forex: ForexInfo.txt

See [Data and Folder Preferences](#) to change data's location folders.

[Futures Data Dictionary:](#) [^Top](#)

In order for Trading Blox to read your data, there needs to be a dictionary file for **Futures** and **Forex** data. This Dictionary file can be in the default **Trading Blox Data/Dictionary** folder, or it can be in the same folder as the data files. These Dictionary files contains the market specific symbol information for each symbol. The information in the Futures Dictionary is the data that is used to calculate the net outcome value of each Futures Trade.

Futures Dictionary Editor (C:\TradingBlox\Data\Dictionary)

Symbol	Description	Format	File/Symbol	Exchange	Trading Months	Contract Size	Currency	Big Point	Margin	Close I
AD	Australian Dollar-CME(F...	ASCII	AD_B.TXT	CME	HMUZ	AUD100000	USD	100,000...	5,400.00	ED:SF
AEX	EDE Index(Amsterdam)...	ASCII	AEX_B.TXT	EDE	FGHJKMNQUVXZ	200	EUR	200.00	0.00	
BO	Soybean Oil-CBT (Floor...	ASCII	BO2_B.TXT	CBT	FKKNQUVZ	60000lb	USD	600.00	2,025.00	S:SM
BP	British Pound-CME(Floo...	ASCII	BP_B.TXT	CME	HMUZ	GBP 62500	USD	62,500.00	3,915.00	AD:EC
C	Corn-CBT (Floor+Electr...	ASCII	C2_B.TXT	CBT	HKNUZ	5000 bu	USD	50.00	2,025.00	
CC	Cocoa-CSCE	ASCII	CC_B.TXT	CSCE	HKNUZ	10 tonnes	USD	10.00	2,660.00	
CD	Canadian Dollar-CME(FL...	ASCII	CD_B.TXT	CME	HMUZ	CAD 100000	USD	100,000...	4,455.00	AD:EC
CL	Crude Oil-Light-NYMEX...	ASCII	CL2_B.TXT	NYMEX	FGHJKMNQUVXZ	1000 barrels	USD	1,000.00	8,100.00	HO:HL
CT	Cotton #2-NYCE	ASCII	CT_B.TXT	NYCE	HKNVZ	50000 lbs	USD	500.00	2,520.00	

Trading Blox Futures Symbol Dictionary Tablee

If you don't have a **Futures**, or a **Forex Dictionary** available, Trading Blox will display data error information in **Log Window**, if the Log Window is open, showing no files could be found.

[Stock Dictionary:](#)

If you don't have a Stock Dictionary in your stock data folder, Trading Blox will look for a Master file. If found, it will assume Metastock format, and if not found it will assume Text format. If text format, it will load all the files ending in ".txt" or ".CSV" into Trading Blox. When no Stock Dictionary is used the following columns displayed in a Stock Portfolio display will not show

Stock files with a Dictionary:

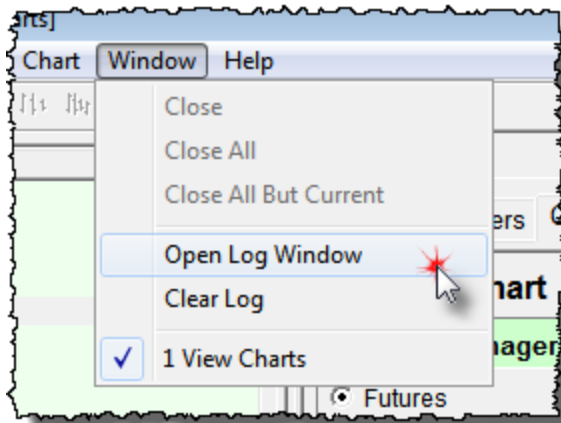
Symbol	Description	Broker ...	Exchange	Sector/Group1	Industry/Group2	File Name	Folder /
☒ AAPL_5902	Apple Inc	AAPL	NASDAQ(GS)	CG	EE	AAPL_5902.T...	Data\Stocks\Stocks100\
☒ ABBV_112378	AbbVie Inc	ABBV	NYSE	HC	DM	ABBV_11237...	Data\Stocks\Stocks100\
☒ ABT_5231	Abbott Laboratories	ABT	NYSE	HC	MA	ABT_5231.TBD	Data\Stocks\Stocks100\
☒ ACN_19525	Accenture Plc	ACN	NYSE	TE	IT	ACN_19525.T...	Data\Stocks\Stocks100\
☒ AIG_5759	American Group	AIG	NYSE	FI	FI	AIG_5759.TBD	Data\Stocks\Stocks100\

Stock Dictionary Data Information**Stock files without a Dictionary:** [^Top](#)

Symbol	Description	Broker ...	Exchange	Sector/Group1	Industry/Group2	File Name	Folder /
☒ AAIT	AAIT	AAIT	Unknown				
☒ ABFS	ABFS	ABFS	Unknown				
☒ ACMP	ACMP	ACMP	Unknown				
☒ ACD	ACD	ACD	Unknown				
☒ AEO	AEO	AEO	Unknown				

Stock Data Information without a Dictionary.

When a Dictionary is listed, but one or more of the symbols listed in the selected portfolio are not listed in the Dictionary, an error record will be displayed in the main screen's **Log Window** area:

**Log Window Menu Controls**

Missing symbol information will show symbols selected in the portfolio, but not found in the dictionary. Any portfolio selected symbols not included in a Dictionary, when a Dictionary is present, will be shown near the top of the portfolio list without any of the usual Dictionary information. Symbols listed at the top that appear like what is shown next, will not be loaded and available during testing:

Interest Rates Files: [^Top](#)

The rates files must be located in the Forex Rates folder, which is in the Data folder. Rate files are required for Forex cost of carry, and for the system wide base currency only, earning interest and paying margin. For example, if the base currency is USD, the **USD_Rates.txt** file will be used for earning interest on excess cash as well as paying margin on stocks on a negative cash balance. The Lending Rate is used when earning interest, and the Borrowing Rate is used for paying margin.

The file can be sparsely populated, and for any given test date, the last available date in the file will be used for the rates. The format for this file is as follows: **Date, Lending Rate, Borrowing Rate**. If only one rate is in the record, it is the Lending Rate, and the Borrowing Rate is the Lending Rate plus .5%. Note that the concepts of Borrowing and Lending are from the consumer's perspective.

All rate files use a name like "**XXX_Rates.txt**" or "**XXX_Rates.csv**". All rate files must be in the Forex Rates folder in the Data folder. The characters "XXX" to the left of "**_Rates.txt**" part in the example above is the ISO currency code required to find the correct rate file so the use of the information for calculation like computing cost of carry will be accurate. Be sure to include all required ISO currency codes in the **ISOCurrencies.csv** file location in the Dictionaries folder.

Example Forex Carry Rate File:

```
Date, Lend, Borrow
20081008,0.015,0.0205
20081029,0.010,0.0155
20081031,0.0045,0.0155
20081216,0.0015,0.0125
```

Column Descriptions:

Date	Lending Rate	Borrowing Rate
20081008	0.015	0.0205
20081029	0.010	0.0155
20081031	0.0045	0.0155
20081216	0.0015	0.0125

Exchange Definition File: [^Top](#)

The exchanges.csv file contains the abbreviation and description of all the exchanges that are used in Trading Blox. A warning will appear if the exchange is used by a market, but missing from this file. To add an exchange to this file simply open the file in excel or notepad, add the exchange and description, and save. Restart Trading Blox.

Or you can press the Examine button next to the Exchange dropdown in the Futures Dictionary. The exchanges are the same for futures and stocks, so this will work for both. The **Exchange.csv** file is located in the Dictionaries folder, as defined in preferences, default is Trading Blox/Data/Dictionaries.

This **Exchange Definition** file is a user maintained file.

Currency Definition File: [^Top](#)

The **ISOCurrency.csv** file is located in the Data/Dictionaries folder and contains all the currencies that can be used by Trading Blox. The file has a version header, column header, and record format is comma delimited.

Version,7

Currency Code,Currency Description

ADF,Andorran Franc

ADP,Andorran Peseta

AED,United Arab Emirates Dirham

AFA,Afghanistan Afghani

ALL,Albanian Lek

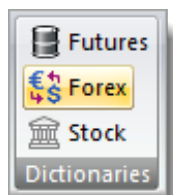
Links:**See Also:**

Edit Time: 6/11/2020 10:17:23 AM

Topic ID#: 155

Section 1 – Forex Dictionary Editor

All traders who wish to trade markets that do not use the same currency as the trader's base currency, will need to have each Forex Currency, Cross-Rate File and a current Quote Rate File that matches the base currency to convert prices to the trader's base currency.



Forex Dictionary Editor

Access to the Forex Dictionary Editor is located on the Main Screen's Dictionary Section where the [Forex Editor](#) button is displayed.

When Trading Blox is installed, it loads a group of Forex, and Quote Rate files. The currency files loaded need to be updated daily to ensure accurate performance results. The Quote Rate files change less frequently, but the current rate change news is a good reminder to see if new records can be added to those files.

You can access the Forex Dictionary editor by selecting the [Home menu's Dictionary](#) section [Forex](#) button.

, will need to keep the Forex Currency, the Cross-Rate, and the Quote Rate files current.

You can access the Futures Dictionary editor by clicking on the [Futures Dictionary](#) button displayed in the [Dictionary](#) section of the [Home](#) menu:

See [Data and Folder Preferences](#) to change the default data folder from the sample data to your own data source. When changing your data source, be sure to update the [Forex Dictionary](#) folder paths, and move the interest rates files to the new location.

To add a new market, press the New Market button, enter the symbol. For example, to add the Korean Won to the Forex Dictionary starts with selecting the symbol "QF3" in **CSI's Unfair Advantage** software. **QF3** is the **CSI** symbol for the **Korean Won** Cash file. Add the symbol **KRWUSD** in the Forex Editor's Symbol field. This order of characters is required to trade any KRW Futures or Stocks. Then select the CSI File Name to add the data file the **KRWUSD** symbol pair.

Tim Arnold:

The file names don't matter as much as the symbol name. Be sure the forex symbols are XXXYYY (EURUSD) format. In this way Trading Blox knows what the file represents. If your system wide base currency is USD, and you want to trade CADJPY, then make sure you have a CADUSD or JPYUSD symbol included in your forex dictionary, with a linked data file.

I typically include all the daily files as EURUSD format symbol, so TB can find them and use them for cross rates. Then if you want to add intraday symbols to use in a portfolio, you can define the symbol as EURUSD-1H or something like that, with the intraday time frame as the post-fix. It'

generally more accurate to have the daily files for converting profit back to USD, because that is likely how the broker would convert back as well.

Symbol	Description	Format	File/Symbol	Tick Size	Min Tick
AUDUSD	Australian Dollar / U.S. Dollar	ASCII	AD60000\$.CSV	0.0001	
BRLUSD	Brazilian Real / U.S. Dollar	ASCII	US50000\$.CSV	0.0001	
DKKUSD	Danish Krone / US Dollar	ASCII	QE40000\$.CSV	0.0001	
EURUSD	Euro / U.S. Dollar	ASCII	EU90000\$.CSV	0.0001	
GBPUSD	British Pound / U.S. Dollar	ASCII	QF40000\$.CSV	0.0001	
MXNUSD	Mexican Peso / US Dollar	ASCII	QF50000\$.CSV	0.0001	
NOKUSD	Norwegian Kroner / US Dollar	ASCII	QE#0000\$.CSV	0.0001	
RUBUSD	Russian Ruble / U.S. Dollar	ASCII	USE0000\$.CSV	0.0001	
USDBRL	U.S. Dollar / Brazilian Real	ASCII	US50000\$.CSV	0.0001	
USDCAD	U.S. Dollar / Canadian Dollar	ASCII	QE20000\$.CSV	0.0001	
USDCHE	U.S. Dollar / Swiss Franc	ASCII	QE10000\$.CSV	0.0001	
USDCNY	U.S. Dollar / Chinese Yuan	ASCII	US60000\$.CSV	0.0001	
USDHKD	U.S. Dollar / Hong Kong Dollar	ASCII	QE70000\$.CSV	0.0001	
USDJPY	U.S. Dollar / Japanese Yen	ASCII	QE90000\$.CSV	0.01	

Selected Market Specifications

Format: Tick Unit:

Symbol: Min Tick:

Broker: Spread:

Description: Commission:

File Name:

Folder:

Controls

Forex Dictionary Editor

Parameter:	Description:
Symbol: ^Top	The symbol for the market. These symbols should be composed of six letter combinations of ISO Standard Currency Codes with the first three letters as the ISO code for the base currency, and the last three as the ISO code for the quote currency. The symbol can include additional characters, such as AUDCAD-5 to represent a 5 minute bar version of the symbol.
Description: ^Top	The name of the market longhand. When creating a new market Trading Blox will fill in this field automatically if the six letter symbol is constructed of valid ISO codes.
Data Format: ^Top	The data format: ASCII or MetaStock.
ASCII File: ^Top	For ASCII text format, the name of the file must be input here.
Metastock Symbol: ^Top	For MetaStock this should be the symbol, which is used to extract information from the MetaStock MASTER file.
File Name: ^Top	The name of the file where the data for this market is located. Press the File Name button to locate the file using a browser interface.

Parameter:	Description:
Folder: ^Top	The folder where the file is located. The "@" represents the default location as set in the Forex Data Folder in preferences. Press the Folder button to locate this folder using a browser interface.
Examine: ^Top	Press this button to open the data file in notepad or excel for editing or viewing.
Examine Base Rates: ^Top	Press this button to open the base currency interest rates file for editing or viewing. The rates file is located in the Forex Data Folder, and has an ISO_Rates.txt format. Example USD_Rates.txt for the USD rates.
Examine Quote Rates: ^Top	Press this button to open the quote currency interest rates file for editing or viewing. The rates file is located in the Forex Data Folder, and has an ISO_Rates.txt format.
Pip Size: ^Top	The minimum trading increment or PIP. Usually 0.01 or 0.0001.
Spread in Pips: ^Top	The spread used for both entry and exit slippage. This value is designed to support simulation of retail Forex brokerage slippage which is generally a fixed number of pips. The exact value depends on the broker and the market.
Reverse Conversion: ^Top	This box, if checked, will reverse the currency conversion for use with non-USD denominated futures, stocks, and other currency conversions.
Find Symbol: ^Top	Used primarily to find Metastock file names. In the example above, the Metastock symbol is AUDCAD, and the Find Symbol button will determine from the master file that the Metastock file name is F1.dat.
Tutorials: ^Top	
https://www.tradingblox.com/tradingblox/howto-ua.htm	
https://www.tradingblox.com/tradingblox/howto-ua-newmarket.htm	
https://www.tradingblox.com/tradingblox/CSIUA/howtouastocks.doc	
https://www.tradingblox.com/tradingblox/CSIUA/Trading%20Blox%20CSI%20Forex.doc	

Links:**See Also:**

1.1 Forex Editing Details

Enter topic text here.

Links:
See Also:

Edit Time: 6/10/2020 11:27:39 AM

Topic ID#: 250

1.2 Interest Rate Files

Interest rates files are most often sparsely populated. The time between any given testing date Trading Blox will encounter is to use the most recent previous rate-date for Forex Carry, Interest Earned, and Margin Paid purposes.

The data in these interest rate files is dependent upon the user to update these files manually and as new interest rate records become available. To get the best value to use, it is best to get the carry cost data from your **Forex Broker**, or a known **Forex Broker** like [Oanda - Interest Rates](#).

In the earlier portion of the **Interest Rate Files**, there are many records where there is only one interest rate for each of the very early date records. Those single interest rate value are a **Lend Rate**. The **Borrow Rate** can then be computed by adding 0.5% to the **Lend Rate**. This calculation method is included for historical reasons.

The best approach is to use two rates so that the **Lend Rate** and **Borrow Rates** are in the interest rate files. The first column is the **Lend Rate** and the second column is the **Borrow Rate**. These are the rates are used for **Forex Carry** costs.

In addition, the system wide currency rate files, **USD_Rates.txt** in most cases, is used for **Interest Earned** and **Margin Paid**. The **Lend Rate** is the **Interest Earned** and the **Borrow Rate** is the **Margin Paid**, for any given date.

Links:
See Also:

1.3 Forex Symbol List

Links:
See Also:

Edit Time: 9/12/2020 2:20:21 PM

Topic ID#: 249

Section 2 – Futures Dictionary Editor

The Futures Dictionary stores each contract's unique information.

Each Future symbol must have a complete record listed in the Futures Dictionary. Once the information is entered into the dictionary, the portfolio process will have the ability to display that symbol's information. Any Future Contract record in the portfolio list that is complete can be used for testing as long the information list is current.



**Futures Dictionary
Button**

You can access the Futures Dictionary editor by clicking on the [Futures Dictionary](#) button displayed in the [Dictionary](#) section of the [Home](#) menu:

Futures Dictionary Editor (D:\Data\Dictionary)

Symbol	Description	Format	File/Symbol	Exchange	Trading Months	Contract Size	Currency	Big Point	Margin
AD	Australian Dollar(Flo...	ASCII	AD_B.CSV	CME	HMUZ	100000	USD	100,000...	2,013.00
AEX	EOE Index(Amsterda...	ASCII	AEX_B.CSV	EOE	FGHJKMNQVXZ	200	EUR	200.00	3,400.00
BAX	Canadian Bankers A...	ASCII	BAX_B.CSV	ME	HMUZ	1000000	CAD	2,500.00	698.00
BL2	Wheat-Milling(MATI...	ASCII	BL2_B.CSV	EURON...	FHKUXZ	50	EUR	50.00	813.00
BO2	Soybean Oil-CBT (FL...	ASCII	BO2_B.CSV	CBT	FHKNUQVZ	60000	USD	600.00	2,025.00
BP	British Pound(Floor+...	ASCII	BP_B.CSV	CME	HMUZ	62500	USD	62,500.00	1,997.00
BR5	Brazilian Real-CME ...	ASCII	BR5_B.CSV	CME	FGHJKMNQVXZ	100000	USD	100,000...	10,750.00
C2	Corn-CBT (Floor+Ele...	ASCII	C2_B.CSV	CBT	HKNUZ	5000	USD	50.00	2,362.00
CC2	Cocoa-CSCE(Floor+...	ASCII	CC2_B.CSV	CSCE	HKNUZ	10	USD	10.00	1,125.00

Selected Market Specifications

Symbol: Contract Size:
 Description: Big Point Value:
 Broker Symbol: Broker BPV:
 Format: ☒ ASCII Text ☐ Metastock ☐ IB Margin:
 Display Digits:
 File Name: Tick Unit:
 Folder: Minimum Tick:
 Exchange: Round Lot:
 Trading Months: Order Sort Value:
 Currency: Group 1:
 Group 2:
 Closely Correlated: Custom 1:
 Loosely Correlated: Custom 2:

Controls

Futures Dictionary Editor - Click the Area to View Information

The Futures Dictionary Editor has been updated. Consider reviewing the details of each of the four sections. When you click on a section, that section's topic will appear with details about each item. For example, this version of the Futures Dictionary has an ability to synchronize all the symbol records in the dictionary by using the "[Synch UA All](#)" button. This button will synchronize all the symbols so they match the current data requirements. The "[Synch UA](#)" button is still available to synchronize individual symbol records.

When the Futures Editor is displayed, it is easy to add, remove or change the information for any future symbol. With all information needed for each record accessible, it will make adding, changing an existing record, or removing a symbol record that is no longer needed.

Example for Adding and Aditing a Symbol: [^Top](#)

Here is an example of adding a Foreign Market, such as **KOS**. The **Kospi** is denominated in **KRW**, so the **USDKRW** is needed as a Forex rate file. Before adding the **KOS** symbol, check the [Forex Dictionary](#) to see if the **USDKRW** and linked to the appropriate data file to convert the **KRW** back to your system wide base currency.

When the **USDKRW** cross-rate file is in the [Forex Dictionary](#), adding the **KOS** symbol will enable Trading Blox to find the the the cross-rate file to use for its calculations.

Forum Tutorials:

<https://www.tradingblox.com/tradingblox/howto-ua.htm>

<https://www.tradingblox.com/tradingblox/howto-ua-newmarket.htm>

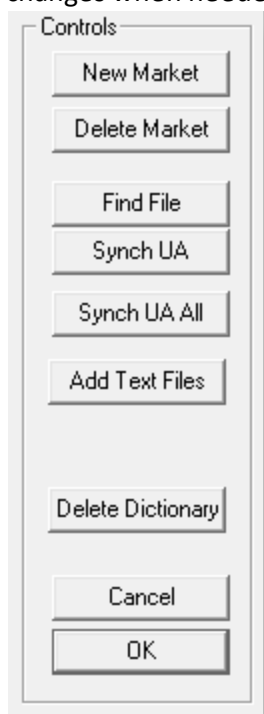
<https://www.tradingblox.com/tradingblox/CSIUA/howtouastocks.doc>

<https://www.tradingblox.com/tradingblox/CSIUA/Trading%20Blox%20CSI%20Forex.doc>

Links:
Historical Data Setup , Futures Dictionary Editor , Futures Market Specifications , Futures Symbol Data
See Also:

2.1 Dictionary Controls

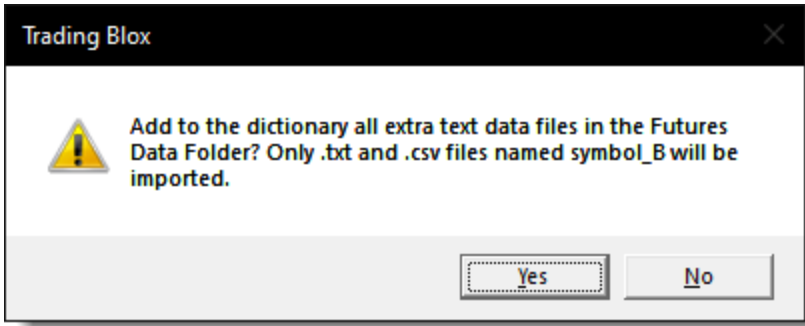
These controls enable the Futures Dictionary to add, remove, update and make Future Symbol changes when needed.



**Futures Dictionary Editor
Controls**

Click on a button display to read the actions of the selected button.

Control Button:	Description:
New Market: ^Top	Add a new market to the Futures Dictionary.
Delete Market: ^Top	Removes a market from the Futures Dictionary.
Find File: ^Top	Finds a market in the Futures Dictionary.
Synch UA: ^Top	Updates the data in Unfair Advantage into the selected symbol in the Dictionary. Review the Synch Futures Dictionary details shown here.
Synch UA ALL: ^Top	Updates all the data in Unfair Advantage into each record in the Dictionary.
Add Text Files: ^Top	When you click this button, the following dialog will appear:

Control Button:	Description:
	 Add Text Files to the Futures Dictionary If you click on the Yes button, Trading Blox will add to the dictionary all extra text data files in the Futures Data Folder. The text file it will add are limited to those with a ".txt" and ".csv" file suffix. The named files that use the symbol_B format will be imported (i.e. CS_B.CSV) . The additional information would be that this imports all text files in the folder, sub-folders and short cut folders, assuming the symbol does not exists in the dictionary. Files in the dictionary will be skipped. The symbol_B formatted files are Future files that show the market symbol, and then append an "_B" when it creates the market's symbol name. CSI allows this option so the file names do not change regardless of the back adjusting method that is being used. So the idea is that if a new market is added to the CSI download destination portfolio, it is easy to add that to new Futures' file to the Futures dictionary. When a new Futures symbol is added to the dictionary, update the data for that symbol use the Synch UA button shown in the control panel above. By updating the trading information to the dictionary, the added attributes that are missing, will provide the information that Trading Blox will need during testing and signal generation. Review the Synch Futures Dictionary details shown here.
Delete Dictionary: ^Top	Removes the Dictionary from its current location.
Cancel: ^Top	Closes the Futures Dictionary without saving any changes.
OK: ^Top	Save the changes in the Futures Dictionary an closes the editor.

Links:**See Also:**

Edit Time: 9/12/2020 2:20:57 PM

Topic ID#: 247

2.2 Market Specifications

The editor uses the Market Specification controls to select an instrument in the current Futures Dictionary to review and edit. When a symbol is entered and displayed in the "[Symbol](#)" field, the contract's exchange values are displayed in the "**Futures Market Symbol Values**" details display. Selected markets display their specification details in the [Symbol Specification Table](#).

Market Specification Editing & [Futures Symbol Specification Table](#):

Selected Market Specifications

Symbol:

Description:

Broker Symbol:

Format: ☒ ASCII Text ☐ Metastock ☐ IB

File Name:

Folder:

Exchange:

Trading Months:

Currency:

Closely Correlated:

Loosely Correlated:

Futures Market Specifications

Editor Item Name:	Description:
Symbol: ^Top	The symbol for the market. The symbol in a futures portfolio must match this symbol.
Description: ^Top	The name of the market longhand.

Editor Item Name:	Description:
Broker Symbol: ^Top	The symbol used by your broker for this market.
Data Format: ^Top	The data format: ASCII or MetaStock.
File Name: ^Top File Name	This is the actual file name of the data file that will be linked to this symbol. Press the File Name button to locate the file using a browser interface. Examine Press this button to open the data file in notepad or excel, for editing or viewing.
Folder Location: ^Top Folder	This is the actual folder location of the data file. Press on the Folder Location button to locate the folder using a browser interface. The "@" sign is used to represent the default data folder as set in preferences. In this way the Dictionary is transportable to another installation of Trading Blox. If you hard code the folder location using C:/xxx then it will not be transportable, but you can specify any location even outside of the Trading Blox folder or the default data folder as set in preferences. Examine Press this button to open the exchanges file in notepad or excel, for editing or viewing.
Exchange: ^Top	The exchange that the market trades on. Examine Press this button to open the exchanges file in notepad or excel, for editing or viewing.
Trading Months: ^Top	The months that the market trades using standard futures month letters. This is used to determine how many times per year account for contract rolls, when you have the Account for Contract Rolls global parameter set to true. The more months you have here, the more times the system will simulate a roll, and account for commission and slippage. Note that if you have the delivery month in your data, then this is not used at all. But if you don't have the delivery month in your data, and you want to account for contract rolls, a roll we be assumed every x bars where x is 250 divided by the number of months entered here. Good for daily data, but not for intraday or weekly data.

Contract Size	100000
Big Point Value	100,000.00
Broker BPV	0.00
Margin	2,013.00
Display Digits	4
Tick Unit	0.0001
Minimum Tick	1
Round Lot	1
Order Sort Value	0
Group 1	Currency
Group 2	Currency_G2
Custom 1	
Custom 2	

Futures Market Selected Symbol Values

Editor Item Name:	Description:
Contract Size: ^Top	
Big Point Value: ^Top	The value of a 1.0 price movement (see Determining Big Point Value for a detailed explanation). The Big Point Value should be expressed in the currency of the underlying contract.
Broker BPV: ^Top	Brokers BPV, when it is different from the published BPV.
Margin: ^Top	The margin used to simulate margin calls while testing.
Display Digits: ^Top	The number of significant decimal digits displayed for this market.
Tick Unit: ^Top	The unit of the tick. This is either a decimal like 1.0, 0.1, 0.01, etc. or a fraction like 1/4, 1/8, 1/32 etc.
Minimum Tick: ^Top	The number of tick units which constitutes a minimum tick. For example, if the minimum tick is 0.025 or twenty five thousandths, the tick unit is thousandths, 0.001 and the minimum tick value is 25.
Round Lot: ^Top	The round lot for trading. If you enter a value of 100 here, Trading Blox will only trade in increments of 100.
Order Sort Value: ^Top	This value determines the sort order of the markets for order generation. The sample data sets these values to a number corresponding to the market open time using U.S. Eastern Standard Time. This results in order files that are in the same order as the market open. For example, the 820 cited above for AD -

Editor Item Name:	Description:
	Australian Dollars corresponds to the 8:20 AM market opening of the AD on the Chicago Mercantile Exchange.
Group 1: ^Top	This string is the group1 to which the market belongs. It can be any string or number, up to 32 characters. When more than one market have the same group, you can access certain instrument properties in scripting such as group margin and group risk, etc.
Group 2: ^Top	This string is the group2 to which the market belongs. It can be any string or number, up to 32 characters. When more than one market have the same group, you can access certain instrument properties in scripting such as group margin and group risk, etc.

Links:
See Also:

Edit Time: 9/12/2020 2:21:47 PM

Topic ID#: 246

2.3 Symbol Data Details

The Futures Dictionary Editor displays a table of all the Futures Symbol loaded into the Futures' Dictionary.

Futures Dictionary Editor (D:\Data\Dictionary)

Symbol	Description	Format	File/Symbol	Exchange	Trading Months	Contract Size	Currency	Big Point	Margin	(^
AD	Australian Dollar-(Flo...	ASCII	AD_B.CSV	CME	HMUZ	100000	USD	100,000...	2,013.00	
AEX	EOE Index(Amsterda...	ASCII	AEX_B.CSV	EOE	FGHJKMNQUVXZ	200	EUR	200.00	3,400.00	
BAX	Canadian Bankers A...	ASCII	BAX_B.CSV	ME	HMUZ	1000000	CAD	2,500.00	688.00	
BL2	Wheat-Milling-(MATI...	ASCII	BL2_B.CSV	EURON...	FHKUXZ	50	EUR	50.00	813.00	
B02	Soybean Oil-CBT (Fl...	ASCII	B02_B.CSV	CBT	FHKNUVZ	60000	USD	600.00	2,025.00	
BP	British Pound-(Floor...	ASCII	BP_B.CSV	CME	HMUZ	62500	USD	62,500.00	1,997.00	
BR5	Brazilian Real-CME ...	ASCII	BR5_B.CSV	CME	FGHJKMNQUVXZ	100000	USD	100,000...	10,750.00	
C2	Corn-CBT (Floor+Ele...	ASCII	C2_B.CSV	CBT	HKNUZ	5000	USD	50.00	2,362.00	
CC2	Cocoa-CSCE(Floor+...	ASCII	CC2_B.CSV	CSCE	HKNUZ	10	USD	10.00	1,125.00	

Futures Data Table Records

Futures Dictionary Data Information:

Each future symbol for each symbol can have information for each of the data name items listed in this table.

	A	B	C	D	E	F	G	H	I
						Trading	Contract		
1	Symbol	Description	Format	File/Symbol	Exchange	Months	Size	Currency	Big Point

J	K	L	M	N	O	P	Q	R
	Close	Loose				Minimum	Order	
Margin	Correlations	Correlations	Decimals	Tick	Unit	Tick	Sort	Group1

S	T	U	V	W	X	Y	Z	AA
	Broker		Round				Broker	Bar
Group2	Symbol	Folder	Lot	Metastock Symbol	Custom 1	Custom 2	BFV	Size

Futures Dictionary Data Column Names

In some instances where the data source being used doesn't have information for a column, that field of information will be left blank.

CSI's Backadjust Settings: [^Top](#)

When a symbol is added to the dictionary, the information for every symbol is available in Unfair Advantage (UA) Market Specification table. To add a symbol to the portfolio name you selected, click on the **Add-Symbol** menu option. When dialog shown next appears, type in a symbol's letters and soon a UA will respond with its record number:

Futures Dictionary Data Information:

Select Data Series - U.S. T-Bond (combined)-CBT

Market ☒ Futures ☐ Stock ☐ Other

Symbol **US [#44]** Market Specs

Futures NFG Perpetual Nth Nearest Back-Adjust Gann Cash/Econ Option

Back-Adjust pricing when contracts roll, based on roll rules, eliminating gaps. **US_0@EAB (US_B) ?**

Rolling Criteria:

☐ Open Interest ☐ Roll By Percent 100

☐ Volume ☐ Min Contracts Out 1

☐ OI and Volume ☐ Max Contracts Out 4

☒ OI or Volume ☐ Roll By Trading Days Before Expiry 0

☐ Date (Calendar Days) ☐ Use Trading Days

12 Days From Start of the Month

1 Months Prior

Expiry Dates

202006	20200619
202009	20200921
202012	20201221

☐ Apply Rule

Group Rule:

No rule specified

Contract Months To Include

		X			X			X			X
F	G	H	J	K	M	N	Q	U	V	X	Z

Roll Timing

When Known (Roll the Next Day)

Representative Prices

Close New Close Old Same Day Roll Day Adjust

Accumulation Method

Back Adjusted

Confirmation Signals

Roll On Second Consecutive Trigger

☒ Use Advanced Adjuster

☒ Use Walk Forward Detection

☐ Proportional / Ratio Adjust Roll Pricing

Dates

From As Available

To As Available

File Export Preferences

Portfolio

Futures New

File Format

☒ ASCII

☐ Excel ☐ PRO

☐ CSI ☐ CSIM

☐ MS Format

Period

☒ D ☐ W ☐ M

☐ Q ☐ S ☐ Y

☐ Build Chart @ Startup

Post Processing

☐ Detrend 2nd Contract

☐ Invert

☐ Indicator

CSI's UA-Add/Edit Futures Data Symbol

CSI Futures File Naming: [^Top](#)

Futures Dictionary Data Information:

Edit Portfolio

Name:

Directory:

Active Stock Portfolio

☐ Manage Activity Help

☐ Daily ☒ Monthly

☒ Remove Delisted Files

General | Data Display / Rounding / Ranging | Historical Adjustments

Aggregate Data | **Back Adjusting** | CSI Format | CSIM/MetaStock Format

ASCII/Excel Files | ASCII/Excel Fields | ASCII/Excel | Other Settings

Warning, these settings only apply to actual futures contracts. Stocks and synthetic series are named according to the symbol for that series.

The file name is assembled out of the fields specified. For example, SCYA means Symbol (S) followed by Century (C) followed by Year (Y) followed by Delivery Month (A).

R must be Primary when used and is only applicable w/ Active Stock Ranked Portfolios

Name By: ☐ Omit C and Y on Cash/Forwards

Please choose from the following

S for Symbol	C for Century (19 or 20)	. for Extension
E for Exchange	Y for Year (i.e. 92)	_ for Separator
Z for Name	N for Delivery Month Number	
# for CSI number	A for Delivery Month Letter	
P for Period (DWM)	K for Strike	
R for Rank	F for Fixed 12 Char Symbol	
s for Sector	M for MetaStock Short Name	

Extension

☒ .PRN

☐ .TXT

☒ .CSV

☒ Use generic filename override for continuous series (Symbol + _B, _N, _P, _G)

☐ Use the CSI number for the symbol when writing stock files

☐ Export Futures into Symbolized Directories (ensure uniqueness by using #_S file name fields)

☐ Make TradeStation Compatible by creating DOPs w/ an Attributes.ini for Description Details

☐ Make TradingBlox Compatible by creating a StockInfo file for Description Details Resort

☐ Restrict Filenames to Standard ASCII characters

OK Cancel

Futures Data File Name Format

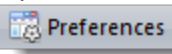
Synch Dictionary Futures: [^Top](#)

For users of **CSI's Futures Data**, the information can be added automatically once the symbol is entered into the Symbol Field shown in the above image.

After the symbol is entered into the symbol field, and with Unfair Advantage installed, click on

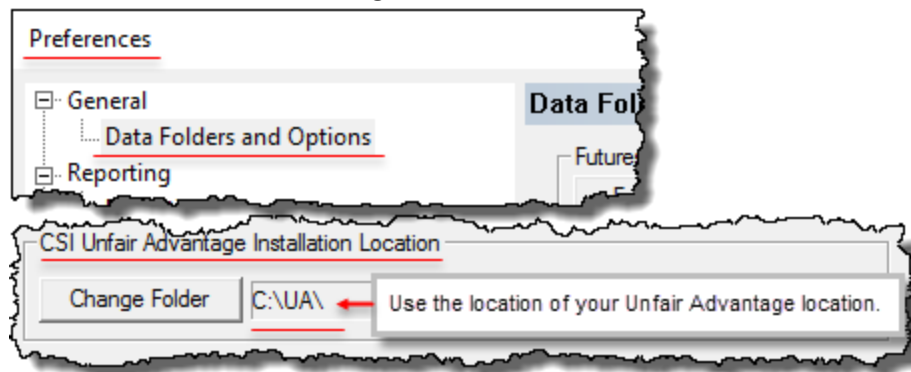
Synch UA

the button and Trading Blox will fill in the fields for that symbol. If the information doesn't get filled in, and you know the symbol is available from CSI, check the

Unfair Advantage path location in the Trading Blox  Preferences section. When the

Futures Dictionary Data Information:

Preferences dialog appears, be sure the information listed in this area is the location where your installation of Unfair Advantage is installed:



Should symbol listed in the portfolio appear in the main screen "Log Window", the most common problem is the file name, or the file is missing.

Links:**See Also:**

Section 3 – Stock Dictionary

The Stock Dictionary [does not have a graphical editor](#) like Futures Dictionary and Forex Dictionary because the data format for stocks is simpler and more consistent across different data vendors. Instead, the Stock Dictionary is defined using an ASCII Text file called **StockInfo.txt** which is located in the [Dictionary Folder](#). Alternately the file can be located in the Stock Data Folder. Trading Blox will look in the Stock Data Folder first, and if the file is not found, it will look in the Dictionary folder.

Before you create your Stock Portfolio in CSI's Unfair Advantage, review the [CSI Symbol Num](#) information.

This file is optional. If you don't have a stockinfo.txt file present in the Stock Data Folder or the Dictionaries Folder Trading Blox will look for all Metastock files and all text files ending in ".txt" or ".csv". It will take these files and create symbols. A file named "IBM.txt" will create a symbol called **IBM** with a name of **IBM** and read the data.

Trading Blox will look for all data files in all linked folders, and make them available for trading. It will then look for the Stock Dictionary file and append any additional information. So you can have a large number of stock files, and a small dictionary with extra data. This also means you can add industries and groups to Metastock data.

Important Note:

If you have a Stock Dictionary, then only those stocks in the dictionary will be available for testing. If you add new data files to your data folders they will not be available for testing until you have added the stock to the Stock Dictionary. If you are using CSI this process is easy, in that you can use the Create Stock Dictionary button in preferences. If you don't need the description, sector, industry, and round lot, etc then you can delete the Stock Dictionary (**stockinfo.txt**) file. Trading Blox will then load up all text files for processing.

Format:

The format of the stockinfo.txt file is:

Symbol,Description,Exchange,Sector/Group1,Industry/Group2,Country,Data Vendor ID,Broker Symbol,Active Status,Round Lot,Currency

If you want to use a header, be sure to indicate the file is version 6 or greater on the first line. If this version information is missing, or less than version 6, Trading Blox will not accept the header.

Example:

Version, 6

Symbol,Description,Exchange,Sector,Industry,Country,Data Vendor ID,Broker Symbol,Active Status,Round Lot,Currency
GV,Goldfield Cp,AMEX,IG,HC,USA,3243,GV,A,1,USD
CGV,CGG Veritas,NYSE,BM,OE,USA,11512,CGV,A,1,USD

For each of the instruments listed, there needs to be a corresponding data file. See [Historical Data Setup](#) for more information.

Volume Multiplier:

Be sure to set the Volume Multiplier in the [Preferences' Editor](#). So if your stock data reports a volume of 1,000, we can translate that into 100,000. This is the way CSI reports the data. If your data provider uses some other multiplier, or reports the actual data, you would want to use a different setting in preferences. The default Volume Multiplier = 100.

The volume is used by Trading Blox to check Minimum Volume and Maximum Percent Volume Per Trade, both global variables.

Currency Conversion:

Trading Blox will look first to the Stock Dictionary for the currency of the stock. If there is no currency it will use the system wide base currency. Be sure to have the corresponding forex pair available in the forex data folder when doing currency conversions.

Create a Stock Dictionary using CSI UA Details:

Set the correct folder location of the CSI installation in the Trading Blox Preferences

Data Folders and Options

Futures Data Folder
Futures Data | Data\Futures\

Stock Data Folder
Stock Data | Data\Stocks\
Dividend Folder | Data\Dividends\

Forex Data Folder
Forex Data | Data\Forex\

Data Dictionaries and Utility Files
Dictionaries | Data\Dictionaryes\

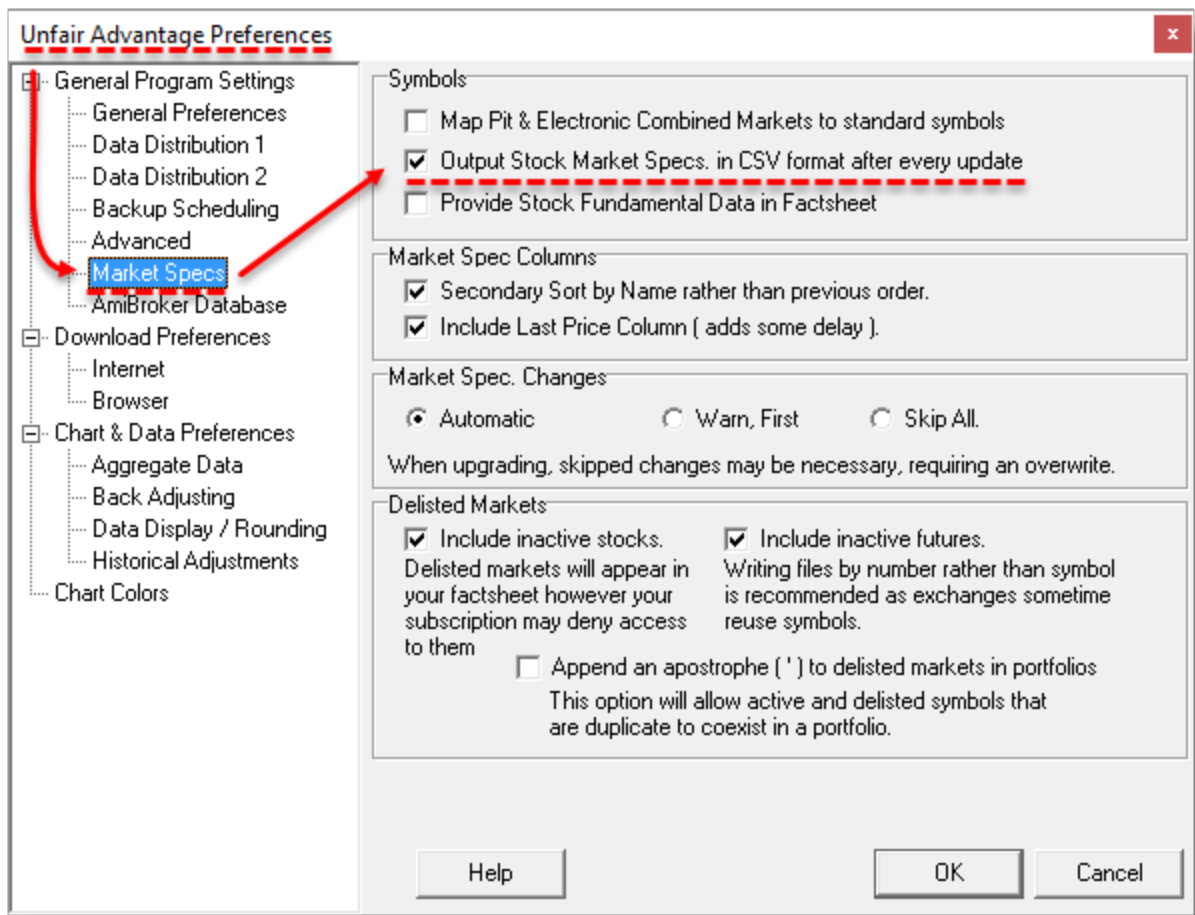
Orders Folder
Orders | Orders\

CSI Unfair Advantage Installation Location
Change Folder | C:\UA\

Dictionary building is dependent upon being able to find CSI's UA file: sdbfacts.csv

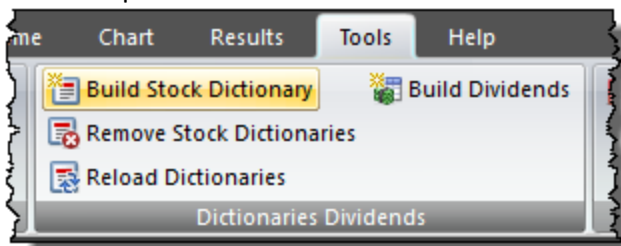
Preferences Data Folder and Options - CSI UA Path Setting

CSI UA Download Market Data file:



Then press the "Get Data" button in CSI UA to update your data and download this file.

Now Trading Blox will be able to build a stock dictionary file from the CSI market data using this button in preferences:



When this button is used, Trading Blox will build the Stock Dictionary. When used with CSI's Stock Data service, it will included the sector, industry, and description for CSI data. There isn't an option to build a Stock Dictionary with other data services.

When the **Build Stock Dictionary** process is running, it will delete your old stock dictionary and build a new one. It will build the file for all data files in your data folder. If you add new data files into your data folder later, you will need to create a new stock dictionary. Only stocks in the stock

dictionary will be available for testing. If you add a new data file later, it will not be available for testing until added to the stock dictionary, or the stock dictionary is removed.

Trading Blox will allow access to all the stocks in the Stock Data folder when the Stock Dictionary is removed. However, it will not provide information about the sector and industry.

Trader's Round Table Forum Tutorials:

<https://www.tradingblox.com/tradingblox/howto-ua.htm>

<https://www.tradingblox.com/tradingblox/howto-ua-newmarket.htm>

<https://www.tradingblox.com/tradingblox/CSIUA/howtouastocks.doc>

<https://www.tradingblox.com/tradingblox/CSIUA/Trading%20Blox%20CSI%20Forex.doc>

Links:
See Also:

Edit Time: 9/12/2020 2:23:00 PM

Topic ID#: 5030

3.1 CSI_Stock-Symbol_Num

It is not uncommon to find the same Stock symbol for two or more companies. This happens when the symbol from the first company with the symbol is retired after an acquisition, or the termination of business. Their Stock Market data is not deleted. Instead it is usually available using the same symbol. CSI data supports the same symbol access, but it also provides a CSI-Number that can be appended to the Symbol so a user can access a specific company using the same symbol Plus the CSI-Number as a more accurate data access history.

When a second company wants to go public and request the same symbol that was assigned to a previous company that is no longer active on the exchange, the second company's request in most cases will be granted to use the same symbol. The previous company's data history can be used with the later company using the same symbol, so an additional factor is needed to ensure the history returned is for the company needed.

Same Stock Symbol + Number Example:

The stock symbol ABI has belonged to three different companies, and all three are now delisted.

Name	Date modified	Type	Size
 ABI'_8075.CSV	3/24/2020 1:55 PM	Microsoft Excel Comma ...	285 KB
 ABI'_19145.CSV	3/24/2020 1:56 PM	Microsoft Excel Comma ...	190 KB
 ABI'_108003.CSV	3/24/2020 1:54 PM	Microsoft Excel Comma ...	95 KB

Same Symbol. different record numbers and different companies

ABI American Bankers Insurance Group #8075

ABI Applera Corp #19145

ABI Safety First Trust Principal-Protected #108003

In the three same symbol example above, each ABI symbol has an a Single-Quote Symbol as its last character. Symbols with a single quote mark are for delisted symbols. This means if the ABI symbol were to be used by a fourth company, and it was actively trading, the single quote mark would not be appended to the symbol.

To ensure the data you have in your stock portfolio is the company name you expect. For CSI's Data Service using Unfair Advantage, this means use the symbol with the CSI symbol number to identify the specific company that is using the symbol you add to you list of stocks. Adding the CSI Number to stock symbols ensures the company you selected will be in your portfolio and it will be updated with the correct market data while the company is still operating.

Edit Portfolio

Name:

Directory:

Stock Index Components:

☐ Manage Monthly

☒ Delete Dropped Symbol Files

General | Data Display / Rounding / Ranging | Historical Adjustments

Aggregate Data | Back Adjusting | CSI Format | CSIM/MetaStock Format

ASCII/Excel Files | ASCII/Excel Fields | ASCII/Excel | Other Settings

Warning, these settings only apply to actual futures contracts. Stocks and synthetic series are named according to the symbol for that series.

The file name is assembled out of the fields specified. For example, SCYA means Symbol (S) followed by Century (C) followed by Year (Y) followed by Delivery Month (A).

Name By: ☐ Omit C and Y on Cash/Forwards

Please choose from the following

<u>S</u> for Symbol	C for Century (19 or 20)
E for Exchange	Y for Year (i.e. 92)
Z for Name	N for Delivery Month Number
<u>#</u> for CSI number	A for Delivery Month Letter
P for Period (DWM)	F for Fixed 12 Char Symbol
K for Strike	M for MetaStock Short Name
_ for Separator	. for Extension

Extension

☐ .PRN

☐ .TXT

☒ .CSV

☒ Use generic filename override for continuous series (Symbol + _B, _N, _P, _G)

☐ Use the CSI number for the symbol when writing stock files.

☐ Export Futures into Symbolized Directories (ensure uniqueness by using #_S file name fields)

☐ Make TradeStation Compatible by creating DOPs w/ an Attributes.ini for Description Details.

Adding CSI record number to the companies market symbol.

This tab display in the Edit Portfolio section of Unfair Advantage allows the user to connect symbols to CSI company identifiers. When Trading Blox builds its Stock Dictionary, it will use the record symbol and the connected company number to identify the data file that Unfair Advantage is maintaining in you data folder.

Note:

Earlier versions of Unfair Advantage didn't put an underscore character "_" between the symbol and the CSI number. If your files don't have this underscore number, you should enter the underscore character between the "S" and the "#" symbol.

Links:

See Also:

Links:

Edit Time: 9/12/2020 2:23:15 PM

Topic ID#: 244

Portfolio Editor

Part

IX

Part 9 – Portfolio Editor

Trading Blox allows users to create customized portfolios of futures, stocks, or forex. All portfolios can be tested historically, and used to generate daily, weekly or monthly orders that enter and exit each market.

Each system requires a portfolio manager section where you can select the portfolio of instrument that system use for testing and or trading. This flexibility allows you to easily test various instrument selections, and it can also allow you to test multiple systems, each with a different portfolio, in the same suite at the same time.

To create new portfolios, change the markets in an existing portfolio, or delete portfolios use the Portfolio Editor.

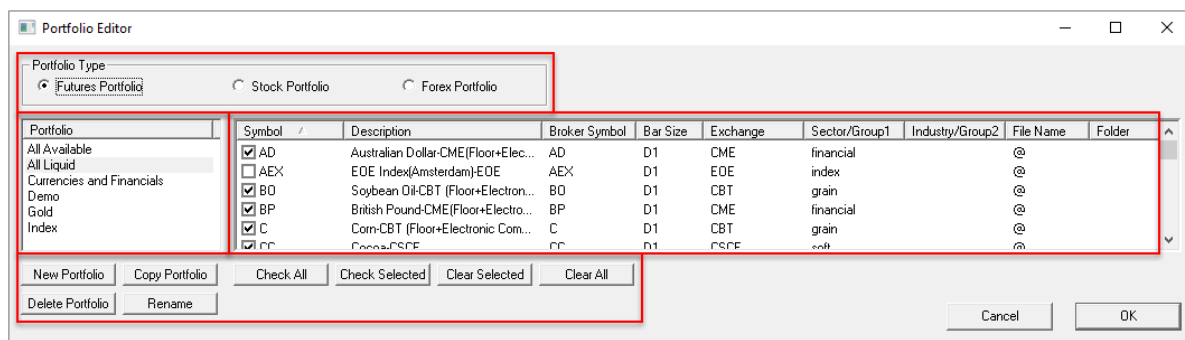
Trading Blox uses the same editor for all three of its supported instrument classes:



Trading Blox Supported Instrument Types.

To display the program's Portfolio Editor, press **F2 Portfolio Key** function key when the main Trading Blox screen is displayed, or click the Trading Blox [Home Menu Tab Group's Portfolio Editor Control](#) and the portfolio editor will appear.

When the Portfolio Editor displays, there is an option to select the type of portfolio. Portfolios can be used in all of the three instrument types. All the instrument types use the same layout. However, each portfolio type will only display portfolio names and symbol information for the portfolio type selected. Instrument data for all three portfolio types shows the same categories of data, but the portfolio names and instrument data information will always be specific for the portfolio type selected.



Forex, Futures and Stock Portfolio Editor

Portfolio Editor Operation:

Creating a new portfolio or using a sample provided is a simple task of selecting a portfolio name and then adding or removing which instrument symbols will be used within the new portfolio.

When a new portfolio name is added to the list of portfolios, the Portfolio Editor will automatically sort the portfolio names in alphabetical order.

To understand all the options and the outlined areas for editing a portfolio, read down through the following tables. To understand a specific area or control button, just move your mouse over the area of interest and will it change color. This change in color is a signal that area is linked to other information. Click on the color changed area the help file will jump to that location where the information is described.

Source of available instruments:

The only instruments that available to a portfolio are those in the current data folder location shown in the Trading Blox [Preferences Editor Menu](#) section: [Data Folders and Options](#) (add link). In this section of Trading Blox all three data type folders and dictionaries can be sourced or their locations can be changed.

Sample Instrument data files provided when Trading Blox was installed will be available in their default locations. However, this instrument information is good for learning how to use Trading Blox, but when it comes time for active trading, it will be necessary to have a daily data provider provide a timely source of information. This important because the "Sample Data" isn't updated soon enough to enable new information that are needed for reliable system trading.

When you are learning to use Trading Blox the sample data will remove the need to find data for testing, and to also see how the reporting and order signal and position reporting is performed. It will also provide information for all three instrument types on which you can create portfolios.

Portfolio Operation:

Portfolios provide a list of symbols that will be made available for testing. All test require at least one symbol that the software can exercise through its system logic. Each symbol in a portfolio will have its data records aligned by date. This date alignment allows Trading Blox to each instrument on the same data before the next date is tested for any of the other symbols. This ability to first sequentially test all the instruments on the same date and time is unique. Most trading system software will test an entire instrument from its first date to its last date prior to testing the next available instrument. This sequential testing of instruments instead of date leaves prevents many of the normal operations a trader would make in active trading. Sequential date testing allows for all the active trading operations in its testing and for each new data/time record of each instrument in the portfolio.

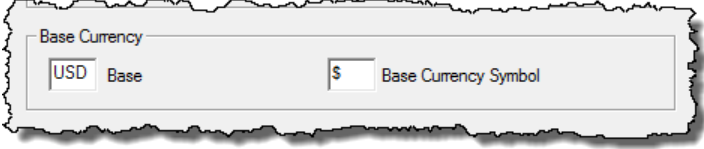
Symbols not in the portfolio can be accessed, but that ability needs to be created for the special task in mind that required a needed instrument to not be included in the portfolio.

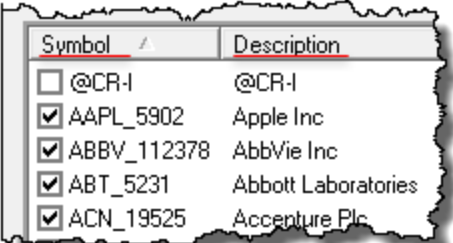
Changing Data Folder Locations or Symbol:

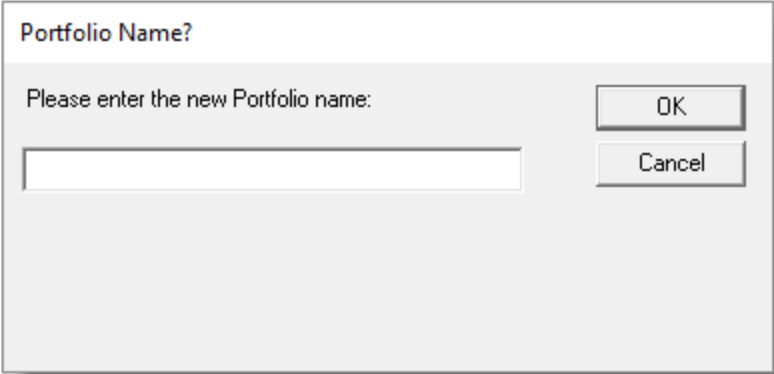
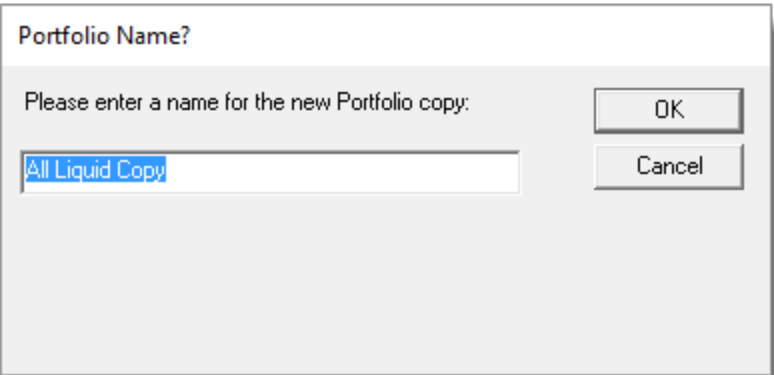
Forex and Futures will require their dictionary information be updated after new symbol information is added, removed, or changed. Any symbol change, folder disk path change, and data file name change must be updated in a dictionary in order for the change of that file to be available to Trading Blox.

Portfolio Type: ^Top	Description:
Futures Portfolio: ^Top	Futures are made accessible to a portfolio after they appear in the Futures Dictionary. Futures not in the dictionary will not be shown in the Portfolio Editor's Instrument list area. When a change is made to a Future's file name, the symbol can appear in the portfolio list of instruments, but it won't be available to testing. This failure is caused by a change in name. If a symbol that is in the portfolio is shown as being missing in the Low Window, go to that instrument record in the Futures Dictionary and fix the file name using the information in the Futures Dictionary Editor
Stocks Portfolio: ^Top	Stock Dictionaries are not required. If there isn't a dictionary, all the stock files found in the Preferences Editor Data Folder Stock setting will be displayed. When a dictionary is being used, only the files listed in the dictionary will be displayed. When a Stock Dictionary is not being used, new stock type files added or removed will appear when the new file is added, and will disappear when the file is removed. When a new file is added, it won't be automatically added to a stock portfolio. When a stock file is removed, and that symbol was assigned to a portfolio, the missing file's symbol will still be selected in the portfolio, but the file will not be available for testing. Missing files will be displayed in an Log Window report. Trading Blox Stock File Search Structure Process: Stocks are the only instruments that can have multiple dictionaries when it comes time to locate stock symbols. This means that when Trading Blox starts, it looks for files using the path settings established in the Preferences Data and Folder section. In each folder and each symbol it finds it will add it to a list of symbols that it will use to determine if it can be in a Stock Dictionary, or when a stock dictionary is not used, it will be displayed in the Portfolio Editor's Instrument Symbol list section. In this next set of images, the path is set to D:\Data\Stocks. That path has five sub-folders and a large number of symbol files. All the files in all the folders, plus all the symbols in the Stock Set's portfolio files will be record so they can be used in a Stock Dictionary, when that is the process being created, or as a list of symbols in the Portfolio Editor's Instrument Symbol list section.

Portfolio Type: ^Top	Description:
	Example Stock Folder and Files Structure Example Stock Set Default Stock Portfolio Example All the files and folders in all the places shown in the example above, which includes the symbols listed in the Stock Set Portfolio files, will be listed in the Instrument Symbol list section when a Stock Dictionary is not used. When a dictionary is being assembled, the same search structure will be used to locate symbols. If you do not have a Stock Dictionary (stockinfo.txt or stockinfo.csv file), it will look for a Metastock MASTER file with that name. If it finds the MASTER file, it will assume Metastock data and use that format process to read the data. If it does not find a MASTER file, it will assume ASCII Text files and look for all the .txt or .csv files in the specified folder, and all the folders contained with the top level Stock folder. Once all the folders have been searched, it will then search all the Stock set portfolio files and collect the symbols in those files. When there isn't a stock dictionary file, and the above symbol information is done its symbol search, Trading Blox will create a list of symbols from the file names and set files and display that list in the Stock option of the Portfolio Editor symbol listing. If a symbol is listed in a Stock file portfolio file, and there is no data file with the same name, that item in the list will be unable to load that symbol name information. Trading Blox flags portfolio symbols that don't have data files when it is first started and creates a warning message in the main screen's Log Window display area when the Log Window is

Portfolio Type: ^Top	Description:
	enabled. When a Stock Dictionary is used, (pending)
Forex Portfolio: ^Top	All Forex files that are needed for the instruments traded on exchanges that don't use the same currency as the setting listed in the Global Preferences Base Currency data field will require a Forex cross-rate file so the currency conversion can be applied:  Currency Symbol General Reporting Global Preference Setting Forex data needs to be updated when the symbol files are updated so that the currency conversion can use the cross rate value for the same date as the symbol date. When a Forex file is not current, the last conversion rate in the cross-rate file will be used for currency conversion.

Portfolio List Areas: ^Top	Description:
 Portfolio Name List for: Forex, or Futures, or Stock	All portfolio type use this Portfolio List to display the names of their portfolios. To change the instruments assigned to a portfolio name, select the portfolio first, and then change the selection of symbols. All changes to all portfolios will be preserved when the editor is closed using the Portfolio Editor's OK button. Portfolio changes will not be saved when the Portfolio Editor's Cancel button is used to close the editor.
 Portfolio Instrument Data List for: Forex, or Futures, or Stocks	All available instruments for a selected portfolio type will be listed in this section of the Portfolio Editor. Futures and Forex instruments require a record in their Forex or Futures Dictionary in order for the instrument to be available, and also to be available for testing when the selected portfolio is selected for use by suite's system for testing or order generation. Stock portfolios don't require a dictionary for them to be available. However, when a stock symbol is added, changed or removed and a stock dictionary is available, the stock dictionary will need to be recreated again in order for the changes in the stock instrument files to be recognized.

Portfolio Controls: ^Top	Description:
New Portfolio: ^Top	This button will creates a new Portfolio with no markets selected.  A New Portfolio Name Dialog.
Copy Portfolio: ^Top	This button will make an exact copy of the portfolio that retains the symbols selection listed in the source portfolios.  Selected Portfolio Copied with the word "Copy" as a suffix.
Check All: ^Top	To select all the markets in the Portfolio List Area, click this button name in the .
Check Selected: ^Top	Any markets that you have selected with the mouse so that their record shows as being highlighted, will be selected when you click on this button.
Clear Selected: ^Top	Any markets that you have selected with the mouse so that their record shows as being highlighted, will be deselected when you click on this button.
Clear All: ^Top	Click on this button will remove all the markets in the selected portfolio.
Delete Portfolio: ^Top	Any click on this button will remove the portfolio name that is selected. This option does not ask you to confirm the deletion of the portfolio.
Rename: ^Top	To change the name of the selected portfolio, click this button and a dialog will open providing a text area where you can alter the selected portfolio name.

Portfolio Controls: ^Top	Description:
	Portfolio Name? Please enter the new Portfolio name: All Available OK Cancel Rename Selected Portfolio

Close Editor: ^Top	Description:
OK: ^Top	Exiting the editor using this button will saves all the changes made since opening and it will close the portfolio editor.
Cancel: ^Top	To discard or ignore any changes made since the editor was opened this time will close the portfolio editor and ignore all changes.

Links:
See Also:

Section 1 – Sample Portfolios

Trading Blox comes with six sample portfolios for the U.S. futures markets, four sample stock portfolio of U.S. equities for the *NASDAQ 100*, and three sample forex portfolios.

The following table shows which futures markets that have been included in some of the Futures portfolios.

Instrument	Trending	Currency & Financials	Turtle Futures	Demo
British Pound	☒	☒	☒	☒
Canadian Dollar		☒	☒	☒
Cocoa			☒	
Coffee	☒		☒	
Corn	☒		☒	☒
Cotton				☒
Crude Oil	☒		☒	☒
DM/Euro	☒	☒	☒	☒
Eurodollar	☒	☒	☒	☒
Gold			☒	
Heating Oil			☒	☒
HG Copper			☒	
Japanese Yen	☒	☒	☒	☒
Mexican Peso	☒	☒	☒	☒
Natural Gas			☒	
Palladium				☒
Pork Bellies				☒
Silver			☒	
Soybeans			☒	
Soybean Oil			☒	☒
Soybean Meal			☒	
Sugar	☒		☒	
Swiss Franc		☒	☒	
10 year T-Note		☒		☒
Unleaded Gas			☒	☒
U.S. T-Bonds		☒	☒	
Wheat			☒	

Links:

Links:**See Also:**

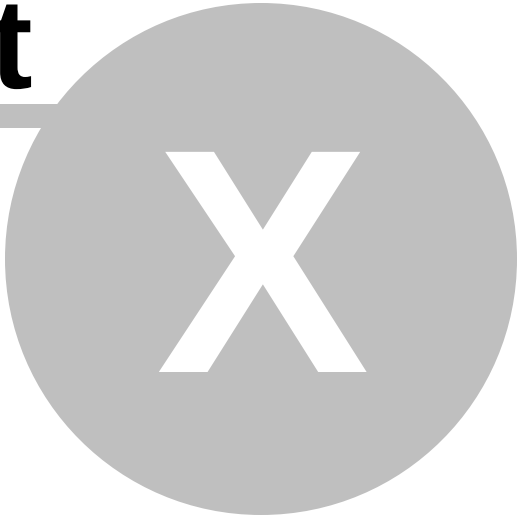
Last Edit: 9/12/2020

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 199

Historical Testing

Part



X

Part 10 – Historical Testing

Trading Blox has the most sophisticated historical simulation engine available. This engine lets you accurately test trading systems while accounting for all the costs and issues that are part of real trading.

The distinguishing features of Trading Blox's *Test Simulation Engine* include:

Available Simulation Tests:	Description:
Multiple Simulation Suites	Trading Blox supports a concept called suites. A suite lets you keep track of different tests at the same time. Each suite is a self-contained simulation specification that includes all the parameters for a complete test including which systems are part of the simulation, their equity allocation, the simulation parameters for each system, as well as the values for global parameters like commission rates, slippage assumptions, the starting balance, etc.
Multi-Market Simulations	Trading Blox simulations include all the markets in a test portfolio to accurately reflect the effect of simultaneous trades in different markets.
Multi-System Simulations	Trading Blox simulations can include many systems each trading simultaneously using an allocation from the common equity pool.
Multi-Portfolio Simulations	Each system can trade a separate and distinct portfolio of futures, stocks or forex markets.
Stepped-Parameter Simulations	Any of the simulation or system-level parameters can be varied during one simulation to show an accurate reflection of the effect of changes of that parameter on a single tests results. See Parameter Stepping for more details of this important feature.
Robust Accounting	Trading Blox keeps track of statistics and simulation results so you can better understand your systems.
True Dynamic Money Management	Unlike other products, Trading Blox's money management is not a post-simulation process applied to a list of fixed trades. Trading Blox's money management runs dynamically during the trade simulations so your trading system can use the value of the current equity and risk to control system entries, exits, and stops for increased control.
Contract-Roll Accounting	Trade simulations using back-adjusted data do not normally account for the effect of increased commissions and slippage due to rolling futures contracts over to new contract months when a position is held for more than a

Available Simulation Tests:	Description:
	few days. Trading Blox's Contract Roll Accounting allows you to accurately simulate the costs of contract rolling in the simulation.
Volatility-Based Slippage	While Trading Blox supports the common fixed slippage used by other testing packages, it also supports the simulation of slippage using a more accurate volatility-based slippage formula that takes into account the changes in market volatility on the slippage.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 156

Section 1 – Parameter Stepping

Altering the input value of even a single rule or test condition will often cause changes that ripple throughout the system, directly affecting its performance characteristics, often in dramatic and unforeseen ways. To assist users in gaining a clearer understanding of this important concept, Trading Blox incorporates an extremely powerful feature called *Parameter Stepping*. For any parameter, *Parameter Stepping* allows a range of values to be stepped through and tested one increment at a time - all within the course of a single test.

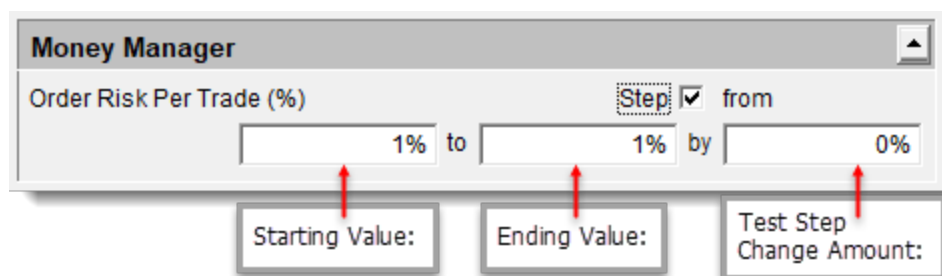
This document and the Trading Blox interface refers to the simulation of a unique combination of parameter input values as a *Test*. Thus, every simulation session (activated by depressing the Run Simulation button), will contain at least one *Test*, or in the case of *Parameter Stepping*, many *Tests*.

For example, if you want to see how changing the value of the Risk per Trade parameter affects overall profitability, you could conduct a number of separate (single-run) test sessions, changing the value of the Risk per Trade each time. Using Parameter Stepping, you could execute multiple parameter runs within a single test session, as follows:

First check the parameter stepping check box:



When the checkbox is clicked, it will add two more edit fields. The field to left side of the image is the place where the starting value is entered. The center field is the place to enter the maximum stepped value. The field on the right side of the image is field that determines the amount each step of the test will increment the parameter's test value:



You can change the values in the edit boxes to control the stepping range and increment. The following shows the values that will result in six separate stepped parameter runs ranging from 2% to 3% in step increments of 0.2%, with values of 2%, 2.2%, 2.4%, 2.6%, 2.8% and 3%:

Money Manager

Order Risk Per Trade (%) Step ☒ from

2% to 3% by 0.2%

After running the test, the Summary Results for this test will contain a Run Summary List which shows the major performance measures for each of the values of the parameter. This list can be sorted by clicking any of the column headings with the mouse:

Summary Results		Trade Chart								
#	Order Risk Per Trade (%)	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades	
1	2.0%	\$360,368,922...	35.89%	0.38	0.74	0.18	94.8%	102.3	4526	
2	2.2%	\$151,660,018...	30.52%	0.32	0.70	0.15	96.9%	102.3	4469	
3	2.4%	\$93,644,815....	27.62%	0.28	0.68	0.12	98.6%	102.3	4363	
4	2.6%	\$184,324,559...	31.71%	0.32	0.72	0.11	98.3%	102.3	4258	
5	2.8%	\$39,275,789....	22.55%	0.23	0.67	0.08	99.5%	102.3	4179	
6	3.0%	\$20,735,523....	18.96%	0.19	0.66	0.06	99.5%	102.3	4080	

Stepped Test Performance Summary Table

Non-numeric parameters that use pop-up menus for True and False, or other values, can also be stepped by selecting the **"Step True to False"** or other stepping item:

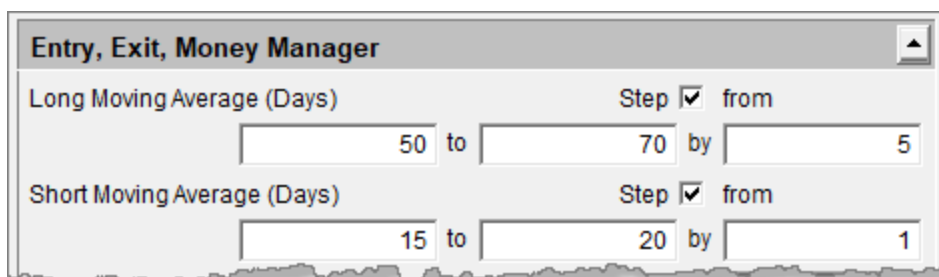
True
False
Step True to False

This will result in two tests, one with the value set to True and one with it set to False.

For Integer type parameters only, there is an option to step by percent or fixed value. So in the following example, you could step from 50 to 70 by 5% or by 5. Use the percent sign in this case to indicate a percent step. Note that this is different than the above example which is stepping by a fixed value but the value happens to be a percent.

It is also possible to step through a range of values for multiple parameters, all within a single simulation. Keep in mind, though, that the total number of unique *Tests* within a Parameter Stepping simulation is equal to the number of steps in each range of values, multiplied together. Because of this multiplicative effect, automated Parameter Stepping simulations can grow quite large.

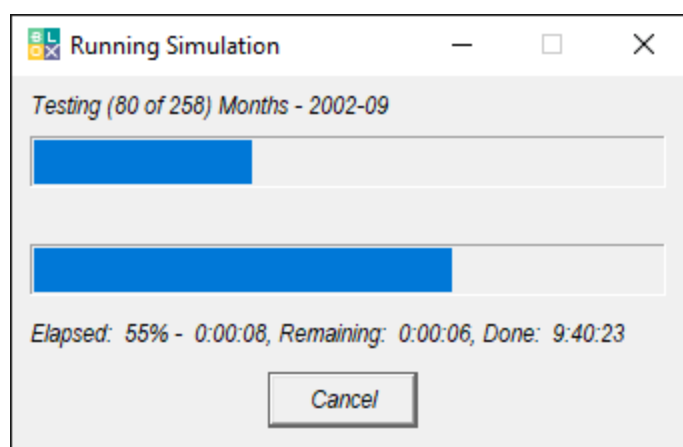
Consider the following combination:



Simultaneous Two Parameters Stepping Example.

The Long Moving Average has five steps (50, 55, 60, 65 and 70), and the Short Moving Average has six (15, 16, 17, 18, 19, and 20). Based on these instructions, the total number of runs that Trading Blox will automatically execute is 5×6 , or 30. If a third condition with only five (3) steps is added, then the number of runs will equal $5 \times 6 \times 5$, or 150.

Once a simulation has begun, Trading Blox shows a progress dialog. This dialog has two progress bars which indicate how much of the simulation has completed:



Progress of Stepped Test Processing.

Stepped test dialog also shows the percentage of the simulation and elapsed time, the time remaining until the simulation is complete and the time when the simulation is estimated to be done.

Trading Blox first loads the price data into memory before applying the trading rules. Price data is loaded only once at the beginning of the simulation, even when the simulation will test hundreds of parameter combinations. The symbols for each of the futures, stock or forex market data which are loaded are shown above the top progress bar as the price data is read.

After the data is read, the top progress bar shows the month being tested as Trading Blox applies the trading rules to month after month of historical data for each unique combination of parameter values. The bottom progress bar displays which test (out of the total number of tests), is currently being tested. In the above example, the progress bar shows that the simulation is currently processing test number 3 out of the 101 that will be tested as part of the simulation.

For simulations where there is no stepping and only a single test, the bottom progress bar will move only twice - once when the data has been read and once after the test is complete.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 183

Section 2 – Specifying the Test Dates

More details about specifying test dates is available here: [Simulation Parameters Date Section](#).

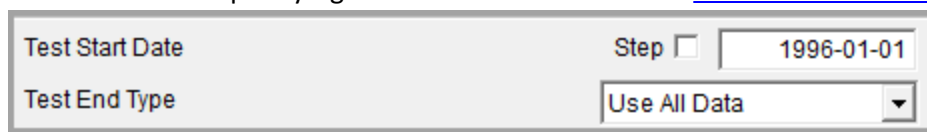
A screenshot of a software interface showing two input fields. The first field is labeled 'Test Start Date' and contains the text 'Step' followed by a checkbox and a date '1996-01-01'. The second field is labeled 'Test End Type' and contains a dropdown menu with the text 'Use All Data' and a downward arrow.

Image shows the "Use All Data" option selection. Other methods are available.

Test dates for a historical simulation test can be specified in various ways.

The date for the start date and end date for testing a historical simulation can be specified. The start date must be earlier than the end date using the following formats:

YYYY-MM-DD

YYYYMMDD

YYMMDD

MM/DD/YY

MM/DD/YYYY

These date formats can also be used for data files and when entering actual positions in the Order Generation Preferences.

The test will start on the first weekday equal to or after the *Test date entered* through the last day of available data, or the test end date specified in the *to* edit box, whichever is first.

It is possible to test multiple start and end days in the same test by stepping through values in the [Global Parameters](#).

If your test start date is near the beginning of, or before, the start of data for some instruments, those instruments will start trading when their indicators have primed. The priming period is determined by adding the max prime required for indicators and the max prime required for look-back parameters.

Section 3 – Order Filter

In certain circumstances, Trading Blox will filter an order without placing with the broker object even though it might have been filled based on the order type and price data of the day.

Orders will be filtered during "Order Processing" according to the following criteria:

Filtering Criteria:	Descriptions:
Volume	Checks the 5-day EMA of the volume versus the global settings. There is a max percent volume per trade, and a minimum volume required. If you set both of these globals to zero, they will be disabled.
Equity	Check that there is enough equity for the trade. For futures this is determined by adding up the margin required for the new trade and checking existing margin requirements vs. available equity. For stocks this is determined by adding up the full price of the stock. Available equity is determined by either the test equity or the Order Generation Equity if non-zero and generating orders. The Max Margin/Equity parameter is used here. If the margin is greater than the max amount times the available equity, then the trade is filtered. Set to zero to disable any margin checking. Note that if many orders are placed on the same day, they could all be filled even though it will use more equity than available.
Risk Manager	If you have a Risk Manager block, Trading Blox executes the CanAddUnit script. If the CanAddUnit script Rejects the order, then it will be filtered out.
Money Manager	If the unitSize script Rejects the order, then it will be filtered out.

If a trade is filtered due to any of the above, there will be an indication in the [Filtered Trade Log](#) file which is located in the Results folder if you set the Log Filtered Trades option to true. You can access all results files from the File/Results menu. These types of filters will be noted with an "Order Processing" type, and the date will be the Instrument Date.

Orders can also be filtered during the fill process, and these will be noted with a "Fill Processing" type and the date will be the Test Date.

The "Fill Processing" reject reasons:

Rejection Reasons:	Descriptions:
No trading data	An order was placed for a date without trading data.
Lock Limit Day	With futures by default the simulation will not buy on a up lock limit day or sell on a down lock limit day. Lock limit is defined as H=L. This feature can be turned off in global preferences.

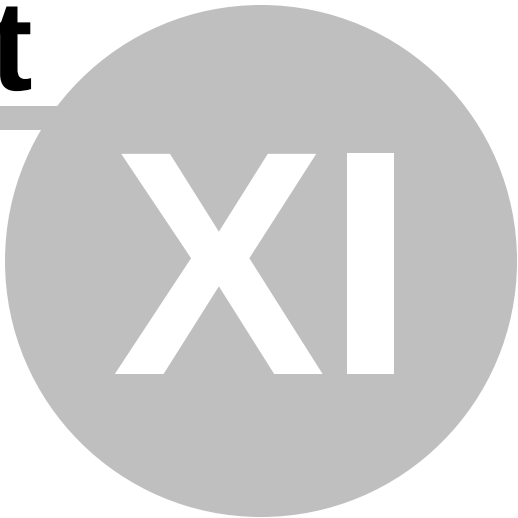
Rejection Reasons:	Descriptions:
Position is locked by Broker Positions	If you have a broker position for this market and system, then new entry or exit orders cannot be filled.
Open past stop price	The order cannot fill if the open price is already past the protective stop price.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 182

Reporting Information

Part



XI

Part 11 – Reporting Information

Each of the three reporting type sections serves a different approach. All can be edited to fits the reporting needs of each type of simulation test, and each will create a Summary Reporting Performance Summary with the information selected for the reporting type that matches the simulation type settings.

Reporting Types: ^Top	Description:
Trading Blox Preferences Reporting General section allow three different setting levels of reporting. Here are test reporting setting groups users find most useful:	
Single Test Reporting	Settings for a Single Test Report most often has all the available reporting preference settings enabled. It is a single test step simulation where none of parameters in the suite change value during the entire test run.
Small Simulation Reporting	Small Simulation Reporting tests only have a small number of parameters step test. Usually these numerical steps are limited to a small range of values. Small Simulation Reports need less information than a Single Test Report, but usually need more information than a Large Simulation Report. Small Simulation Reporting will automatically go into effect when a suite simulation parameter stepping exceeds the test count entered for Large Test Reporting Threshold setting shown in the General Reporting preference section. Default value limit for Small Simulation Reporting is "8" test steps, which can be changed.
Large Simulation Reporting	Tests that have many parameters stepping, or the parameters have a large range of values to test, will create a large number of performance test record results during a simulation. These large test simulation are more informative when they provide more overall summary information instead of a lot details.

In each three different types of simulation testing there are four sections where settings can be enabled or disable. Single and Small step test usually will enable the report to contain more information than the Large step count report. Settings in each of the different reporting sections there are four groups of settings that control what is reported.

Note:

At startup of Trading Blox the default values assigned to the software's internal parameters can be changed. Changing them is a simple process that is explained once the [Trading Blox Parameter Settings](#) (link) topic is displayed.

ReportSettingTypes: ^Top	Description:
Graphs	Information and examples about the many graphs displayed on a test report.
Monte Carlo Simulations	Monte Carlo simulations provide a repetitive randomization method to determine the robustness of a system of a system and how that system would have performed had the many test sequences had been changed many thousands of times. In the link above, the seed value for the randomization is a default value that can be changed.
Text Tables	These text base reports show the summary information of the trading period where the instruments and system performance are summarized.
Trade and Equity Logs	This section shows the types and names of the logs that are part of a simulation test. These logs allow a user to validate and see in detail how the performance progressed through the testing period.
Filter Results	Filter Results is only displayed in the Large Simulation Reporting section. This section can use the three main statistical calculations to reduce the information displayed at the end of parameter test.

Edit Time: 3/7/2022 11:51:36 AM

Topic ID#: 192

Section 1 – Graphical Reporting

All reporting items on the left side of this image are options in all three of the test type reports. Control of which type of report is selected by Trading Blox is determined by the settings in the [Reporting General](#) section :

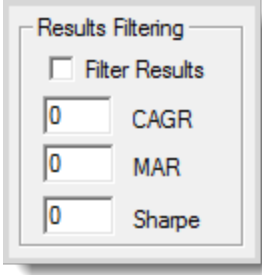
Graphical Information Reporting Information Options

Options on the right side of the image are only available when a simulation test enables parameter stepping in the testing suite. Top two items are available in the [Small](#) and [Large Simulation Reporting](#) setting sections, but [Results Filtering](#) is only available when a Large Simulation Report is executing.

Graphical Reporting: ^Top	Description:
Log Equity Graph	Graph of Total Equity and Closed Equity on a logarithmic scale.
Linear Equity Drawdown Graph	Graph of Total Equity and Closed Equity on a linear scale with drawdowns.
Custom Graphs	This option enables the display the Custom graphs that are created in block scripts to appear at the end of a test in the Custom Graph section of the Summary Test Results report . Graphs displayed are created in a user blox script location. These scripts collect test information in a BPV-variable during a test so it can be displayed in the Custom Graph. For example, Margin Equity as a percentage of Total Equity can be added as one of the graphs displayed to reveal the level of margin that was used during the test.
Monthly Returns Graph	Graph of the Monthly Returns.
Monthly Returns Distribution Graph	Graph showing the distribution of the Monthly Returns.
Total Risk Profile Graph	Graph of the total risk to Total Equity and Closed Equity .

Graphical Reporting: ^Top	Description:
R-Multiple™ Distribution Graph	Graph showing the distribution of trades by R-Multiple™.
R-Multiple™ Contribution Graph	Graph showing the profit contribution of trades by R-Multiple™.
Parameter Step Graph	
Muti-Parameter Graph	

[Large Simulation Reporting](#) will allow a user to enter the minimum value for any or all of the statistical calculation options shown in this section. Entering information in any of the three data fields will be the basis for which reports are saved for reviewing at the end of a test. Once the values to filter are entered, the Filter Result option must be enabled. When left disabled, the end of test filtering of the test step performance will be ignored.

Large Simulation Filters: ^Top	Filter Settings:	Description:
	Filter Results	Enable or Disable Option - Controls the simulation's ability to reject test step results that are below any of the three values set in each of the statistical fields.
	[CAGR-value]	CAGR Enter minimum value to accept reporting CAGR value.
	[MAR-value]	MAR Enter minimum value to accept reporting MAR value.
	[Sharpe-value]	Sharpe Enter minimum value to accept reporting Sharpe value.

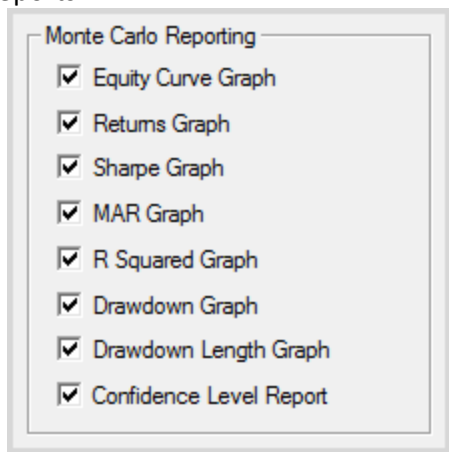
List of All Graphic Images: (Fix-It)

LinearEquityDrawdownGraph_P1.png
 LogarithmicEquityGraph_P1.png
 MonthlyReturnsGraph_P1.png
 MonthlyDistributionGraph_P1.png
 RiskGraph_P1.png
 <User Custom Graphs>

Section 2 – Monte Carlo Reporting

Trading Blox [Monte Carlo Simulation Results](#) create a large group population samples on which to estimate the probability of the simulation test results. Output results are controlled by the enabling or disabling of the options available in each of the simulation reporting methods, and by the [Monte Carlo Options](#) in the [Preference Editor](#).

These can be activated for Single Run and Small Tests, but are not available for Large Simulation Reports.



There are seven Monte Carlo Simulations that influence how the Monte Carlo simulation results create:

Monte Carlo Reporting: ^Top	Description:
Equity Curve Graph	A log-scale of the different equity graphs overlaid on the test equity curve.
Returns Graph	A curve of the distribution frequency and accumulated percentiles of the various returns. This graph won't be useful unless you check Sample with Replacement as the returns will be the same without that checked
Sharpe Graph	Same as above but with CAGR% for each alternate equity curve.
MAR Graph	Same as above but with MAR for each alternate equity curve.
R Squared Graph	Same as above but with R-Squared for each alternate equity curve.
Drawdown Graph	Same as above but with the maximum drawdown, second largest, and third largest for each alternate equity curve.
Drawdown Length Graph	Same as above but with the longest drawdown, second longest, and third longest drawdown for each alternate equity curve.
Confidence Level Report	A text-based percentile report which will be run as part of the normal performance settings. It shows the values of the performance statistics at the percentiles specified by the confidence level above.

Monte Carlo Reporting: ^Top	Description:
	NOTE: You must also have Trading Performance and Statistics checked for this report to work.

List of All Graphic Images: (Fix-It)

MonteCarloDrawdownGraph_P1.png
MonteCarloDrawdownLengthGraph_P1.png
MonteCarloEquityGraph_P1.png
MonteCarloMARGraph_P1.png
MonteCarloReturnGraph_P1.png
MonteCarloRSquaredGraph_P1.png
MonteCarloSharpeGraph_P1.png
MonthlyDistributionGraph_P1.png
MonthlyReturnsGraph_P1.png

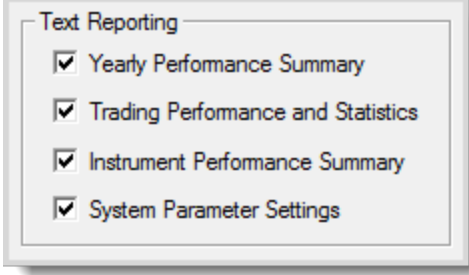
RMultipleContributionGraph_S2_P1.png
RMultipleDistributionGraph_S2_P1.png

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 173

Section 3 – Text Reporting

Text table reports can be added to any of the three type of [test reporting sections](#). However, they are most useful in with [Single Test Reporting](#). In some cases where [Small Simulation Reporting](#) only creates a few test steps they can be informative. Including them with a [Large Simulation Report](#) demands a lot of time unless the [Results Filtering](#) limits the output to only a few test.

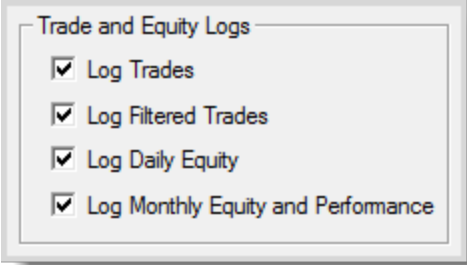
Text Reporting: ^Top	Description:
	Trading Blox text based reports are tables that show these table sections.
Yearly Performance Summary	A summary of the trading performance by year.
Trading Performance and Statistics	General performance statistics.
Instrument Performance Summary	Summary performance statistics broken down by instrument.
System Parameter Settings	Listing of the settings for every parameter of each active system.

Text Reporting Table Areas: (Fix-It)

- Stepped Parameter Summary Performance
- Yearly Performance Summary
- Instrument Performance Summary
- Test Period for parameter run 1.
- Trading Performance
- Win/Loss Statistics
- Equity Management
- Global Simulation Parameters
- Monte Carlo Confidence Level Statistics
- Preferences
- System Parameter Settings
- (User Custom Statistics)
- (User Custom Table Information)

Section 4 – Trade and Equity Logs

There are six different Trading Blox logs that can be created with a [Test Suite Summary Results](#) report. In addition, there isn't a software limit on the number of custom logs or reports that a user can add to [Simulation Test Results](#).

Trade and Equity Logs: ^Top	Description:
	Six different logs can be produced from a Suite's Simulation test:
Log Trades	This will log all trades in the Trade Log, and save trade details to the chart. For large tests, you can save memory and increase speed by keeping this box unchecked. When the system does not have to save trade details for every run, you can run much larger tests with available memory.
Log Filtered Trades	This option when checked logs the reason for all filtered or rejected trades into the Filtered Trade Log. It will indicate if a trade was filtered because of equity, volume, portfolio manager, risk manager, lock limit day, or open through the stop.
Log Daily Equity	This will log daily equity in the Daily Equity Log.
Log Monthly Equity and Performance	This will log monthly performance in the Monthly Equity Log.
Print Output	The file to which the PRINT statement goes when using the PRINT statement in Blox Basic Scripting. See the Print Output File.
Custom Reports	When the need to generate a report of a log process is required, Trading Blox can create custom reports and logs using Trading Blox Basic scripts in any type of blox module. Use custom reports when reporting information needs to be part of the system. Custom reports can be any Windows acceptable name and placed in a user created or selected folder.

Log Files Created: (Fix-It)

Daily Equity Log.csv
 Filter Log.csv
 Monthly Equity Log.csv
 Print Output.csv

Trade Log.csv
Custom Output Reports
Custom Order Reports

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 228

Importing-Exporting Suites

Part

XIII

Part 12 – Importing-Exporting Suites

Trading Blox makes it easy to import and export complete systems. This is especially useful when there is a need to move a test ready system to a Trading Blox installation that is only for generating orders, or when your broker provides the support to run your system at his location so you won't have to work to the market's schedule each day.

Trading Blox can export the system using Encryption so there is no chance it can be changed. Decryption is also possible when there is a need to access the code modules to make changes.

To Export or Import a complete system follow the details on these pages:

- o [Importing Suite Files](#)
- o [Exporting Suite Files](#)

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 158

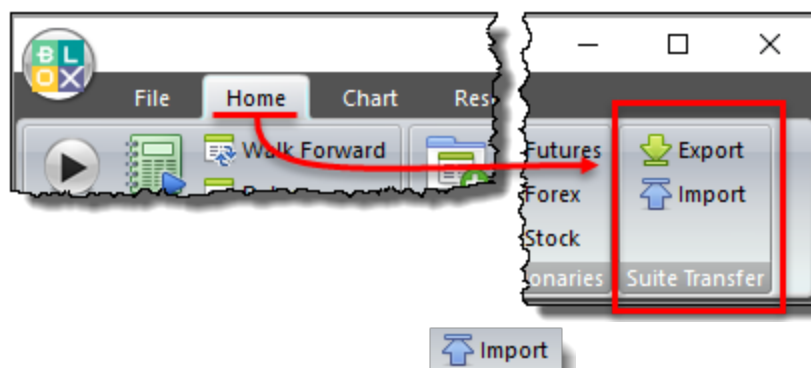
Section 1 – Importing Suites

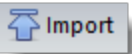
Import Suite Purpose:

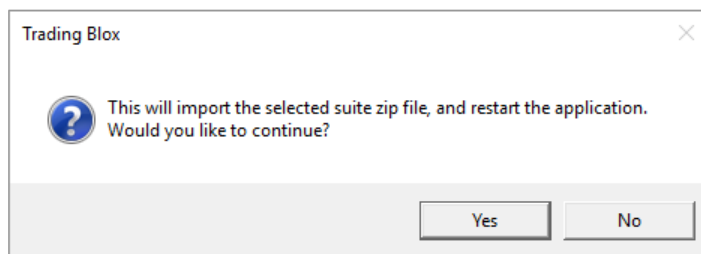
Trading Blox Suites contains numerous files and settings that are required to enable the system perform as it was intended when it was created. In order to prevent a lot of checking and parameter setting effort, Suites imported using the **Suite Import** menu option will retain all the settings and the files assigned when the suite was last tested.

Import Suite Instruction:

1. Save any open reports or Log Window information before you start the importing process. This is important because when you give consent to import the file, Trading Blox will exit from the screen while it imports the file's components. Once the imported information is installed, Trading Blox will automatically restart. When it restarts it will show the Suite name of the imported suite in the Suite List.
2. Click on and then Copy the Import Suite Zip file package you wish to import into Trading Blox.



3. Click on Home Menu and then click  suite option.
4. This next message will appear informing that you are about to import a Suite file collection.

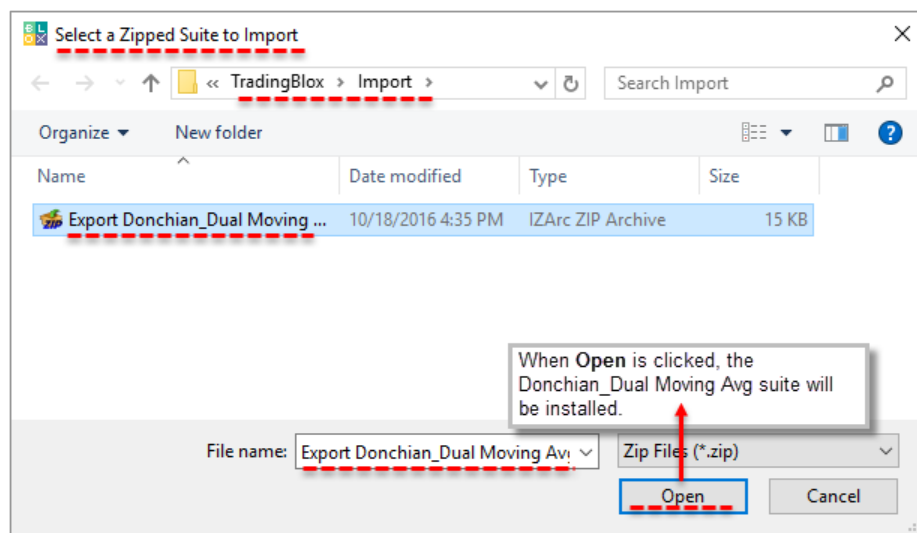


Home Menu Import File Will Exit and Restart after Suite is Imported.

5. Click the **Yes** button if you want to proceed at this time. Click **No**, if you want to save more displayed information before Trading Blox exits and restarts.

Import Suite Instruction:

6. If you click **Yes** a similar window shown below will appear providing you with a location where you can paste the suite file package you just copied. Click anywhere on the open white space and the Paste the file package into the open dialog.



Home Menu Suite Transfer Import File Open Dialog

7. You should see the name of the suite import package file in the white area of the dialog and in the Open File name area.
8. When you click the **Open** button, Trading Blox will unpack and place all the files in the Suite-Zip file into their proper Trading Blox folders. When the file installation finishes, Trading Blox will exit and then restarts again.
9. After Trading Blox has closed and then reappeared again, you should see the Suite file name you just imported.

Caution**Non-Encrypted Files:**

Files exported with the same name as those that are normally provided with a Trading Blox installation will be overwritten when the Suite is imported in another installation. This concern is for users who might have made modifications to some of the Blox or System module files and didn't change the name of file in some way that it is not longer the default name.

Special files that are not normally part of the Trading Blox process, like special index or weighting files that are not contained with a Blox module will not be exported and then will not be available for export.

Encrypted Files:

Caution

To import an encrypted file it must first be placed in the Import Folder where Trading Blox is installed.

Never change the name of an encrypted ".tbz" file once it is created. Names use are linked to the system name and blox and changing the name will prevent the system from being able to allow the file to appear.

If the names used are found during importing they will conflict with other blox or systems already in the Trading Blox installation location.

When you have a successful import Trading Blox will make the system available for testing. Encrypted files are accessible for viewing or for editing.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 159

Section 2 – Exporting Suites

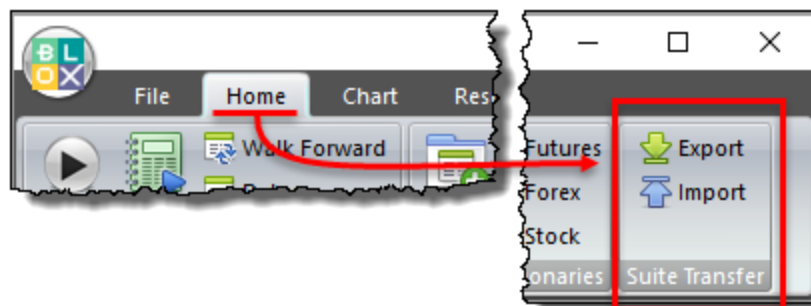
Export Suite Purpose:

Suite construction most often contains numerous files and settings that can have a significant effect on how a system performs and the results it creates.

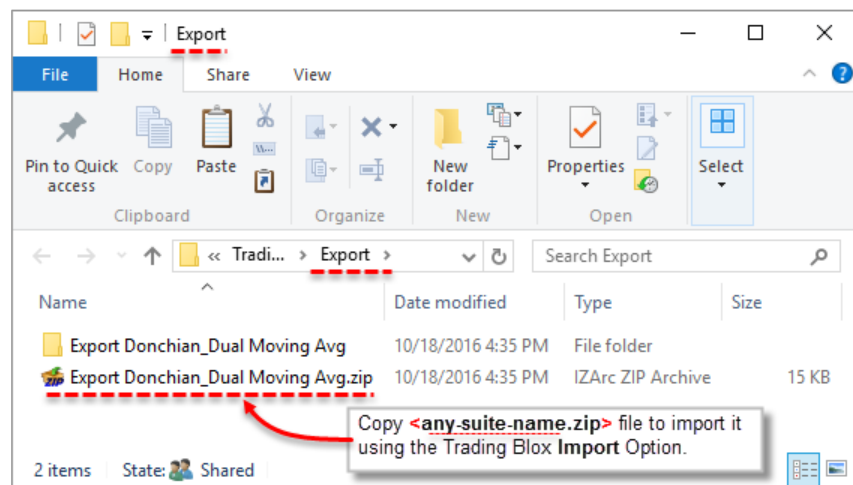
In order to reduce the time it takes to transfer Suites to another Trading Blox installation, the Export method will package all the files for transport in the Zip file it creates.

Export Suite Instruction:

1. Click on the Suite that is you want to transfer to another Trading Blox installation.
2. Click on the Home Menu Suite Transfer Export control.



3. When this next dialog appears, select the suite Zip file name and copy it using Ctrl-C or by right clicking and selecting the copy menu item.



Home Menu Export Suite Zip File Access Dialog

4. Now add that file to your email, or other methods for sending that file to the other Trading Blox installation where it is to be imported.

Caution:

Files exported with the same name as those that are normally provided with a Trading Blox installation will be overwritten when the Suite is imported in another installation. This concern is for users who might have made modifications to some of the Blox or System module files and didn't change the name of file in some way that it is not longer the default name.

Special files that are not normally part of the Trading Blox process, like special index or weighting files that are not normally part of a Suite/System, not be exported. When special files must be part of the suite new installation, those will have to copied and pasted manually.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 136

Summary Results

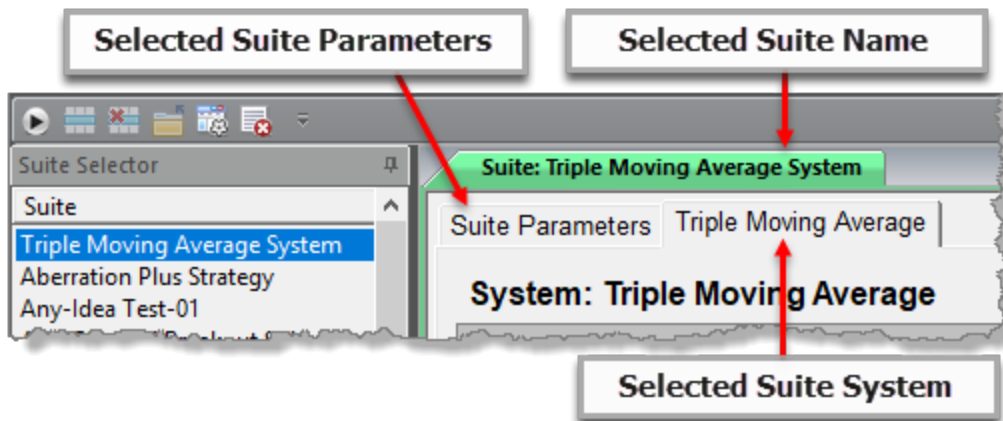
Part

XIII

Part 13 – Summary Results

Selected Suite Control Access Tabs:

All Suites that are assigned one or more systems, will have these tabs. Each tab shown in this image will show the controls that effect how a Suite test will be controlled during testing.



Selected Suite Test Control Access Tabs

Each suite with a selected system will display three or more tabs when it is selected. One tab in a suite allows access to the Suite Parameters that control how the systems are tested. The other tabs will allow access to a System's Parameters that can change how the system is run. When a suite has more than one selected system, an additional tab will appear for each of the system names selected.

Summary Test Results:

The test Results are displayed after a historical simulation test is finished.

When a test has been performed, Trading Blox will display its test result summary report. In that report, the settings for reporting enabled in the [Preferences Reporting Section](#) settings will be displayed.

In the next image shown below, the test results show a stepped parameter test. This type of test is used to see the results of how a parameter that starts at one value and ends at another value performed at each of the steps tested.

Suite: Triple Moving Average System Results: Triple Moving Average System - 1										
Summary Results										
	Long Moving Average (days)	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE DD	Longest ...	Trades	
1	100	\$62,787,741....	24.90%	0.33	0.71	0.39	76.1%	57.2	2257	
2	110	\$58,204,498....	24.47%	0.32	0.70	0.39	76.8%	58.1	2231	
3	120	\$96,960,696....	27.43%	0.36	0.76	0.40	76.2%	55.8	2199	
4	130	\$44,830,604....	22.98%	0.30	0.69	0.40	76.4%	58.1	2157	
5	140	\$57,089,888....	24.36%	0.33	0.71	0.41	74.1%	52.6	2129	

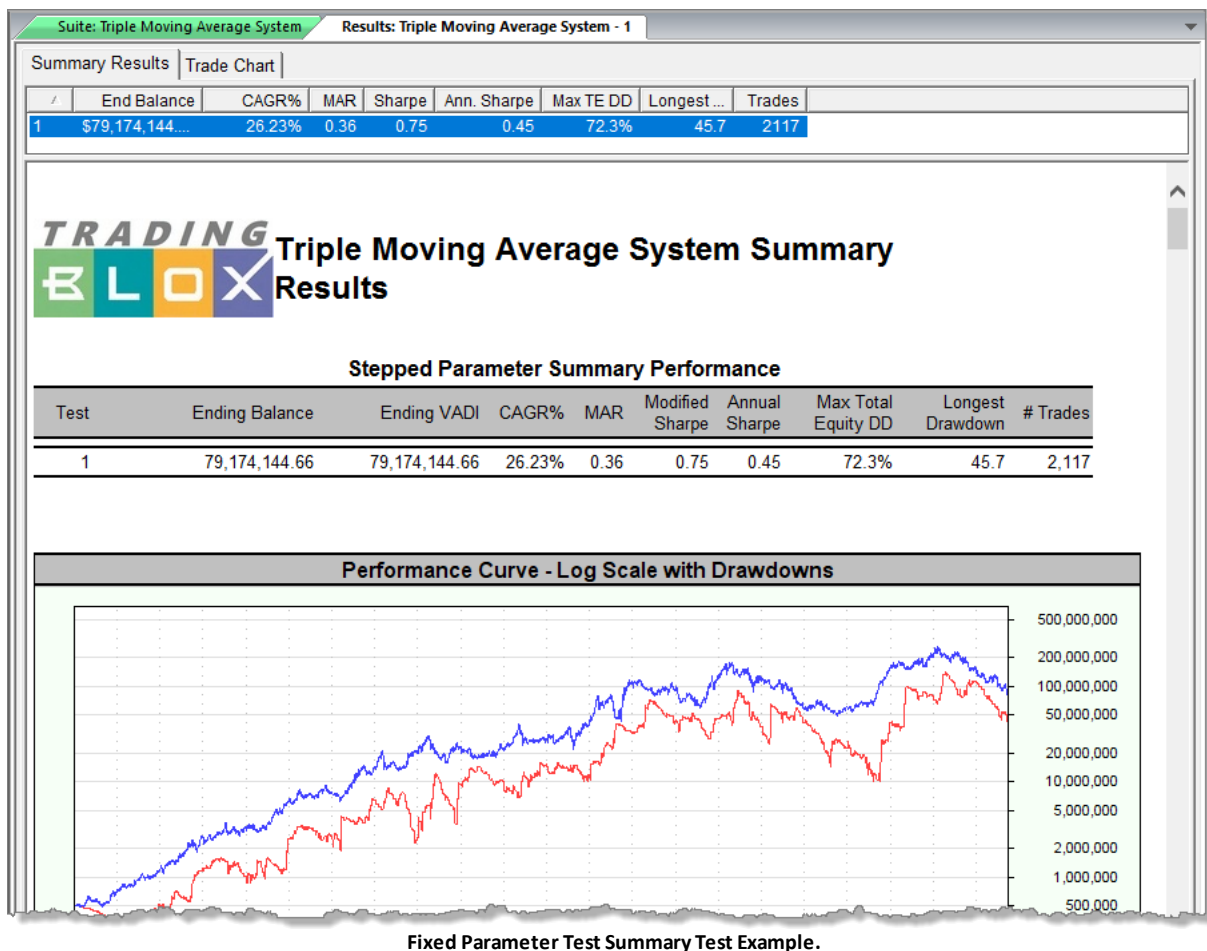


Triple Moving Average System Summary Results

Stepped Parameter Summary Performance										
Test	Long Moving Average (days)	Ending Balance	Ending VADI	CAGR%	MAR	Modified Sharpe	Annual Sharpe	Max Total Equity DD	Longest Drawdown	# Trades
1 Details	100	62,787,741.79	62,787,741.79	24.90%	0.33	0.71	0.39	76.1%	57.2	2,257
2 Details	110	58,204,498.96	58,204,498.96	24.47%	0.32	0.70	0.39	76.8%	58.1	2,231
3 Details	120	96,960,696.30	96,960,696.30	27.43%	0.36	0.76	0.40	76.2%	55.8	2,199
4 Details	130	44,830,604.09	44,830,604.09	22.98%	0.30	0.69	0.40	76.4%	58.1	2,157
5 Details	140	57,089,888.23	57,089,888.23	24.36%	0.33	0.71	0.41	74.1%	52.6	2,129
6 Details	150	83,609,468.62	83,609,468.62	26.56%	0.37	0.76	0.46	72.3%	45.7	2,115
7 Details	160	58,012,058.19	58,012,058.19	24.45%	0.32	0.72	0.43	76.1%	56.3	2,087
8 Details	170	65,850,937.54	65,850,937.54	25.18%	0.35	0.73	0.43	72.5%	44.8	2,081
9 Details	180	63,381,070.87	63,381,070.87	24.96%	0.32	0.73	0.42	77.5%	53.6	2,060
10 Details	190	56,129,532.45	56,129,532.45	24.26%	0.31	0.71	0.42	78.5%	54.5	2,049
11 Details	200	55,245,397.28	55,245,397.28	24.17%	0.32	0.71	0.41	76.4%	46.4	2,023

Single Parameter Multiple Step Test Example

This next image shows what a typical fixed parameter test results will look like:



All summary test reports will have the following tabs where information used in the test is generated and displayed.

Results Tabs

Each *Simulation Results* window has an associated test results tab. Trading Blox automatically selects the tab which corresponds to the most recent test after that test finishes. The tabs are named after the Simulation Suite

Test Summary List

The top of the *Summary Results* tab contains a list of the results for each of the distinct parameter combinations run for the simulation. For simulations with no parameter steps, this list will have only a single line. Clicking on any of the lines in the *Test Summary List* takes you to the top of the detailed results for that test.

Detailed Test Report

The bottom of the *Summary Results* tab contains a detailed test report. This report is generated using the HTML format, the same format as internet web pages. This allows much more sophisticated formatting than the simple ASCII Text file output used by most system testing programs. Each of these HTML-based reports are also saved as files in the [Results Folder](#) for later

viewing. You can define which reports show up in the Detailed Test Report section using the [Report General](#) section of the Trading Blox Preferences.

Note that this HTML version of the summary results is redundant to the Test Summary Listing above. You can right click on this list to export directly to excel. This report does not print for runs greater than 5000 because the resulting file would be too large for windows to handle.

Report Section Tabs

Each *Test Results* window has one tab for the *Summary Results* and one for the *Trades*. Clicking on a tab will take you to that section.. The next section, [Test Trades Report](#), shows the *Trades* Tab.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 212

Test Results

Part



Part 14 – Test Results

The Summary Test Results are shown after a test completes and can be found on the Summary Results tab of the main Results Window (see the [Test Results](#) user interface overview for more information).

A copy of the Summary Test Results is also stored in the Results folder. Each time a test is run, a separate Summary Test Results file (and folder containing images) with a unique file name is created. Results are never overwritten, so that a permanent record of each test is maintained.

Trading Blox names each Summary Test Results file according to the unique date and time at which the test was executed. The naming convention is as follows:

<Suite Name> yyyy-mm-dd_hh_mm_ss.htm

For example, a file with the name "Test 2003-04-01_14_53_01.html" was run on April 1, 2003, and completed at 14:53:01 (or 2:53 PM).

You can move these files out of this folder, or send them to others. However, you should also move the corresponding folder name to the same place because it contains the images for the reports. The test .htm file is named according to the suite. The corresponding folder is named "Test" with the same timestamp as the .htm file. For the above example, the folder would be named "Test 2003-04-01_14_53_01.html".

If you run many tests with many different parameter combinations Trading Blox will generate many results files. If you find your hard drive space is getting low, you can delete older unused results files and their associated directories.

The reports in Trading Blox include:

Report Item Name:	Descriptions:
Stepped Parameter Performance	To quickly see the effects of parameter changes on system results
Equity Graphs	Log-scale and linear-scale equity graphs to see system performance over time
Monthly Return Graphs	Shows the monthly returns as well as the distribution of monthly returns by Percentage .
Total Risk Profile	Shows the amount of risk or portfolio heat carried over time
R-Multiple™ Graphs	Shows the distribution of trades by R-Multiple™ as well as the profit contribution of the trades by R-Multiple™ .
Yearly Performance Summary	Shows the performance of the system broken down by year

Report Item Name:	Descriptions:
Trading Performance Statistics	CAGR%, MAR ratios, etc. Trading Blox has the most extensive trading system performance statistics available
Instrument Performance Summary	A breakdown of the performance by instrument
System Parameter Settings	Shows the exact parameters used to generate the results so you can reproduce results or compare against other results
Trade List with Charts	A complete set of each trade drawn on a chart for quick perusal of each trade responsible for the results

You can specify which of these reports is included in the Summary Test Results using the Preferences for reporting (see: [Report Content Specification](#) for details).

Section 1 – Stepped Parameter Performance

The Stepped Parameter Summary Performance List is displayed at the top of the results list:

Summary Results		Trade Chart								
#	Long Moving Average (Days)	End Balance	CAGR%	MAR	Sharpe	Ann. Sharpe	Max TE ...	Longest ...	Trades	
1	30	\$453,348.94	-3.22%	-0.11	-0.10	-0.20	30.5%	16.3	71	
2	40	\$442,908.03	-3.97%	-0.12	-0.10	-0.20	31.8%	16.0	59	
3	50	\$497,676.04	-0.16%	-0.01	0.08	-0.01	29.4%	10.6	50	
4	60	\$548,124.20	3.12%	0.10	0.27	0.15	30.4%	16.2	40	
5	70	\$537,352.32	2.43%	0.08	0.23	0.12	31.0%	16.3	36	

Stepped Test Summary Reporting Table

Clicking on any row in the Stepped Parameter Summary Performance List will bring the detailed reports for that particular parameter run to the Detailed Test Results area of the [Test Results Window](#).

Reporting Property name descriptions:

Property Name:	Descriptions:
Run	A sequential count that keeps track of the number of distinct parameter combinations being tested.
End Balance	The account balance at the completion of each run.
CAGR%	The compounded annual growth rate (geometric mean return).
MAR	A ratio obtained by dividing the CAGR% by Max Total Equity Drawdown. The higher the MAR ratio, the better the risk-adjusted performance.
Sharpe	This version of the Sharpe Ratio divides annualized return by annualized standard deviation of returns, using monthly data points. Excludes the risk-free-rate in the numerator.
Ann. Sharpe	This version of the Sharpe Ratio divides annual return by standard deviation of annual returns, using actual annual calendar data points (The Sharpe Ratio is the classic measure of return versus. risk. Divides excess return by standard deviation to determine reward per unit of risk. The higher the Sharpe ratio, the better the risk-adjusted performance.)
Max TE DD	Max Total Equity Drawdown - a one-time event that reflects the largest equity retracement-relative to a previous equity high-in the entire simulation. Expressed as a percentage, this statistic is measured peak-to-valley, using Total Equity.
Longest TE DD	Longest Total Equity Drawdown - a one time event that reflects the longest duration drawdown in Total Equity over the life of the simulation. It is measured from previous equity peak to new equity peak, and expressed in months.
# Trades	Number of round-trip trades placed over the course of the entire simulation.
Custom Statistics	If your system includes custom statistics they will show up here.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 4002

Section 2 – Multi-parameter Surface Chart

When you run two or more stepped parameters in the same test, the results will be shown in a contoured surface chart. All combinations of the steps will be graphed on the **x** and **y** axis. The average selected Goodness Measure Index will be shown as various colors.

You can select which goodness measure to use in Preferences, and you can set the scaling to be automatic or manual.

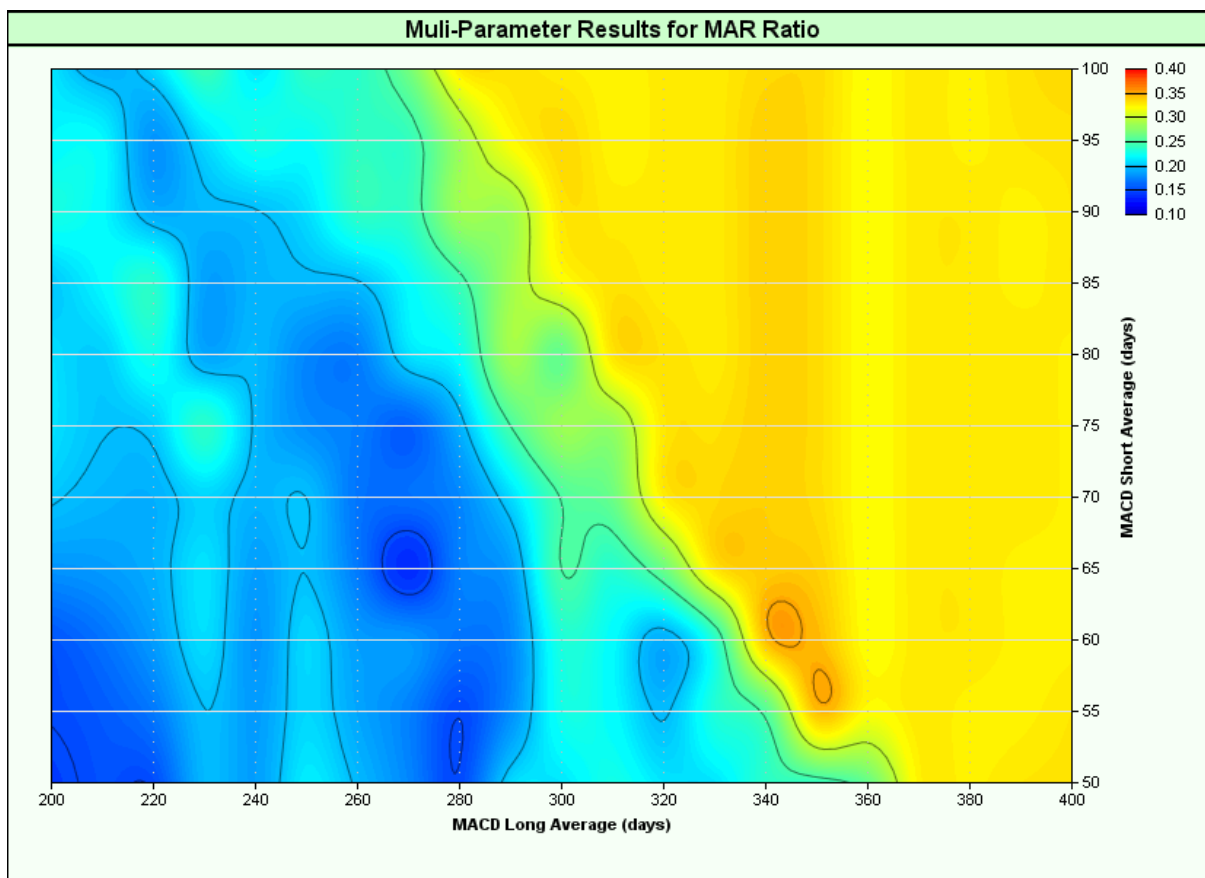
If you run a stepped test with more than two parameters, the Goodness Measure Index will be averaged for all results at a given x and y axis.

Multiple charts can be produced simultaneously in one simulation by use the **test.SetGoodnessToChart()** function in Blox Basic.

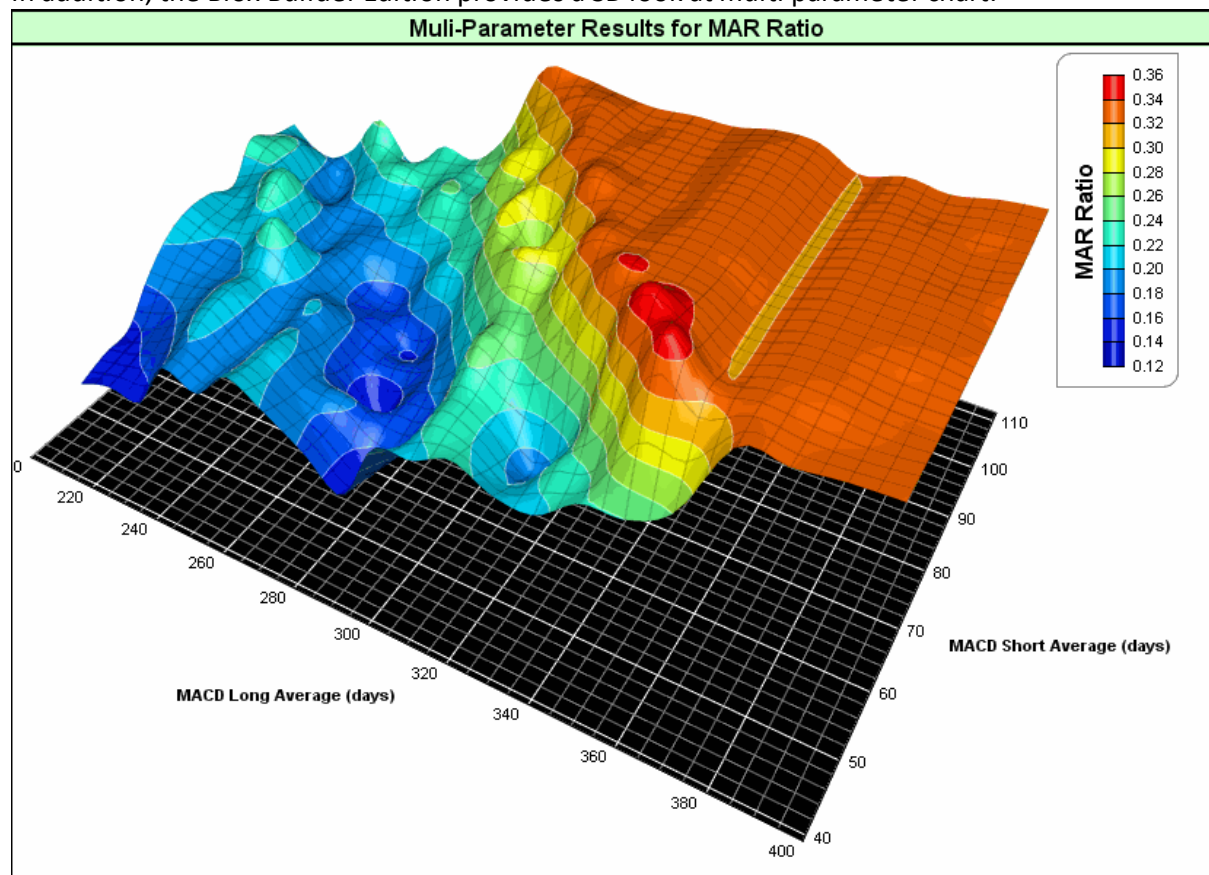
Each chart can be turned on or off in the **TradingBlox.ini** file by setting the Multi Parameter 1 and Multi Parameter 2 parameters to true or false. This is useful if you prefer one chart to the other, and want to turn one off.

Multi Parameter **Chart 1** = **FALSE**

Multi Parameter **Chart 2** = **TRUE**



In addition, the Blox Builder Edition provides a 3D look at multi parameter chart:

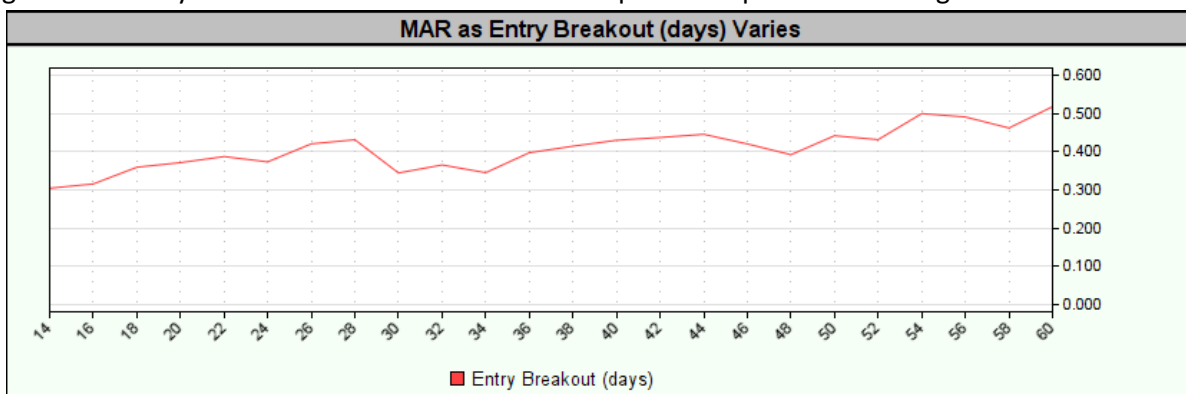


Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 179

Section 3 – Parameter Stepping Graph

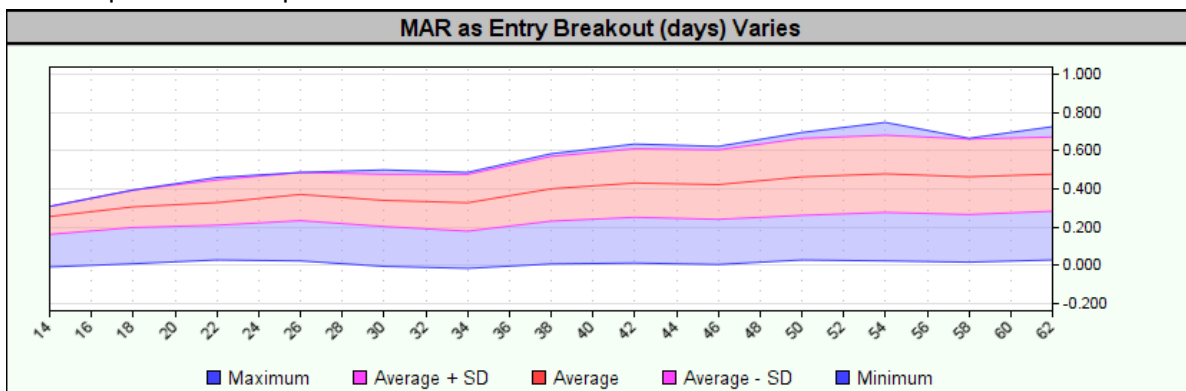
One of the major goals of any system testing effort is the determination of the optimal set of parameters to trade. The Parameter Stepping Graphs allow traders to see how the values of a given summary measure varies as the values for a particular parameter changes.



MAR as Entry Breakout (days) Varies

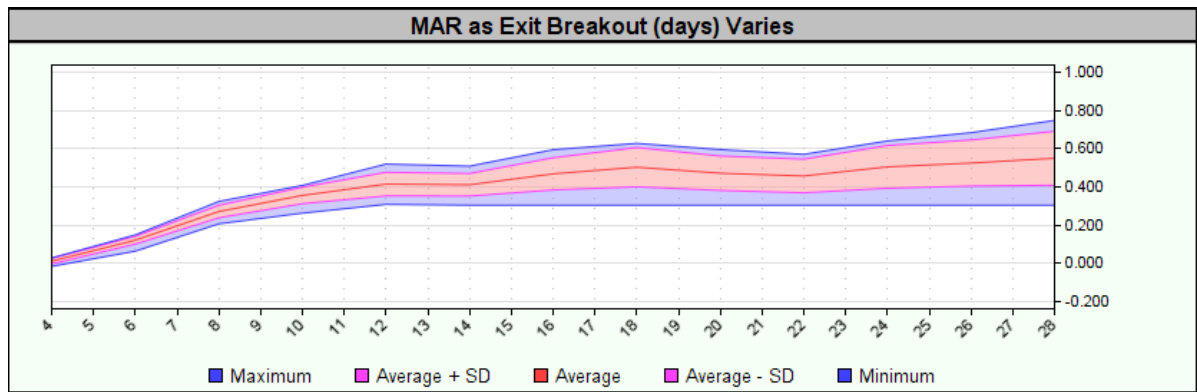
For a single parameter test run, the above Parameter Stepping Graph shows a line indicating the values of the measure for each step in the test. For example, the above graph for a stepping test of the Donchian System's Entry Breakout parameter shows an almost 45% CAGR value at step value of 28 days for the Entry Breakout dropping to 35% as the Entry Breakout moves higher to 31.

For tests with more than one parameter, the Parameter Stepping Graphs shows a band of values for each parameter step.



MAR as Entry Breakout (days) Varies

The above graph shows a dual stepped test using the Entry and Exit Breakout value creates channel results in each parameter's graph. Entry Breakout parameter days stepped from 14 to 60 days, and the Exit Breakout parameter stepped from 4 to 28.



MAR as Exit Breakout (days) Varies

The above graph shows another test that also stepped the Exit Breakout from 4 to 28.

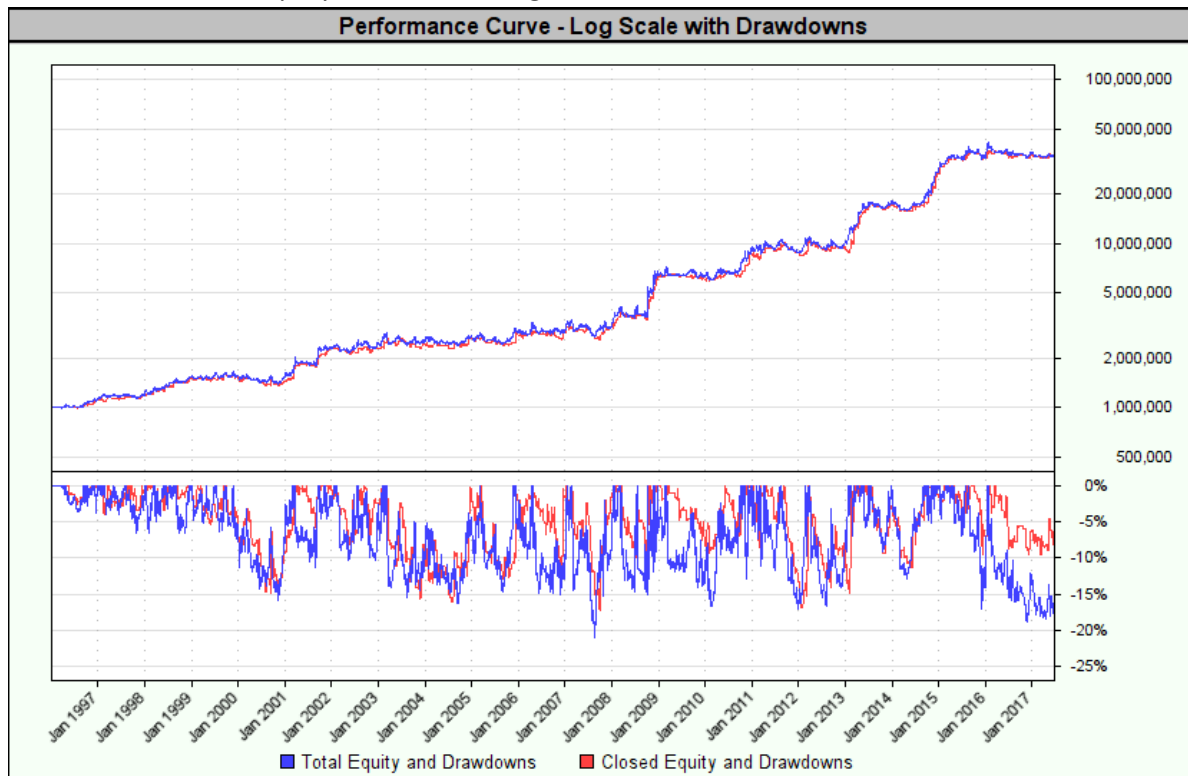
This graph channels means that for each of the steps on the Entry Breakout for the Exit Breakout combined created graph plots that show the average, minimum, maximum and a band that is one standard deviation above and below the average for each value.

A narrow band around the average indicates that the other parameters affected the results in a relatively minor way. A wide band indicates that the other parameters affected the test in a large way. The wider the band, the greater the influence the other parameters had on the test measure.

The preferences item Measure of Goodness Index from the [Reporting General](#) preference page controls the measure used for the Parameter Stepping Graphs. In the above examples, this value was set to CAGR%.

Section 4 – Log Equity Graph

Following the Summary Test Results, a series of graphs is displayed. In this image example both the total and closed equity are shown along with the drawdown:



Total & Closed Logarithmic Equity Curve Results with DrawDown

Closed Equity display is controlled by the settings in the Preference's section [Reporting General](#).

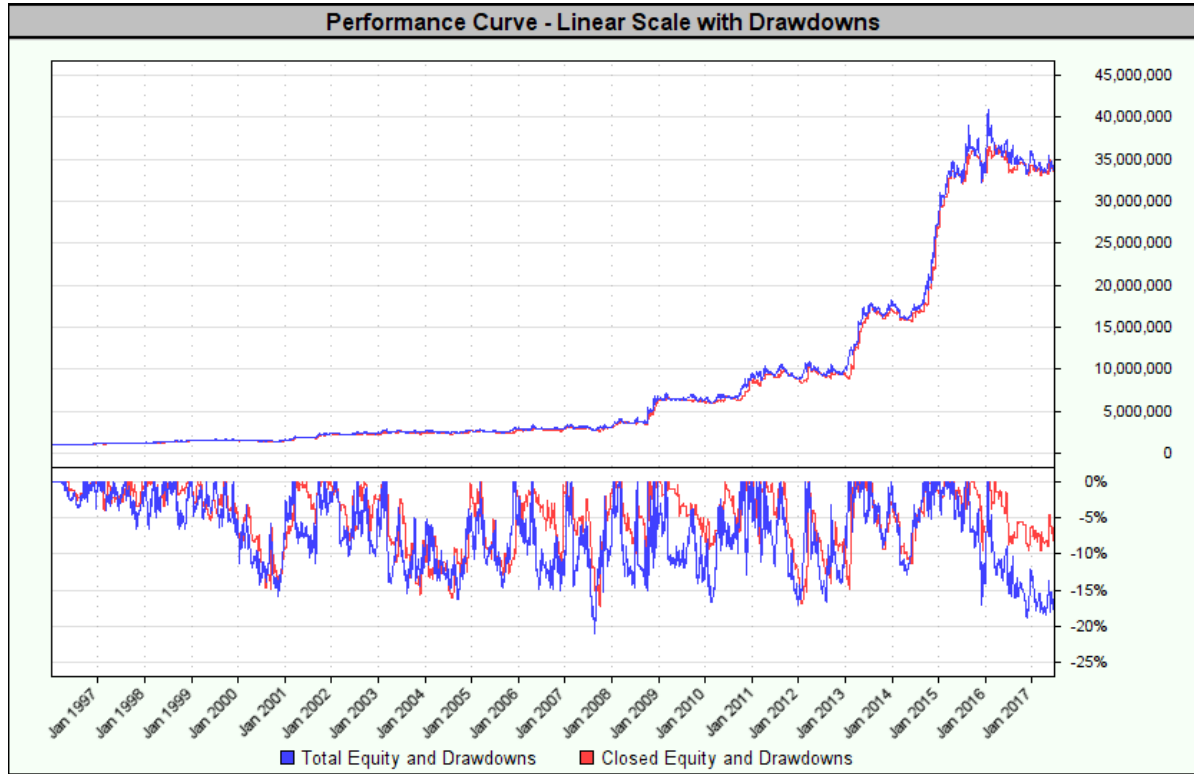
The Log Scale Equity Graph displays the [Total Equity](#) and [Closed Equity](#) on a logarithmic scale (with time on the horizontal axis), providing a view into how the overall account value changes over time, as well as how [Total Equity](#) and [Closed Equity](#) vary in relation to one another over the same time period.

[Total Equity](#) is plotted in blue; [Closed Equity](#) is plotted in red. In the world of finance, red is often used to represent a loss. Here, it is used only to distinguish between the [Total Equity](#) and [Closed Equity](#), and has no other connotation.

On a logarithmically-scaled Equity chart, a given distance (on the vertical axis) always represents an equal percentage change. For instance, on the chart below, the distance from \$100K to \$1M is the same as the distance from \$1M to \$10M. They both represent the same percentage change (in this case 1,000%).

Section 5 – Linear Equity Drawdown Graph

The Linear Equity Graph shows the total and closed equity profile over time measured on a linear scale with the comparative drawdowns over the same period plotted below that graph:



Total & Closed Linear Equity with DrawDown Curves

Closed Equity display is controlled by the settings in the Preference's section [Reporting General](#).

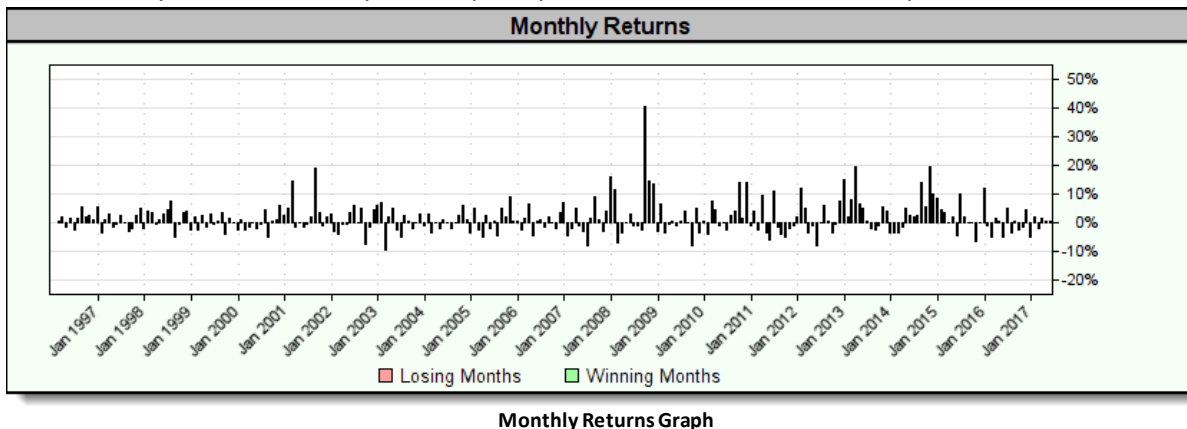
On the Linear Equity Graph, both [Total Equity](#) and [Closed Equity](#) are plotted over time on the graph. This graph has a vertical scale that is linear, or arithmetic, as opposed to logarithmic scale of the [Log Scale Equity Graph](#).

On a linear-scaled Equity chart, a given distance (on the vertical axis) always represents the same absolute change in system wide base currency. For instance, on the chart above, the distance between \$10,000,000 - \$20,000,000 is the same as the distance between \$20,000,000 - \$30,000,000. While linear scale Equity Curve graphs tend to lose resolution in the earlier years of an extended study (as can be seen above), most of us are accustomed to looking at linear representations, and they do have their uses.

In the bottom half of the graph, [Total Equity](#) and its Drawdown are plotted in blue; [Closed Equity](#) and its Drawdown are plotted in red. Drawdown is a retracement from a previous Equity peak, and is expressed here in percentage terms. This type of graph is also sometimes referred to as an Underwater Equity curve.

Section 6 – Monthly Returns Graph

The Monthly Returns graph shows the returns for each month of the test. The graph is automatically resized so that you can quickly see each month's individual performance.



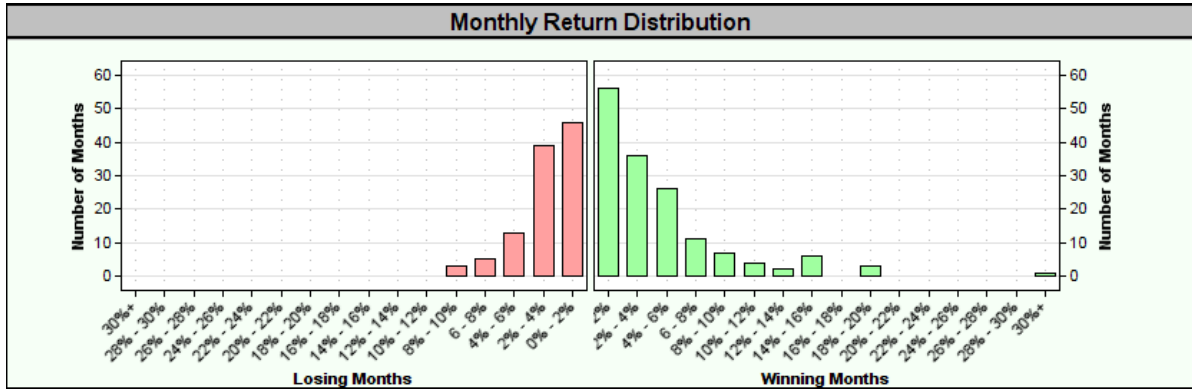
Monthly Returns Graph

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 177

Section 7 – Monthly Returns Distribution Graph

The Monthly Return Distribution graph shows the frequency distribution of the returns. Each bar shows the number of times that the monthly returns fall within a specified range. In the above example, you can see that there were 10 months where the returns were between 8% and 9%.



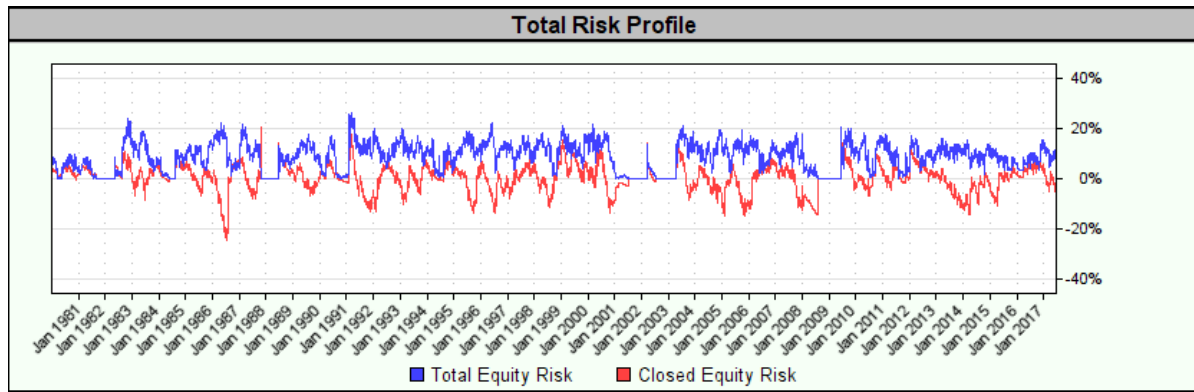
Monthly Returns Distribution Graph

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 176

Section 8 – Total Risk Profile Graph

Total Risk examines the relationship between [Total Equity](#) and [Closed Equity](#) to Open Risk (respectively). The Total Risk Profile graph displays Total Equity Risk and Closed Trade Risk plotted over time.



Total Risk Profile

The following definitions will help illustrate the difference between [Total Equity](#) and [Closed Equity](#) risk.

Property Name:	Definitions:
Closed Equity	Starting Balance plus the cumulative profit (or loss) from all closed-out trades. (Sometimes referred to as Closed Trade Equity.)
Open Equity	Total profit (or loss) of all open positions.
Total Equity	Closed Equity + Open Equity
Open Risk	The (currency) distance from the current price to the closest stop, for all open positions. It is based on the assumption that all open positions will be exited at their current designated stops (though some or all of these positions may prove profitable in the future).
Closed Risk	The (currency) distance from the entry price to the closest stop (either entry or trailing), for all open positions. If the trailing stop is profitable then the closed risk on a position is zero.
Locked-In Profits	Open Equity - Open Risk. (The profit on a trade in progress becomes "locked-in" when the trailing stop moves favorably past the entry price.)
Total Equity Risk	Open Risk / Total Equity
Closed Equity Risk	Closed Risk / Closed Equity

Consider the following example of Total Equity Risk:

Total Equity	\$150,000
Closed Equity	\$100,000

Open Equity	\$50,000
Open Risk	\$40,000
Locked-in Profits	\$10,000
Closed Risk	\$0
Total Equity Risk	Open Risk / Total Equity = \$40,000 / \$150,000 = 26.67%

Closed Equity Risk, while similar to **Total Equity Risk**, is impacted only by locked-in losses (Thus, it is **0%** in the example above). The plot of **Closed Equity Risk** will only assume a positive value when **Closed Risk** is positive. **Closed Risk** is positive when the [Locked-In Profits](#) are less than zero. If there are [Locked-In Profits](#) then the **Closed Risk** is zero and the **Closed Equity Risk** is zero.

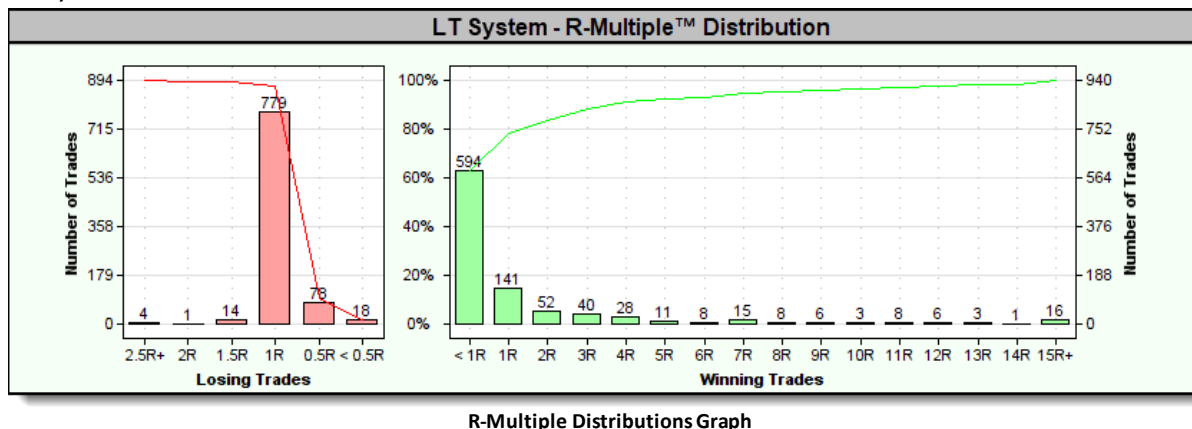
Now consider how **Closed Equity Risk** differs:

Total Equity	\$150,000
Closed Equity	\$100,000
Open Equity	\$50,000
Open Risk	\$60,000
Locked-in Profits	-\$10,000
Closed Risk	\$10,000
Closed Equity Risk	Closed Risk / Closed Equity = \$10,000 / \$100,000 = 10.00%

Section 9 – R-Multiple Distribution Graph

The concept of an R-Multiple™ was pioneered in 1993 by trader Chuck Branscomb, who explains that this technique is "The most important way to look at systems." The idea came about as a way to equate all markets and get away from looking at expectation in currency terms." R-Multiple™s were popularized by Dr. Van Tharp in his book, "Trade Your Way to Financial Freedom".

An R-Multiple™ is simply the profit or loss for a given trade divided by the entry risk. The entry risk is defined as the difference between the entry price and the stop price at the time of the entry.



R-Multiple Distributions Graph

The R-Multiple™ Distribution graph accounts for every closed trade in the simulation, both winning (green) and losing (red) trades. For the test reflected in the graph above, there were a total of 807 trades. Of 292 winning trades, 105 fell in the range between $0R < 1R$. Continuing to the right, 47 winning trades had R-Multiple™s between $1R < 2R$, and 38 trades fell in the bin of trades in the range $2R < 3R$, etc. Note that 14 winning trades had R-Multiple™s of $15R$ or greater.

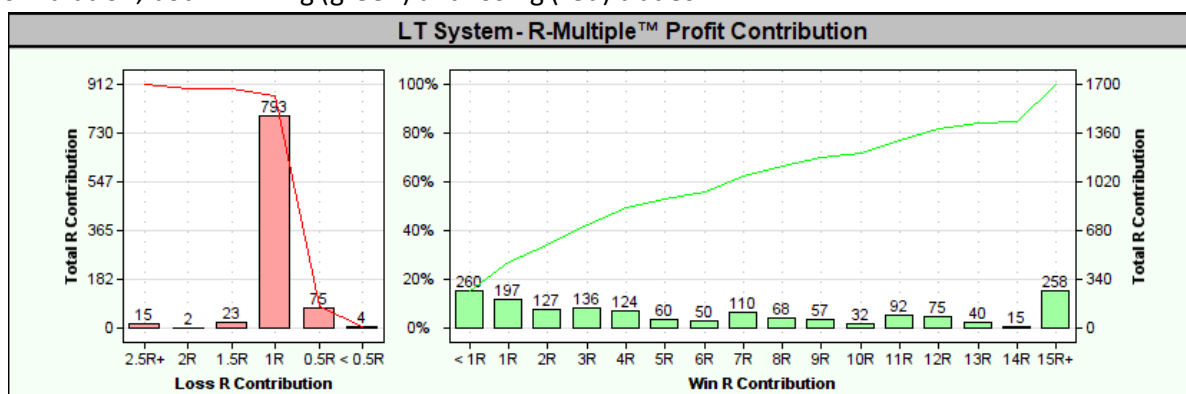
The histogram of Losing Trades (above, left) shows that the vast majority of losers were of a magnitude of $-1.5R$ or less, with 393 trades (out of a total of 515 losing trades) less than $-1.5R$.

The thin lines above the bars of the histogram are cumulative plots. Note that two scales apply to the vertical axes: on the outside is the Number of Trades, and on the inside is the Cumulative Percentage. For Winning Trades, the starting point of the cumulative plot ($<1R$ on the horizontal axis) contains 105 trades. But the next data point (the $1R < 2R$ bin on the horizontal axis) grows to 152 trades ($105+47$) on the right vertical axis, and encompasses slightly more than 50% of all winning trades, according to the left vertical axis.

As can be clearly seen, the Losing Trades were relatively well contained: The cumulative plot line shows at a glance that roughly 90% of losers were of a magnitude of $-1.5R$ or less.

Section 10 – R-Multiple Contribution Graph

The [R-Multiple™](#) Profit Contribution graph in the series shows Profit Contribution as a function of R-Multiple™s. Like the [R-Multiple™ Distribution graph](#), it accounts for every closed trade in the simulation, both winning (green) and losing (red) trades.



R-Multiple Contributions Graph

Each bar of the histogram simply represents the sum of the [R-Multiple™](#) value of all the trades in that particular bin.

For instance, the [R-Multiple™](#) Profit Contribution for the 14R bin below (which contains all trades in the range **14R < 15R**) ...is 57. This comes from 4 trade with slightly more than 14 each, **4 trades x 14R = 56R**.

As in the previous graph, the thin lines above the bars of the histogram are cumulative plots. Again, note that two scales apply. On the right vertical axis is the Total R Contribution, on the left vertical axis is the Cumulative Percentage, and they be interpreted in the same way as in the preceding example: For the test reflected in the chart below, about 60% of the total profits-in R-came from the cumulative contribution of the bins of 8R or greater.

Section 11 – Yearly Performance Summary

The Yearly Performance Summary shows the performance for each individual year of the test.

Yearly Performance Summary							
Year	Days	Closed Balance	End Total Equity	End VADI	Total Equity Gain	Gain %	# Trades
1996	365	1,115,804.31	1,134,037.56	1,134,037.56	134,037.56	13.40%	10
1997	365	1,177,029.94	1,233,032.29	1,233,032.29	98,994.73	8.70%	30
1998	365	1,516,282.77	1,551,677.04	1,551,677.04	318,644.74	25.80%	31
1999	365	1,522,242.37	1,576,775.48	1,576,775.48	25,098.44	1.60%	62
2000	366	1,451,887.74	1,542,421.72	1,542,421.72	-34,353.75	-2.20%	67
2001	365	2,312,215.97	2,337,467.81	2,337,467.81	795,046.09	51.50%	70
2002	365	2,289,045.45	2,424,464.75	2,424,464.75	86,996.93	3.70%	66
2003	365	2,422,931.36	2,563,120.93	2,563,120.93	138,656.19	5.70%	97
2004	366	2,683,932.01	2,702,329.67	2,702,329.67	139,208.73	5.40%	86
2005	365	2,771,772.05	2,885,446.96	2,885,446.96	183,117.29	6.80%	92
2006	365	2,891,585.97	2,994,483.97	2,994,483.97	109,037.01	3.80%	98
2007	365	3,090,525.87	3,135,728.38	3,135,728.38	141,244.41	4.70%	111
2008	366	6,273,818.28	6,526,135.71	6,526,135.71	3,390,407.33	108.10%	157
2009	365	6,019,715.24	6,252,145.45	6,252,145.45	-273,990.26	-4.20%	96
2010	365	8,679,982.70	9,198,279.31	9,198,279.31	2,946,133.86	47.10%	106
2011	365	8,782,147.21	8,885,224.97	8,885,224.97	-313,054.34	-3.40%	135
2012	366	9,495,589.71	10,209,413.15	10,209,413.15	1,324,188.18	14.90%	99
2013	365	17,062,126.88	18,058,014.64	18,058,014.64	7,848,601.49	76.90%	106
2014	365	26,753,093.89	28,349,422.25	28,349,422.25	10,291,407.62	57.00%	87
2015	365	33,425,915.49	33,857,635.17	33,857,635.17	5,508,212.92	19.40%	89
2016	366	34,323,721.36	35,373,336.32	35,373,336.32	1,515,701.15	4.50%	89
2017	174	34,497,933.03	34,497,933.03	34,497,933.03	-875,403.29	-2.50%	50

Yearly Performance Summary

The following table lists the columns and a description of their contents:

Property Name:	Descriptions:
Year	The Calendar Year for which the line of information applies.
Days	The number of calendar days in the trading year. This column will have a value of 365 for complete years (366 for Leap Years, less for partial years at the beginning or end of the simulation).
Closed Balance	The end-of-year Closed Equity balance. Composed of the Starting Balance, plus the cumulative profit (or loss) from all closed-out trades during the calendar year.
End Total Equity	Total value of the investment account for the year (includes open positions using current market prices).
Total Equity Gain	Equals End Total Equity - Starting Balance. (After the 1st calendar year, Starting Balance is simply End Total Equity from the previous year.)

Property Name:	Descriptions:
Gain%	Total Equity Gain divided by Starting Balance for the calendar year. (After the 1st calendar year, Starting Balance is simply End Total Equity from the previous year.)
Trades	The number of trades for the calendar year.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 4008

Section 12 – System Parameter Settings

The System Run Parameters section lists the specific values of the parameters for a given test. This information appears at the bottom of the System's Summary.

This first image shows the parameters as they appear when the suite has a single system:

System Parameter Settings	
Donchian w Rule Labels Parameters	
Allocation Percent	100.00%
Order Risk Per Trade (%)	2.00%
Entry Breakout (days)	22
Entry Offset (ATR)	-0.40
Stop (ATR)	1.80
Exit Breakout (days)	14
Exit Offset (ATR)	0.00
ATR (days)	39
Hold Initial Stop	True
Use Reversal Exit	True
Ratchet Exit Stop	True
Max Margin Equity (%)	100.00%
Check Margin on Fill	True
Portfolio	Futures:All Liquid
MACD Long Average (days)	300
MACD Short Average (days)	75
Trade Direction	Trade All

System Parameter Settings in a Single System Suite

This next image shows the parameters two sets of parameters in a multiple system suite:

System Parameter Settings			
Bollinger Breakout System Parameters		Donchian w Rule Labels Parameters	
Allocation Percent	100.00%	Allocation Percent	100.00%
Order Risk Per Trade (%)	1.00%	Order Risk Per Trade (%)	1.00%
Close Average (days)	80	Entry Breakout (days)	22
Entry Threshold (std. dev.)	2.00	Entry Offset (ATR)	-0.40
Exit Threshold (std. dev.)	1.00	Stop (ATR)	1.80
Portfolio	Futures:All Available	Exit Breakout (days)	14
Trade Direction (Long/Short/All)	Trade All	Exit Offset (ATR)	0.00
		ATR (days)	39
		Hold Initial Stop	True
		Use Reversal Exit	True
		Ratchet Exit Stop	True
		Max Margin Equity (%)	100.00%
		Check Margin on Fill	True
		Portfolio	Futures:All Liquid
		MACD Long Average (days)	300
		MACD Short Average (days)	75
		Trade Direction	Trade All

System Parameter Settings in a Multiple System Suite

When a multiple system suite has systems that have are assigned by the [Allocation Manager](#) **0%**, those system parameters are not displayed in the summary report, and the system names with zero allocation are not listed.

When a system in a multiple system suite is given an allocation of zero percent, Trading Blox will remove that system from the list of selected systems. To retain the selected systems in a suite, and allow one or more of them to influence the performance of the other systems, add the [Virtual System](#) blox to each of the systems that could be assigned a **0%** allocation so it won't be removed from the suite's list of selected systems.

Any system using the [Virtual System](#) parameter set to **TRUE**, will stay in the suite of selected systems, but there performance will not add or detract from the active systems where the Virtual System blox is set to **FALSE**. By adding the Virtual System blox to the systems in a suite, the enabling of various system combinations can easily be tested tested.

This next system parameter list of systems shows the account allocation for each of the five systems selected for the testing suite. Three of the systems have their [Virtual System](#) set to **TRUE** and show an allocation of **100%**. The other two systems show the [Virtual System](#) is set to **FALSE**, so those system's are included in the summary reporting numbers.

System Parameter Settings			
ADX System Parameters		Bollinger Breakout System Parameters	
Allocation Percent	100.00%	Allocation Percent	60.00%
Size of Unit (contracts or shares)	1	Order Risk Per Trade (%)	1.00%
ATR (days)	39	Close Average (days)	80
ADX Periods to use (bars)	10	Entry Threshold (std. dev.)	2.00
ADX Trend Limit (adx)	50.00	Exit Threshold (std. dev.)	1.00
DI Period to use (bars)	16	Virtual System	False
Use and Hold ATR Stops	True	Portfolio	Futures:All Available
ATR Stop (fraction)	1.80	Trade Direction (Long/Short/All)	Trade All
Virtual System	True		
Portfolio	Futures:All Liquid		
Trade Direction (Long/Short/All)	Trade All		
Donchian w Rule Labels Parameters		RSI Trend Catcher Parameters	
Allocation Percent	40.00%	Allocation Percent	100.00%
Order Risk Per Trade (%)	1.00%	Order Risk Per Trade (%)	1.00%
Entry Breakout (days)	22	RSI Length (Bars)	130
Entry Offset (ATR)	-0.40	Entry RSI Threshold	56.00
Stop (ATR)	1.80	Exit RSI Threshold	51.00
Exit Breakout (days)	14	Stop Width (ATR)	5.00
Exit Offset (ATR)	0.00	ATR Bars	20
ATR (days)	39	Hold Initial Stops	True
Hold Initial Stop	True	Virtual System	True
Use Reversal Exit	True	Portfolio	Futures:All Liquid
Ratchet Exit Stop	True	MACD Long Average (days)	300
Max Margin Equity (%)	100.00%	MACD Short Average (days)	75
Check Margin on Fill	True	Trade Direction	Trade All
Virtual System	False		
Portfolio	Futures:All Liquid		
MACD Long Average (days)	300		
MACD Short Average (days)	75		
Trade Direction	Trade All		
Triple Moving Average Parameters			
Allocation Percent	100.00%		
Order Risk Per Trade (%)	1.00%		
Max Margin Equity (%)	100.00%		
Check Margin on Fill	True		
Buy On Crossover	True		
Long Moving Average (days)	150		
Medium Moving Average (days)	70		
Short Moving Average (days)	20		
Initial Stop Type (ATR/LMA)	ATR Stop		
ATR Average (days)	20		
Stop (ATR)	2.50		
Close through Short MA	True		
Hold Initial Stops	False		
Virtual System	True		
Portfolio	Futures:All Liquid		
Trade Direction (Long/Short/All)	Trade All		

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 4010

Section 13 – Instrument Performance Summary

The Instrument Performance Summary shows the performance for each individual market for each portfolio in the test.

Instrument Performance Summary													
Symbol	Wins	%	Losses	%	Trades	Win Months	%	Loss Months	%	Avg. Win %	Avg. Loss %	Avg. Trade %	% Profit Factor
AEX	43	38.10%	70	61.90%	113	186	72.10%	72	27.90%	1.34%	0.83%	0.00%	1
BL2	11	57.90%	8	42.10%	19	171	66.30%	87	33.70%	0.61%	0.43%	0.17%	1.97
BO2	13	41.90%	18	58.10%	31	161	62.40%	97	37.60%	1.57%	0.73%	0.23%	1.55
DA	30	68.20%	14	31.80%	44	192	74.40%	66	25.60%	0.25%	0.13%	0.13%	4.11
DJ	34	44.70%	42	55.30%	76	212	82.20%	46	17.80%	1.11%	0.59%	0.17%	1.52
EBL	26	47.30%	29	52.70%	55	202	78.30%	56	21.70%	0.69%	0.82%	-0.11%	0.75
EBM	20	71.40%	8	28.60%	28	191	74.00%	67	26.00%	0.94%	0.72%	0.46%	3.25
EBS	2	66.70%	1	33.30%	3	198	76.70%	60	23.30%	3.06%	0.81%	1.77%	7.54
ED	8	88.90%	1	11.10%	9	164	63.60%	94	36.40%	2.15%	0.73%	1.83%	23.54
EM	6	54.50%	5	45.50%	11	163	63.20%	95	36.80%	1.46%	0.24%	0.69%	7.24
EMD	34	56.70%	26	43.30%	60	206	79.80%	52	20.20%	1.19%	0.71%	0.37%	2.21
ES	32	52.50%	29	47.50%	61	189	73.30%	69	26.70%	0.89%	0.73%	0.12%	1.35
FC	31	55.40%	25	44.60%	56	190	73.60%	68	26.40%	0.86%	0.61%	0.20%	1.73
FF	6	60.00%	4	40.00%	10	154	59.70%	104	40.30%	2.56%	0.14%	1.48%	27.19
GC2	38	63.30%	22	36.70%	60	200	77.50%	58	22.50%	1.39%	0.76%	0.60%	3.14
GI	45	48.40%	48	51.60%	93	210	81.40%	48	18.60%	0.97%	0.50%	0.21%	1.82
JT	48	40.70%	70	59.30%	118	220	85.30%	38	14.70%	0.49%	0.10%	0.14%	3.56
JY	36	45.00%	44	55.00%	80	193	74.80%	65	25.20%	1.51%	0.73%	0.27%	1.68
KW2	24	55.80%	19	44.20%	43	183	70.90%	75	29.10%	0.90%	0.73%	0.18%	1.54
LB	25	50.00%	25	50.00%	50	197	76.40%	61	23.60%	0.86%	0.36%	0.25%	2.41
LCO	45	48.90%	47	51.10%	92	202	78.30%	56	21.70%	1.33%	0.74%	0.27%	1.73
LGO	45	52.30%	41	47.70%	86	192	74.40%	66	25.60%	1.46%	0.74%	0.41%	2.17
MP	13	56.50%	10	43.50%	23	183	70.90%	75	29.10%	1.64%	0.76%	0.60%	2.82
NK	60	50.40%	59	49.60%	119	191	74.00%	67	26.00%	0.77%	0.76%	0.01%	1.04
O2	12	38.70%	19	61.30%	31	160	62.00%	98	38.00%	0.62%	0.36%	0.02%	1.09
RB2	67	50.40%	66	49.60%	133	196	76.00%	62	24.00%	0.91%	0.73%	0.09%	1.26
RM2	20	62.50%	12	37.50%	32	227	88.00%	31	12.00%	1.00%	0.46%	0.45%	3.63
RY	36	48.60%	38	51.40%	74	207	80.20%	51	19.80%	0.72%	0.45%	0.12%	1.53
SB2	22	62.90%	13	37.10%	35	162	62.80%	96	37.20%	0.84%	0.74%	0.26%	1.94
SM2	27	65.90%	14	34.10%	41	197	76.40%	61	23.60%	0.55%	0.74%	0.11%	1.43
TU2	9	50.00%	9	50.00%	18	164	63.60%	94	36.40%	2.10%	0.72%	0.69%	2.91
VX	13	59.10%	9	40.90%	22	238	92.20%	20	7.80%	1.55%	0.74%	0.61%	3
YG	19	61.30%	12	38.70%	31	183	70.90%	75	29.10%	0.52%	0.42%	0.15%	1.93
YI	16	50.00%	16	50.00%	32	176	68.20%	82	31.80%	1.13%	0.46%	0.34%	2.46
YM	24	53.30%	21	46.70%	45	208	80.60%	50	19.40%	1.10%	0.73%	0.25%	1.73

Instrument Performance Summary

The following table lists the columns and a description of their contents:

Property Name:	Descriptions:
Symbol	The symbol for the market.
Wins	The number of winning trades and the percentage of the total trades for a given market.
Losses	The number of losing trades and the percentage of the total trades for a given market.
Trades	The total number of trades for a given market.
Win Months	The number of months and the percentage of the total testing months a given market is profitable
Loss Months	The number of months and the percentage of the total testing months a given market is unprofitable

Property Name:	Descriptions:
Avg. Win %	The average winning trade return.
Avg. Loss %	The average losing trade return.
Avg. Trade%	The average trade return including both winning and losing trades.
% Profit Factor	The Percent Profit Factor for the given market in isolation.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 162

Section 14 – Trading Performance and Statistics

The following section provides detailed trading performance statistics, as well as a list of the assumptions used for the particular test.

Many of the statistics listed below deal with risk and return. [Appendix I - Risk and Return](#) contains a supplement that discusses these concepts in more detail than is permitted here. If you are new to the world of trading system design, or would like an overview of the concepts underlying the use of risk-adjusted return measurements, please skip ahead to [Appendix I - Risk and Return](#) before proceeding with this section.

Trading Performance and Statistics Sections:

[Summary Performance Ratios](#)

[Drawdown Statistics](#)

[Miscellaneous Performance Statistics](#)

[Win/Loss, Profit Factor, and Expectancy](#)

Summary Performance Ratios:

Statistic:	Description:
CAGR% ^ Top	Compounded Annual Growth Rate; this is the annual percentage rate at which an account must continually grow over the period of the study, in order to attain the ending value from the starting value. Synonymous with Geometric Mean Return.
RAR% ^ Top	Regressed Annual Return. Slope of the linear regression of all the points in the equity curve. See page 186 of Way of the Turtle for more information.
Maximum Total Equity Drawdown% ^ Top	Max Total Equity Drawdown (Max DD) is a one-time event that reflects the largest retracement relative to a previous equity high in the entire simulation. It is based on a daily, marked-to-the-market assessment. Expressed as a percentage, this statistic is sampled peak-to-valley, using Total Equity . It conveys the maximum pain an investor would have had to endure over the life of the study in order to achieve the resulting return.
MAR Ratio ^ Top	Often referred to as a pain-to-gain ratio, this risk-adjusted return metric was developed by Managed Accounts Review for ranking Commodity Trading Advisors (CTAs). ~~~~~ ~~~~~ $\text{MAR Ratio} = \text{CAGR} / \text{Max Total Equity Drawdown}$ ~~~~~ ~~~~~

Statistic:	Description:
R-Cubed (R3) ^ Top	Robust Risk/Reward Ratio. uses the RAR% in the numerator and the length-adjusted average maximum draw down in the denominator. The average maximum draw down is computed by taking the five largest draw downs and dividing by 5. The length adjustment is made by taking the average maximum draw down length in days and dividing it by 365 and then multiplying that number by the average maximum draw down. See page 188 of Way of the Turtle for more information.
Modified Sharpe Ratio ^ Top	The Sharpe Ratio is the classic measure of return vs. risk. Developed by Nobel Laureate and (now) Stanford professor William F. Sharpe, it divides excess return by standard deviation to determine reward per unit of risk. (The higher the Sharpe ratio, the better the risk-adjusted performance.) This modified version divides annualized return by annualized standard deviation of returns, using monthly data points. Exclusion of the risk-free-rate in the numerator makes this measure less sensitive to changes in leverage: <pre> ~~~~~ ~~~~~ Return = 12 * Average Monthly Return Risk Free Rate (excluded) Annualized Std Dev = Square Root (12) * Std Dev Monthly Returns Modified Sharpe Ratio = Annualized Return / Annualized Std Dev ~~~~~ ~~~~~ </pre>
Annual Sharpe Ratio ^ Top	This more traditional form of the Sharpe Ratio divides excess annual return by the standard deviation of annual returns to determine reward per unit of risk. Uses actual annual calendar returns: Return: Compounded Annual Growth Rate Risk Free Rate: Computed from the annual risk free rate as set in the TradingBlox.ini file converted TO a compounded daily rate times 365.251. Std Dev: Standard Deviation of Annual Returns <pre> ~~~~~ ~~~~~ Annual Sharpe Ratio = (Return - RFR) / Std Dev ~~~~~ ~~~~~ </pre>

Statistic:	Description:
Robust Sharpe Ratio ^ Top	The robust sharpe ratio is RAR% divided by the annualized standard deviation of the monthly returns. See page 189 of Way of the Turtle for more information
Annual Sortino Ratio ^ Top	Identical in form to the Annual Sharpe Ratio, but uses only downside deviation (negative data points) in its denominator, whereas the Sharpe Ratio uses overall standard deviation (which contains both upside and downside deviation). The Sortino Ratio is typically favored by those who believe that the performance of trend-following CTAs, and others, is unfairly penalized by the Sharpe Ratio's inclusion of upside volatility in its calculation. The higher the Sortino Ratio, the better: Risk Free Rate (RFR): Computed from the annual risk free rate as set in the TradingBlox.ini file converted to a compounded daily rate times 365.251. Return: Compounded Annual Growth Rate Semi-Deviation: Std Dev of negative Annual returns ~~~~~ $\text{Annual Sortino Ratio} = (\text{Return} - \text{RFR}) / \text{Semi-Deviation}$ ~~~~~ Note: In the calculation of standard deviation for the Sortino Ratio, "n" equals the entire population and includes zeroes for positive returns.
Monthly Sharpe Ratio ^ Top	Identical in form to the Annual Sharpe Ratio, but based on monthly data points: Return: Average of the monthly returns. Risk Free Rate: Converts the annual risk free rate as set in the TradingBlox.ini file to a compounded monthly return. Std Dev: Standard Deviation of Monthly (calendar) Returns ~~~~~ $\text{Monthly Sharpe Ratio} = (\text{Return} - \text{RFR}) / \text{Std Dev}$ ~~~~~
Daily Sharpe Ratio ^ Top	Identical in form to the Annual Sharpe Ratio, but based on daily data points: Return:

Statistic:	Description:
	Average of the daily returns. Risk Free Rate: Converts the annual risk free rate as set in the TradingBlox.ini file to a compounded daily return. Std Dev: Standard Deviation of Monthly (calendar) Returns Daily Sharpe Ratio = (Return - RFR) / Std Dev
Daily Geometric Sharpe Ratio ^ Top	Identical in form to the Annual Sharpe Ratio, but based on daily data points: Return: Compounded Daily Return. Risk Free Rate: Converts the annual risk free rate as set in the TradingBlox.ini file to a compounded daily return. Std Dev: Standard Deviation of Monthly (calendar) Returns Daily Sharpe Ratio = (Return - RFR) / Std Dev
Monthly Sortino Ratio ^ Top	Identical in form to the Annual Sortino Ratio, but based on monthly data points: (See Annual Sortino Ratio)
Calmar Ratio ^ Top	Another gain-to- pain ratio, used to determine return relative to drawdown (downside) risk. The higher the Calmar ratio, the better: The Calmar Ratio is identical to the MAR Ratio with the exception of their denominators. MAR uses the Max DD over the life of the simulation, based on a daily, marked-to-the-market assessment. While the Calmar's denominator uses the Max DD over the life of the simulation, it is based on the worst month-end to month-end Total Equity data points. Note: Unlike some forms of the Calmar Ratio, which consider only the last 3 years of returns, Trading Blox uses the entire duration of the simulation. Calmar Ratio = CAGR / Max Drawdown (monthly data points)

Drawdown Statistics

Statistic Name:	Description:
Longest Total Equity ^ Top Drawdown	A one time event that reflects the longest duration drawdown in Total Equity over the life of the simulation. It is measured from previous equity peak to new equity peak, and expressed in months.
Maximum Monthly Total Equity DD% ^ Top	Like Maximum Drawdown, this is a one-time event that reflects the largest retracement relative to a previous equity high in the entire simulation. It is based on a month-end to month-end, marked-to-the-market assessment. Expressed as a percentage, this statistic is calculated peak-to-valley using Total Equity .
Average Max Drawdown (%) ^ Top	The average of the 5 largest percent drawdowns based on total equity.
Average Max Drawdown Length ^ Top	The average of the 5 largest total equity drawdowns by test days, converted to months.
Maximum Closed Equity Drawdown% ^ Top	A one-time event that reflects the largest retracement relative to a previous equity high in the entire simulation. It is based on a daily, marked-to-the-market assessment. Expressed as a percentage, this statistic is calculated peak-to-valley using Closed Equity .
Average Closed Equity Drawdown% ^ Top	The average of all Closed Equity drawdowns. If Monday is a Closed Equity peak, Tuesday shows a 2% retracement, and new peak is hit on Wednesday, then this one-day, 2% drawdown is stored, and averaged with all other drawdowns. This is significantly different from the way the other drawdown statistics are calculated.

Miscellaneous Performance Statistics:

Statistic Name:	Description:
Round Turn per Million ^ Top	The number of contracts traded per million currency on average. This number takes the average number of contracts traded divided by the average equity in millions. Trading Blox computes the round turns per million as follows: At the end of each month, the total round turns for the month are multiplied by 1,000,000 divided by the starting total equity for the month. The total turns per million is then incremented by this monthly number. <pre> ----- turnsPerMillion = monthRoundTurns * (ONE_MILLION / monthStartingTotalEquity) turnsPerMillionTotal = turnsPerMillionTotal + </pre>

Statistic Name:	Description:
	<pre>turnsPerMillion ----- ----- At the end of the test, the average turns per million is computed as follows: ----- ----- averageTurnsPerMillion = (turnsPerMillionTotal / totalTestingMonths) * 12 ----- ----- </pre>
End Account Balance ^ Top	The ending balance after all trades are closed out at the end of the simulation (at this point Closed Equity and Total Equity are equal).
Highest Total Equity ^ Top	Highest Total Equity achieved at any point during the simulation, using current market prices to value open positions.
Highest Closed Equity ^ Top	Highest Closed Equity achieved at any point during the simulation. (Sometimes referred to as Closed Trade Equity.)
Total Commissions ^ Top	Total commissions paid out for all positions.
Total Slippage ^ Top	The total slippage (in currency) incurred, for all trades. See "Slippage Percent" for definition.
Total Forex Carry ^ Top	The money earned or paid as a result of Forex Carry (see: Forex Carry Calculations for details)
Earned Interest ^ Top	Total amount of interest (in currency), hypothetically earned on cash, or on T-Bill interest in a futures account.
Margin Interest ^ Top	Total amount of interest (in currency), hypothetically paid for money borrowed on margin.

Win/Loss, Profit Factor, and Expectancy:

Statistic Name:	Description:
Wins ^ Top	Number of winning trades; also displayed as a percentage of Total Trades.
Losses ^ Top	Number of losing trades; also displayed as a percentage of Total Trades.
Total Trades ^ Top	Total number of trades.
Winning Months ^ Top	Number of profitable months; also displayed as a percentage of total months in the simulation.

Statistic Name:	Description:
Losing Months ^ Top	Number of unprofitable months; also displayed as a percentage of total months in the simulation.
Total Win Dollars ^ Top	The total won on profitable trades, in system wide base currency.
Total Loss Dollars ^ Top	The total lost on unprofitable trades, in system wide base currency.
Average Risk Percent ^ Top	The average percent of equity risked per trade.
Average Win Percent ^ Top	The average winning trade as a percentage of equity.
Average Loss Percent ^ Top	The average losing trade as a percentage of equity.
Average Trade Percent ^ Top	The average trade as a percentage of equity. This number will be positive for winning systems and negative for losing systems.
Profit Factor ^ Top	Total Win Dollars / Total Loss Dollars
Profit Factor Ratio ^ Top	Average Win Dollars / Average Loss Dollars
Percent Profit Factor ^ Top	Total Wins in Percent / Total Losses in Percent - This is a more useful statistic than Profit Factor since it does not weight later trades more heavily than earlier trades like Profit Factor does. To compute Total Wins in Percent, we add up the percentage won for each trade that is a winning trade as a percentage of the account equity at the time the trade was initiated. To compute Total Losses in Percent, we add up the percentage lost for each trade that is a losing trade as a percentage of the account equity at the time the trade was initiated.
Expectation ^ Top	Sometimes known as Expectancy, this statistic shows how much one expects to gain for every amount bet or risked on a given trade. Numbers greater than 0.0 are winning systems, less than 0.0 are losing systems. Average Gain of All Trades / Average Risk of All Trades, or, $\frac{(\text{TotalWinPercent} - \text{TotalLossPercent})}{\text{TotalRiskPercent}}$
Margin to Equity Ratio ^ Top	The summary margin to equity ratio is the MarginEquityTotal / TotalTradingDays, where the MarginEquityTotal is a sum of each day's

Statistic Name:	Description:
	margin to equity percent. The equity used for this is system total equity. So this number is the average margin to equity percent over the course of the test, using the system total equity as the equity basis.

Edit Time: 2/18/2022 3:37:18 PM

Topic ID#: 4009

Log Files

Part

XV

Part 15 – Log Files

Trading Blox also creates several detailed logs when a test is run. Results from test are stored in file names that match the Log Name Preference setting. Log files are stored in the [Results](#) folder.

You can quickly open this folder from the File Menu using the [Results Menu Group](#). You can also navigate directly to the Results folder and open each of the output files. Be sure to close them or Save As before running another test or Trading Blox will complain.

Depending on the Preferences in the [Report General](#), and [Trade and Equity Logs](#) enabled for reporting, data may, or may not be logged to these files. These files are described in detail later in this section.

Log Name:	Descriptions:
Trade Log	A complete list of each trade in text format.
Filtered Trade Log	A complete list of each filtered trade.
Daily Equity Log	A daily log of the major equity values for each trading day.
Monthly Log	Shows the equity for each month.
Print Output File	A file in which all PRINT statements go when using the PRINT statement in Blox Basic Scripting.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 4201

Section 1 – Trade Log

The Trade Log contains an entry for every single trade in a test. The file is kept in the Results folder along with the other result files. The name of the file is "Trade Log.csv". An entry is placed in the trade log each time a trade closes. No open trades are logged here. The trades are logged in order of their completion. In order that it will always reflect the most recently executed test, the Trade Log is overwritten each time a new test is run.

Note that if "Log Trades" is disabled in Preferences (see [Report General](#)) nothing will be logged in this file. For large tests with thousands of parameter combinations, disabling this option will result in faster tests.

The Trade Log includes the following columns for any given trade:

Log Column Name:	Description:
Test	The test number when doing multiple parameter runs
System Number	System number based on position in System List
System Name	The system name
Symbol	Future/Stock symbol
Unit	Unit Number
Position	Position: Long or Short
Entry Date	Date the entry was logged
Entry Time	Time the entry was logged
Entry Order	Entry order price (price which initiated entry)
Stop Price	Predefined stop price
Risk %	Measurement of risk in currency, based on the entry and stop
Quantity	Number of shares or contracts
Entry Fill	The "actual" entry price after accounting for slippage
Entry DPP	The Dollars Per Point (actually system wide currency per point) used on the day of entry.
Entry Stock Split Ratio	The Stock split ratio on the day of entry.
Exit Date	Date the trade was closed
Exit Time	Time the trade was closed
Exit Order	Exit order price (price which initiated exit)
Exit Fill	The "actual" exit price after accounting for slippage

Log Column Name:	Description:
Exit DPP	The Dollars Per Point (actually system wide currency per point) used on the day of exit.
Dividends	The dividends accrued on this trade, if the dividend files are available.
Commission	The total commission for this trade.
Profit	Listed in system wide base currency
Profit Percent	Profit as a percentage of Base Equity. The base equity used is from the end of the day prior to the trade entry.
<u>Closed Equity</u>	New account balance after adding this trade's profit
<u>Total Equity</u>	<u>Total Equity</u> at time of exit
Draw down	Drawdown
Days in Trade	Number of days between entry and exit
R Multiple	The R Multiple of the trade
Custom Value	The custom value set for the trade in scripting
Rule Label	The Rule Label as set in scripting

Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 4202

Section 2 – FilteredTradeLog

The log also contains entries for each trade that was Filtered, for whatever reason. This is extremely useful if you find yourself asking, "Why didn't my system trade on this day when I expected it to?" There is a preference to have filtered trades included in this output or not. Be sure to turn this on if you want to see the filtered trades.

Lines for trades not taken could look like the following:

- For system 8, symbol ADBE on 1995-03-31, trade attempted but Money Manager set a unitSize of zero
- For BIIB on 1995-03-31, the volume of 3666 is less than required 10000
- For FLEX on 1995-11-20, cannot add 99300.00 additional margin to 978255.01 existing with only 998343.31 equity.

There are many reasons why a trade may not be taken, and you will see the reason in this file.

Log Column Name:	Description:
Test Number	The test number when running multiple parameter runs
Type of filter	Either Order Processing or Fill Processing
System Name	System Name
Symbol	Instrument symbol
Position	Either Long or Short
Quantity	The potential quantity for the order
Date	The date this order was rejected. Note that this is the instrument date for Order Processing types and the test date for the Fill Processing types.
Message	The reason the order was rejected.

See the [Order Filter](#) for additional information.

Section 3 – Daily Equity Log

The Daily Equity Log contains an entry for every single day of the test showing summary equity information as well as a limited summary of the open positions for each day. You can use the Equity Log to determine if a trade that appears like it should have been taken wasn't taken because Purchase Equity or Margin Equity was too high. The file is named "Daily Equity Log.csv" and is stored in the Results folder.

An entry is placed in the Equity Log for each trading day in a given historical test. This daily entry includes data on all open positions, for each day of the simulation. In order that it will always reflect the most recently executed test, the Equity Log is overwritten each time a new test is run.

The Equity Log includes the following columns:

Log Column Name:	Description:
Test Number	The test number
Date	Includes trading dates only
Total Equity	Total Equity is Closed Equity plus Open Equity
Closed Equity	Closed Equity at month end
Drawdown	Drawdown in Total Equity
Open Equity	Open Equity is the value of open positions marked to market
Cash	Closed Equity minus Purchase Equity
Purchase Equity	Equity used for Stock purchases
Margin Equity	Equity used as Margin for Futures purchases
Total Slippage	Total Slippage to date
Total Dividends	Dividends earned test to date.
Total Earned Interest	Interest earned test to date.
Total Margin	Margin paid test to date.
Total Commissions	Commissions test to date.
Total Carry	Forex carry cost test to date.

Section 4 – Monthly Equity Log

The Monthly Equity Log contains an entry for each month that is tested, providing a month-by-month view into the results. It is named "Monthly Equity Log.csv" and is stored in the Results folder. The log entry is placed at the end of the month. This log is overwritten each time a new test is run such that it will always reflect only the most recently executed historical simulation test.

Monthly Equity Log includes the following columns:

Log Column Name:	Description:
Test	Test number
Year	Year of the month entry
Month	Month of the entry
Days	Number of days in the month
Closed Balance	<u>Closed Equity</u> Balance at month end
<u>Total Equity</u>	<u>Total Equity</u> at month end
Total Equity Gain	<u>Total Equity</u> gain from the previous month in system wide base currency
Gain%	Percentage <u>Total Equity</u> gain from the previous month
C Risk %	Closed Equity Risk
T Risk %	Total Equity Risk
CT DD%	Closed Equity Drawdown
TE DD%	Total Equity Drawdown
#Trades	Number of Trades for this month

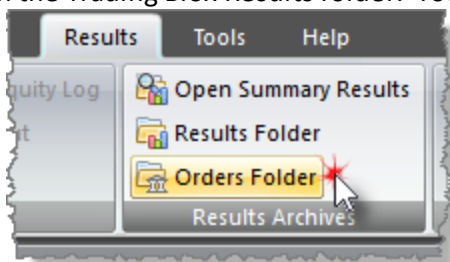
Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 4203

Section 5 – PrintOutput.csv

This file is used to record the output from all **PRINT** statements in Trading Blox Builder Edition. The [PrintOutput.csv](#) is a text file that is re-written every time a test is run. Most often the data contained in this file is comma delimited for easy use in other programs like Excel. It is also easily opened with the Windows Notepad.

Print Output.csv file is stored in the Trading Blox Results folder. You can quickly open this folder



from the [Results Menu Group](#):

Be sure to close all log files before you run another test so Trading Blox complain with a reminder to close the a log file. Another option is to open the copy of a log file that is saved in the test results folder. Log files saved in locations that are not in the Results folder don't need to be closed, but spreadsheet programs might not allow two different files with the same name to be open at the same time.

Monte Carlo Simulations

Part



Part 16 – Monte Carlo Simulations

A Monte Carlo simulation is one way of determining the robustness of a system, and in answering the question:

"What if the past had been just slightly different"?

Trading Blox's Monte Carlo feature simulates new equity curves that are similar to the actual test equity curve, but different in certain random ways. A typical Monte Carlo simulation might generate hundreds or thousands of new equity curves. How many it will generate is adjustable changing the value in the [Preference's Monte Carlo Parameters](#).

Trading Blox supports two different randomization algorithms for the simulation of the Monte Carlo equity curves:

Reordering and Sampling with Replacement.

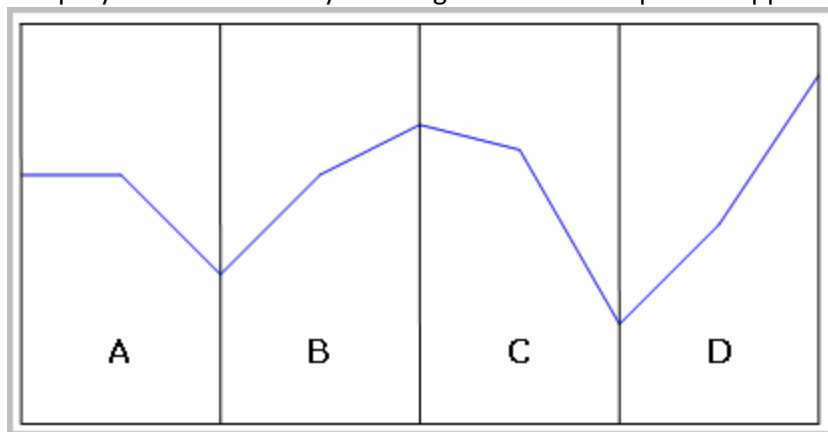
In each case, the new equity curves are generated by taking portions of the actual test equity curve and changing the order in how they happened.

In real trading, bad days occur together with a frequency that is much higher than one might attribute to purely random chance. This is because at the end of large trends, many markets seem to get carried along and then reverse at the same time. The period of a system's maximum draw-down is usually a statistically improbable series of down periods.

Trading Blox allows you to specify the number of days which make up each portion using the preference **"Sample Grouping Days"**. This Monte Carlo feature allows you to generate equity curves which maintain some of the auto-correlation found in actual the equity curves. Without this feature, Monte Carlo simulations tend to underestimate the potential for large or lengthy draw-downs because the randomization process will result in equity curves that don't often show the lengthy periods of negative returns since they are statistically unlikely.

Reordering

The equity curve is randomly rearranged so that each portion appears in a different order:



Consider an equity curve like the above divided into four sections. If you simply reorder then you could have orders:

ABDC

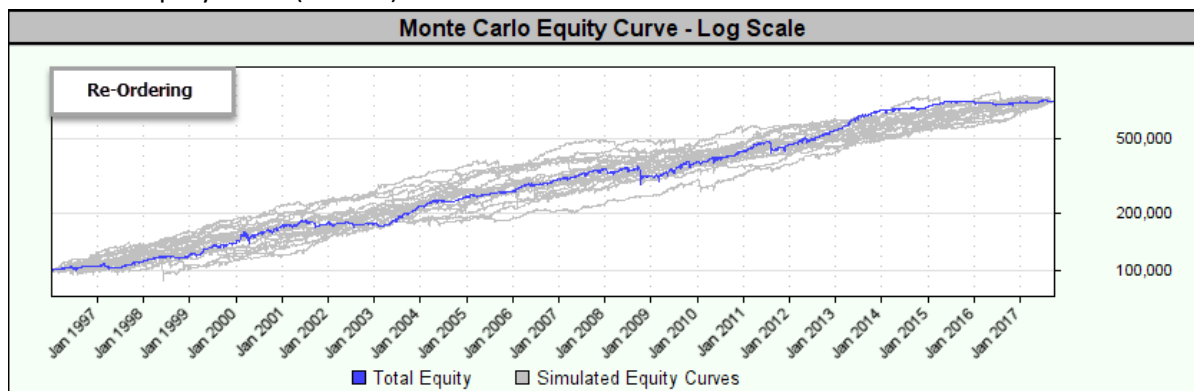
ABCD
ACBD
ACBC
BCDA
BDCA
DCAB
DCBA
CDBA

etc.

Notice how an equity curve that contains AC adjacent to each other, i.e. ACDB, BACD, etc.. This placement will have a larger draw-down than the original equity curve because the reordered curve combines two down periods which were not adjacent in the original equity curve.

However since each of the individual sections of the equity curve represents a net percentage change in equity, any reordering of the equity curves will result in exactly the same endpoint and therefore will result in the same CAGR%.

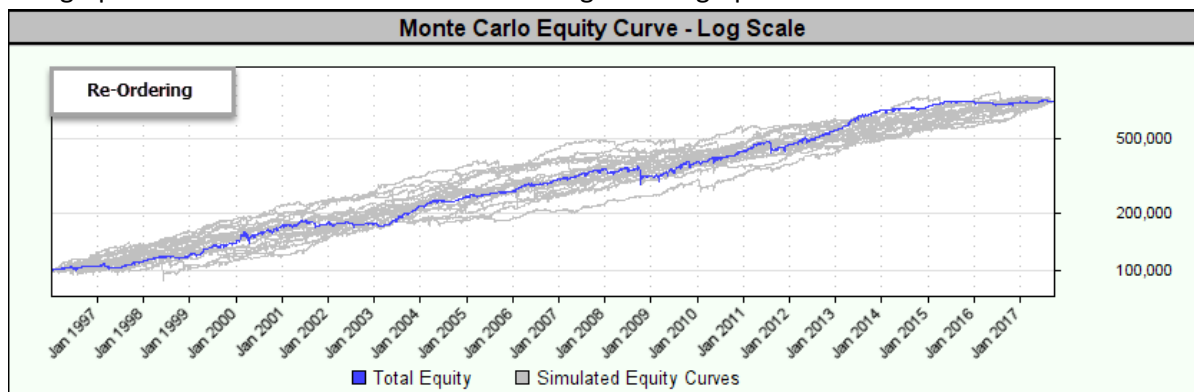
The following shows the results of plotting the reordered equity curves (in gray) against the actual simulated equity curve (in blue):



Monte Carlos Equity Curve - Re-Ordering

Notice how all the curves each end at the same point when Re-Ordering is the sampling process.

This graph shows the Monte Carlo - Re-Ordering CAGR% graph:



Monte Carlos Equity Curve - Re-Ordering - CAGR%

Sampling with Replacement

The other method of simulating a new equity curve is called "Sampling with Replacement". This method generates a new equity curve by randomly selecting from the actual test equity curve with the additional provision that portions of the original equity curve can be used more than once.

Thus you could have excellent curves like:

BDBD

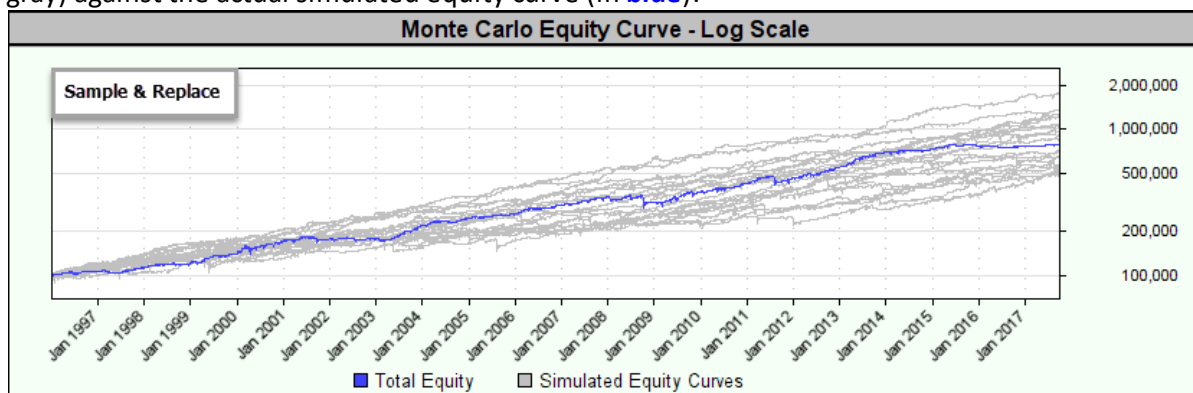
BDDB

BBBB

or even:

DDDD

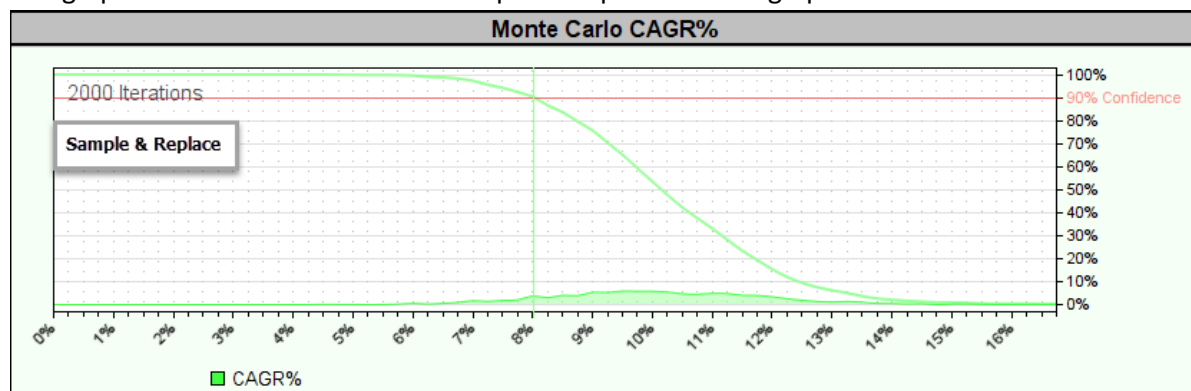
The following shows the results of plotting the equity curves from Sampling with Replacement (in gray) against the actual simulated equity curve (in blue):



Monte Carlos Equity Curve - Sample & Replace

Notice how each of the curves has a different endpoint representing a different CAGR%.

This graph shows the Monte Carlo - Sample & Replace CAGR% graph:



Monte Carlos Equity Curve - Sample & Replace - CAGR%

Many of the Monte Carlo graphs plot the distribution and cumulative distribution figures for various test measures, CAGR%, MAR Ratio, Maximum Draw-down etc.

Note the red line in each of the CAGR% graphs. The red line labeled 90% confidence is where the red line intersects with the cumulative equity curve on the graph. This means that 90% of the simulation's tests showed an equity curve with a CAGR% greater than percentage value shown by the vertical green line.

Some Cautions

Beginning traders and system testers often look at particular test results and think that they mean more than they actually represent. The future will not look like the past but will probably look something like the past. A simulation graph like the above helps one think in terms of the reality that the future will be one of many different possibilities, some better than the test results some worse than the test results.

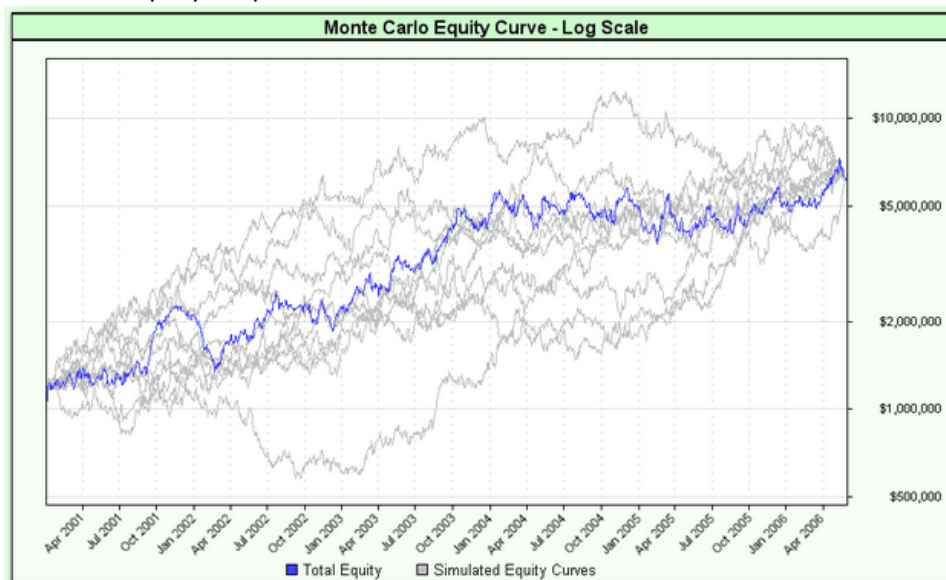
The confidence levels used by Trading Blox indicate only that a certain percentage of the results of the simulation are better for the given measure. The use of the term Confidence Level is common in the industry for this purpose. The use of the term does not imply that the test indicates that the future will result in a certain probability of a particular outcome. In other words, just because a test shows a 95% confidence level of a certain return or other measure does not mean that Trading Blox is predicting that there is a 95% chance of exceeding that measure in actual trading.

The confidence level can be set by the [Monte Carlo](#) preference settings. Some traders prefer to use 90% or 95% confidence levels.

Section 1 – Equity Curve Graph

The Monte Carlo Equity Graph plots a number of the Monte Carlo simulation alternate equity curves on the same graph with the tests actual equity plotted in blue.

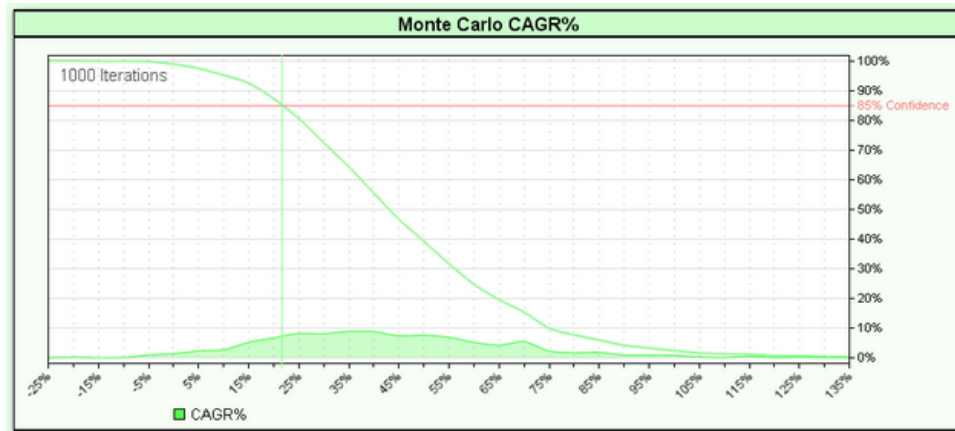
This is a Monte Carlo Equity Graph:



The number of equity curves to plot can be set with the "Maximum Equity Curves to Graph" preference found here: [Monte Carlo](#). The above graph shows a value of 10 for "Maximum Equity Curves to Graph".

Section 2 – Returns Graph

The Monte Carlo Returns Graph shows a histogram of the distribution and cumulative distribution for the returns expressed as CAGR%.



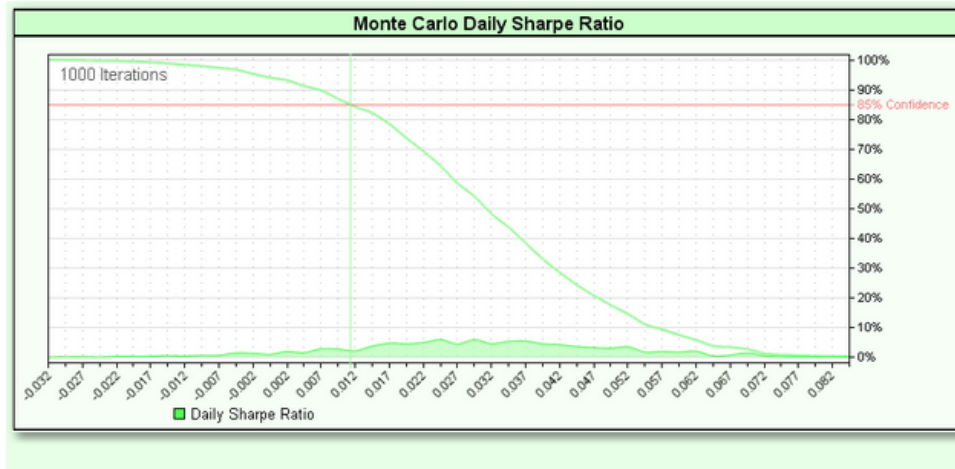
In the above example, the 90% confidence level corresponds to about 22% CAGR, meaning that 85% of the simulated equity curves showed a return of 22% or better. The distribution is centered around 35% but shows some simulated curves with a CAGR% as low as -25% and as high as 125%.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 174

Section 3 – Sharpe Graph

The Monte Carlo Sharpe Graph shows a histogram of the distribution and cumulative distribution for the Daily Geometric Sharpe Ratio of the simulated equity curves.

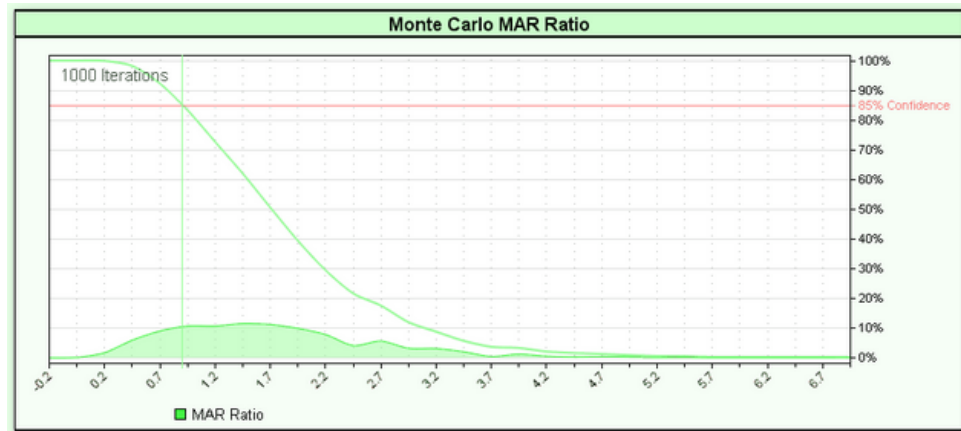


Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 200

Section 4 – MAR Graph

The Monte Carlo MAR Graph shows a histogram of the distribution and cumulative distribution for the MAR ratio of the simulated equity curves.



Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 168

Section 5 – R Squared Graph

The Monte Carlo [R Squared](#) Graph shows a histogram of the distribution and cumulative distribution for the [R Squared](#) values for the simulated equity curves.

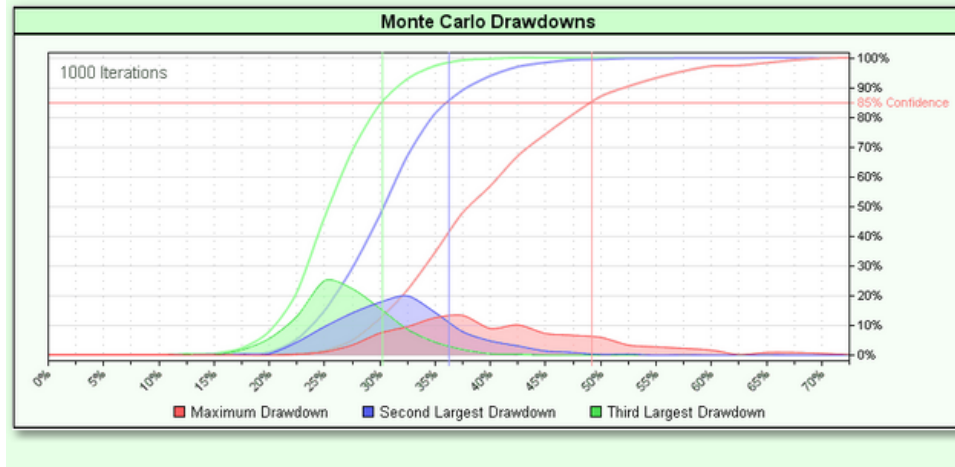


Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 198

Section 6 – Drawdown Graph

The Monte Carlo Drawdown Graph shows a histogram of the distribution and cumulative distribution for the drawdowns for the simulated equity curves.

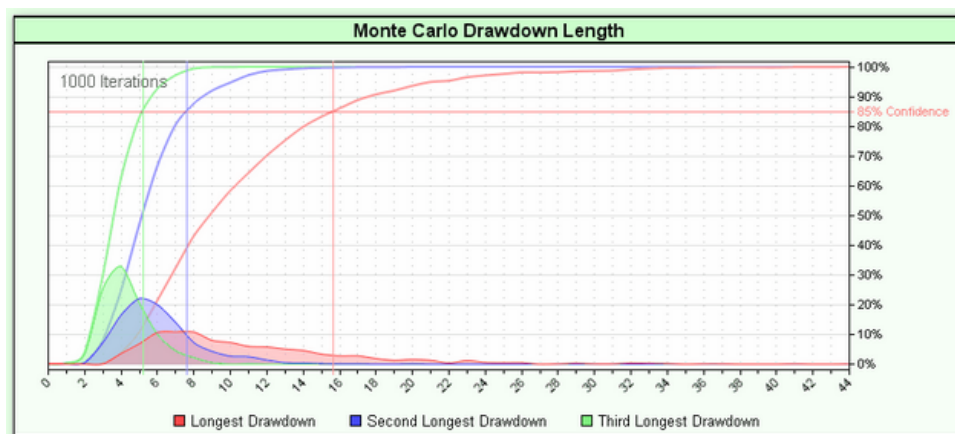


Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 128

Section 7 – Drawdown Length Graph

The Monte Carlo Drawdown Length Graph shows a histogram of the distribution and cumulative distribution for the lengths of the drawdowns (in months) for the simulated equity curves.



Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 129

Section 8 – Confidence Level Report

The Monte Carlo Confidence Level Report shows the values for various measures at the confidence level specified in [Monte Carlo Preferences](#).

Monte Carlo Confidence Level Statistics	
85% Return	33.09%
85% Sharpe	0.03
85% MAR	0.72
85% R Squared	0.769
85% Maximum Drawdown	49.12%
85% Second Largest Drawdown	35.96%
85% Third Largest Drawdown	30.22%
85% Longest Drawdown	15.8
85% Second Longest Drawdown	7.7
85% Third Longest Drawdown	5.2

CAUTION: The confidence levels used here mean that 85% of the simulated equity curves had measures which were at the indicated level or better. This is not a predictive number. It does not mean that there is an 85% chance that a given measure will be exceeded in the future.

Built-in Systems and Blox

Part



Part 17 – Built-in Systems and Blox

Trading Blox comes with several complete trading systems (depending on which version of the software you own). Each of these systems can be tested separately or mixed together with any of the other trading systems. For example, it is possible to run a test with the Turtle System using 50% of the equity and a Triple Moving Average System using the other 50%. Trading Blox also includes the ability to test each system using a separate data set. You can trade the Turtle System using one portfolio and the Triple MA system using another portfolio. Each system has its own parameters which can be varied and tested separately or in conjunction with those from the other systems.

To select a system or systems, check the box next to it:



Allocation Manager example of a two systems suite.

You can then set the allocation percentages in the [Global Parameters Allocation Manager](#).

Trading Blox Builder also has other built-in Blox that are used in our systems for money management, portfolio sizing, risk, etc. Remember that one of the most powerful features in Trading Blox Builder is the ability to use the same Blox in different systems! For information on how to customize our built-in systems or build your own, please see the Blox Builder manual. Each system topic includes a description of the system, the parameters, and the code.

Section 1 – Money Managers

Basic Money Manager

True to its title, this Money Manager is very simple. The number of contracts or shares will be the number you input here.

Basic Money Manager (Fixed Unit Size)

Value enter will be used to determine the quantity of shares, or contracts to have in an order.

Fixed Fractional Money Manager

This Money Manager sets the unit size to a certain percent of your total trading equity. This Block is used by many of our built-in systems.

Fixed Fractional Money Manager

The position size is computed using the following formula:

$$\text{Position Size} = \frac{(\text{Percent Risk per Trade} \times \text{Trading Equity})}{(\text{Entry Risk Points} \times \text{Dollars per Point})}$$

If you have no stops, there will be no risk. If there is no risk, there can be no trades with this money manager!

Multi Money Manager

This money manager allows you to choose from three choices of money management:

Multi-Money Manager Blox

1. **Single Contract** : The number of contracts or shares will be one.
2. **Fixed Fractional**: Described above. Remember that you must have stops in your system to get trades with this method of money management.
3. **Volatility Adjusted**: This method can be used without stops, and uses the Average True Range to size positions. The greater the volatility, the smaller the position size.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 171

Section 2 – Risk Managers

Trading Blox provides a variety of risk limiting controls that can be added to a system as a source of ideas on which to build custom blox that are more tailored to the trader's goals.

Each of the risk managing blox provide their own approach to filtering orders, or by reducing the distance between the protective price and the close of each date,

Risk Managing Blox:	Description:
<u>Correlation Risk Manager</u>	This Risk Manager is used in the ADX, MACD, and Turtle built-in systems to limit the maximum number of correlated units.
<u>Correlation Risk Manager Check Fills</u>	This Risk Manager is used in the ADX, MACD, and Turtle built-in systems to limit the maximum number of correlated units.
<u>Group Risk Manager</u>	Entry orders are rejected based on violations of threshold values for each of the group risk sections enabled.
<u>Total Risk Limiter</u>	This risk manager allows you to select a maximum amount of risk you want at any given time, and then a method of reducing positions if your risk is too great. If you choose "Reduce Positions" this block will sell a portion of your units to reduce to below the maximum. If you select "Move Stops" the system will set new stops based on the maximum risk and place orders accordingly.
<u>Unit Limiter</u>	This block rejects entry orders for new units if the number of current positions in the same direction in closely or loosely correlated markets is greater than or equal to the user set maximum.

Order Fill Predictability:

Success in limiting new entry orders to reduce risk, limit position or unit correlations and the number of open position is highly dependent upon the order entry order's selected execution method. Success also requires the orders be controlled ahead of them being reported and which orders will be filled in the next trading session. Knowing an order is going to be filled ahead of it being released is limited to market orders that only need a trading session to execute. Orders that add price and or session timing conditions cannot be predicted with any reasonable level of certainty.

Entry Order Categories:	Execution Conditions:
Market: On-Open On-Close	These orders are easy to apply order restrictions because their ability to be filled by the market is certain in most situations. All that has to happen for these orders to be filled is for the market to price another trading session and the order will be filled.
Conditional: On-Stop On-Limit On-Stop Open At-Limit Open On-Stop Close At-Limit Close	All of these order execution types require trading session conditions that are not possible to reliably predict their ability for being executed. Without predictability rigid order counts will create below expectation limiting results. Controlling these types of orders so they allow results that are closer to the parameter values used to control orders can be closer when there is a percentage overage allowance, but the results will the control will allow values above and below the parameter values applied.

2.1 Correlation Risk Manager

This Risk Manager is used in the ADX, MACD, and Turtle built-in systems to limit the maximum number of correlated units.

Correlation Risk Manager Parameters:

Correlation Risk Manager		
Max Closely Correlated Markets	Step ☐	3
Maximum Loosely Correlated Markets	Step ☐	6
Max Directional Units	Step ☐	12

Parameter:	Description:
Max Closely Correlated Markets	Sets the max number of units the system can have on in closely correlated markets.
Maximum Loosely Correlated Markets	Sets the max number of units the system can have on in loosely correlated markets
Max Directional Units	Sets the max number of units the system can have on in each direction, Long or Short.

Correlation Risk Manager Control Logic - CAN ADD UNIT:	
<pre> ' ===== ' Correlation Risk Mgr ' CAN ADD UNIT SCRIPT - START ' ===== ' ~~~~~ ' Check correlations, and total directional positions ' ~~~~~ If order.position = LONG THEN ' LONG ORDER Correlation Filter If instrument.closelyCorrelatedLongUnits >= maxCloselyCorrelatedMarkets OR instrument.looselyCorrelatedLongUnits >= maxLooselyCorrelatedMarkets THEN order.Reject("Too many correlated long units") ENDIF ' ----- ' LONG Position Count Filter If system.totalLongUnits >= maxDirectionalUnits THEN order.Reject("Too many long directional units") ENDIF ELSE' o.position = LONG ' ----- ' SHORT ORDER Correlation Filter If instrument.closelyCorrelatedShortUnits >= maxCloselyCorrelatedMarkets OR </pre>	

Correlation Risk Manager Control Logic - CAN ADD UNIT :

```
        instrument.looselyCorrelatedShortUnits >=
maxLooselyCorrelatedMarkets THEN
        order.Reject( "Too many correlated short units" )
    ENDIF
    ' -----
    ' SHORT Position Count Filter
    If system.totalShortUnits >= maxDirectionalUnits THEN
        order.Reject( "Too many short directional units" )
    ENDIF
ENDIF' o.position = SHORT
' ~~~~~
' =====
' CAN ADD UNIT - END
' Correlation Risk Mgr
' =====
```

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 123

2.2 Correlation Risk Manager - Check Fills

This version of the Correlation Risk Manager has the same functionality of the Correlation Risk Manager plus the addition of an ability check the process after the order has been filled.

Correlation Risk Manager - Check Fills Parameters:

Correlation Risk Manager - Check Fills	
Max Closely Correlated Markets	Step ☐ 3
Maximum Loosely Correlated Markets	Step ☐ 6
Max Directional Units	Step ☐ 12

Parameter:	Description:
Max Closely Correlated Markets	Sets the max number of units the system can have on in closely correlated markets.
Maximum Loosely Correlated Markets	Sets the max number of units the system can have on in loosely correlated markets
Max Directional Units	Sets the max number of units the system can have on in each direction, Long or Short.

Correlation Risk Manager - Check Fills Control Logic - CAN ADD UNIT:

```

' =====
' Correlation Risk Mgr wFills
' CAN ADD UNIT SCRIPT - START
' =====
' ~~~~~
' Check LONG correlations, and Total LONG Position Count
' ~~~~~

If order.position = LONG THEN
    ' LONG ORDER Correlation Filter
    If instrument.closelyCorrelatedLongUnits >=
maxCloselyCorrelatedMarkets OR
    instrument.looselyCorrelatedLongUnits >=
maxLooselyCorrelatedMarkets THEN
        ' Remove Order and Send Rejection Msg to Filter Log
        order.Reject( "Too many correlated long units" )
    ENDIF
' -----
' LONG Position Count Filter
If system.totalLongUnits >= maxDirectionalUnits THEN
    ' Remove Order and Send Rejection Msg to Filter Log
    order.Reject( "Too many long directional units" )
ENDIF
ELSE ' o.position = LONG

```

Correlation Risk Manager - Check Fills Control Logic - CAN ADD UNIT:

```

' ~~~~~
' Check SHORT correlations, and Total SHORT Position Count
' ~~~~~
' SHORT ORDER Correlation Filter
  If instrument.closelyCorrelatedShortUnits >=
maxCloselyCorrelatedMarkets OR
    instrument.looselyCorrelatedShortUnits >=
maxLooselyCorrelatedMarkets THEN
    ' Remove Order and Send Rejection Msg to Filter Log
    order.Reject( "Too many correlated short units" )
  ENDIF
' ~~~~~
' SHORT Position Count Filter
  If system.totalShortUnits >= maxDirectionalUnits THEN
    ' Remove Order and Send Rejection Msg to Filter Log
    order.Reject( "Too many short directional units" )
  ENDIF
ENDIF ' .position = SHORT
' ~~~~~
' =====
' CAN ADD UNIT - END
' Correlation Risk Mgr wFills
' =====

```

CAN FILL ORDER:

```

' =====
' Correlation Risk Mgr wFills
' CAN FILL ORDER SCRIPT - START
' =====
' ~~~~~
' Check correlations, and total directional positions
' ~~~~~
' Only filter Order Correlationons on Entry Fill.
' There are no count or Correlcation filter for exits
If order.isEntry THEN
  ' ~~~~~
  ' Check LONG Order correlations, and Total LONG Position Count
  ' ~~~~~
  If order.position = LONG THEN
    ' ~~~~~
    ' LONG Order-Filled Correlation Filter
    ' ~~~~~
    If instrument.closelyCorrelatedLongUnits >=
maxCloselyCorrelatedMarkets OR
        instrument.looselyCorrelatedLongUnits >=
maxLooselyCorrelatedMarkets THEN
        order.Reject( "Too many correlated long units" )
    ENDIF
  ' ~~~~~
  ' Check Total SHORT Position Count

```

CAN FILL ORDER:

```

' -----
If system.totalLongUnits >= maxDirectionalUnits THEN
    order.Reject( "Too many long directional units" )
ENDIF
ELSE
' -----
' Check SHORT Order correlations
' -----
If instrument.closelyCorrelatedShortUnits >=
maxCloselyCorrelatedMarkets OR
    instrument.looselyCorrelatedShortUnits >=
maxLooselyCorrelatedMarkets THEN
    order.Reject( "Too many correlated short units" )
ENDIF
' -----
' Check Total SHORT Position Count
' -----
If system.totalShortUnits >= maxDirectionalUnits THEN
    order.Reject( "Too many short directional units" )
ENDIF
ENDIF
ENDIF
' ~~~~~
' =====
' CAN FILL ORDER - END
' Correlation Risk Mgr wFills
' =====

```

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 124

2.3 Group Risk Manager

Entry orders are rejected based on violations of threshold values for each of the group risk sections enabled.

Group Risk Manager Parameters:

Group Risk Manager

Use Risk True

Max Group Long Risk (%) Step ☐ 20%

Max Group Short Risk (%) Step ☐ 10%

Use Units False

Max Group Long Units Step ☐ 100

Max Group Short Units Step ☐ 100

Use Quantity False

Max Group Long Quantity Step ☐ 1,000

Max Group Short Quantity Step ☐ 1,000

Legend: True, False, Step True to False

Parameters:	Descriptions:
Use Risk	True enables Max Group Risk parameters.
Max Group Long Risk %	Max risk for Long instruments in the group.
Max Group Short Risk %	Max risk for Short instruments in the group.
Use Units	True enables Max number of Long and Short Group Units.
Max Group Long Units	Max number of Long units in the group.
Max Group Short Units	Max number of Short units in the group.
Use Quantity	True enables Max number of Long and Short Group positions.
Max Group Long Quantity	Max quantity of Long positions.
Max Group Short Quantity	Max quantity of Short positions.
Group Risk Limiter Control Logic - CAN ADD UNIT :	
' =====	

Group Risk Limiter Control Logic - CAN ADD UNIT:

```

' Group Risk Manager
' CAN ADD UNIT SCRIPT - START
' =====
' ~~~~~
' LONG Order Filtering Logic
' -----
' Check the long positions in this group to see if we can add a new
unit
  If order.position = LONG AND
    ( ( useRisk AND instrument.groupLongRisk >= maxLongRisk *
system.tradingEquity ) OR
      ( useUnits AND instrument.groupLongUnits >= maxLongUnits ) OR
      ( useQuantity AND instrument.groupLongQuantity >=
maxLongQuantity ) ) THEN
    order.Reject( "Long group risk greater than maximum" )
  ENDIF
' -----
' SHORT Order Filtering Logic
' -----
' Check the short positions in this group to see if we can add a new
unit
  If order.position = SHORT AND
    ( ( useRisk AND instrument.groupShortRisk >= maxShortRisk *
system.tradingEquity ) OR
      ( useUnits AND instrument.groupShortUnits >= maxShortUnits ) OR
      ( useQuantity AND instrument.groupShortQuantity >=
maxShortQuantity ) ) THEN
    order.Reject( "Short group risk greater than maximum" )
  ENDIF
' ~~~~~
' =====
' CAN ADD UNIT - END
' Group Risk Manager
' =====

```

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 153

2.4 Total Risk Manager

This risk manager allows you to select a maximum amount of risk you want at any given time, and then a method of reducing positions if your risk is too great. If you choose "Reduce Positions" this block will sell a portion of your units to reduce to below the maximum. If you select "Move Stops" the system will set new stops based on the maximum risk and place orders accordingly.

Total Risk Limiter Parameters:

Parameters:	Descriptions:
Maximum Risk Threshold %	Enter the maximum amount of risk as a percentage that this blox should limit.
Reduction Algorithm	Blox provide two methods for reducing the percentage of risk being published by the open risk to current trading equity amount. Option 1: Reduce Position: This option will reduce the position's current quantity using the percentage value entered into the "Maximum Risk Threshold %." Option 2: Move Stops: Distance between the current close and the position's protective risk price determines how many position points are at risk. Risk points are converted to a monetary value so that it can be expanded by the number of quantity contained in the position. Once the amount of risk in monetary terms is understood it is divided by the System's Trading Equity series element for the each trade day so that a current open risk rate can be determined.
Note: The following expression determines the amount by which risk is reduced when the threshold is breached: $\text{Reduction Amount} = (\text{Portfolio Risk} - \text{Max Risk Threshold}) / \text{Portfolio Risk}$ Example: When current risk is 40% and the maximum threshold is 30%, then risk will be reduced by: (40% - 30%) / 40% = 25%. Reducing all current position sizes by 25%, or reducing each stop offset distance (current price to the current exit) by 25% brings the position risk back to 30%.	

Total Risk Manager Control Logic - INITIALIZE RISK MANAGER:

```

' =====
' Total Risk Limiter
' INITIALIZE RISK MANAGER SCRIPT - START
' =====
' Clear Total Risk Accumulator
totalRisk = 0
' =====
' INITIALIZE RISK MANAGER SCRIPT - END
' Total Risk Limiter
' =====

```

COMPUTE INSTRUMENT RISK

```

' =====
' Total Risk Limiter
' COMPUTE INSTRUMENT RISK SCRIPT - START
' =====
' Add the instrument risk to the total risk.
totalRisk = totalRisk + instrument.currentPositionRisk
' =====
' COMPUTE INSTRUMENT RISK SCRIPT - END
' Total Risk Limiter
' =====

```

COMPUTE RISK ADJUSTMENTS:

```

' =====
' Total Risk Limiter
' COMPUTE RISK ADJUSTMENTS SCRIPT - START
' =====
' ~~~~~
If system.tradingEquity > 0 THEN
'   Compute the current risk.
riskPercent = totalRisk / system.tradingEquity

'   If the risk is above our threshold...
If riskPercent > maximumRiskThreshold THEN
    reductionPercent = (riskPercent - maximumRiskThreshold) _
                      / riskPercent
ELSE
    reductionPercent = 0.0
ENDIF
ELSE
    reductionPercent = 0.0
ENDIF
' ~~~~~
' =====
' COMPUTE RISK ADJUSTMENTS SCRIPT - END
' Total Risk Limiter
' =====

```


Total Risk Manager Control Logic - INITIALIZE RISK MANAGER:**ADJUST INSTRUMENT RISK:**

```

' =====
' Total Risk Limiter
' ADJUST INSTRUMENT RISK SCRIPT - START
' =====
' ~~~~~
' REDUCE POSITION QUANTITY
' -----
' If we need to reduce risk
If reductionPercent > 0.0 THEN
  If reductionAlgorithm = REDUCE_POSITIONS THEN
    ' Reduce the position by this amount.
    broker.AdjustPositionOnOpen( 1.0 - reductionPercent )
  ENDIF
  ' -----
  ' LONG POSITION - MOVE STOP CLOSER
  ' -----
  If reductionAlgorithm = MOVE_STOPS THEN
    If instrument.position = LONG THEN
      ' Adjust the stops for each unit.
      For index = 1 TO instrument.currentPositionUnits
        ' Determine the current risk.
        risk = instrument.close - instrument.unitExitStop[ index ]
        ' Determine the stop that corresponds with the reduced
risk.
        newStop = instrument.close - ((1 - reductionPercent) *
risk)
        ' Set the new stop.
        instrument.SetExitStop( index, newStop )
        broker.ExitUnitOnStop( index, newStop )
      Next
    ENDIF ' Long
  ' -----
  ' SHORT POSITION - MOVE STOP CLOSER
  ' -----
  If instrument.position = SHORT THEN
    ' Adjust the stops for each unit.
    For index = 1 TO instrument.currentPositionUnits
      ' Determine the current risk.
      risk = instrument.unitExitStop[index] - instrument.close
      ' Determine the stop that corresponds with the reduced
risk.
      newStop = instrument.close + ((1 - reductionPercent) *
risk)
      ' Set the new stop.
      instrument.SetExitStop( index, newStop )
      broker.ExitUnitOnStop( index, newStop )
    Next
  ENDIF ' Short
ENDIF ' Algorithm Move Stops

```

Total Risk Manager Control Logic - INITIALIZE RISK MANAGER:

```
ENDIF ' There is a reduction required
| ~~~~~
| =====
| ADJUST INSTRUMENT RISK SCRIPT - END
| Total Risk Limiter
| =====
```

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 227

2.5 Unit Limiter

This block rejects entry orders for new units if the number of current positions in the same direction in closely or loosely correlated markets is greater than or equal to the user set maximum.

Unit Limiter Parameters:

Unit Limiter	
Max Closely Correlated Markets	Step ☐ 5
Maximum Loosely Correlated Markets	Step ☐ 10

Parameter:	Description:
Max Closely Correlated Markets	Integer value that controls the maximum number of closely correlated markets allowed at any time during the test period.
Maximum Loosely Correlated Markets	Integer value that controls the maximum number of loosely correlated markets allowed at any time during the test period.

Unit Limiter Control Logic - CAN ADD UNIT SCRIPT :

```

' =====
' Unit Limiter
' CAN ADD UNIT SCRIPT - START
' =====
' ~~~~~
' If this is a proposed new Long unit...
' -----
If order.position = LONG THEN

    ' Compute the total correlated units.
    closelyCorrelatedUnits = instrument.currentPositionUnits _
                          + instrument.closelyCorrelatedLongUnits
    looselyCorrelatedUnits = closelyCorrelatedUnits _
                          + instrument.looselyCorrelatedLongUnits

    ' If the total correlated units are below their respective limits...
    If closelyCorrelatedUnits >= maxCloselyCorrelatedMarkets AND
        looselyCorrelatedUnits >= maxLooselyCorrelatedMarkets THEN
        ' Remove Order and Send Rejection Msg to Filter Log
        order.Reject( "Correlated long units greater than maximum" )
    ENDIF
ELSE ' o.position = LONG

    ' -----
    ' If this is a proposed new Short unit...
  
```

Unit Limiter Control Logic - CAN ADD UNIT SCRIPT :

```

' -----
'   Compute the total correlated units.
closelyCorrelatedUnits = instrument.currentPositionUnits _
                        + instrument.closelyCorrelatedShortUnits
looselyCorrelatedUnits = closelyCorrelatedUnits _
                        + instrument.looselyCorrelatedShortUnits

'   If the total correlated units are below their respective
limits...
If   closelyCorrelatedUnits >= maxCloselyCorrelatedMarkets AND
      looselyCorrelatedUnits >= maxLooselyCorrelatedMarkets THEN

      '   Remove Order and Send Rejection Msg to Filter Log
      order.Reject( "Correlated short units greater than maximum" )
ENDIF
ENDIF '   ' o.position = LONG
'   ~~~~~
'   =====
'   CAN ADD UNIT SCRIPT - END
'   Unit Limiter
'   =====

```

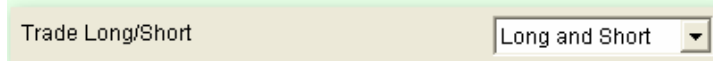
Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 231

Section 3 – Portfolio Managers

Trade Direction Portfolio Manager

The *Trade Long/Short* parameter controls which types of trades the system will take. The possible values are: *Long and Short*, *Long Only*, *Short Only*, and *Step All Values*. Choosing *Step All Values* will result in a three parameter run test with each of the other three values. Most of our built-in systems use this Portfolio Manager.



Strength Filter

The *Strength Filter* filters a pool of candidates which the systems can trade from the current portfolio. While this parameter is normally used for stocks, it can be applied to futures or forex.

The *Strength* used in the *Strength Filter* is simply a volatility-adjusted measure of the price increase (or decrease) of an instrument over a finite time period. The parameter *Strength Days* defines this measurement period (in days).

Filter Threshold tells Trading Blox how many instruments to filter out of the system's portfolio and lay aside as candidates for possible entries by the system. Instruments are ranked based on the formula:

$$\text{Strength} = \frac{(\text{price today} - \text{price } \textit{Strength Days ago})}{(\text{the value of ATR } \textit{Strength Days ago})}$$

Strength Filter has two sections: *Long* which selects for strength, and *Short* which selects for weakness.

The *Long* section filters instruments (individual futures, stocks, or forex) based on price strength. It culls the highest strength instruments over the time period specified by *Strength Days*, and sets aside the number of candidates specified by the *Filter Threshold* value.

The *Short* section filters for low price strength. It culls the lowest strength instruments over the time period specified by *Strength Days*, and sets aside the number of candidates specified by the *Filter Threshold* value.

The *Strength Filter* simply provides a pool of candidates which the systems can trade. The particular instruments ultimately traded out of the pool provided by the strength filter are a function of the entry rules for that system.

If Use Strength Filter is set to *True* or *Step True* to *False* then four additional parameters will be shown:



Use Strength Filter	☐ True
Filter Threshold - Long	Step ☐ 100
Filter Threshold - Short	Step ☐ 100
Strength Days - Long	Step ☐ 250
Strength Days - Short	Step ☐ 250

Filter Threshold - Long

Applicable only if *Use Strength Filter* is True, this parameter defines the number of instruments to be filtered out as candidates for long trades.

Filter Threshold - Short

Applicable only if *Use Strength Filter* is True, this parameter defines the number of instruments to be filtered out as candidates for short trades.

Strength Days - Long

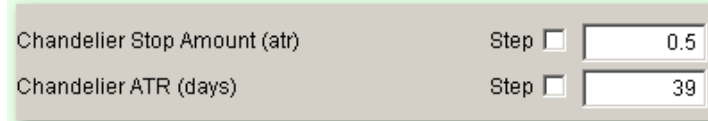
Applicable only if *Use Strength Filter* is True, this parameter defines the length of time in days over which performance is examined to determine strength for long trades.

Strength Days - Short

Applicable only if *Use Strength Filter* is True, this parameter defines the length of time in days over which instrument performance is examined to determine weakness for short trades.

Section 4 – Chandelier Exit

The Chandelier Exit system is only an Exit Block, so it does not place entry orders. It does place stop exit orders based on ATR if a new high or low is hit during an open trade.



Chandelier Stop Amount (atr)	Step ☐	0.5
Chandelier ATR (days)	Step ☐	39

You can input (or step) values for the number of days to calculate ATR and the ATR stops it uses to place stop orders.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 120

Section 5 – Gap Against Exit

This Exit Block will exits all Long open positions, when today's high is below yesterday's close, and when today's Low is below yesterday's Close for a Short position trade.

It is designed to provide some protect against loss when prices move against the trade's direction.



Gap Against Exit - Exit

There is only one parameter that acts like an On or Off switch.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 147

Section 6 – ADX System

The Average Directional Index (ADX) was developed by J. Welles Wilder to help traders determine the *strength* of the trend. Two components that help you determine the *direction* of the trend are the Positive Directional Indicator (+DI) and Negative Directional Indicator (-DI).

The built-in ADX System uses two factors to signal and entry:

- The strength of the trend (ADX) needs to be greater than a certain threshold
- The cross over of the +DI and -DI

It uses a fraction of the ATR plus/minus the close to set the stop price.

The ADX System uses the following parameters:



Parameter	Value	Step
ATR (days)	39	☐
ADX Periods to use (bars)	10	☐
ADX Trend Limit (adx)	50	☐
DI Period to use (bars)	16	☐
Use and Hold ATR Stops	True	☐
ATR Stop (fraction)	1.8	☐

Average True Range (days)

Sets the number of days to use when calculating the Average True Range

ATR Stop (fraction)

Sets the fraction (.5, 1.5, 3) of ATR to use for the stop width

ADX Periods to use (bars)

Sets the number of bars to use in calculating the ADX

ADX Trend Limit (adx)

Sets the minimum threshold for entering a trade

DI Period to use (bars)

Sets the number of bars to use in calculating the +DI and -DI

Entry Script

```

IF adxIndicator > adxTrendLimit AND
positiveDirectionalIndicator > negativeDirectionalIndicator AND
instrument.position <> LONG THEN

    IF useATRStops THEN
        broker.EnterLongOnOpen( instrument.close - averageTrueRange *
atrStop )
  
```

```
        ELSE
            broker.EnterLongOnOpen
        ENDIF

ENDIF

IF adxIndicator > adxTrendLimit AND
   positiveDirectionalIndicator < negativeDirectionalIndicator AND
   instrument.position <> SHORT THEN

    IF useATRStops THEN
        broker.EnterShortOnOpen( instrument.close + averageTrueRange *
atrStop )
    ELSE
        broker.EnterShortOnOpen
    ENDIF

ENDIF
```

Adjust Stops

```
' .....
' Enter stop if "Use ATR Stops" is true
' .....
IF useATRStops THEN

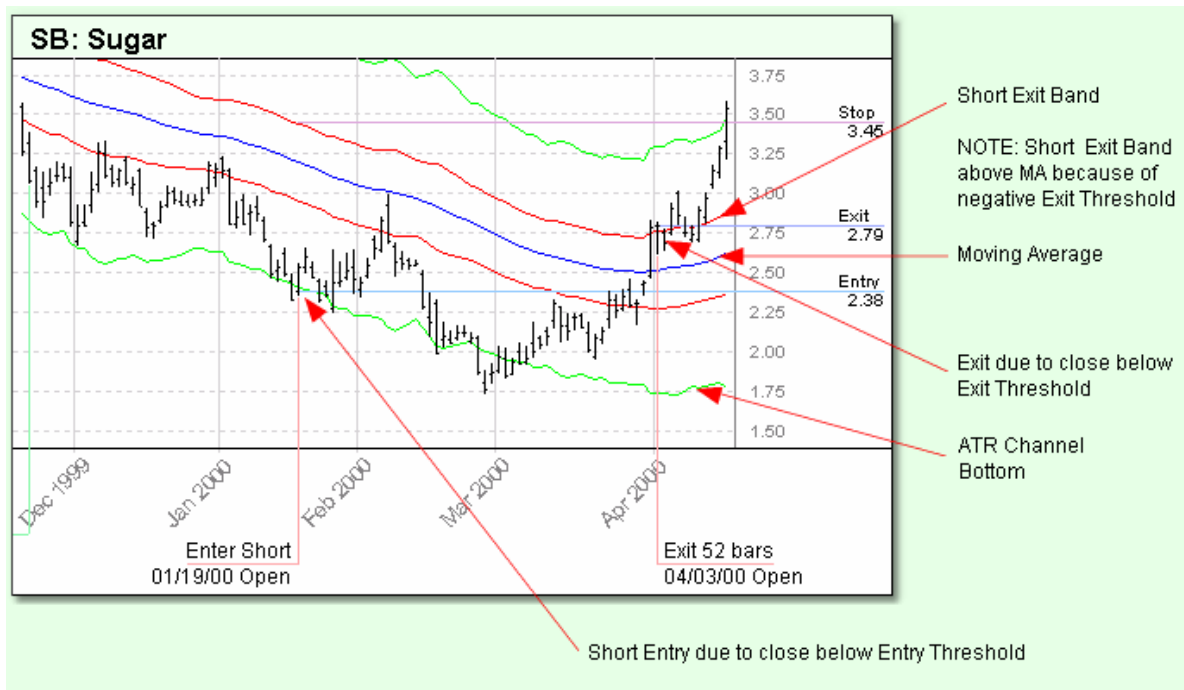
    broker.ExitAllUnitsOnStop( instrument.unitExitStop )
ENDIF
```

Section 7 – ATR Channel Breakout System

This system is a variation on the [Bollinger Breakout System](#) which uses [Average True Range](#) instead of standard deviation as a measure of the volatility which defines the width of the channels or bands.

A variation of the *ATR Channel Breakout System* was popularized as the PGO system by trader Mark Johnson on [Chuck LeBeau's System Trader's Club](#) forum and elsewhere. This version, the ATR Channel Breakout System, is more flexible and permits the testing of different entry and exit thresholds.

The system is a form of breakout system that buys on the next open when the price closes above the top of the ATR Channel, and exits when the price closes back inside the channel. Short entries are the mirror opposite with selling taking place when the price closes below the bottom of the *ATR Channel*.



The system trades based on a volatility-band breakout where volatility is measured using [Average True Range](#) (ATR). The center of the *ATR Channel* is defined by an *Exponential Moving Average* of the closing prices using a number of days defined by the parameter *Close Average Days*. The top and bottom of the *ATR Channel* are defined using a fixed-multiple of ATR from the moving average specified by the parameter *Entry Threshold*.

The system enters at the open following a day that closes over the top of the *ATR Channel* or below the bottom of the *ATR Channel*. The system exits following a close below the *Exit Channel* which is defined using a fixed-multiple of ATR from the moving average specified by the parameter *Exit Threshold*.

This ATR Channel Breakout system is similar to the [Bollinger Breakout System](#) except that it uses [Average True Range](#) instead of standard deviation as a measure of the volatility which defines the width of the channel.

For example, an *Entry Threshold* of 3 and an *Exit Threshold* of 1 would cause the system to enter the market when the price closed more than 3 ATR above the moving average and to exit when the price subsequently dropped below 1 ATR above the moving average. NOTE: *Exit Threshold* can be a negative number which will cause the system to exit only after the price comes some amount through the moving average.

The ATR Channel Breakout system includes four parameters which affect the entries:

Entries and Exits		
ATR (Days)	Step ☐	39
Close Average (Days)	Step ☐	80
Entry Threshold (Fraction)	Step ☐	2
Exit Threshold (Fraction)	Step ☐	1

ATR Days

The number of days in the [Exponential Moving Average](#) for the [Average True Range](#) itself.

Close Average Days

The number of days in the [Exponential Moving Average](#) of daily closes which forms the center of the ATR channel.

Entry Threshold

The width of the channel in ATR. This defines both the top and bottom of the channel. The system buys or sells to initiate a new position when the closing price crosses the price defined by this threshold.

Exit Threshold

If set to zero, the system will exit when the price closes below the moving average. If set to some higher number the system will exit when the price closes below the given threshold. A negative Exit Threshold means that the exit channel is below the moving average for a long position.

Entry Script

```

' -----
' Enter orders if channel is breached
' -----
IF instrument.position <> LONG AND
   instrument.close > channelTop THEN

    broker.EnterLongOnOpen( exitTop )
ENDIF

```

```
IF instrument.position <> SHORT AND
  instrument.close < channelBottom THEN

  broker.EnterShortOnOpen( exitBottom )
ENDIF
```

Exit Script

```
' .....
' Exit orders if exit threshold is breached
' .....

IF instrument.position = LONG AND
  instrument.close < exitTop THEN

  broker.ExitAllUnitsOnOpen
ENDIF

IF instrument.position = SHORT AND
  instrument.close > exitBottom THEN

  broker.ExitAllUnitsOnOpen
ENDIF
```

Adjust Stops

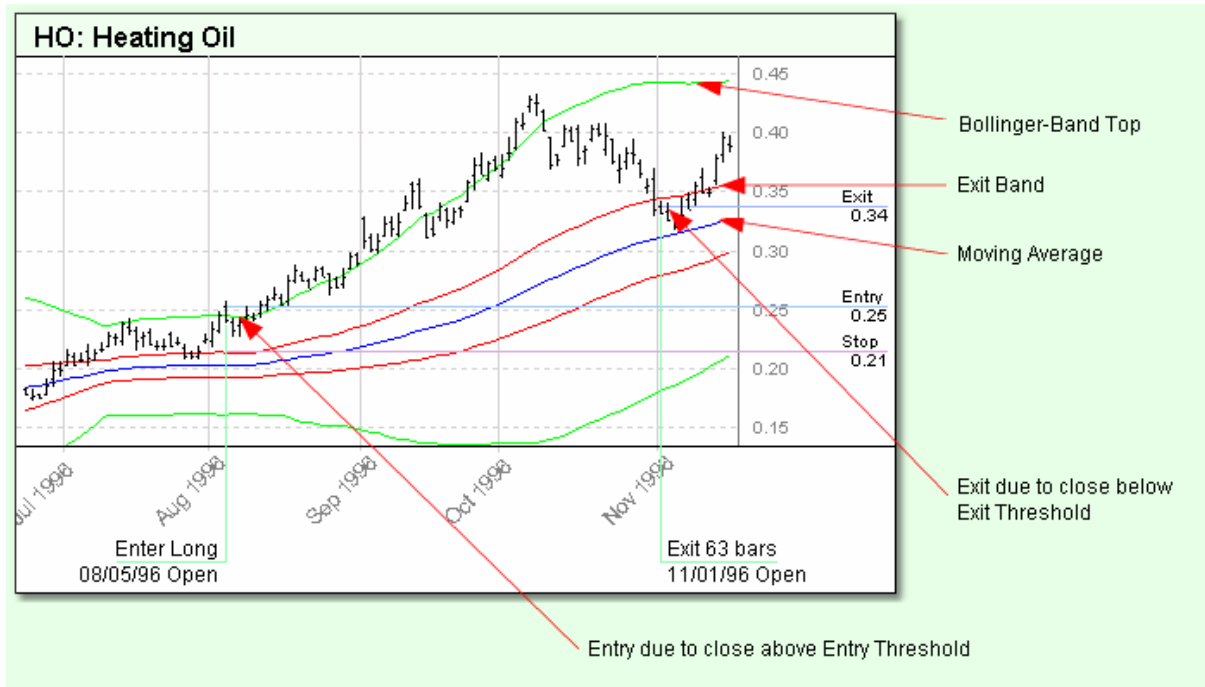
```
' .....
' Adjust stops so money manager knows core equity
' .....

IF instrument.position = LONG THEN
  instrument.SetExitStop( exitTop )
ENDIF

IF instrument.position = SHORT THEN
  instrument.SetExitStop( exitBottom )
ENDIF
```

Section 8 – Bollinger Breakout System

This system was described by Chuck LeBeau and David Lucas in their 1992 book: "*Technical Traders Guide to Computer Analysis of the Futures Markets*". The system is a form of breakout system that buys on the next open when the price closes above the top of the [*Bollinger Band*](#) and exits when the price closes back inside the band. Short entries are the mirror opposite with selling taking place when the price closes below the bottom of the *Bollinger Band*.



The center of the *Bollinger Band* is defined by an [Simple Moving Average](#) of the closing prices using a number of days defined by the parameter *Close Average Days*. The top and bottom of the *Bollinger Band* are defined using a fixed-multiple of the [standard deviation](#) from the moving average specified by the parameter *Entry Threshold*.

The system enters at the open following a day that closes over the top of the *Bollinger Band* or below the bottom of the *Bollinger Band*. The system exits following a close below the *Exit Band* which is defined using a fixed-multiple of the [standard deviation](#) from the moving average specified by the parameter *Exit Threshold*.

The value of the *Exit Band* on the day of entry is used as the stop for the purpose of determining position size using the standard *Fixed Fractional* position sizing algorithm.

The Bollinger Breakout System system includes three parameters that affect the entry and exit:

Entries and Exits	
Close Average (days)	Step ☐ 80
Entry Threshold (std. dev.)	Step ☐ 2
Exit Threshold (std. dev.)	Step ☐ 1

Close Average

The number of days in the [Simple Moving Average](#) which forms the center of the Bollinger Band channel.

Entry Threshold

The width of the channel in standard deviation. This defines both the top and bottom of the channel. The system buys or sells to initiate a new position when the closing price crosses the price defined by this threshold.

Exit Threshold

If set to zero, the system will exit when the price closes below the moving average. If set to some higher number the system will exit when the price closes below the given threshold. A negative Exit Threshold means that the exit channel is below the moving average for a long position.

For example, an *Entry Threshold* of 3 and an *Exit Threshold* of 1 would cause the system to enter the market when the price closed more than 3 standard deviations above the moving average and to exit when the price subsequently dropped below 1 standard deviation above the moving average.

Entry Script

```

' -----
' Enter orders if they channels are breached
' -----
IF instrument.position <> LONG AND
  instrument.close > channelTop THEN

  broker.EnterLongOnOpen( exitTop )
ENDIF

IF instrument.position <> SHORT AND
  instrument.close < channelBottom THEN

  broker.EnterShortOnOpen( exitBottom )
ENDIF

```

Exit Script

```

' -----
' Exit orders if exit threshold is breached
' -----
IF instrument.position = LONG AND

```

```
    instrument.close < exitTop THEN

    broker.ExitAllUnitsOnOpen
ENDIF

IF instrument.position = SHORT AND
    instrument.close > exitBottom THEN

    broker.ExitAllUnitsOnOpen
ENDIF
```

Adjust Stops

```
' .....
' Adjust stops so money manager knows core equity
' .....
IF instrument.position = LONG THEN
    instrument.SetExitStop( exitTop )
ENDIF

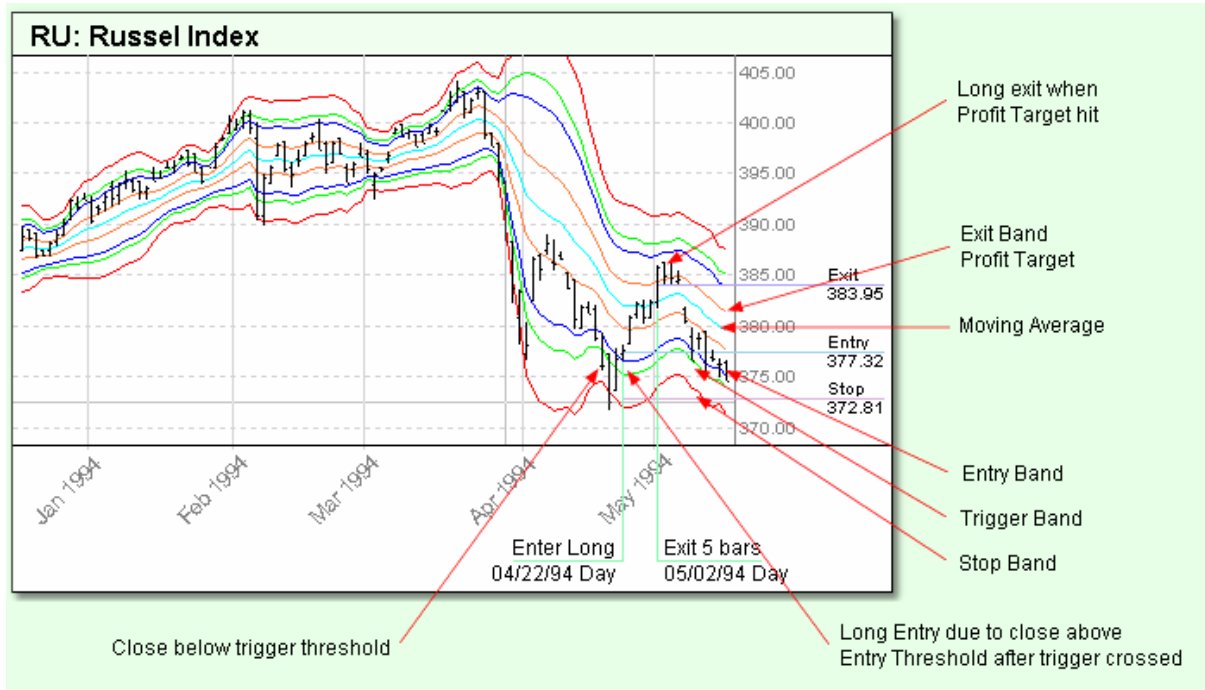
IF instrument.position = SHORT THEN
    instrument.SetExitStop( exitBottom )
ENDIF
```

Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 113

Section 9 – Bollinger Counter-Trend System

This system trades counter-trend within Bollinger Bands.



The center of the *Bollinger Band* is defined by an [Simple Moving Average](#) of the closing prices using a number of days defined by the parameter *Close Average Days*. The top and bottom of the *Bollinger Band* are defined using a fixed-multiple of the [standard deviation](#) from the moving average specified by the parameter *Entry Threshold*.

The system sells short when the price drops below the entry threshold after having exceeded the trigger threshold. It then exits using a profit target limit order at the price defined by the parameter *Exit Threshold*.

The system buys long when the price rises above the entry threshold after having gone below the trigger threshold. It then exits using a profit target limit order at the price defined by the parameter *Exit Threshold*.

The trigger is deactivated if the price crosses to the other size of the band and triggers in the opposite direction before a trade is entered. This might occur due to risk, equity, or volume constraints.

The Bollinger Counter-Trend System includes six parameters that affect the entry and exit:

Entries and Exits

Close Average (days)	Step ☐	80
Trigger Threshold (std. dev.)	Step ☐	3
Entry Threshold (std. dev.)	Step ☐	2
Exit Threshold (std. dev.)	Step ☐	1
Stop Threshold (std. dev.)	Step ☐	2.6
Hold Initial Stops		False

Close Average

The number of days in the moving average which forms the center of the Bollinger Band channel.

Trigger Threshold

The amount of standard deviation above or below the moving average which the price must reach before triggering a possible entry.

Entry Threshold

Expressed in terms of standard deviation from the moving average, the point below which the system will initiate a short position. When the price hits the trigger threshold and then subsequently goes below the entry threshold a short position is entered, vice versa for long entries.

Exit Threshold

The profit target for winning trades. The system will exit if the price reaches the exit threshold.

Stop Threshold

The number of standard deviation which will result in a stop loss.

For example, a *Trigger Threshold* of 4, *Entry Threshold* of 3, *Exit Threshold* of 1, and *Stop Threshold* of 4 would cause the system to enter the market on a stop when the price hit the 3 standard deviations above the moving average after having previously exceeded 4 standard deviations above the moving average. The system would exit when the price subsequently dropped below 1 standard deviation above the moving average or once again reached the 4 standard deviation level resulting in a stopped out loss..

NOTE: Exit Threshold can be a negative number which will cause the system to exit only after the price comes some amount through the moving average.

Hold Initial Stops

If set to True, stops will remain at the value on entry. This will prevent stops from moving as the Bollinger Band moves up and down with the moving average. If set to False, then the stop moves to the current value of the Stop Threshold multiplied by the current standard deviation and added to (or subtracted from) the current Close Average value.

Entry Script

```

' -----
' Enter orders if triggered
' -----
IF instrument.position <> LONG AND
  longTriggered THEN

  broker.enterlongonstop( channelBottom, stopBottom )
ENDIF

IF instrument.position <> SHORT AND
  shortTriggered THEN

```

```

    broker.entershortonstop( channelTop, stopTop )
ENDIF

```

Exit Script

```

' -----
' Reset the triggers if we are in a position
' -----
IF instrument.position = LONG THEN
    ' Reset triggers
    longTriggered = FALSE
    shortTriggered = FALSE

    ' exit on limit order
    broker.ExitAllUnitsAtLimit( exitBottom )
ENDIF

IF instrument.position = SHORT THEN
    ' Reset triggers
    longTriggered = FALSE
    shortTriggered = FALSE

    ' exit on limit order
    broker.ExitAllUnitsAtLimit( exitTop )
ENDIF

```

Adjust Stops

```

' -----
' Enter stop if "holdstops" is true
' -----
IF holdStops THEN

    broker.ExitAllUnitsOnStop( instrument.unitExitStop )
ENDIF

```

After Instrument Day

```

' -----
' Check the triggers
' -----
' Check the short trigger.
IF NOT shortTriggered AND
    instrument.position = OUT AND
    instrument.high > triggerTop THEN

    shortTriggered = TRUE
    longTriggered = FALSE
ENDIF

' Check the long trigger.
IF NOT longTriggered AND
    instrument.position = OUT AND

```

```
instrument.low < triggerBottom THEN  
    shortTriggered = FALSE  
    longTriggered = TRUE  
ENDIF
```

Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 114

Section 10 – Donchian System

The Donchian System is based on the Turtle system. It uses the Turtle logic, except it is single unit, does not use the Last Trade is Winner rule, does not use correlations, and uses a MACD Portfolio Manager to filter trades.

The Donchian System trades on breakouts similar to a Donchian Dual Channel system. There are two breakout figures, a longer breakout for entry, and a shorter breakout for exit.

The Donchian system uses a stop based on the *Average True Range* (ATR).

Note:

Turtle's concept of N has been replaced by the more common and equivalent term *Average True Range* (ATR) in this document.

Entry Breakout (days)

A trade is entered when the price hits the high or the low of the preceding X-days. For example, Entry Breakout = 20 means that a long position is taken if price hits the 20-day high; A short position is taken if the price hits the 20-day low.

Entry Offset (ATR)

If set to zero, this parameter has no effect. If *Entry Offset in ATR* is set to 1.0, a long position isn't entered until price hits the normal breakout price, plus 1.0 ATR. Likewise, a short position won't be entered until the price hits the normal breakout price, minus 1.0 ATR. Either a positive or negative value can be specified for this parameter. A positive value effectively delays entry until the specified point after the breakout threshold chosen; a negative value would enter before the breakout threshold chosen.

Stop (ATR)

This parameter defines the distance from the entry price to the initial stop, in terms of ATR. This system by default uses the order entry price, not the fill price, as a basis of the stop price. Since ATR is a measure of daily volatility and the Turtle System stops are based on ATR, this means that the Donchian System equalizes the position size across the various markets based on volatility.

According to the original Turtle Rules, long positions were stopped out if price fell 2 ATR from the entry price. Conversely, short positions were stopped out if the price rose 2 ATR from the entry price. Unlike the *Exit Breakout* based stop, which moves up or down with the X-day high or low, the stop defined by *Stop in ATR* is a "hard" stop that is fixed above or below the entry price upon entry. Once set, it does not vary throughout the course of the trade

Note that trades are liquidated when price hits either the Exit Breakout, Entry Breakout for the opposite direction, or the Stop in ATR, whichever is closest to the price at the time.

Exit Breakout (days)

Trades in progress are exited when the price hits the high or the low of the preceding X-days. This concept is the identical to Entry Breakout, but the logic is reversed: Long trades are exited when price hits the X-day low, and short trades are exited when the price hits the X-day low. The Exit Breakout moves up (or down) with price. It protects against adverse price excursions, and also serves as a trailing stop that acts to lock in a profit when the trend reverses.

Note:

Trades are liquidated when price hits either the Exit Breakout, Entry Breakout for the opposite direction, or the Stop in ATR, whichever is closest to the price at the time.

These options can be enabled or disabled with the Hold Initial Stops and Use Reversal Exit parameters. If the initial stop is held, then the initial stop price will be used to exit during the trade. If using the reversal exit, then the trade will be exited if the price hits the entry breakout for the opposite direction.

Exit Offset (ATR)

If set to zero, this parameter has no effect. If Exit Offset in ATR is set to 1.0, a long position isn't exited until price hits the normal breakout price, minus 1.0 ATR. Likewise, a short position won't be exited until the price hits the normal breakout price, plus 1.0 ATR. Either a positive or negative value can be specified for this parameter. A positive value effectively delays exit until the specified point after the breakout threshold chosen; a negative value would exit before the breakout threshold chosen.

ATR (days)

Defined the number of days for the ATR calculation. This is an exponential moving average of the True Range. 39 represents a 20 day Wilder ATR.

MACD Long Average (days)

This is the number of days for the long moving average portion of the MACD indicator.

MACD Short Average (days)

This is the number of days for the short moving average portion of the MACD indicator. The MACD itself is the Short Moving Average minus the Long Moving Average. The system will allow Long trades when the MACD is greater than zero and allow Short trades when the MACD is less than zero.

Money Manager

This system uses the Fixed Fractional [Money Manager](#)

Section 11 – Dual Moving Average System

This system uses two moving averages, one short and one long. The system trades when the short moving average crosses the long moving average.



The system optionally uses a stop based on [Average True Range](#) (ATR). If the ATR stop is used, the system will exit the market when that stop is hit.

If the ATR stop is not used, the system does not have an explicit stop and will always be in the market, making it a reversal system. It will exit a position only when the moving averages cross. At that point, it will exit and enter a new position in the opposite direction. In this case, the positions are sized based only on ATR using a custom money manager.

If an ATR stop is not used, then the entry risk is essentially infinite. This will cause the R-Multiple™s relatively meaningless since all gains will be less than the infinite risk of entering without any stop.

The Dual Moving Average System includes five parameters that affect the entries:

Entries and Exits	
Long Moving Average (Days)	Step ☐ 70
Short Moving Average (Days)	Step ☐ 20
Use ATR Stops	False
ATR (Days)	Step ☐ 39
Stop (in ATR)	Step ☐ 2

Long Moving Average

The number of days in the long moving average.

Short Moving Average

The number of days in the short moving average.

Use ATR Stops

If set to TRUE then the system will enter a stop based on a certain number of ATR from the entry point.

ATR Days

The number of days used for the ATR calculation. This parameter is visible and active only if *Use ATR Stops* is TRUE.

Stop

The stop width expressed in terms of ATR. This parameter is visible and active only if *Use ATR Stops* is TRUE.

If the *Use ATR Stops* is FALSE there are no stops, but the system uses a theoretical 1 ATR stop for the purposes of position sizing.

Entry Script

```
VARIABLES: currentclose, stopPrice TYPE: Price
```

```
' Get the current close.
```

```
currentClose = instrument.close
```

```
' If we are not long and the faster moving averages are above the slower one.
```

```
IF instrument.position <> LONG AND  
shortMovingAverage > longMovingAverage THEN
```

```
  ' If we are using ATR stops...
```

```
  IF useATRStops THEN
```

```
    ' Compute the stop price.
```

```
    stopPrice = currentClose - (stopInATR * averageTrueRange)
```

```
    ' Enter a long on the open using this stop.
```

```
    broker.EnterLongOnOpen( stopPrice )
```

```
  ' NOT using ATR stops...
```

```
  ELSE
```

```
    ' Enter a long on the open with no stop.
```

```
    broker.EnterLongOnOpen
```

```
  ENDIF
```

```
ENDIF
```

```
' If we are not short and the faster moving averages are below the slower one.
```

```

IF instrument.position <> SHORT AND
  shortMovingAverage < longMovingAverage THEN

  ' If we are using ATR stops...
  IF useATRStops THEN
    ' Compute the stop price.
    stopPrice = currentClose + (stopInATR * averageTrueRange)

    ' Enter a short on the open using this stop.
    broker.EnterShortOnOpen( stopPrice )

  ' NOT using ATR stops...
  ELSE
    ' Enter a short on the open with no stop.
    broker.EnterShortOnOpen

  ENDIF
ENDIF

```

Exit Script

```

' Exit position on open if moving averages cross
IF instrument.position = LONG AND
  shortMovingAverage < longMovingAverage THEN
  broker.ExitAllUnitsOnOpen
ENDIF

IF instrument.position = SHORT AND
  shortMovingAverage > longMovingAverage THEN
  broker.ExitAllUnitsOnOpen
ENDIF

```

Adjust Stops

```

' .....
' Enter stop if "Use ATR Stop" is true
' .....

IF useATRStops THEN

  broker.ExitAllUnitsOnStop( instrument.unitExitStop )
ENDIF

```

Section 12 – MACD System

The Moving Average Convergence/Divergence indicator is a centered oscillator that shows the difference between two moving averages, typically 12 and 26 days. These values, like any other parameter, can be altered to show the MACD over a different period of time.

The MACD system buys when the MACD goes above zero and sells when it goes below zero. So it's always in the market. Very similar in function to the Dual Moving Average System.

The MACD system uses the following parameters:



Entries and Exits	
Short Moving Average (days)	Step ☐ 50
Long Moving Average (days)	Step ☐ 150
Average True Range (days)	Step ☐ 39
ATR Stop (fraction)	Step ☐ 0.5

Short Moving Average Days

Number of days used to calculate the short moving average

Long Moving Average Days

Number of days used to calculate the long moving average

Average True Range Days

Number of days used to calculate the Average True Range

ATR Stop

ATR Stop (fraction): Fraction of the ATR used for stops

Entry Script

```

IF macdIndicator > 0 AND
  instrument.position <> LONG THEN

  IF useATRStops THEN
    broker.EnterLongOnOpen( instrument.close - averageTrueRange *
atrStop )
  ELSE
    broker.EnterLongOnOpen
  ENDIF

ENDIF

IF macdIndicator < 0 AND
  instrument.position <> SHORT THEN

```

```
        IF useATRStops THEN
            broker.EnterShortOnOpen( instrument.close + averageTrueRange *
atrStop )
        ELSE
            broker.EnterShortOnOpen
        ENDIF
    ENDIF
```

Adjust Stops

```
' .....
' Enter stop if "holdstops" is true
' .....

    IF useATRStops THEN

        broker.ExitAllUnitsOnStop( instrument.unitExitStop )
    ENDIF
```

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 165

Section 13 – RSI Trend Catcher System

The RSI Trend Catcher uses the Relative Strength Index to generate buy and sell signals.

It uses a threshold range above which it will buy and below which it will sell. The range it determined as:

Top of range - entered by user as entryThreshold

Bottom of range - (100 - entryThreshold)

It sets the stops stopWidth fraction of the ATR from the close.

It uses a threshold range for exits. Above which it exits a short position, and below which it exits a long position.

The RSI Trend Catcher System uses the following parameters:



Entries and Exits		
RSI Length (Bars)	Step ☐	130
Entry RSI Threshold	Step ☐	56
Exit RSI Threshold	Step ☐	51
Stop Width (ATR)	Step ☐	5
ATR Bars	Step ☐	20

RSI Length (Bars)

Number of bars used to calculate the RSI

Entry RSI Threshold

The top of the entry range above which the system will enter a long position. 100 - entryThreshold is the bottom of the entry range, below which the system will enter a short position.

Exit RSI Threshold

The bottom of the exit range, below which the system will exit a long position. 100 - exitThreshold is the top of the exit range, above which the system will exit a short position.

Stop Width (ATR)

Fraction of the ATR used for stop. Stops are based off the Close

ATR Bars

Number of bars used to calculate the Average True Range

Entry Script

```

' -----
' Enter position if RSI crosses threshold.
' -----
VARIABLES: exitStop TYPE: Floating

IF instrument.position <> LONG AND
  relativeStrengthIndex > entryThreshold THEN

  ' Compute the stop.
  exitStop = instrument.close - (stopWidth * averageTrueRange)

  ' Enter a long on the open with this stop.
  broker.EnterLongOnOpen( exitStop )
ENDIF

IF instrument.position <> SHORT AND
  relativeStrengthIndex < (100 - entryThreshold) THEN

  ' Compute the stop.
  exitStop = instrument.close + (stopWidth * averageTrueRange)

  ' Enter a long on the open with this stop.
  broker.EnterShortOnOpen( exitStop )
ENDIF

```

Exit Script

```

' -----
' Exit Position if RSI crosses Threshold
' -----

IF instrument.position = LONG AND
  relativeStrengthIndex <= exitThreshold THEN

  ' Exit the position.
  broker.ExitAllUnitsOnOpen
ENDIF

IF instrument.position = SHORT AND
  relativeStrengthIndex >= (100 - exitThreshold) THEN

  ' Exit the position.
  broker.ExitAllUnitsOnOpen
ENDIF

```

Adjust Stops

```

' -----
' Enter stop if "holdstops" is true
' -----
IF holdStops THEN

```

```
broker.ExitAllUnitsOnStop( instrument.unitExitStop )  
ENDIF
```

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 197

Section 14 – Stochastic System

The Stochastic Oscillator is a centered momentum oscillator. It is a percentile that shows a recent close in relationship to a high and low over a certain period of time. Stochastics were originally designed for stocks in a trading market. Although the parameters can be tweaked to give good results in almost any market, the default values reflect good results with stocks over the past 10 years.

Our built-in stochastic system uses convergence for signals. Positive convergence of a certain strength (called a swing factor), during a certain number of bars, and rising generate a long signal. The opposite exits a long position. Our system does not take short trades at all.

The conditions used to generate a long signal are as follows:

- There is a positive convergence that develops below a %K(Full) of 22.
- The %K(Full) is above 22 (this can be customized) and has started rising above %K(Slow) over the past 4 days.
- The convergence must take place over a certain time.
- The convergence must include highs and lows of a certain profundity to avoid meaningless signals.

The conditions for selling are much the same, except the convergence must be negative and falling below 78 (also customizable). Below are examples of both signals. Note the positive and negative convergences and rising or falling %K(Full).



The following values are used for the Stochastic System. The values shown are the default values.



Parameter	Step	Value
%K Days	☐	24
%D Days	☐	3
%K Full MA Days	☐	3
ATR Days	☐	39
ATR Stop	☐	3
Stochastic Entry Factor (Days)	☐	50
Stochastic Swing Factor	☐	4
OverBought Level (stochastic)	☐	78
Oversold Level (stochastic)	☐	22
Maximum Units	☐	7
Crossover Lookback (days)	☐	4
Swing Exit Bars (days)	☐	35

%K Days

The number of days used to determine %K(Fast).

%D Days

The number of days used to determine %D, a moving average of %K.

%K Full MA Days

The number of days used to smooth %K(Full).

ATR Days

The number of days used to calculate [average true range](#).

ATR Stop

A factor used to determine stops based on ATR.

Stochastic Swing Factor (Days)

The number of days high and low stochastics are considered in determining convergence.

Stochastic Swing Factor

The depth, in stochastic units, that highs and lows must be from each other. If this is too high, only very large fluctuations will produce signals. If this is too small, insignificant variations will generate signals.

Overbought Level

The range, in stochastic units, that is considered overbought.

Oversold Level

The range, in stochastic units, that is considered oversold.

Maximum Units

The maximum units that may be in the long position at any one time.

Crossover Lookback (Days)

The number of days that a crossover between %K(Slow) and %K(Full) is looked for.

Swing Exit Bars (Days)

Akin to Stochastic Swing Factor (Days), except only used to exit a long position.

Entry Script

```
' We are looking for a specific pattern here
' A convergence at a certain over-sold level, with a cross over
IF stochasticHigh2 > overSoldLevel AND
   stochasticHigh1 > overSoldLevel AND
   stochasticLow1 < overSoldLevel AND
   stochasticLow2 < overSoldLevel AND
   stochasticKSlow > stochasticKFull AND
   stochasticKSlow[crossoverLookback] <
stochasticKFull[crossoverLookback] AND
   stochasticLow1 > stochasticLow2 AND
   stochasticHigh1 < stochasticHigh2 AND
   instrument.bar - stochasticLow2Bar < swingEntryBars THEN

   IF instrument.totalUnits < maxUnits THEN

       IF useATRStops THEN
           broker.EnterLongOnOpen( instrument.close - averageTrueRange *
atrStop )
       ELSE
           broker.EnterLongOnOpen
       ENDIF

   ENDIF
ENDIF
```

Exit Script

```
' We are looking for an exit point -- opposite logic to the entry point
IF stochasticHigh2 > overBoughtLevel AND
   stochasticHigh1 > overBoughtLevel AND
   stochasticLow1 < overBoughtLevel AND
   stochasticLow2 < overBoughtLevel AND
   stochasticKSlow < stochasticKFull AND
   stochasticKSlow[crossoverLookback] >
stochasticKFull[crossoverLookback] AND
   stochasticLow1 > stochasticLow2 AND
```

```

    stochasticHigh1 < stochasticHigh2 AND
    instrument.bar - stochasticLow2Bar < swingExitBars THEN

    broker.ExitAllUnitsOnOpen
ENDIF

```

Adjust Stops

```

' -----
' Enter stop if "Use ATR Stops" is true
' -----

IF useATRStops THEN

    FOR unitIndex = 1 to instrument.totalUnits
        broker.ExitUnitOnStop( unitIndex,
    instrument.unitExitStop[ unitIndex ] )
    NEXT

ENDIF

```

After Instrument Day Script

```

IF lookingForHigh = TRUE THEN
    IF stochasticFull > stochasticHigh1 THEN
        'Finds new high

        stochasticHigh1 = stochasticFull
        stochasticHigh1Bar = instrument.bar
    ENDIF

    IF stochasticHigh1 - stochasticFull > swingFactor THEN
        'The high found is significant based on swingFactor

        lookingForLow = TRUE
        lookingforHigh = FALSE

        stochasticLow2 = stochasticLow1
        stochasticLow2Bar = stochasticLow1Bar

        stochasticLow1 = stochasticFull
        stochasticLow1Bar = instrument.bar
    ENDIF
ENDIF

IF lookingForLow = TRUE THEN

    IF stochasticFull < stochasticLow1 THEN
        stochasticLow1 = stochasticFull
        stochasticLow1Bar = instrument.bar
    ENDIF

    IF stochasticFull - stochasticLow1 > swingFactor THEN

```

```
    lookingForHigh = TRUE
    lookingForLow  = FALSE

    stochasticHigh2 = stochasticHigh1
    stochasticHigh2Bar = stochasticHigh1Bar

    stochasticHigh1 = stochasticKFull
    stochasticHigh1Bar = instrument.bar
ENDIF
ENDIF
```

Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 208

Section 15 – Triple Moving Average System

This system uses three *moving averages*, one short, one medium, and one long. The system trades long when the short *moving average* is higher than the medium *moving average* and the medium *moving average* is higher than the long *moving average*. When the short *moving average* is back below the medium *moving average*, the system exits. The reverse is true for short trades.



For this reason, unlike the Dual Moving Average system, this system is not always in the market. The system is out of the market when the relationship between the short MA and medium MA does not match the relationship between the medium MA and long MA.

For example, considering Long trades, if the short MA is over the medium MA but the medium MA is under the long MA, the system is out of the market. Likewise if the medium MA is over the long MA but the short MA is not over the medium MA the system is out of the market.

This means that the system can initiate trades due to either:

- Short MA is above the medium MA for Long entries or to below for Short entries. This is the most common case.
- Medium MA is above the long MA where the short MA is already over the medium MA for Long entries, or to below the long MA where the short MA is already under the medium MA for Short entries. This will happen when the market has been descending or ascending for a long time and then reverses direction. It takes longer for the medium MA to move to the other side of the long MA since they are both slower moving averages than the short MA.

The system optionally uses a stop based on [Average True Range](#) (ATR). If the ATR stop is not used, the system uses the value of the long moving average as the stop for the purpose of position sizing.

In the event of a stop out, the system will reenter whenever the above conditions are true, even if this is the following day's open.

The system includes seven parameters that affect the entries:

Entries and Exits	
Long Moving Average (days)	Step ☐ 150
Medium Moving Average (days)	Step ☐ 70
Short Moving Average (days)	Step ☐ 20
Use ATR Stops	False
ATR Average (days)	Step ☐ 39
Stop (ATR)	Step ☐ 2.5
Close through Short MA	True

Long Moving Average

The number of days in the long moving average.

Medium Moving Average

The number of days in the medium moving average.

Short Moving Average

The number of days in the short moving average.

Use ATR Stops

If set to TRUE then the system will enter a stop based on a certain number of ATR from the entry point.

ATR Average

The number of days used for the ATR calculation. This parameter is visible and active only if *Use ATR Stops* is TRUE.

Stop

The stop width expressed in terms of ATR. This parameter is visible and active only if *Use ATR Stops* is TRUE.

If the *Use ATR Stops* is FALSE the system computes a stop at the price of the long moving average for the purposes of position sizing. In this case, the stop is active only for the first day.

Close through Short MA

If set to True, Trading Blox won't take a trade unless the close is also on the right side of the short moving average. For example, with this parameter set to True, in addition to the short moving average being over the medium moving average and the medium moving average being over the long moving average, the close must be *above* the short moving average in order to trigger a Long position entry.

Entry Script

```

' Get the current close.
currentClose = instrument.close

' If we are not long and the faster moving averages are above the slower
ones.
IF mediumMovingAverage > longMovingAverage AND
   shortMovingAverage > mediumMovingAverage AND
   ((NOT closeThroughShortMA) OR currentClose > shortMovingAverage) AND
   instrument.position != LONG THEN

    ' Compute the stop.
    IF stopType = ATR_STOP THEN
        stopPrice = currentClose - (stopInATR * averageTrueRange)
    ELSE
        stopPrice = longMovingAverage
    ENDIF

    ' Enter a long at the open using this stop.
    broker.EnterLongOnOpen( stopPrice )
ENDIF

' If we are not short and the faster moving averages are below the slower
ones.
IF mediumMovingAverage < longMovingAverage AND
   shortMovingAverage < mediumMovingAverage AND
   ((NOT closeThroughShortMA) OR currentClose < shortMovingAverage) AND
   instrument.position != SHORT THEN

    ' Compute the stop.
    IF stopType = ATR_STOP THEN
        stopPrice = currentClose + (stopInATR * averageTrueRange)
    ELSE
        stopPrice = longMovingAverage
    ENDIF

    ' Enter a short at the market using this stop.
    broker.EnterShortOnOpen( stopPrice )
ENDIF

```

Exit Script

```

' If we are long and the short moving average crosses the medium.
IF instrument.position = LONG AND
   shortMovingAverage < mediumMovingAverage THEN

    ' Exit on the open
    broker.ExitAllUnitsOnOpen
ENDIF

```

```
' If we are short and the short moving average crosses the medium.  
IF instrument.position = SHORT AND  
    shortMovingAverage > mediumMovingAverage THEN  
  
    ' Exit on the open  
    broker.ExitAllUnitsOnOpen  
ENDIF
```

Adjust Stops

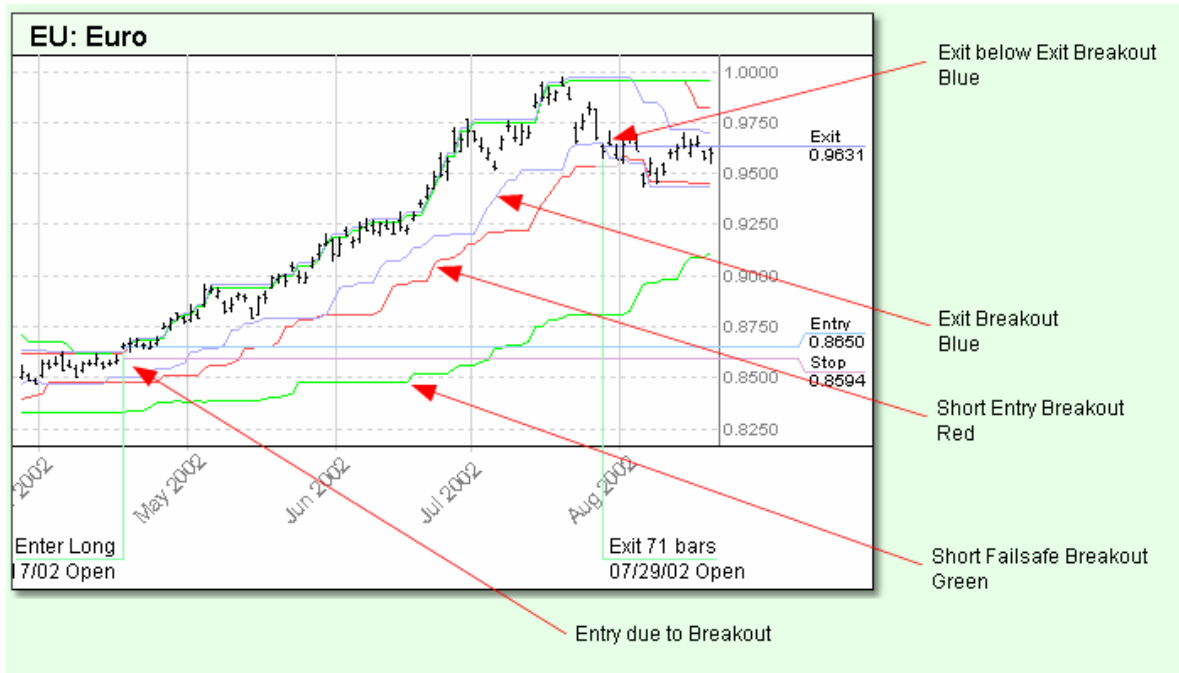
```
' .....  
' Enter stop if "holdstops" is true  
' .....  
IF holdStops THEN  
  
    broker.ExitAllUnitsOnStop( instrument.unitExitStop )  
ENDIF
```

Edit Time: 2/18/2022 3:35:42 PM

Topic ID#: 229

Section 16 – Turtle System Rules

The rules to the Turtle System as implemented in Trading Blox are described in the Original Turtle rules document at the site: www.originalturtles.org. Please consult that document for a complete description of the Turtle System rules.



The Turtle System trades on breakouts similar to a Donchian Dual Channel system. There are two breakout figures, a longer breakout for entry, and a shorter breakout for exit. The system also optionally uses a dual-length entry where the shorter entry is used if the last trade was a losing trade.

The Turtle system uses a stop based on the [Average True Range](#) (ATR).

Note that the Turtle concept of N has been replaced by the more common and equivalent term *Average True Range* (ATR) in this document.

Parameter	Value
Trade If Last Is Winner	True
Entry Breakout (days)	20
Entry Failsafe Breakout (days)	55
Entry Offset (ATR)	0
Unit Add (ATR)	0.5
Stop (ATR)	2
Exit Breakout (days)	10
Exit Offset (ATR)	0
Max Instrument Units	4
ATR (days)	39

Trade Long/Short

This parameter tells Trading Blox whether or not trades in the short direction are to be taken.

Trade if Last is Winner

When this parameter is set to False (unchecked and disabled), Trading Blox looks back at the last entry breakout for that instrument and determines if it would have been a winner, either actually, or theoretically. If the last trade was, or would have been a winner, then the next trade is skipped, regardless of direction (long or short).

The last breakout is considered to be the last breakout in that market regardless of whether or not that particular breakout was actually taken, or was skipped because of this rule. (Trading Blox looks back only at "regular" breakouts, and not Entry Failsafe Breakouts.)

The direction of the last breakout-long or short-is irrelevant to the operation of this rule, as is the direction of the trade currently being considered. Thus, a losing long breakout or a losing short breakout, whether hypothetical or actual, would enable the subsequent new breakout to be taken as a valid entry, regardless of its direction (long or short):

Some traders believe that two large, consecutive wins are unlikely, or that a profitable trade is more likely to follow a losing trade. Trading Blox allows you to test this idea by setting this parameter to False.

Entry Breakout (days)

A trade is entered when the price hits the high or the low of the preceding X-days, as adjusted by the *Entry Offset*. For example, Entry Breakout = 20 means that a long position is taken if price hits the 20-day high; A short position is taken if price hits the 20-day low.

Entry Failsafe Breakout (days)

This parameter works in concert with Trade if Last is Winner, and is used only if Trade if Last is Winner = False (as is shown in the partial screen shot above).

For example, consider the following set of parameters and values:

With these settings, if a 20-day breakout entry was recently signaled, but was skipped because the prior trade was a winner (either actually, or theoretically), then if the price breaks out above or below the 55-day extreme high or low, an entry is initiated for that position regardless of the outcome of the prior trade.

Entry Failsafe Breakout keeps you from missing very strong trends due to the action of the Trade if Last is Winner rule.

Entry Offset (ATR)

If set to zero, this parameter has no effect. If *Entry Offset in ATR* is set to 1.0, a long position isn't entered until price hits the normal breakout price, plus 1.0 ATR. Likewise, a short position won't be entered until the price hits the normal breakout price, minus 1.0 ATR. Either a positive or negative value can be specified for this parameter. A positive value effectively delays entry until the specified point after the breakout threshold chosen; a negative value would enter before the breakout threshold chosen.

Unit Add (ATR)

This parameter defines the price at which additions to an existing position are made. The Turtles entered single Unit positions at the breakouts, and added to those positions at 1/2 ATR intervals following trade initiation. (Adding to existing positions is often referred to as "pyramiding.")

Following the initial breakout entry, Trading Blox will continue to add a Unit (or Units, in the case of a large price move in a single day), at each interval defined by *Unit Add in ATR*, as price progresses favorably, right up to the maximum permitted number of Units, as specified by the various *Max Units* rules (explained below).

During historical simulation tests, the theoretical entry price is adjusted up or down by *Slippage Percent* and/or *Minimum Slippage*, to obtain the simulated fill price. So each interval is based on the simulated fill price of the previous order. So if an initial breakout order slipped by 1/2 ATR, the new order would be moved to account for the 1/2 ATR slippage, plus the normal unit add interval specified by *Unit Add in ATR*.

The exception to this rule is when multiple Units are added in a single day during a trade in progress. For example, with *Unit Add in ATR* = 0.5, the initial breakout order is placed and incurs slippage of 1/2 ATR. Several days later, two more units are added on the same day. In this case, the order price of both the 2nd and 3rd Units is adjusted up by 1/2 ATR (to 1 full ATR past the breakout), based on the slippage incurred by the 1st Unit. Ordinarily, in the case where several Units have been added (each on a separate day), the order price of each Unit is adjusted by the cumulative slippage (in N) of all the Units that preceded it on the trade in progress.

Notes:

- Users who generate orders will need to adjust the order price of any Units that are to be added, based on the cumulative slippage (in ATR) of all preceding Units. Trading Blox does not account for slippage and actual fills.

- The initial Unit, as well as any Units added, are all sized based on a combination of the current value of ATR, and the current available account balance. (This differs slightly from the Original Turtle Trading Rules in that the Turtles were only given an updated value of ATR once a week). So do not be surprised, for example, if the initial Unit size of a Wheat trade is 7 contracts, and the size of the next Unit added to this trade is 5 contracts.
- As per the Original Turtle Rules, the prices for stops of previous units are raised as new units are added. However, since the Unit Add in ATR parameter can be larger than the Stop in ATR parameter, it is possible that prices for stops would be raised above the entry price. In keeping with the spirit of the Turtle System Trading Blox does not allow stops to be raised above the initial entry price when adjusting stops based on the addition of new units..

Stop (ATR)

This parameter defines the distance from the entry price to the initial stop, in terms of ATR. Since ATR is a measure of daily volatility and the Turtle System stops are based on ATR, this means that the Turtle System equalizes the position size across the various markets based on volatility.

According to the original Turtle Rules, long positions were stopped out if price fell 2 ATR from the entry price. Conversely, short positions were stopped out if the price rose 2 ATR from the entry price.

Unlike the *Exit Breakout* based stop, which moves up or down with the X-day high or low, the stop defined by *Stop in ATR* is a "hard" stop that is fixed above or below the entry price upon entry. Once set, it does not vary throughout the course of the trade, unless Units are added, in which case the for earlier units are raised by the amount specified by *Unit Add (ATR)*.

Trades are liquidated when price hits the stop defined by either the *Stop in ATR*, the *Entry Breakout* for the opposite direction, or the *Exit Breakout* (see above), whichever is closest to the price at the time.

In this system the initial entry stop for the trade entry day is based on the order price. This is for ease of placing the stop once the order is filled. Note that the stop is adjusted based on the actual fill price for the following day.

Exit Breakout (days)

Trades in progress are exited when the price hits the high or the low of the preceding X-days as adjusted by the *Exit Offset*. This concept is the identical to Entry Breakout, but the logic is reversed: Long trades are exited when price breaks out below the X-day low, and short trades are exited when price breaks out above the X-day low.

The Exit Breakout moves up (or down) with price. It protects against adverse price excursions, and also serves as a trailing stop that acts to lock in a profit when the trend reverses.

Trades are liquidated when price hits the stop defined by either the *Stop in ATR*, the *Entry Breakout* for the opposite direction, or the *Exit Breakout* (see above), whichever is closest to the price at the time.

Exit Offset (ATR)

If set to zero, this parameter has no effect. If Exit Offset in ATR is set to 1.0, a long position isn't exited until price hits the normal breakout price, minus 1.0 ATR. Likewise, a short position won't be exited until the price hits the normal breakout price, plus 1.0 ATR. Either a positive or negative value can be specified for this parameter. A positive value effectively delays exit until the specified point after the breakout threshold chosen; a negative value would exit before the breakout threshold chosen.

Max Instrument Units

This parameter defines the maximum number of Units that can be held at one time, in any single futures market, or any single stock. For instance, *Max Instrument Units* = 4 means that no more than 4 Units of Coffee may be held at one time; this includes the initial Unit, plus 3 Units added.

Entry Orders:

```

VARIABLES: longEntryPrice, shortEntryPrice, unitIncrementATR,
unitEntryPrice TYPE: Price
VARIABLES: entryOffsetATR TYPE: Price
VARIABLES: unitsOn, unitIndex TYPE: Integer
VARIABLES: brokerPrice TYPE: Price

' Calculate the unit increment, stop amount, and entry offset
unitIncrementATR = unitAdd * averageTrueRange
stopWidth = stopInATR * averageTrueRange

'Total units currently on for this instrument
unitsOn = instrument.totalUnits

'-----
'Place initial entry orders when out of the market
'-----
' If we are not long (we are short or out) then place orders for tomorrow
' We will place orders for all potential units in case it's a big day
IF instrument.position <> LONG THEN

    ' Determine whether to use the regular entry high or the failsafe
    longEntryPrice = offsetAdjustedEntryHigh
    IF ( NOT tradeIfWinner ) AND ( lastTradeProfit > 0 ) THEN
        longEntryPrice = offsetAdjustedFailsafeHigh

    ' Loop over the potential units and place an order
    FOR unitIndex = 1 TO maxUnits
        brokerPrice = longEntryPrice + ( ( unitIndex - 1 ) *
unitIncrementATR )
        broker.EnterLongOnStop( brokerPrice, brokerPrice - stopWidth )
    NEXT
ENDIF

' If we are not short (we are long or out) then place orders for tomorrow
' We will place orders for all potential units in case it's a big day
IF instrument.position <> SHORT THEN
```

```

    ' Determine whether to use the regular entry low or the failsafe
    shortEntryPrice = offsetAdjustedEntryLow
    IF ( NOT tradeIfWinner ) AND ( lastTradeProfit > 0 ) THEN
shortEntryPrice = offsetAdjustedFailsafeLow

    ' Loop over the potential units and place an order
    FOR unitIndex = 1 TO maxUnits
        brokerPrice = shortEntryPrice - ( ( unitIndex - 1 ) *
unitIncrementATR )
        broker.EnterShortOnStop( brokerPrice, brokerPrice + stopWidth )
    NEXT

ENDIF

'-----
'Add additional Units when already in the market
'-----
'If long and can add more units
IF ( instrument.position = LONG ) AND ( unitsOn < maxUnits ) THEN

    ' Loop over potential units to add, placing orders at maximum of
regular entryHigh or unit increment
    FOR unitIndex = unitsOn + 1 TO maxUnits
        unitEntryPrice = instrument.unitEntryFill[ unitsOn ] + ( unitIndex
- unitsOn ) * unitIncrementATR
        brokerPrice = max( offsetAdjustedEntryHigh, unitEntryPrice )
        broker.EnterLongOnStop( brokerPrice, brokerPrice - stopWidth)
    NEXT
ENDIF

'If short and can add more units
IF ( instrument.position = SHORT ) AND ( unitsOn < maxUnits ) THEN

    ' Loop over potential units to add, placing orders at minimum of
regular entryLow or unit increment
    FOR unitIndex = unitsOn + 1 TO maxUnits
        unitEntryPrice = instrument.unitEntryFill[ unitsOn ] - ( unitIndex
- unitsOn ) * unitIncrementATR
        brokerPrice = min( offsetAdjustedEntryLow, unitEntryPrice )
        broker.EnterShortOnStop( brokerPrice, brokerPrice + stopWidth )
    NEXT
ENDIF

```

Entry Order Filled:

```

VARIABLES: stopAmount, stopPrice, originalATR TYPE: Price
VARIABLES: unitFillPrice, newStopPrice TYPE: Price
VARIABLES: newUnitExitStop, originalExitStop, newExitStop TYPE: Price
VARIABLES: entryDayIndex TYPE: Integer
VARIABLES: unitsOn, unitIndex TYPE: Integer

```

```

'-----

```

```

' Get the total number of units on now, including this new unit
unitsOn = instrument.totalUnits

' Calculate the new exit stop for this new unit, based on the fill price
if instrument.position = LONG then newUnitExitStop = order.fillPrice -
order.entryRisk
if instrument.position = SHORT then newUnitExitStop = order.fillPrice +
order.entryRisk

' Set exit stop for this new unit added
instrument.setExitStop( unitsOn, newUnitExitStop )

'-----
' If more than one unit on, then
' Reset the stops for all units, except the current one, based on this new
unit
' For each unit, look for the original ATR and calculate the original stop
price
' Add to the original stop
'-----
IF unitsOn > 1 THEN
    IF instrument.position = LONG THEN
        FOR unitIndex = 1 TO unitsOn - 1

            ' Compute the ATR TYPE: of the entry for this unit.
            entryDayIndex = instrument.bar -
instrument.unitEntryDayIndex[ unitIndex ]
            originalATR = averageTrueRange[ entryDayIndex ]

            ' Get the fill price.
            unitFillPrice = instrument.unitentryfill[ unitIndex ]

            ' Compute the original exit stop.
            originalExitStop = unitFillPrice - stopInATR * originalATR

            ' Now compute the new stop based on the original ATR.
            newStopPrice = originalExitStop + ( unitsOn - unitIndex ) *
unitAdd * originalATR

            ' Constrain the stop to the fill price or the new unit stop.
            newExitStop = min( newStopPrice, unitFillPrice,
newUnitExitStop )

            ' Give the instrument this new stop.
            instrument.setExitStop( unitIndex, newExitStop )
        NEXT
    ENDIF

    IF instrument.position = SHORT THEN
        FOR unitIndex = 1 TO unitsOn - 1

            ' Compute the ATR TYPE: of the entry for this unit.
            entryDayIndex = instrument.bar -

```

```

instrument.unitEntryDayIndex[ unitIndex ]
    originalATR = averageTrueRange[ entryDayIndex ]

    ' Get the fill price.
    unitFillPrice = instrument.unitentryfill[ unitIndex ]

    ' Compute the original exit stop.
    originalExitStop = unitFillPrice + stopInATR * originalATR

    ' Now compute the new stop based on the original ATR.
    newStopPrice = originalExitStop - ( unitsOn - unitIndex ) *
unitAdd * originalATR

    ' Constrain the stop to the fill price or the new unit stop.
    newExitStop = max( newStopPrice, unitFillPrice,
newUnitExitStop )

    ' Give the instrument this new stop.
    instrument.setExitStop( unitIndex, newExitStop )
NEXT
ENDIF
ENDIF

' -----
' Track theoretical positions
' -----

' Only run these calcs if we have to
IF ( tradeIfWinner = FALSE ) THEN

    ' If this is the first unit on, then set the theoretical position at
this point
    IF unitsOn = 1 THEN
        theoreticalPosition = instrument.position
        theoreticalEntryPrice = order.fillprice
        theoreticalExitStop = order.stopprice
    ENDIF
ENDIF
ENDIF

```

Exit Orders;

```

VARIABLES: brokerPrice, exitPrice, unitCurrentStopPrice TYPE: Price
VARIABLES: unitsOn, unitIndex TYPE: Integer

unitsOn = instrument.totalUnits

' If long...
IF ( instrument.position = LONG ) THEN

    ' Get the best exit price based on exit low and entry low.
    exitPrice = max( offsetAdjustedExitLow, offsetAdjustedEntryLow )

```



```

' Loop over the units and place stop order at the best price,
' including current stop price.
FOR unitIndex = 1 TO unitsOn
    unitCurrentStopPrice = instrument.unitExitStop[ unitIndex ]
    brokerPrice = max( exitPrice, unitCurrentStopPrice )
    broker.ExitUnitOnStop( unitIndex, brokerPrice )
NEXT
ENDIF

' If short...
IF ( instrument.position = SHORT ) THEN

    ' Get the best exit price based on exit high and entry high.
    exitPrice = min( offsetAdjustedExitHigh, offsetAdjustedEntryHigh )

    ' Loop over units and place stop order at best price,
    ' including current stop price.
    FOR unitIndex = 1 TO unitsOn
        unitCurrentStopPrice = instrument.unitExitStop[ unitIndex ]
        brokerPrice = min( exitPrice, unitCurrentStopPrice )
        broker.ExitUnitOnStop( unitIndex, brokerPrice )
    NEXT
ENDIF

```

Adjust Stops:

```

VARIABLES: stopPrice, bestExitPrice, unitCurrentStopPrice TYPE: Price
VARIABLES: unitsOn, unitIndex TYPE: Integer

' Get the total units on.
unitsOn = instrument.totalUnits

' If long...
IF ( instrument.position = LONG ) THEN

    ' Get the best exit price based on exit low and entry low.
    bestExitPrice = max( offsetAdjustedExitLow, offsetAdjustedEntryLow )

    ' Loop over the units and set the stop to the best price,
    ' including current stop price.
    FOR unitIndex = 1 TO unitsOn

        ' Get the current exit stop.
        unitCurrentStopPrice = instrument.unitExitStop[ unitIndex ]

        ' If the best stop is better than our current stop then
        ' exit on this stop.
        IF bestExitPrice > unitCurrentStopPrice THEN
            instrument.SetExitStop( unitIndex, bestExitPrice )
            broker.exitUnitOnStop( unitIndex, bestExitPrice )
        ENDIF
    NEXT
ENDIF

```

```

NEXT
ENDIF

' If short...
IF ( instrument.position = SHORT ) THEN

    ' Get the best exit price based on exit high and entry high.
    bestExitPrice = min( offsetAdjustedExitHigh, offsetAdjustedEntryHigh )

    ' Loop over the units and set the stop to the best price,
    ' including current stop price.
    FOR unitIndex = 1 TO unitsOn

        ' Get the current exit stop.
        unitCurrentStopPrice = instrument.unitExitStop[ unitIndex ]

        ' If the best stop is better than our current stop then
        ' exit on this stop.
        IF bestExitPrice < unitCurrentStopPrice THEN
            instrument.SetExitStop( unitIndex, bestExitPrice )
            broker.exitUnitOnStop( unitIndex, bestExitPrice )
        ENDIF
    NEXT
ENDIF

```

After Instrument Day:

```

VARIABLES: theoreticalExitPrice TYPE: PRICE

' Calc stop width
stopWidth = stopInATR * averageTrueRange

' -----
' Track theoretical positions
' -----

IF ( tradeIfWinner = FALSE ) THEN

    ' Update theoretical profit mark to market
    IF ( theoreticalPosition = LONG ) THEN
        lastTradeProfit = ( instrument.close - theoreticalEntryPrice )
    ENDIF

    if ( theoreticalPosition = SHORT ) THEN
        lastTradeProfit = ( theoreticalEntryPrice - instrument.close )
    ENDIF

    ' If not long, then watch for long entry

```

```

IF ( theoreticalPosition <> LONG ) THEN
    IF ( instrument.high > offsetAdjustedEntryHigh[1] ) THEN
        theoreticalPosition = LONG
        theoreticalEntryPrice = offsetAdjustedEntryHigh[1]
        theoreticalExitStop = theoreticalEntryPrice - stopWidth
    ENDIF
ENDIF

' If not short, then watch for short entry
IF ( theoreticalPosition <> SHORT ) THEN
    IF instrument.low < offsetAdjustedEntryLow[1] THEN
        theoreticalPosition = SHORT
        theoreticalEntryPrice = offsetAdjustedEntryLow[1]
        theoreticalExitStop = theoreticalEntryPrice + stopWidth
    ENDIF
ENDIF

' If long, then watch for exits
IF ( theoreticalPosition = LONG ) THEN
    theoreticalExitPrice = max( theoreticalExitStop,
offsetAdjustedExitLow, offsetAdjustedEntryLow )
    IF ( instrument.low < theoreticalExitPrice ) THEN
        lastTradeProfit = ( theoreticalExitPrice -
theoreticalEntryPrice )
        theoreticalPosition = OUT
    ENDIF
ENDIF

' If short, then watch for exits
IF ( theoreticalPosition = SHORT ) THEN
    theoreticalExitPrice = min( theoreticalExitStop,
offsetAdjustedExitHigh, offsetAdjustedEntryHigh )
    if ( instrument.high > theoreticalExitPrice ) THEN
        lastTradeProfit = ( theoreticalEntryPrice -
theoreticalExitPrice )
        theoreticalPosition = OUT
    ENDIF
ENDIF

ENDIF

```

Section 17 – Walk Forward

Trading Blox version 5.x has integrated the Walk Forward Parameter rules so they are easier to execute. How that process works, and the steps a user can follow are listed in the linked areas of this table.

Walk Forward Information Sections:

Walk Forward Testing Description	Statistical Measure Selections
Rolling Walk Forward Testing	Walk Forward Start and End Dates
Walk Forward Suite Selection	Walk Forward TestingThreads
System Selection	Walk Forward Statistics Reporter
Walk Forward Stepped Parameters	Walk Forward Simulation Test
Walk Forward Segment Settings	Stepped Parameter Test Summary Report
Walk Forward Goodness Setting	Walk Forward Stats 2 Graph Results
Walk Forward Stats 2 Testing Results	

Walk Forward Testing Description: [^Top](#)

The Walk Forward Test is a two-stage process that creates a series of data testing segments between the start and end dates selected in the Suite's Parameter date fields.

When a Walk Forward test runs, Trading Blox creates a groups of data segments. Each data segment is determined by two of the three parameters displayed in the [Walk Forward Parameters](#) dialog. In the earlier data end of a Walk Forward segment, the size of the segment that is created by the [Walk Forward Optimization Days parameter](#). The data section created by the number of days listed in the [Walk Forward OOS](#) parameter.

The optimal parameters values of the [Walk Forward Parameters](#) optimization process are determined by the statistical methods entered in the [Walk Forward Goodness](#) field. If the [Walk Forward Goodness](#) parameter field is left blank, the MAR statistic will be used to determine the optimal parameter values to test on the [OOS](#) section of a segment.

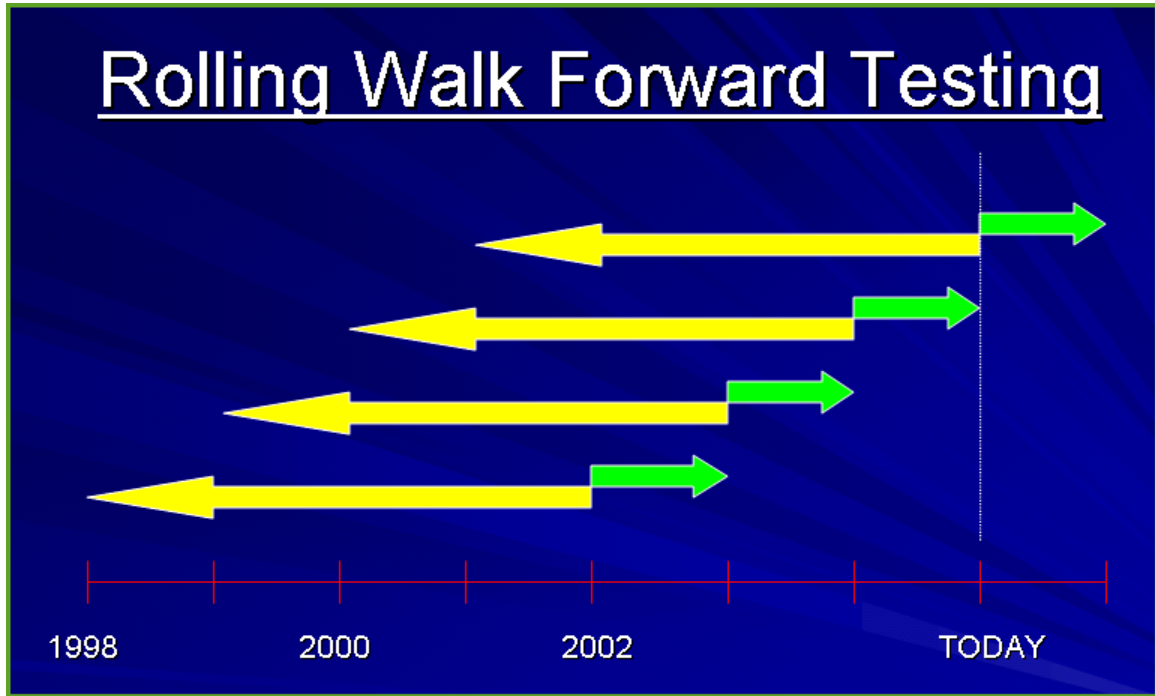
At the completion of the stepped parameter test in the optimizing segment, Trading Blox will apply the optimal parameter values using a normal single test to determine how the optimal parameters perform on the unseen [Out of Sample](#) data section of the tested segment.

Rolling Walk Forward Testing: [^Top](#)

This next chart shows arrows that make up the two sections of each data segment used by the Walk-Forward testing process. Yellow arrows point toward the left side of the image show the data range section used for parameter optimization. Green arrows pointing toward the right side of the image are the [Out of Sample \(OOS\)](#) data portion used to validate the optimal parameters.

The logic behind this type of testing is based on the concept that optimal parameters should be

developed on historical data, but not accepted for trading until they prove themselves on unseen data.



Walk-Forward Step Process Example

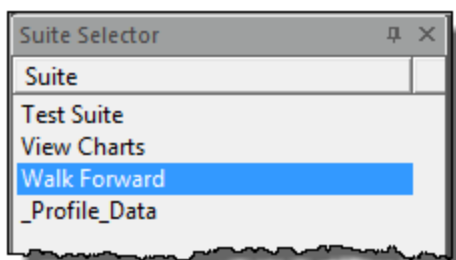
Unseen data can be any instrument's data that is large enough to provide a lot of record variations on which to develop a reliable measure of performance during the parameter optimization stage so it can perform reasonably close to the same performance on an additional large number of unseen record variations. Another way to test a system with OOS instrument records is to run the system after development off-line to see how it performs over the next year or more of time in paper-trading mode.

Walk Forward Simulation Example:

Walk Forward Suite Selection: [^Top](#)

Walk Forward testing can be performed with any suite name that has a system attached. The suite name doesn't need to be **Walk Forward** it can be a different suite name. When a suite is using a different name, there must also be a system with the same name in order for the **Walk Forward Stats 2** reporting blox to have a system in which it is enabled.

Our second step will be the same for all suite names. In this step we need to select a system. Any system that is test ready and has parameters that can step in value will work with Walk Forward optimization.



Any Suite name can be selected, but there must be a system name that matches the suite name when the Walk Forward Stats 2 blox statistical results are needed.

System Selection: [^Top](#)

Our example testing suite name is **Walk Forward**. We will include the **Walk Forward Stats 2** results in our testing example. This means we need a system that is named, Walk Forward so the statistical blox can report its result.



The default Donchian System will be this system we use to perform a Walk Forward Parameter Optimization test. Click the Donchian system that it shows that it is included in our Walk Forward Suite simulation. Our example test uses a Donchian system with the **Donchian SmartFills - EntryExit blox** as its entry exit order module.

Walk Forward Stepped Parameters: [^Top](#)

Our example test will step the breakout entry and exit bar length values as the parameters to optimize. If you are using the same system, click on the Step option for the Entry and the Exit Breakout parameters. Enter the values shown for each parameter when their fields expand:

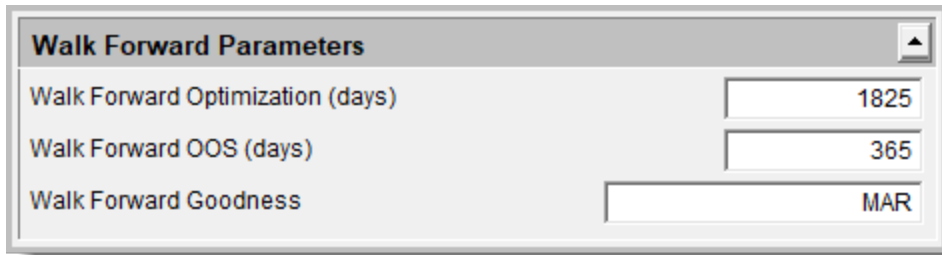
Entry, Exit	
Entry Breakout (bars)	Step ☒ from
22 to 25 by 1	
Entry Offset (atr)	Step ☐ -0.4
Stop (atr)	Step ☐ 1.8
Exit Breakout (bars)	Step ☒ from
10 to 40 by 5	
Exit Offset (atr)	Step ☐ 0
ATR (bars)	Step ☐ 39

Donchian Entry Exit with Smart Fills Stepped Parameters

If you are using a different system, Enable the Step option for the two parameters shown in your system and select a range you wish to test.

Walk Forward Segment Settings: [^Top](#)

First step is to establish the optimization and OOS data section lengths in the Suite's Walk Forward Parameter settings. This step is where the test in sample optimization data lengths and out of sample validation testing periods are established.



Walk Forward Parameters	
Walk Forward Optimization (days)	1825
Walk Forward OOS (days)	365
Walk Forward Goodness	MAR

Global Parameter Walk Forward Method Control.

In the image above the Suite's Walk Forward Parameter field the optimization days/bars needs the first 1,825 records will be the in sample optimization period, and the next 365 instrument records will be out of sample (OOS) will be the parameter validation period. Each additional segment will adjust its starting period forward by the number of OOS days and repeat the test again so as to generate a new OOS performance record.

Note:

Most instrument markets trade five days a week, and there are fifty-two weeks in a year in which there are often about eight holidays in each year that means there are about 252 trading days in a year. Using the above assumption, the optimization days value of 1,825 data records shown above will cover a time period of about 7.25 years. In the second parameter value of 365 days that creates an OOS sample Walk Forward time range of about 1.45 years. To have four independent test segments, the number of records required for each segment is 8.7 years for each test segment. To have enough data range between dates, using the same parameter values requires $(1,825 + (365 \times 3)) = 2,920$ records, or about 11.6 years of data.

Walk Forward Goodness Setting: [^Top](#)

Set the goodness measure in the Walk Forward Goodness parameter. This can be any built-in statistic or custom statistic added with [test.AddStatistic](#). The optimal run from the optimization runs is determined by the maximum goodness value.

In our Walk Forward test example will use the MAR statistic as the basis for selecting the best parameter values.

Statistical Measure Selections: [^Top](#)

If you wish to use a different statistical measure, then change the determined index number before clicking OK.

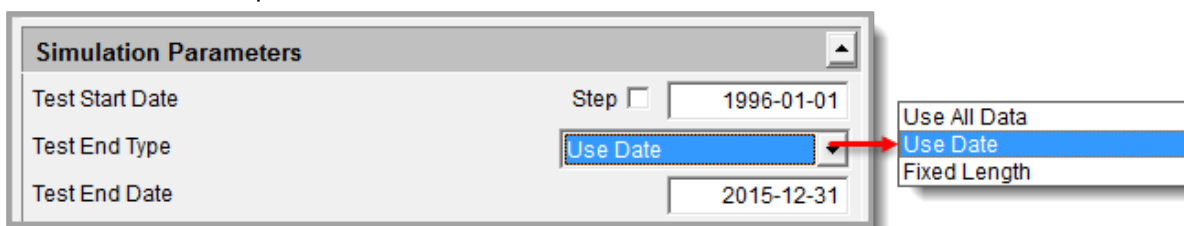
Statistical Measure Selections: [^Top](#)

The logic of the Walk Forward optimization process is designed to select a statistical measure's highest value. While that is how Walk Forward will return results, it is possible with some statistical measures to select the peak where the values on each side of a peak are significantly lower. Selecting the pinnacle value is a measure that shows a lot of volatility might not work out well because it is unlikely the selected parameters will perform as well during variations.

Graphs of the statistical measure show the measure's volatility when displayed. If the measure is volatile, consider a different statistical measure or consider changes to the logic so its performance can be from the top of a more flattened curve with gentle slopes.

Walk Forward Start and End Dates: [^Top](#)

Check the date range set for the Walk Forward simulation. Walk Forward will perform all three of the date selection options, but if you wish to control how many segments, or want only one, then use the date information [Walk Forward Segment Settings](#) to help you establish how this Walk Forward test is performed.



Suite's Simulation Start & End Date Control Parameters

Our Walk Forward example will use the Walk Forward Stats 2 blox in the system name: Walk Forward.

Controlling Stops:

Saving of open positions from one OOS test to the next has limitation because any variables that might be tracking trailing stops in any manner are not brought over to the next testing segment.

Walk Forward Testing Threads: [^Top](#)

Each thread in a stepped simulation sets up an entire copy of the systems, blocks, objects, and variables, and runs the before and after simulation scripts for processing. You can access the parameter test number, and the thread index, to control the scripting as needed. The threads are run simultaneously, and in no particular order, so you can't share values between threads as you don't know which parameter test is running first or second within a thread group. Print out the parameter test number and thread index for a large test to see what I mean.

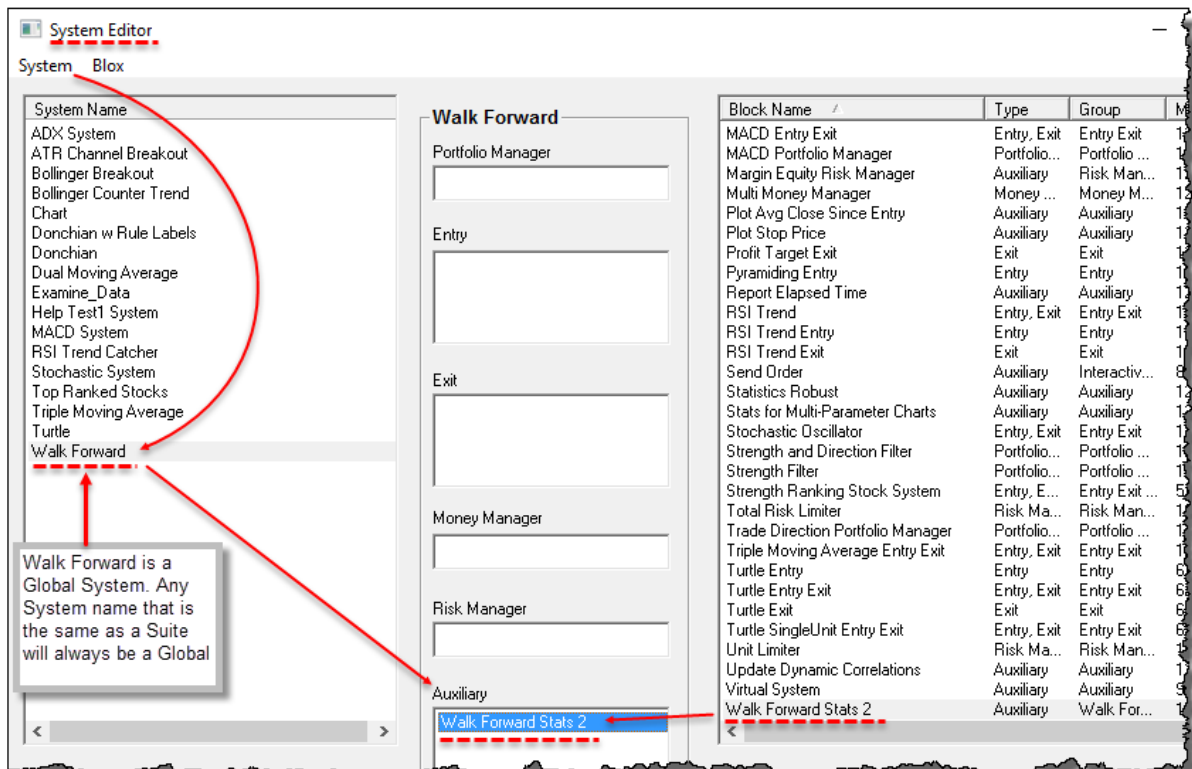
The way to save global values is either in the registry (for single values) or a file (for arrays). Take a look at the Walk Forward Stats 2 blox for an example of saving the stitched equity curve. In this case the out of sample runs (`test.walkForwardStatus=2`) do not execute at the same time, so we can output the equity values in order. If you are exporting values from regular test runs, keep in mind they can be out of order (test 4 running before test 1). You may run into conflicts attempting to write to the file from multiple threads at the same time, so the work

Walk Forward Testing Threads: [^Top](#)

around to that in the past has been to write to a different file for each thread, and then stitch them back together in the correct order in the After Test script section.

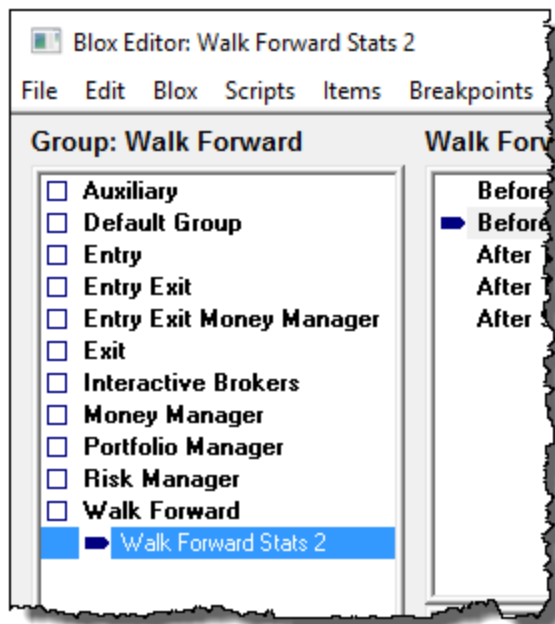
Walk Forward Statistics Reporter: [^Top](#)

Adding the Walk Forward Stats 2 blox statistical information to a Walk Forward test is similar to how others blox are added to a system, but there is one exception. To get started, enter the System Editor using the menu or by pressing F3:



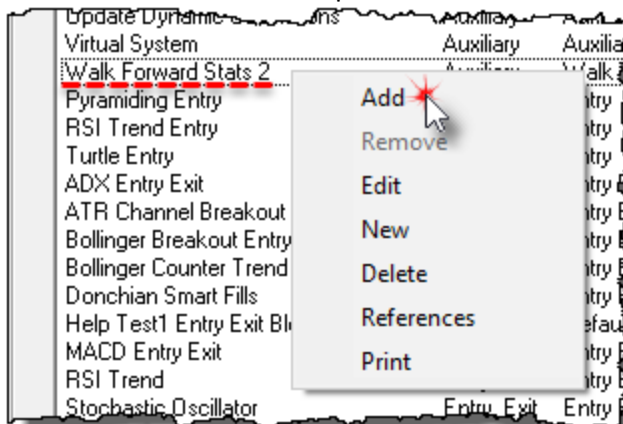
In the left column near the bottom is the **Global System** name, **Walk Forward**, created for this testing example. Click on that system name. If the **Walk Forward Stats 2** blox is not attached, follow the next steps to add it.

This Walk Forward Stats 2 blox is Installed when Trading Blox is installed. It is designed to report statistical and graphic results created during a Walk-Forward Parameter Optimization simulation.



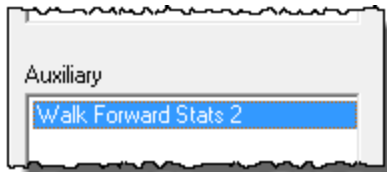
To include this additional statistical module it must be attached to a system with the same name as the Walk Forward testing suite. Any system that has the same name as a Suite, Trading Blox designates that system as a Global System. This statistical module will only work with a Global Suite. When it is included in a Walk Forward test the Walk Forward Stats 2 statistical results will be included in the [Stepped Parameter Test Summary Report](#), and the .

To find the Walk Forward Stats 2, open the System Editor and locate the blox name. Blox is designated as an Auxiliary blox. When the blox is found right click to bring up the pop-up menu, and then click on the Add option:



Add Walk Forward Stats 2 to Global Suite

When the add operation is complete the blox name will appear in the Auxiliary section of the System Editor.



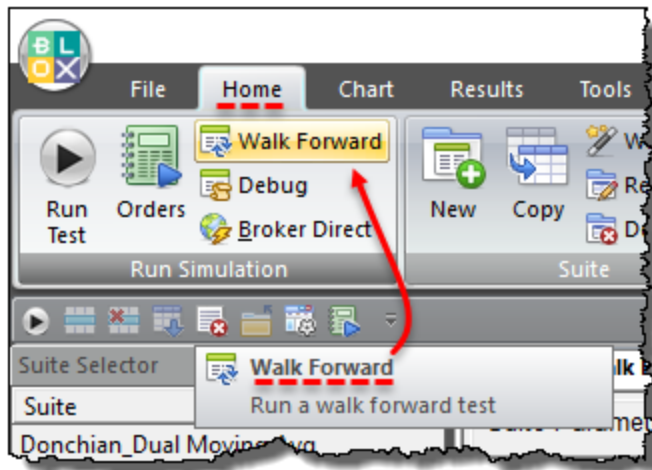
Walk Forward Stats 2 Blox is Added as an Auxiliary blox

Walk Forward Stats 2 Global System Only Blox: [^Top](#)

If you created a different Suite name to use with your system and want to see the Walk Forward Stats 2 blox results, Do Not allow the Walk Forward Stats 2 blox to be attached to a standard simulation system as the blox' internal error checking will cause Trading Blox to halt with an error and report the blox is attached in the wrong place. When Walk Forward Stats 2 is needed, attach it to a global system name that is the same as the selected test suite name.

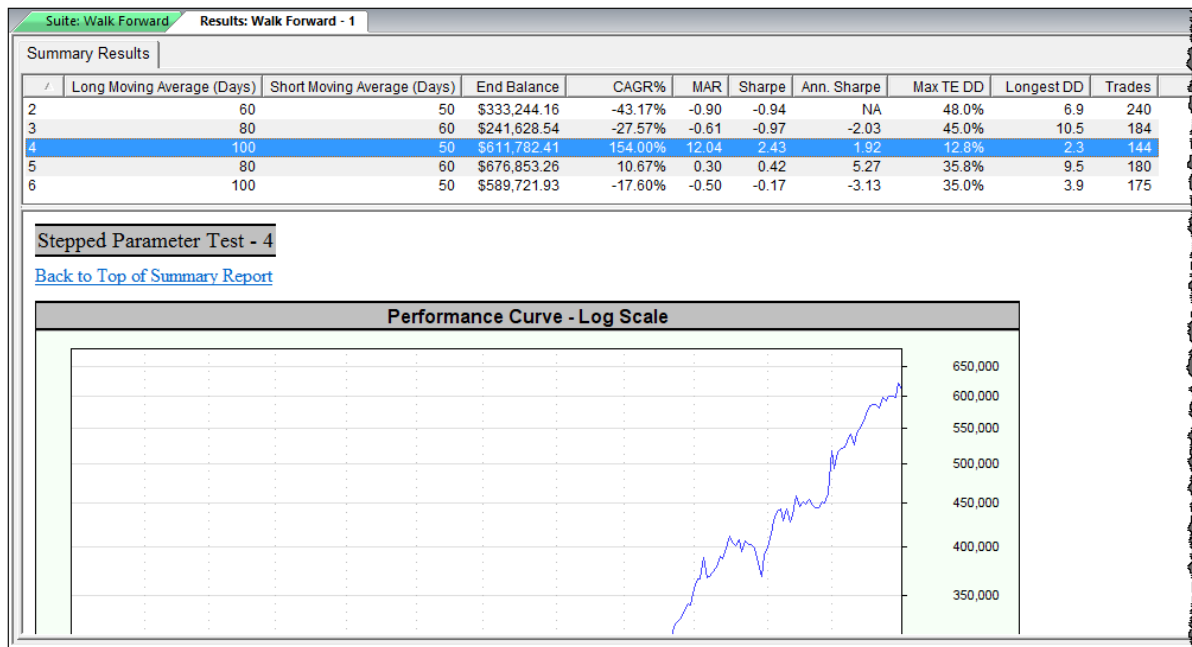
Walk Forward Simulation Test: [^Top](#)

Go to the Home main menu and in the Run Simulation section click on the Walk Forward button to begin the Walk Forward simulation test:



Stepped Parameter Test Summary Report: [^Top](#)

This table shows Summary Performance result from a simple Walk Forward Simulation example using the values shown here:

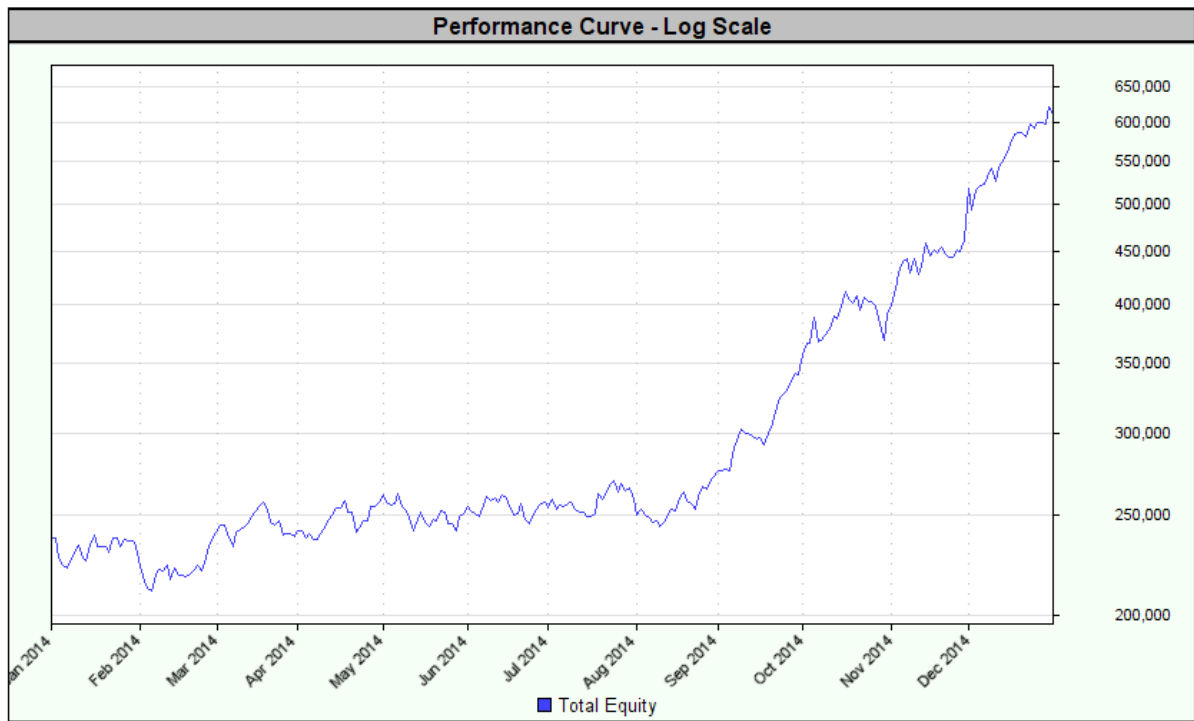


Walk-Forward Stats Performance Summary Example

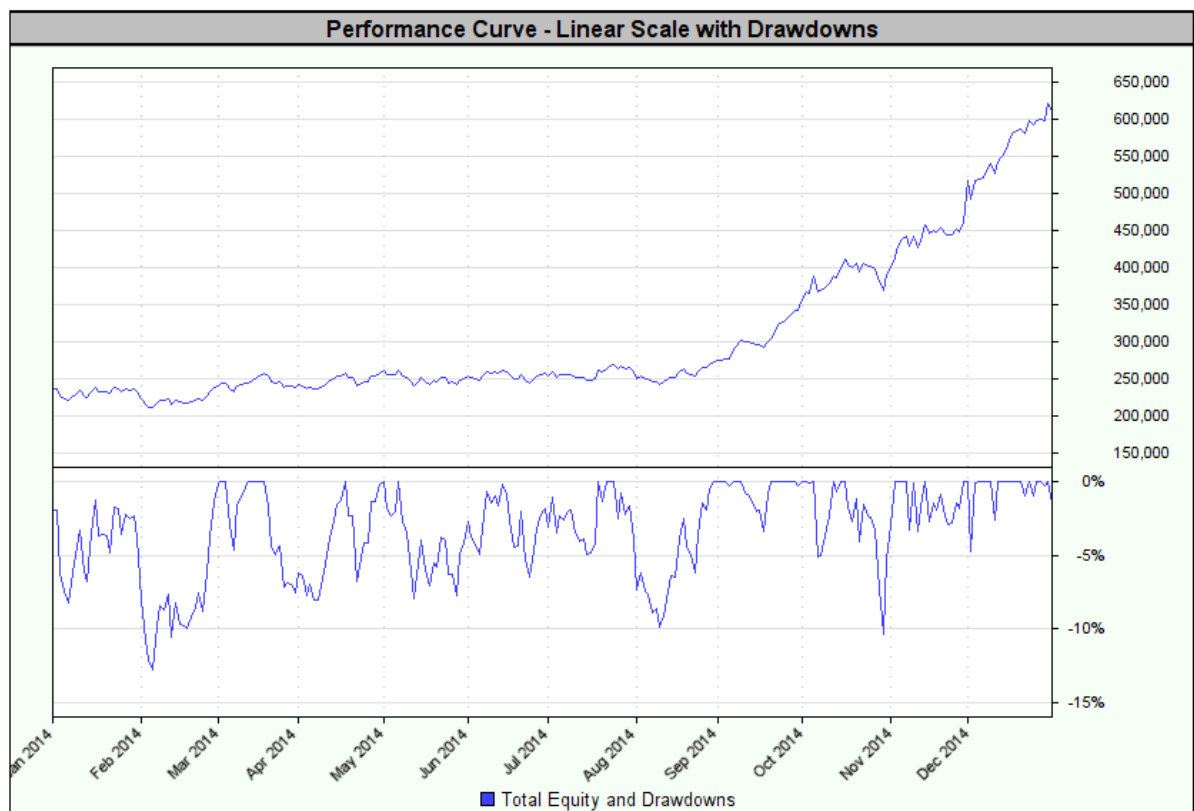
The above image performance table shows each parameter step section for each of the six simulation steps performed. Each step in the test will also show a Stepped Parameter Summary Test report for each of the stepped periods that will appear when any of the numbered test rows are clicked with the mouse.

Walk Forward Stats 2 Graph Results: [^Top](#)

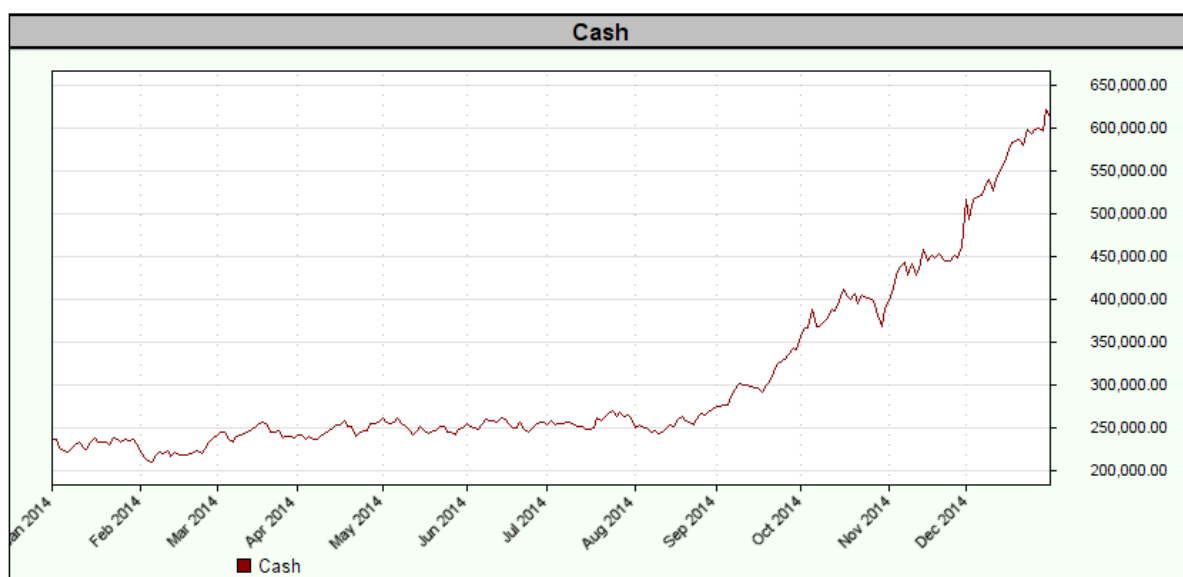
Walk Forward Stats 2 blox attached to the Global System does not have any parameters, how it will display three graphs of how each of the stepped test performed. Each of the three graph examples shown here will appear in each of the stepped performance section shown in the Stepped Performance Summary Report heading area shown above:



Walk Forward Stats 2 Performance Curve Log Scale



Walk Forward Stats 2 Performance Curve Linear Scale with Drawdown



Walk Forward Stats 2 Cash

Walk Forward Stats 2 Testing Results: [^Top](#)

When the Trading Blox [Log Window](#) is active, this additional information will be available in the Log Window display.

```
Starting Instrument Data Load at 09:05:03
Ending Instrument Data Load at 09:05:04
Finished Loading Instruments. Elapsed time: 1 seconds.

Starting Simulation Setup at 09:05:04

28 Test Optimization for dates 20060102 to 20101231
Parameter test 17 will be used for out of sample test due to best goodness of 1.533831
Out of Sample test number 1 for dates 20110101 to 20111231
28 Test Optimization for dates 20070102 to 20111231
Parameter test 19 will be used for out of sample test due to best goodness of 0.953879
Out of Sample test number 2 for dates 20120101 to 20121230 with starting equity 420258.474522
28 Test Optimization for dates 20080102 to 20121230
Parameter test 14 will be used for out of sample test due to best goodness of 0.591911
Out of Sample test number 3 for dates 20121231 to 20131230 with starting equity 363317.638015
28 Test Optimization for dates 20090101 to 20131230
Parameter test 25 will be used for out of sample test due to best goodness of 0.083212
Out of Sample test number 4 for dates 20131231 to 20141230 with starting equity 280854.118846
28 Test Optimization for dates 20100101 to 20141230
Parameter test 20 will be used for out of sample test due to best goodness of 0.115529
Out of Sample test number 5 for dates 20141231 to 20151230 with starting equity 356956.081263
28 Test Optimization for dates 20110101 to 20151230
Parameter test 25 will be used for out of sample test due to best goodness of -0.098728
Out of Sample test number 6 for dates 20151231 to 20160916 with starting equity 317767.967484

Ending Test Run at 09:05:31
Finished Test Run. Elapsed time: 27 seconds.
```

Walk Forward Stats 2 Log Window Test Report

Section 18 – Stats for Multi Parameter Charts

This block sets the statistics to use for the multi parameter charts. Enter the names of the statistics, either built in or custom, separated by commas or spaces.

Code Example:

```
' This function assigns the goodness stats to use for multi-parameter
charts.
' These are the built-ins functions, and Custom stats can be used as
well.
' Some of the other statistics that can be used are "End Balance, CAGR%,
MAR,
' Sharpe, Ann. Sharpe, Max TE DD, Longest DD, Trades"
test.SetGoodnessToChart( goodnessString )
```

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 207

Section 19 – Top Ranked Stocks

The Top Ranked Stocks system periodically ranks the stocks in the portfolio according to one of four user chosen indicators and allocates an equal dollar share of system trading equity to each stock ranking at or above the percentile set by the user. This simple system is long only and trades only on bars when the entire portfolio is reallocated.

The rankings are determined by the user's choice of one of four available indicators along with a parameter for look-back bars:

- Percent Change – The percentage change in closing price over the look-back number of bars.
- Price vs. EMA - The current closing price as a percentage of the look-back bar EMA.
- Momentum in ATRs – The difference in current closing price versus closing price of look-back bars ago expressed as a percentage of the look-back bar ATR.
- Rate of Change – Built-in Rate of Change of closing prices indicator over the look-back bars.

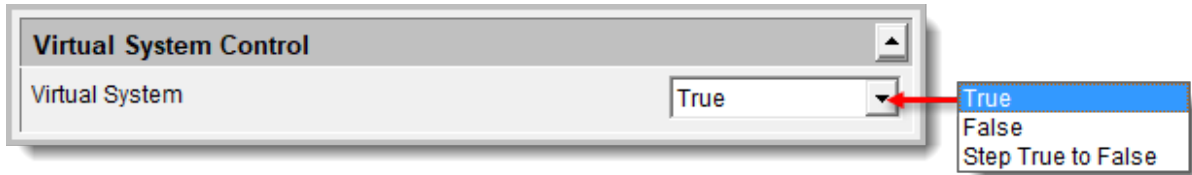
By default, stocks are ranked from highest to lowest indicator values. The Invert Ranking parameter may be set to “True” to rank stocks from lowest to highest.

The user sets the value of the Reallocate Every N Bars parameter to determine the number of bars that elapse between reallocation of the portfolio. At the beginning of the simulation, the portfolio of stocks is ranked and an initial set of positions chosen. After that, every time the set number of bars elapses, the portfolio is re-ranked and positions are reallocated on the next open.

If a stock was included for the last holding period, and after re-ranking is still included, then the size of the position is adjusted on the next open to represent an equal dollar share of current system trading equity. If a stock is held long but no longer ranks high enough to be included, it is exited on the next open. If there is no position in a stock that now ranks high enough for inclusion, then it enters long on the next open with an equal dollar allocation to all other positions.

Section 20 – Virtual System

The Virtual System blox enables a system to execute with two different results.



Virtual System control parameter options.

When Virtual System is set to TRUE, all the test results from the system will show nothing. When set to FALSE, the test of the system will have the normal results that would have happened had the Virtual System blox not been attached to the system.

Virtual control gives the user the ability to have an additional system attached to a suite and not have that system have any impact on the results from the orders it generates. Think of the Virtual System control as the ability to have a system that can be referenced in a suite without it changing the results with its testing results.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 240

Generating Orders

Part

XVII

Part 18 – Generating Orders

Orders are created by Trading Blox Entry functions. When the entry function generates an order to enter the market, it automatically sends the order to the money manager in the system so a quantity can be added.

To generate orders, click on the [Orders Button](#) in the [Home Menu](#) group, or press the **F7** key. Both methods will run the selected suite and its systems, and generate a report of all the active positions and the current orders for the last data date. Orders listed on the report are for the next market's trading day, or when intraday data is being use, the next price bar record.

from you suite and its system, click on the Orders button, or press the F7 key. menu item. Trading Blox will run a primer simulation based on the test start date, and then generate new orders for the next bar. The next bar is the bar after the last bar -- for intraday data this would be the next hour or minute, and for daily data this would be the next day. The order date and time will be noted on the order file. The results will be stored in the Orders folder as well as displayed.

The next bar is determined by the Test-End date selection and the latest record in the portfolio. When some instruments don't have data that is as late as the latest date of the instruments in a portfolio.

The amount of equity available for order generation will be determined by the ending equity of a test run for the period Start Date to End Date (all data). The test start and end date are used to run a test just prior to Order Generation. This pre test is used to get theoretical positions and equity. Typically the start date is set to the date actual trading has started, so starting equity matches the start date. This start date is left constant as orders are generated in the futures.

Another option is to set the Test Start Date a few years in the past, to get some priming and track record. With this approach, set the Test Starting Equity to an amount such that the Test Ending Equity is equal to the amount at the brokerage. Plus or minus 5% is fine. And as the test ending equity varies over time from the brokerage account, only adjust when outside of a threshold, and then adjust slippage or commission as desired to get the ending equity back in line. The test starting equity can also be adjusted, or capital adds and draws can be used. All positions and orders will likely change as a result of this, so not something to do very often.

Using Capital Adds Draws seems like the least intrusive way to deal with the equity variations, but keep in mind that this could lead to drift from the system design. Imagine constantly adding capital, because the system is making more money than the simulation. Rather than treat the symptom, perhaps treat the cause, which is likely an unreasonably low slippage or commission assumption. Once these are corrected, the system might perform differently than expected or desired or designed.

The filter for volume, margin, and Risk Manager (Can Add Unit script) are run as orders are generated. So orders generated will be pre filtered for these factors. Note that scripts like After Instrument Open, Can Fill Order, and After Instrument Day, will not be run for the order generation bar. The instrument.tradeDayOpen is also not available for the order generation bar.

The name of the file will be: Orders <Timestamp> .mht. The Timestamp has the format YYYY-MM-DD_HH_MM_SS.

The order file will look like the following. It shows the open positions at the top, and the orders for tomorrow at the bottom.

For each order, the Current Position is listed as well: it shows either LONG, SHORT, or OUT; where OUT represents no current open position in the market.

An additional file, "Orders.csv" is created (and overwritten) for every order run. You can open this file in excel, and filter your orders as necessary.



Suite: TestSuite

System: Donchian 3

Data Updated: 2017-04-21

Current open positions 2017-04-24											
Position	Quantity	Market	Unit	Entry	Entry Date	Exchange	Close	Total Profit	Bars	Roll Info	Rule
Short	38	BON7	1	31.91	2017-03-27	CBT	32.16	-5,700	18 N7	on 20170412	
Short	55	CN7	1	361 2/8	2017-04-21	CBT	363 6/8	-6,875	0		
Short	39	WN7	1	429 4/8	2017-04-20	CBT	421	16,575	1		

New orders for 2017-04-24												
Order	Quantity	Market	Unit	Price	Execution	Exit Stop	Exit Limit	Exchange	Position	Close	Distance	Rule
Short Entry	11	GCM7	1	1241.5	on Stop	1269.0	Undefined	COMEX	Out	1289.1	3.47	2017-04-21
Short Exit	38	BON7	1	32.33	on Stop	N/A	N/A	CBT	Short	32.16	0.26	2017-04-21
Short Exit	38	BON7	1	32.32	on Stop	N/A	N/A	CBT	Short	32.16	0.25	2017-04-21
Short Exit	55	CN7	1	372 6/8	on Stop	N/A	N/A	CBT	Short	363 6/8	1.61	2017-04-21
Short Exit	55	CN7	1	379 4/8	on Stop	N/A	N/A	CBT	Short	363 6/8	2.83	2017-04-21
Short Exit	39	WN7	1	445 4/8	on Stop	N/A	N/A	CBT	Short	421	2.93	2017-04-21
Short Exit	39	WN7	1	449 4/8	on Stop	N/A	N/A	CBT	Short	421	3.41	2017-04-21

Data has been updated through: 2017-04-21

Report Created: 2017-05-03_09_58_22
Copyright Trading Blox, LLC 2017

Delivery Month

If the data source includes the Delivery Month, the month of the positions will also be listed. The delivery month of the orders generated will also be listed. If they do not match, ie if you have a position in a back month while the data has rolled to the new front month, an * will appear next to the order price to highlight this roll. A note will be printed in the positions section that you should roll this position from the old month to the new month. The roll alert message and * will only show for 1 day. Thereafter it is assumed that the position has rolled from the old position to the new position.

All prices are in current month prices, including the entry price of the position, and all the new order prices. Trading Blox assumes the position will be rolled on the open, because the data has already rolled. It is important to do this in that you want to 'trade what you test.' So if your historical testing assumed a certain roll algorithm, and your data is setup for such, you should roll your positions at the same time.

Distance

Distance is the number of ATR from the last Close price to the Order price. This gives you some idea of how much the market would need to move to fill this order, so you can place the orders into the market as necessary.

The ATR used is the Wilder 20-day ATR (39 day Exponential Moving Average of the True Range).

Last Bar

The Last Bar is the date of the last bar of data for the market. If this date is not the current date, or as expected due to holidays, then you should double check your data. It's possible the data for a particular market was not updated on schedule. In this case the Order Report is for April 6 (Monday) and all the markets have a Last Bar of April 3 (Friday).

Message

The Message is a custom value set in Blox Scripting. This can be any message (string value) as needed by the system.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 148

Section 1 – Broker Positions

We don't recommend using this feature; If used, the Order Generation Equity must be updated every day.

We recommend using a simulation to determine the positions, fills, new orders and current equity. With a simulation, you can ensure the brokerage account matches the positions and orders created with a simulation. Without the diligence to ensure your system information matches your brokerage account, you will find the system information will not be what you need as information from both sources drifts in different directions .

Just as important, it's a lot of work to manually update position records without the risk errors that can easily accumulate.

An issue that happens often, is the failure you will not get the exact fill that a historical simulation might assume when trading an actual account. This is to be expected given all the details each trading day creates. It is also easy to forget the update a contract roll change when the underlying data source has rolled to a new contract. Other times, your broker may fill an incorrect order, or you may enter an order incorrectly.

To enable these position differences and have them inserted into the test information, use the [Use Broker Positions](#) global parameter to enable the simulation to use the information in this Broker Positions as an information source that will reduce the difference between the simulations and the brokerage account history.

The Broker Positions screen lets you tell Trading Blox what your actual positions are so that Trading Blox can generate orders using those existing positions rather than using positions based on historical testing assumptions. It should be noted that a preferred approach for many traders is to match their actual trading to the simulation. So in other words, let the simulation create the positions and make sure your broker positions match the system.

Our primary reason we do not recommend using this Broker Positions feature is the added complexity and its manual updating requirement is likely to have errors at some point. The Broker Positions tool is here for you, but you will need to decide if the complexity is worth the effort and risk.

Using Broker Positions:

System	Symbol	Delivery Month	Fill Date	Fill Time	Position	Execution	Fill Price
Donchian	GC		20090206		Long	Open	900
Turtle	MSFT		20040921	0850	Short	Open	27.00
Turtle	msft		20040922	12:00	Out	Bar	27.00

System Name	Donchian	Position	Long	OK Cancel New Position Delete Position
Broker Symbol	GC	Execution Type	Open	
Delivery Month (YYYYMM)		Fill Price	900	
Fill Date (YYYYMMDD)	20090206	Entry Protective Stop Price	890	
Fill Time (HHMM)		Entry Quantity	10	

Broker Position Editor - Available in the main screen's Home menu group.

Enter your actual positions, fills, and quantities that you have with your broker.

For stocks and futures be sure to **enter the actual fill prices** from your broker statement at the time of the fill, not the stock split or futures back adjusted prices. Stock prices will be adjusted by the stock split ratio to match the stock split adjusted data. Likewise the quantity will be adjusted on the stock split date to represent the new adjusted quantity. Futures prices will be adjusted during the test to reflect the back adjusted prices used in trading and on the charts.

These positions get inserted into the test on the **Order Date** specified, and any existing positions will be exited as a result. You can enter a **Zero-Quantity** to lock out any further positions in this instrument. You can enter a position of **OUT** to exit any existing positions on the **Order date**.

NOTE:

Positions entered here are locked -- so subsequent exits in the system code will not cause an exit of these positions. Entering a subsequent position of **"OUT"** will exit the position and leave it locked so no further entries will be filled by the simulation. Enter a position of **"Exit"** to exit the position and unlock the instrument so that further entries and exits by the simulation can occur. The order generation report will still have new entries and exits as needed for this positions. The order generation is not locked, just the simulation fill process.

For Each Broker Position use the item description shown:

Position Element Names:	Description:
System Name	When a Suite is selected and displayed in the Main screens display, the

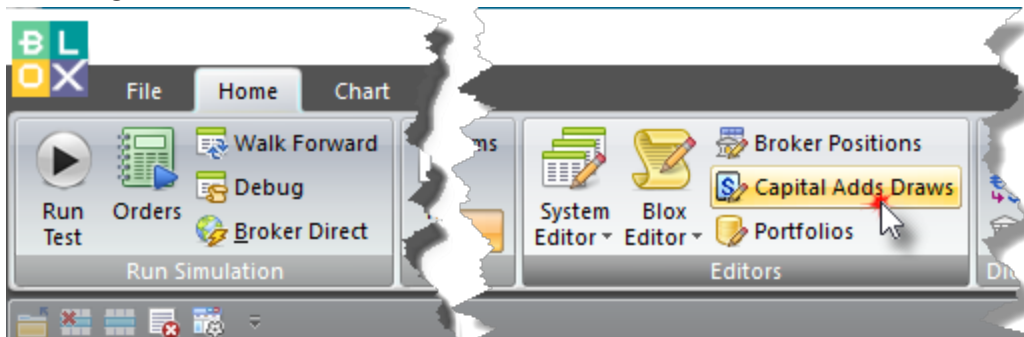
Position Element Names:	Description:
	System Name displayed will be automatically entered. When it isn't filled in automatically, be sure the system name is a perfect match the system name where the position is intended. Failure to or the trade will be ignored. Drop down box populates from selected systems for the current suite.
Symbol	Ensure the entered symbol is an exact match.
Delivery month	This item is optional. It is designed to suppress the alert message once you have rolled from the old month to the new month. The data in the record is only available when it matches the delivery month.
Fill Date	A date entries must be an actual trading day fill-date. This date must be within the test range of that instrument. When it isn't in the test range this information be ignored. Allowed Date formats are: YYYY-MM-DD YYYYMMDD YYMMDD MM/DD/YY MM/DD/YYYY
Fill Time	Fill-Time must be an available time in the data file. For daily data, this item can be ignored. For intraday-data it is used and it must use this time format: HHMM format
Position	Each position direction must be one of the items in the Drop down box list of options: Long Short Out Exit When Out is selected, the position will exit and lock the instrument. Position with Exit selected, will then exits the position and unlock the instrument .
Execution Type	There are three options: OPEN BAR EXIT When set to Open, the position will be 'filled' before any simulated positions for a specific bar. When set to Bar, a simulation open positions for the bar will be filled first, then this broker position will be filled. When set to Close, this broker position will be filled last.
Fill Price	A Fill-Price is required for all Entries and Exits. Price to use is the transaction price of the order. Do not use any commas in the price.
Entry Protective Stop Price	When a Entry price is left blank, the system will assume the Order-Type was not an ON_STOP required price condition. This price is asking for the original entry day protective stop price applied to the order when it was created. The protective stop price used in your system for the remaining position days will likely vary from the original price as the position ages.
Entry Quantity	Actual order quantity size goes here.

NOTE:

- o The position entry delivery month is only presented as long as the data is still priced in that month plus a few days.
- o For 2 days after the roll, a roll alert is presented indicating that the position should be rolled from the old month to the new month.
- o Then after the 2 days the system assumes you have rolled and the entry delivery month is changed to the current data delivery month.

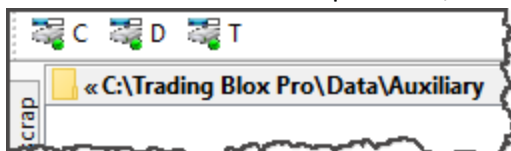
Broker Position Data & Capital Adds & Draw Changes Location:

To access the **Capital Adds & Draws dialog**, click on the **Capital Adds Draw Button** shown in the next image:



Capital Adds & Draws Button

The Broker Position and Capital Add/Draw files are now located in Data/Auxiliary.



Broker Orders & Capital Adds & Draw Data File Location.

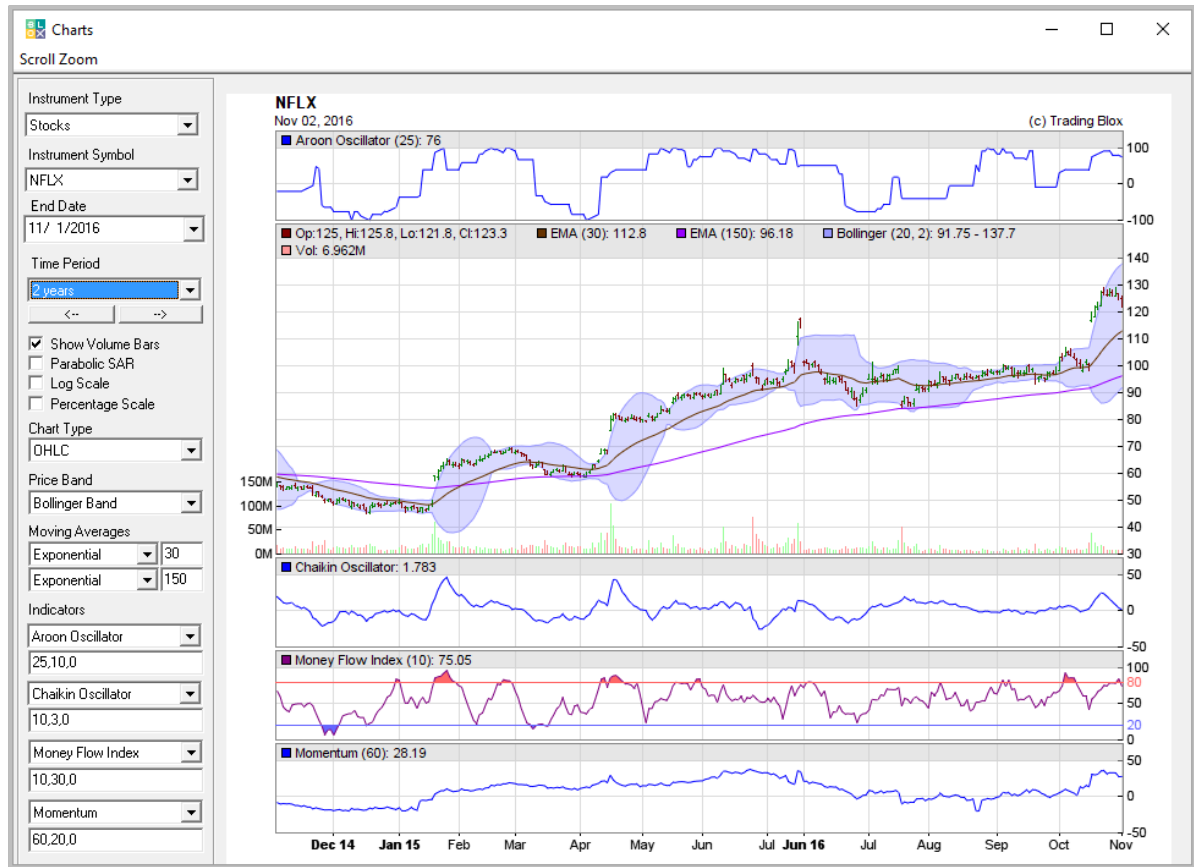
Quick Charts

Part



Part 19 – Quick Charts

The [Tools Menu](#) Charting area provides a [Quick Chart](#) control to display and use a simple charting feature that will display Quick Charts in this Trading Blox Builder Licensed version.



Quick Chart Stock Symbol Example

Quick Chart Options:	Description:
Instrument Type	Stock/Futures/Forex
Instrument Symbol	The symbol as entered into the Dictionary
End Date	Defaults to today, but can be set to any date using the calendar picker
Time Period	Set to the time period for the chart. The back and forward arrows move that amount forward and back in time, to scroll the chart.
Chart Options	Show Volume Bars Parabolic SAR Log Scale

Quick Chart Options:	Description:
	Percentage Scale
Chart Type	Candlestick Closing Price Median Price OHLC Bars Typical Price Weighted Close
Price Band	Bollinger Band Donchian Band Envelope
Moving Averages	Up to two moving averages can be plotted on the chart. Select the type, and enter the bars to use to the right. Simple Exponential Triangular Weighted
Indicators	Up to four indicators can be plotted with the chart. The first one will be on top. Use the space below to enter the parameters such as 14 or 14,20 or 14,20,10 depending on how many parameters the indicator needs. Values to enter are limited to three values. Enter the left-most value first, and then the other one or two values if needed. When more values are entered than needed, the value not needed are ignored.
	Note: These indicators are all computed by the third party charting package and are for visual reference only. They are not necessarily computed the same way as the indicators in the Trading Blox Systems.

Quick Chart Indicators:		
Accumulation/Distribution Aroon Oscillator Aroon Up/Down Avg Directional Index Avg True Range Bollinger Band Width Chaikin Money Flow Chaikin Oscillator Chaikin Volatility Close Location Value Commodity Channel Index	Donchian Channel Width Ease of Movement Fast Stochastic MACD Mass Index Momentum Money Flow Index Neg Volume Index On Balance Volume Performance % Price Oscillator	Pos. Volume Index Price Volume Trend Rate of Change RSI Slow Stochastic Stochastic-RSI TRIX Ultimate Oscillator Volume William's %R

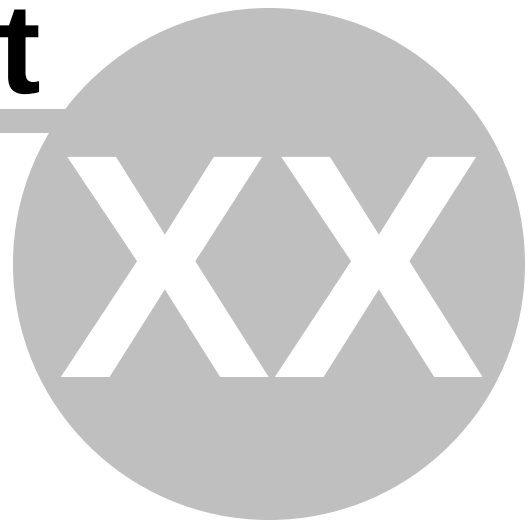
Quick Chart Indicators:		
Detrended Price Osc.	% Volume Oscillator	

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 189

Upgrade Changes

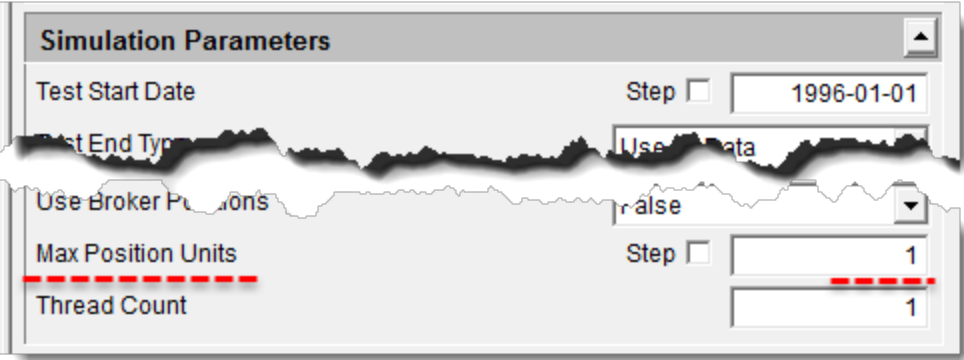
Part



Part 20 – Upgrade Changes

Before each new release of Trading Blox the software is improved. New features are added, correction, if any are implemented, and some times keyword names need to change so that an improvement in a features will operate better.

Trading Blox allows a user to have as many installation of software on the same computer as the user wants. Once Trading Blox is installed, a second copy of the software should be installed in a different folder name so that upgrades to the software can first be performed in an isolated test area that won't disrupt the Trading Blox the installation that is used for trading, or for system development. By adding safe "Sandbox" area where the upgrades can first be tested on your computer, the Suites being used in trading and development won't be affected by changes that need to be made before they can operate in the new version.

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
1	The Global Parameter property 'Minimum Slippage' is now a Futures only parameter. In testing and order generation results, this parameter control is only active for portfolios with Futures instruments. Stock results are not affected by this Global Property setting. If a Suite is exported and imported between version 4 version 5, the setting of this property will be set to zero.
2	Previously, the Max Units property setting was only accessible in by editing the Trading Blox 'Test Settings.ini' file. In that file it was named, Maximum Units. This property is now visible in the Global Parameter Simulation it can now be accessed in the Global Parameter area.  Default value is now 1-Unit. Previous value was 7-Units. Be sure the multiple-unit system being tested are not be limited by this property setting. Note:

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	A position's quantity is the number of shares or contracts are applied to an order. A position's Unit count represents the number of orders that have been applied during the the time current position has been active.
3	The Update indicators script will run from the start of data for priming purposes. In version 4 the Update Indicator script would only begin its execution of this script after the instrument's data record reached test Start Date set for the test or order generation.
4	This feature can be changed by opening the settings file 'TradingBlox.ini' in the Trading Blox installation folder and changing the value of this setting from True to False: Allow Priming Update Indicators=TRUE ' Allows the Update Indicator script to run from the beginning of an instrument's data file. Allow Priming Update Indicators=FALSE ' Allows the Update Indicator script to run after an instrument data record testing reaches the user's entered Start Date value.
5	"Use Start Date Stepping" option is now part of the Start and End date parameter group in the Simulation Parameters section. With both of these period control parameters now part of the same process the operation of how Walk Forward testing operates will be much easier to understand.
?	Max Percent Market Volume is missing This volume number is only used for the AverageVolume property, which is used for the Max Percent Volume per Trade and Minimum Stock Volume global parameters. You can also access these properties as <code>instrument.adjustedVolume</code>, and <code>instrument.averageVolume</code>.
	Version 5.0.2 Released December 8, 2016 -- Blox Editor window can move behind application window. -- F5 from Blox Editor saves changes and runs test. -- Order property default for IB OutsideRTH (order.outsideRTH) is now true by default. -- When connected to IB, all orders sent regardless of data format. -- New TBFXCM data option for free access to intraday forex data. -- Walk Forward OOS test open positions now have rule message denoting they are saved for the next OOS run. -- New IB dashboard showing live open position status. -- New Forgot License button on authentication screen for easy access to Forgot License web process. -- Fixed problem with indicators using open as data type -- IB broker direct auto roll for futures with data from external source like CSI.

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	-- Added BrokerDirectDate() and BrokerDirectTime() functions that return the current broker direct bar. -- Added StopBrokerDirect() function. -- Futures month auto roll with IB broker direct -- Process Weekends .ini flag allows Saturday as trading day. Good for monthly data. Version 5.0.1 Released November 1, 2016 -- Added system.isVirtual property -- Added test.SetGoodnessforWalkForward() function. -- Fixed a few bugs -- Stability improvements -- New broker direct IB features, bring actual fills back from IB to TB, modify existing orders rather than cancel and replace. -- Removed IB scripting connections -- New Data/Auxiliary/ Folder where broker positions and capital add/draw files are located, along with new ib order fill file. Be sure to move required files here from the Dictionaries folder. Version 5.0.0.5 Released October 15, 2016 - Fixed a few graphical issues. - Added symbol selector drop down box to Quick Charts - Added scroll left, scroll right, zoom in, zoom out, short cuts to Quick Charts - R2, RAR, R3, and chart DD is now sensitive to equity manager selector for closed or total equity. - New test progress dialog process. More updates, less CPU usage. - Faster multi threaded data loading - New Tools / Refresh Windows (reset settings) Process - IB Broker Direct updates. Version 5.0.0.0 Released October 12, 2016 Blox Editor

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	- Parameters can now be dependent on other parameters. A dependent parameter will be show and hide based on the values in the control parameter. Set the value of the control parameter to show the dependent parameter. Selectors start at 0. Boolean true is 1 and false is 0. - Indicators can now use other indicators as an input value. - Block/Group tree updates. Block New menu item always available. - Option to create new parameter while editing an indicator. - Option to edit parameter while editing an indicator. System Editor - Control is now menu based. System and Block menu can be used from top menu or right click on item. - Edit of blox now opens the block in a single block editor window. - Option to create a new block from the system editor. Blox Scripting - Added: MaxSynchBars function to return the maximum date synched bars available between two series. - Added: instrument.purchaseEquity - Added: system.purchaseEquity - Added: system.marketOrdersValue - Added: test.purchaseEquity - Added: test.marketOrdersValue - Added: test.testEndOrig for use in WF testing or when scripting has changed the test end date. - Added: order.SystemName - Added: order.BlockName - Added: order.ScriptName - Changed: Update Indicators script now runs for each instrument as the data is primed, rather than waiting until the Test Start date. In this way Custom Indicators can be primed before the test start. This can be disabled in the .ini file.

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	- Added ConnorRSI indicator - Added order.SetUnitPositionMessage function to add a message to the position listing on the order generation report. - Added position message to order generation report. - Added rule to orders on order report - Removed stop from positions on order report – stop is not always used or active - Added order.SymbolWithType for using with LoadSymbol - Added exit limit price to order report - Stop and Limit prices for entry order are N/A if entry day retracement is < 0 (disabled). - LoadIPVFromFile now accepts an optional integer for columns to skip, and/or the use of "" as the series name to skip columns. In this way some columns of a file can be loaded discreetly. - Added test.sortInstrumentList - Parameters are now enforced to be read only. - New error checking routines create warnings for scope conflict - Added zero affinity parameter to chart.setYAxisAutoScale - Removed system.correlation. Use Correlation functions instead. - Added order.symbolIndex for use in LoadSymbol. Returns the instrument.priorityIndex based on the system portfolio. - Added line width parameter to chart.AddLineSeries function. - Added RegressionError function. - Added system.portfolioType property - Added order.SymbolRef which returns the instrument.referenceID for use in LoadSymbol. - Instrument.LoadSymbol can now take the instrument.referenceID as a parameter. Very fast. - Fixed issue with RSI function using small non auto indexed series - Updated External checkbox for variables to be more flexible. Can find a system scoped IPV in another system using LoadSymbol. - Added BuildStockDictionary script function. Use with a futures portfolio only, and in a separate block, system, and suite. Test cannot continue after this call.

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	- Added ReloadPortfolios script function. Use in separate block, system, suite to load up a script created portfolio. Trade Chart - Resize trade list section of the window with resizer bar. - Resize of indicator areas more intuitive with resizer bar. - Resize right column detail width. - Cursor dynamically changes based on options. - Chart menu includes color and trade details preferences - Right click on trade chart for preferences. - Right click on trade details for preferences - Right click on cross hair detail for display preferences - Right click on indicators for temporal plotting options, and also option to edit and/or create a new indicator. Edit of plotting options will be reflected dynamically on the chart. Changes that affect indicator series values require a new test run to update. - Double click on indicator to edit. - Right click on the Indicator header to create a new indicator, selecting the block. - New headers showing the suite, test run, summary result details. - Added Test Dates to Data Dates at top of Crosshair detail section. - New "Scripts: System Name" section for easy access to all scripts. Double click or Right click to access all scripts from all blox in the system. - Double click on any trade Entry Date or Price to access the script that created the Entry order. - Double click on any trade Exit Date or Price to access the script that created the Exit order. - Double click on any trade quantity to access the script "unit size" that set the trade quantity. - New candlestick and bar sizing based on chart zoom - Removed initial stop price marker as confusing for trailing stops - Added candle stick Ask chart type for forex. Default is bid pricing. Right click on chart to change.

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	Summary Results Window • Double click on summary result row to see the chart details for that run. • Summary results and order reports are now all .html extension instead of .htm Results Tab • Right click on Results tab for simulation run specific results like trade log. • Main Results menu items are context sensitive depending on results tab selected. • Option to open a historic Summary Results file Suites • Right click on suite for all “Suite Menu” options • Test start end dates are part of Suite Parameters • Global Parameters renamed Suite Parameters • Click on the suite to focus on the System Parameter Tabs • New Wizard option to create a suite, system, and block all at the same time. • Up / Down buttons to resize the suite listing area. • New “Walk Forward” suite which should contain the GSS Walk Forward System which should contain the Walk Forward Stats 2 block. This block does some stitched summary statistics and can be modified as desired. The Walk Forward Suite will do a WF test if Run WF is used or if just Run Simulation is used. If Order Generation is used on the Walk Forward suite, the last OOS test run is used to generate orders. • New Broker Direct process that runs order generation each x minutes (bar length) for auto trading with FXCM or IB. Systems • Double click on system to edit in System Editor • Right click on system for all “System Menu” options like edit, add, remove, and quick access to all blox in the system. • Button to show/hide the system list. By default, only active systems are displayed.

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	- Click on the system to focus on the System Parameter Tab - Click on the Suite Parameters to focus on the suite parameters tab Suite Parameter - Added: Use Stock Split global parameter to turn on/off the use of the stock split ratio. Useful when using total value ETF where the splits and dividends are all included in the adjusted close. - Added: Max Units global parameter. This sets the maximum units for each position. - Changed: Max Margin Equity global parameter is now test based, and uses total equity and margin for stocks and closed equity and purchase equity for stocks. Also includes the total margin/purchase equity required for any existing market on open orders. - Test start end dates are part of Suite Parameters - Added new options for test end date: Use All Data, Use Date, Use Fixed Length - Removed Max Percent Volume per Trade parameter - Forex Pip based slippage is not Spread Slippage. Uses spread amount in forex dictionary. - Added Broker Direct section for direct connection to FXCM and IB. - Stepping is enabled for the System Allocations. Sliders removed. - Stepping is enabled for the Test Start Date. Removed old date stepping process. - Minimum Slippage is now Minimum Futures Slippage and only applied to futures. System Parameter Tab - Right click on tab for all "System Menu" options. Menus - Moved Results from File to Main menu - New System menu - New Blox menu - New Tools menu - Removed View menu, moved Quick Charts to Windows menu - Updated the keyboard short cuts

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	- New toolbar with Wizard Suite, New Suite, Copy Suite, New System, New Block, and up/down resize of Suite Listing area. Debugger - Added Step Out option to step out of the current script and return to the calling script. - Updated the Step Into and Step Over options. - After Starting test run in debug mode, or moving into Step Into mode, using the Run button will resume running the test to the end. - Using the Stop and Edit Blox option (F4) will stop the test and bring up the current script in the blox editor. Walk Forward Tests - Some updates to the walk forward testing, such as fixing commission issues on saved walk forward positions and some updates to the saved stop and limit prices for back adjusted futures. - Added ability to Generate Orders using the “Walk Forward” suite. - Suite named “Walk Forward” will run a WF test even if F5 is used. Preferences Dialog - Updated Data Folder and Options area as well as Reporting General area - Removed ‘raise negative data series’ as that is not supported Testing - Indicators now only compute when parameter has changed. Saves time on large stepped tests. - Indicators that are not stepping use same memory for all test threads. Saves memory when using large number of threads. - IPV Simulation scoped series use single memory location. Saves memory for static series. - Normal log is copied to date stamped results folder for each test run

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	Broker Direct - Support for auto trading with FXCM and IB. Data is updated from the broker, existing positions are synchronized with simulation, orders are synchronized and updated. This can be done with the normal Order Generation process, batch job, or using the new Broker Direct timer controller. Broker connection can be Offline, Data Only, or Data and Orders as desired. This feature is still in testing. - On broker direct startup, a regular simulation run is completed to get the data updated and ready. Then starting on the next bar, order generation process is run each bar to update data and send orders. The orders are generated based on the second to last bar, because the last bar will always be current live and incomplete. - If Offline is selected, then no connection to broker will be made, and data will not be updated, and order will not be sent. - If Data Only is selected, then an order generation or simulation run will only update the data and run as normal. - If Data and Orders are selected then data will be updated, and with the order generation process, whether manual, batch job, or broker direct, positions will be synchronized and orders will be sent to the broker. - New Reporting Broker Direct Suite Parameter controls creation of order and summary result reports. If set to false no reports are generated. Useful when auto running an intraday system with broker direct. - New StockDataRTH .ini file setting default to true, so that stock data loaded from IB is from 9:30 to 4:00, and the Broker Direct Real Time processing will pause outside of those hours. - Support for all IB data time frames, m1, m2, m3, m5, m10, m15, m20, m30, H1, H2, H3, H4, H8, D1.
	Forex Dictionary Editor - Format is now a drop down, to support more data options like FXCM and IB. - Symbol dropdown can be used to set the pair - Timeframe dropdown can be used to set the time frame of the bars. - Tick Unit dropdown sets the number of decimals to use for display - Minimum Tick sets the minimum ticks on which to trade (IB is different than FXCM)

Item:	Trading Blox 4.x to Trading Blox 5.x Migration Changes:
	• Tick Unit times Minimum Tick is the increment used to format prices sent to the broker. • Spread is only used when global parameter is true. • Commission is per round turn. Futures Dictionary • Listing on the bottom to support dialog resizing. • Added Bar Size for intraday time frame or daily. Stock dictionary • New format • Symbol Description, Exchange, Sector, Industry, Country, Data Vendor ID, Broker Symbol, Active Status, Round Lot, Currency, Format, Bar Size, ISIN, Custom Value

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 232

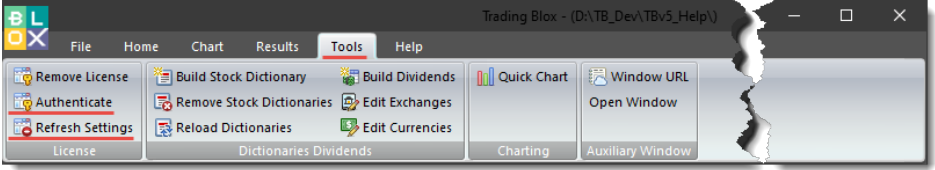
Trouble Shooting

Part



Part 21 – Trouble Shooting

When user's report a problem that has a simple fix that can be added to Trading Blox, the fix will be added here.

Reported Issue:	Solution:
Window is missing	To display the missing window, follow these instructions: Holding-Down Shift-Key while selecting dialog menu item like System or Blox Editor, Trading Blox will reset the dialog window location information so it will appear within the boundaries of the display screen. This approach is more useful when a window is lost than the old way of clicking on the Main-Screen's Tool section and clicking on the Reset Settings.
When a Trading Blox problem appears, or a situation happens when using the Update option that didn't perform, or didn't happen in a way as expected:	Try using the "Tools Menu" tab option to: Perform a new "Authenticate" or "Refresh Settings."  When something doesn't look or feel right with Trading Blox, consider using the Tools Refresh, or the Authenticate options. In most cases, the strange displays will be corrected. If something happened to the Windows Registry, the software authentication Key might need to be applied again. If you don't have it handy, log into the Trading Blox Customer Account area to recapture your registration key: https://www.tradingblox.com/tb/customer-login/

Appendices

Part



Part 22 – Appendices

When the Appendices tab is not opened, this table can be used to jump to each topic in this topic list of topics:

Index	Group Name Index:
1	Application Settings
2	Backtesting
3	Backups
4	Batch Job
5	Block Basic
6	Configuration settings
7	Data Providers
8	Data
9	Determining Big Point Value
10	Error Log and Output File
11	Folders
12	Forex Carry Calculations
13	Forex Dictionary Format
14	Futures Dictionary Format
15	General Reporting
16	Glossary of Terms
17	Interactive Brokers
18	Large Test Reporting
19	Monte Carlo
20	Moving the License
21	Portfolio File Format
22	Risk and Return
23	Single Run Reporting
24	Small Test Reporting
25	System Component Table

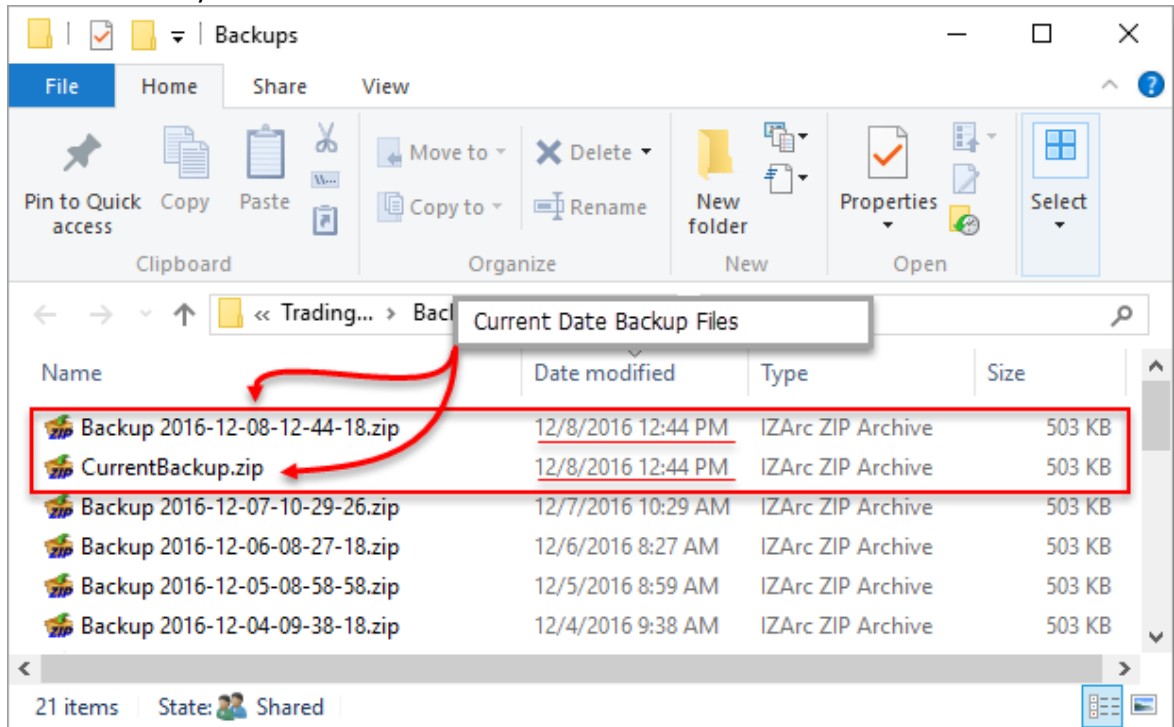
Index	Group Name Index:
26	Technical Support
27	Trade Charts
28	Trade Details
29	TradingBlox Directories
30	Walk Forward

Edit Time: 3/4/2022 12:56:00 PM

Topic ID#: 105

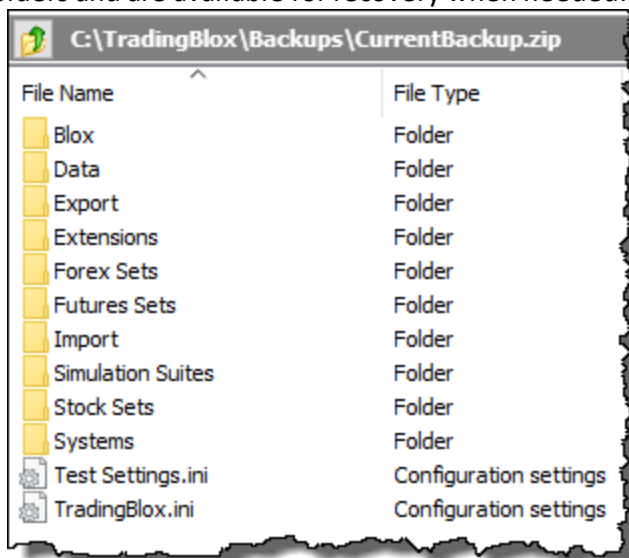
Section 1 – Backups

Backups are created and automatically stored in this location when the application starts for the first time each day.



Trading Blox Trading System Files Backup Information

All the backup files are packed into a Zip file so each day that Trading Blox is run there will be an backup of the files from the last time Trading Blox was used. All the backup files are in these folders and are available for recovery when needed:



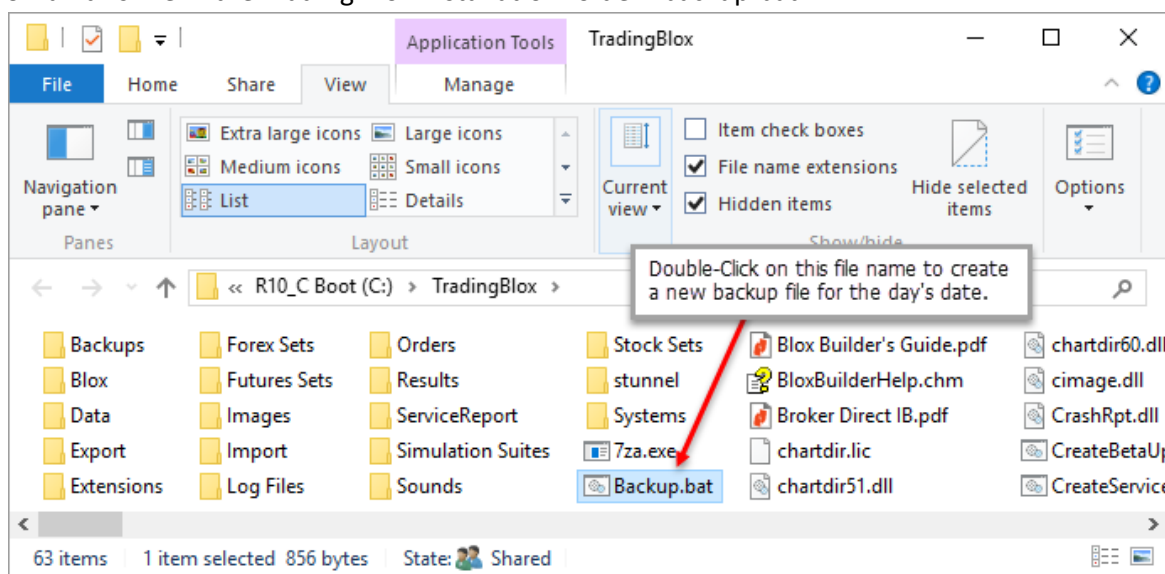
All Trading Blox Files these Folder Names are Backup

All the files in the Zip file are compressed files. Compressed files take less disk space and are easily accessed.

Creating A New Backup:

New backups can be created by either of these two ways any time Trading Blox open:

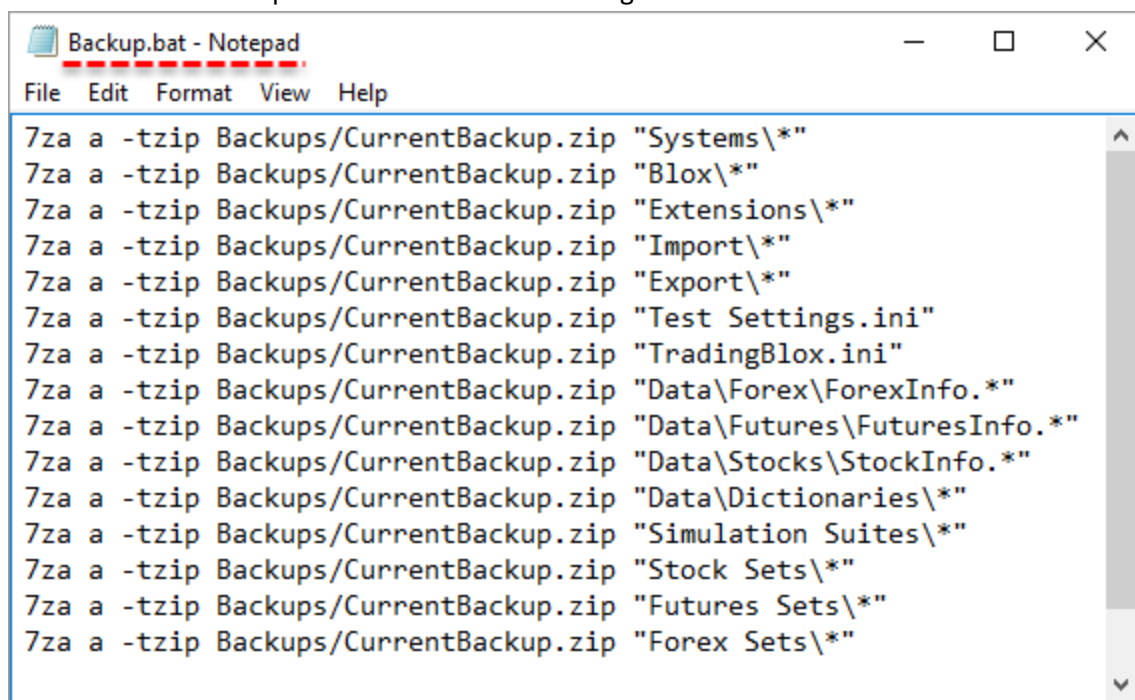
- o Click on the "[Backup Link](#)" on the far left side of the Main Screen [File Menu](#).
- o Run this file in the Trading Blox installation folder: 'backup.bat'



Trading Blox New Backup File Location

When the above backup.bat file is run, it will use its internal list of information to create a backup.

Which files are backed up area listed in this next image:



Trading Blox 'Backup.bat' Folder and File Name Backup scripts.

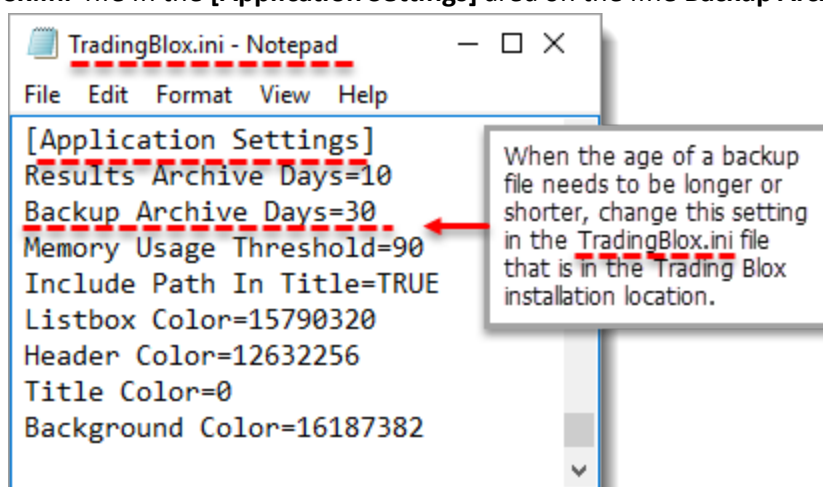
Each of these scripted lines in the image above represent what is backed up in each of the folder locations. These scripts can be easily changed by opening the '**Backup.bat**' file with Windows Notepad.

Adding, Changing, or Deleting while files are Backed Up:

Script information is simple Windows Batch File scripting language that is easy to learn. To learn how to use Window's Batch File language, start with this Microsoft Topic: [Using batch files](#). There are many other sites available with Batch file information and tutorials that will appear after using the search term: "Using Batch Files" in an internet search.

Backed Up File Time Limit:

Backup files older than 30-days are automatically removed. Each backup file is a compressed zip file with a date time stamp. All the folder's and their files are archived in the Backups folder shown near the top of this topic. The number of archive days can be changed in the '**Trading Blox.ini**' file in the **[Application Settings]** area on the line **Backup Archive Days**.



Trading Blox 'Backup.bat' Folder Archive Days Setting.

By default, backups older than 30 days are removed or replaced with newer files.

Section 2 – Batch Job

Trading Blox will support script file automation. Each script file can be created to run Trading Blox for a different reason. This topic provides details for creating and execute a Windows "Batch-File" and Windows "VB-Script" files to accomplish tasks. For example, each task can execute a Suite to generate a new position and order report.

Batch-Script Files, and **VB-Script Files** are fairly simple, user created files that can wake up Trading Blox so it performs the task defined in the script file. Script file commands are keywords that will be recognized by Trading Blox to sequence how the script file operates. The [Script-File-Commands](#) details are defined below.

This is made possible by the use of **Batch-Job** Files that contain script commands that Trading Blox can support the options listed in command table. It also required that the suite name used in the Batch-Job File's script be a name created earlier that will perform what the user intended.

Automation in this context means, any suite in Trading Blox that will produce a test result, or will generate new orders report, can be made to run from double-clicking on a Batch-Job File name. Batch-Job Files can also be added to the Windows Task Scheduler that run on a specific day and time.

A [Batch-Job File](#) is simple text file that contains one or more commands listed to perform a wanted task.

In your Trading Blox installation folder there are three **".bat"** files, and four **".vbs"** files. Both groups are **"Batch Job Files"** that will execute and perform a job. Each type of Batch Job file has its own group of command language that are specific to the type of file you want to create.

The DOS commands are used in **".bat"** files. There is a lot of information about how to use these command on the internet. To locate them use the search term "Dos script commands" in a search engine. A tutorial that covers how to perform task using these commands is available here: "[Batch Script - Commands](#)."

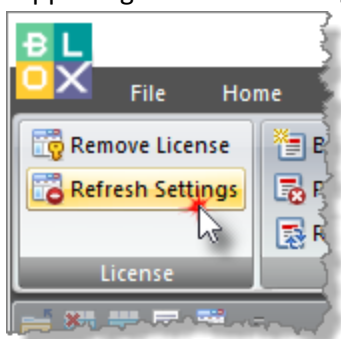
A **".vbs"** file, [VBScript](#) file, is also a text **Batch-Job** File that is modeled on the Microsoft Visual Basic language. [VBScript](#) files have more language commands items and can perform more complex tasks. See the "[VBScript Tutorial](#)."

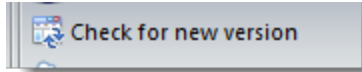
In the Trading Blox Roundtable Customer Support Forum, there is a discussion named: "[Batch-Job](#)" that instructs and reports current task command updates.

The language items in a **Batch File** create a script that can perform a Trading Blox actions. To enable Windows to recognize the saved text file is a **"Batch Executing File"**, the name of the file must have **".bat"** suffix. For example, **"AnyBatchScript.bat"** will run the language items as a **Windows Batch File** script. When naming a new batch-job file, consider using a name that helps to reflect the task process the Batch-Job file will perform. For example, **"UpdateOrders.bat"**, will create a new position and orders report.

These Batch Job Files are installed when Trading Blox is installed:

Windows Batch File Details:	Description:
Backup.bat	Windows Batch Files are a series of commands that execute each command in the order in which its command are listed. This batch file name "Backup.bat" will create a backup of all the files that are contained within Trading Blox installation folder. A this batch job name can be executed with by double-clicking with your mouse, the Batch-Job file name. "CurrentBackup.zip." When this batch file is run, it will look to see if there is a "CurrentBackup.zip" already in the Backups folder. If that file is found, the previous "" will be renamed to a folder named: "Backup Backup 2019-04-26-09-13-54.zip." Date shown in the renamed backup zip file name is the date and time that the new "CurrentBackup.zip" was created.
CreateBetaUpdate.bat	This batch file updates a beta version of Trading Blox.
CreateServiceReport.bat	This batch file create a Service Report File of Trading Blox. Service Reports are used for resolving problems that other methods tried have been unsuccessful.

VBScript Script File:	Description:
RestartTB.vbs	This batch file restarts. Trading Blox after the updated files have been installed.
BetaUpdater.vbs	Only use this Batch-Job file, if you need to update the Beta Version you have running in Trading Blox.
TBResetRegistrySettings.vbs	This batch file resets all the registry settings to their default values when executed. Use this when you find there is something unusual happening with how Trading Blox is performing.  Main Screen License Group - Refresh Settings

<u>VBScript Script File:</u>	Description:
	When this option is used, your license information is required so the software can update its license information.
Updater.vbs	 Use this Help - Version Group option to download the latest release version. When the updated files are downloaded, the UpDater.vbs batch job will executed and unpack and replace the updated Trading Blox files. Check For New Version

To create a Batch-Job File that will run a test, or generate a new orders report, the command Flag options that are supported in Trading Blox are the Flags that need to be in the script that generates a test, or a new orders report.

In the following table are the current supported command Flag options support by Trading Blox. Some of the Flags are used in the examples are below this table.

To generate an automated test or new orders report, the process can be started by double-clicking on the Batch-Job File name. In our examples, the place we use to save a Batch-Job File is the folder location where Trading Blox is installed. User's with experience with Batch-Job automation learn that with more Batch-Job file language items added to a Batch-Job file, they can store their file anywhere because the added language items will know the Trading Blox file location.

You can use windows scheduler for this as well, if you create a .bat file.

Trading Blox Supported Flag Options:	Description:
-bd	This command specifies Broker Direct . ie. <code>tradingblox.exe -suite "IB Suite" -bd</code>
-delay	Test delays are in seconds.
-exit	This flag tells Trading Blox to exit when the test or orders are finished running. Runs silent mode: no breakpoints and no data loading error messages
-hidden	When this flag added Trading Blox will run in a hidden window.
-min	This flag is used, Trading Blox will start in a minimized window.
-noProgress	Hides the progress dialog during a test or an order report execution.
-normal	Use this option after a -hidden was used so the normal display more will be visible.

Trading Blox Supported Flag Options:	Description:
-orders	Generates a new order report (F7) for each suite-name in the list of suites named in the batch job file.
-suite	Suites to use with batch job can be any suite name that is shown in the Suite list of Trading Blox (i.e. "suite1", "suite name 2"). Suite names must be have quotation marks around the suite name. When more than one suite is to run, each suite-name must be separated by a Coma [,]
-test	Runs a standard (F5) test for each suite-name in the list of suites
-wf	This flag will execute a Trading Blox Walk Forward Test in Trading Blox in versions 5.3 and that were released on or later than January 29, 2019. The Walk Forward menu item only does one thing. It runs a Walk Forward Test. In a script, the Trading Blox Batch-Job flag -wf is used to use Global Walk Forward menu items. Walk-Forward Testing requires the Walk-Forward System be elected in the suite. This enables the Reporting Calculations so the testing statistics can appear in the test report. The Walk-Forward System is not a standard blox and it doesn't appear in the Global Suite Parameters list of options because the blox in the system doesn't contain any parameters even though it is a Global Suite System blox. Walk-Forward Testing doesn't normally generate orders because the default test disables orders. Generating Orders in this type of test is a special condition that needs the user to change the default test so it can generate orders. This the system selected for the Walk-Forward Testing needs the user to enter into the Before Test script of the selected system the order generation function <code>test.SetGeneratingOrders(TRUE)</code> before running the Walk-Forward Test.

Single Batch Job File Example:

To run a test, or to generate orders, each of the following Windows Batch Job files show what to include to automate a test, or to generate a new orders report.

NOTE:

In the examples offered here, the Windows Batch Job files are placed in the folder where Trading Blox is installed. As you gain experience in working with an automated execution

Single Batch Job File Example:

process, the automation process can add additional language elements needed to automate changing the Batch Job file's location. For this example, and so the process will be as simple as possible, the file will be saved in the Trading Blox installation folder.

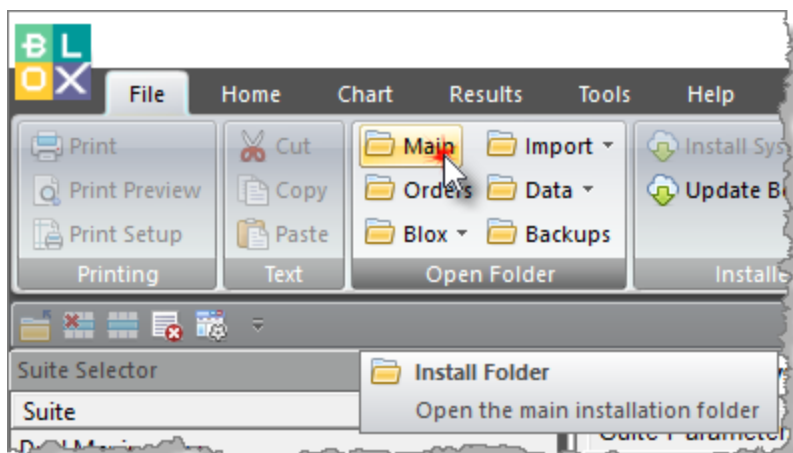
Basic Batch Job Tutorials:

Our first example will create a Windows Job Batch to run a Trading Blox Suite named **"TestSuiteTest"** when the file name is double clicked. To keep this example as simple as possible, the file will be placed in the same folder location where Trading Blox is installed.

- To create this file, open the Windows **Notepad.exe** program.
- Then click the File -> Save As option and use the file-browser dialog to find the folder when Trading Blox is installed.
- When installed folder appears, type the file named: **TestSuiteTest.bat** into the dialog's file-name area.
- Once the file is saved, enter the following text into the text area, save the file and close Notepad (Suite Test Example).

```
tradingblox.exe -suite "TestSuiteTest" -test -exit
```

- To run your new **"TestSuiteTest.bat"** file, click on the main menu's File Tab and then Click on the Main button in the Open Folder group.
- the new file you created will appear near the bottom of the folder's list of file when the folder is set to display items in ascending alphabetical order.



Trading Blox Installation Main-Folder

When the automation is intended to generate new orders, use this version of the Batch-Job file:

```
tradingblox.exe -suite "TestSuiteTest" -orders -exit
```

Multiple Batch Job File Example:

To run multiple .bat files in a row, or at the same time, use a Windows Script Host file. Create a file in notepad with a ".vbs" extension. The following sets the current directory, and runs several bat files one after another.

The first parameter of the WSHShell.Run function is the first Batch-Job file name that will execute. The second, Batch-Job name will execute when the first file ends its execution. The last Batch-Job file name in this example will execute after the second file terminates its execution.

Multiple Batch-Job executions can run a sequence of programmed tasks, and each will wait for the prior Batch-Job to stop running so the next file is able to perform its tasks.

```
Dim WSHShell
Set WSHShell = CreateObject("WScript.Shell")

WSHShell.CurrentDirectory = "C:\Testing\TradingBlox\"

WSHShell.Run "BatchRun1.bat", 0, true
WSHShell.Run "BatchRun2.bat", 0, true
WSHShell.Run "BatchRun3.bat", 0, true
```

This above example runs its tasks hidden, and the ".bat" file also uses the "-hidden" and the "-noProgress" flags in their scripts. This approach enables them to run behind the scenes while the computer is used for other tasks.

If a ".bat" file references a different Trading Blox installation folder, then the wait on return parameter could be false, which would launch all these processes at the same time. A multiple simultaneous license of Trading Blox would be required to run multiple instances at the same time.

Multiple Batch Job File Example:

This is a PowerShell script that updates CSI data daily downloaded data. It also will run a suite to build dividends, and generates orders:

```
Start-Process -wait "C:\UA\EzDownloader.exe" -WorkingDirectory "C:\UA\"

$pathToExe = "C:\DataUpdater\TBDataUpdater\TradingBlox.exe"
$arguments = "-suite BuildDividends -test -exit"

Start-Process -wait -FilePath $pathToExe -ArgumentList $arguments -
WorkingDirectory "C:\DataUpdater\TBDataUpdater\"

$arguments = "-suite FancySystem -orders -exit"
Start-Process -wait -FilePath $pathToExe -ArgumentList $arguments -
WorkingDirectory "C:\DataUpdater\TBDataUpdater\"
```

This script example can easily be modified to update the TB dividends, and Generate orders.

Multiple Batch Job File Example:**Website Script Information Resources:****[PowerShell Commands](#)**

[Batch Script - Commands](#) (Tutorial)

<https://tinyurl.com/yyvett9x>

[https://msdn.microsoft.com/en-us/library/d5fk67ky\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/d5fk67ky(v=vs.85).aspx)

<https://tinyurl.com/yy5q3jtu>

[Windows VBScript commands](#) (Tutorial)

<https://ss64.com/vb/>

Links:

[Windows VBScript commands, Reference on Windows Script Host, Windows VBScript commands, WshScriptExec Object](#)

See Also:

Section 3 – Configuration settings

Trading Blox assigns recommended default settings to the program's properties that are used during a test.

The default values are stored iwhere the **Trading Blox.exe** program is located.

All the default settings that can be changed are the "**TradingBlox.ini** file that any text editor can open the file and display the settings.

There are 14 groups where each group assigns a default setting. The "**TradingBlox.ini** file begins on row 1 and ends on row 257. To read what the default setting is for an element in a group, click on the name of the group in the next table to quickly display that group's setting:

TradingBlox Default Settings Index	
TradingBlox Directories	Data
Walk Forward	General Reporting
Monte Carlo	Single Run Reporting
Small Test Reporting	Large Test Reporting
Trade Charts	Trade Details
Block Basic	Application Settings
Backtesting	Interactive Brokers

Item:	Group Name:	Description:
1	TradingBlox Directories ^Top	Futures Data Directory=D: \Data\Futures\ Stock Data Directory=D: \Data\Stocks\ Forex Data Directory=D:\Data\Forex\ Dividend Data Directory=D: \Data\Dividends\ Dictionary Directory=D: \Data\Dictionaryes\ Orders Directory=Orders\ Results Directory=Results\ CSI UA Directory=D:\Data\UA\
2	Data ^Top	Integrity Check= FALSE Num Extra Data Fields=0 Years Data=5 Load Volume= TRUE Raise Negative Data Series= FALSE Process Weekly Bars= TRUE Process Monthly Bars= FALSE

Item:	Group Name:	Description:
		Process Daily Bars= FALSE Process Weekends= TRUE Load Unadjusted Close= TRUE Reverse Currency Conversion= FALSE Base Currency ISO Code=USD Base Currency Sign =\$ Stock Volume Multiplier=100 Global Heap Size=0 Post Data End Date Warnings= TRUE Forex Friday End Time=2100 Forex Sunday Start Time=2200 Time Zone Adjust=0 IB Futures RTH= FALSE IB Stocks RTH= TRUE Dividend File Suffix=_Div.csv Use IB Forex Bid Ask= FALSE Find Stock Dictionary= TRUE
3	Walk Forward ^Top	Optimization Size=5 Out Of Sample Size=1 Save Positions= TRUE
4	General Reporting ^Top	Small Test Threshold=8 Goodness Index=3 Goodness Min =0 Goodness Max =0 Graph Closed Equity= FALSE Play Report Sound= FALSE Show Order Results= TRUE Add Non Traded Instruments= FALSE Summary Report Test Maximum=5000 Multi Parameter Chart 1= TRUE Multi Parameter Chart 2= TRUE Risk Free Rate=0 Use Population SD= TRUE Display Last Roll Date= TRUE Include Position Stop= FALSE Graph File Suffix=.png Chart Width Multiplier=1 Chart Height Multiplier=1
5	Monte Carlo ^Top	Iterations=2000 Equity Graph Count=20 Sample with Replacement= TRUE Grouping Days=100 Confidence Level=90 Use Fixed Random Seed= TRUE Random Seed=2006
6	Single Run Reporting ^Top	Log Trades= TRUE Log Filtered Trades= TRUE Log Daily Equity= TRUE

Item:	Group Name:	Description:
		Log Monthly Equity=TRUE Graph Log Equity=TRUE Graph Linear Equity=TRUE Custom Graph=TRUE Graph Monthly Returns=TRUE Graph Monthly Distributions=TRUE Graph Total Risk Profile=TRUE Graph System R-Multiple Distribution=TRUE Graph System R-Multiple Contribution=TRUE Graph Multiple Parameters=FALSE Graph Parameter Steps=FALSE Report Yearly Performance=TRUE Report Trading Performance Statistics=TRUE Report Instrument Performance=TRUE Report System Parameter Settings=TRUE Graph Monte Carlo Equity Curve=FALSE Graph Monte Carlo Return=FALSE Graph Monte Carlo Sharpe=FALSE Graph Monte Carlo MAR=FALSE Graph Monte Carlo R Squared=FALSE Graph Monte Carlo Drawdowns=FALSE Graph Monte Carlo Drawdown Length=FALSE Report Monte Carlo Confidence Levels=FALSE Filter Results=FALSE Filter CAGR Threshold=0 Filter MAR Threshold=0 Filter Sharpe Threshold=0
7	Small Test Reporting ^Top	Log Trades=TRUE Log Filtered Trades=FALSE Log Daily Equity=FALSE Log Monthly Equity=FALSE Graph Log Equity=TRUE Graph Linear Equity=TRUE Custom Graph=TRUE Graph Monthly Returns=TRUE Graph Monthly Distributions=TRUE Graph Total Risk Profile=FALSE Graph System R-Multiple Distribution=FALSE Graph System R-Multiple Contribution=FALSE Graph Multiple Parameters=TRUE Graph Parameter Steps=TRUE

Item:	Group Name:	Description:
		Report Yearly Performance= TRUE Report Trading Performance Statistics= TRUE Report Instrument Performance= FALSE Report System Parameter Settings= FALSE Graph Monte Carlo Equity Curve= FALSE Graph Monte Carlo Return= FALSE Graph Monte Carlo Sharpe= FALSE Graph Monte Carlo MAR= FALSE Graph Monte Carlo R Squared= FALSE Graph Monte Carlo Drawdowns= FALSE Graph Monte Carlo Drawdown Length= FALSE Report Monte Carlo Confidence Levels= FALSE Filter Results= FALSE Filter CAGR Threshold=0 Filter MAR Threshold=0 Filter Sharpe Threshold=0
8	Large Test Reporting ^Top	Log Trades= FALSE Log Filtered Trades= FALSE Log Daily Equity= FALSE Log Monthly Equity= FALSE Graph Log Equity= TRUE Graph Linear Equity= FALSE Custom Graph= FALSE Graph Monthly Returns= FALSE Graph Monthly Distributions= FALSE Graph Total Risk Profile= FALSE Graph System R-Multiple Distribution= FALSE Graph System R-Multiple Contribution= FALSE Graph Multiple Parameters= TRUE Graph Parameter Steps= TRUE Report Yearly Performance= FALSE Report Trading Performance Statistics= FALSE Report Instrument Performance= FALSE Report System Parameter Settings= FALSE Graph Monte Carlo Equity Curve= FALSE Graph Monte Carlo Return= FALSE Graph Monte Carlo Sharpe= FALSE Graph Monte Carlo MAR= FALSE Graph Monte Carlo R Squared= FALSE Graph Monte Carlo Drawdowns= FALSE

Item:	Group Name:	Description:
		Graph Monte Carlo Drawdown Length= FALSE Report Monte Carlo Confidence Levels= FALSE Filter Results= FALSE Filter CAGR Threshold=0 Filter MAR Threshold=0 Filter Sharpe Threshold=0
9	Trade Charts ^Top	Grid Color=16443110 Long Color=37632 Short Color=213 Entry Color=16764057 Stop Color=14527197 Exit Color=16751001 Up Bar Color=37632 Down Bar Color=213 Up Candle Color=37632 Down Candle Color=213 Chart Background Color=14745599 Chart CrossHair Color=12632256 Custom Color 1=0 Custom Color 2=139 Custom Color 3=16711680 Custom Color 4=16777215
10	Trade Details ^Top	Entry Equity= TRUE Closed Equity= TRUE Core Equity= FALSE Days= TRUE Drawdown Percent = TRUE Entry Date= TRUE Entry Fill Price = TRUE Entry Order Price = TRUE Entry Stop Price = TRUE Entry Risk= TRUE Exit Date= TRUE Exit Fill Price = TRUE Exit Order Price = TRUE Long or Short = TRUE Profit Dollars= TRUE Profit Percent = TRUE Profit Points= TRUE Profit Position Dollars= TRUE Profit R-Multiple= TRUE Quantity= TRUE Total Equity= TRUE Unit Number= TRUE
11	Block Basic ^Top	Keyword Color=16711680 Function Color=0 Operator Color=0

Item:	Group Name:	Description:
		Comment Color=37632 String Color=65535 Trading Object Color=4718664 Built-In Color=2124031 Temporary Color=0 Block Permanent Color=9109504 Instrument Permanent Color=9109504 Indicator Color=9109504 Parameter Color=13781 Include Default Scripts= FALSE Automatic Blox Script Parsing= TRUE Allow Priming Update Indicators= TRUE Blox Editor Font Size=14 Blox Editor Font=Courier New Use Jelly Fish Editor= FALSE Blox Directory= Parameter Tab Size=6
12	Application Settings ^Top	Results Archive Days=90 Backup Archive Days=30 Memory Usage Threshold=90 Include Path In Title= TRUE Listbox Color=15790320 Header Color=12632256 Title Color=0 Background Color=16187382 Broker Direct Trigger File=DataTrigger.txt Run Silent= FALSE System Edit Area Tabs= FALSE
13	Backtesting ^Top	Stock Margin Short Rate=100 Stock Margin Long Rate=100 Use Round Lot= TRUE Max Stepped Tests=0
14	Interactive Brokers ^Top	Enable IB Emails= FALSE Default IB Port=7496 Default IB Client=1 Synch IB Positions= TRUE Weekly Restart Time=-1 Bypass Warning= FALSE Position Synch Percent =0

Section 4 – Data Providers

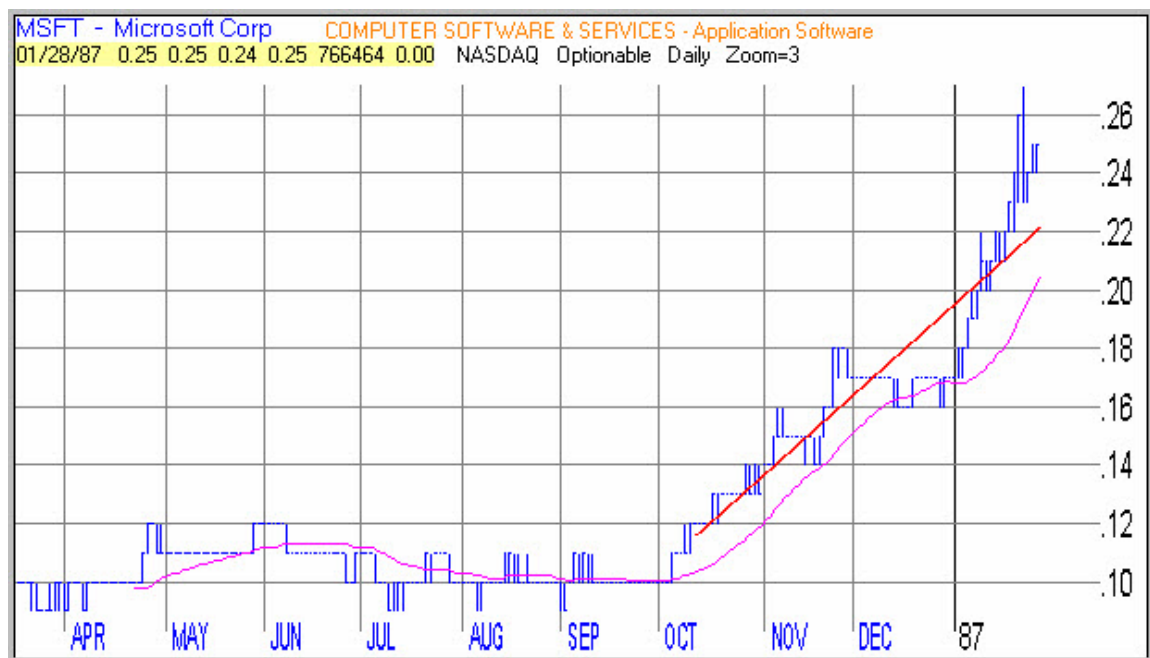
Below is a partial list of a few Data Providers, and their contact information.

[CSI Commodity Systems Inc. - Market Data & Software](#)

Commodity Systems, Inc., known in the industry as CSI, offers data in MetaStock and ASCII Text formats for many different U.S. and World futures and stock markets. CSI is the preferred provider of futures data and is the supplier of the sample data we provide.

Trading Blox includes specific Futures Dictionary files for CSI data for their default back-adjusted futures contracts in ASCII text formats. The default rolling algorithm is to use open interest and roll on the first trigger. This back-adjustment algorithm will result in symbols that have "0_I0B" appended to them. For example, the full CSI symbol for corn "C" using the default back-adjustment algorithm would be "C_0_I0B". The ASCII text file would be named "C_0_I0B.txt" and this symbol would be the symbol used inside the MetaStock MASTER file for that data.

Visit <https://www.tradingblox.com/tradingblox/howto-ua.htm> for a tutorial explaining setting up Trading Blox with data from CSI.



Note how the prices for each day appear to have at most one or two prices. This is because the adjustments for splits have been constrained to \$0.01 which is not enough resolution to show the actual pricing relationships between the different days.

Section 5 – Determining Big Point Value

Simply put, the Big Point Value is the amount of money you will make if you buy one contract and it goes up by one full point. That is one number to the left of the decimal. For example, Gold's price goes up 1 full point. i.e. From 401 to 402, or Soybeans from 4 to 5.

Be your data in the Futures' Dictionary represents the current Big Point Value that is used to represent your Futures data prices.

Different data vendors store the data for the same futures markets in different units.

For example, Data Provider A might store the same line of data for the British Pound in dollars like:

20020703, 1.4886, 1.4944, 1.4882, 1.4898, 4356, 41414

While Data Provider B might store a line of price data for the British Pound in cents like:

20020703, 148.86, 149.44, 148.82, 148.98, 4356, 41414

Big Point Value Mapping reconciles this unitary difference (148.86 vs. 1.4886) for these two vendors. The Big Point Value is multiplied by the prices stored in the price data file to determine the dollar value of a one point move in the underlying.

Using the above example, the British Pound entry in the Futures Dictionary for Data Vendor A would be:

62500

While the entry for British Pound for Data Vendor B would be:

625

These values, and all Big Point Values, derive from the contract size. According to the Chicago Mercantile Exchange, the BP contract size is "62,500 pounds sterling (British pounds)". Therefore if a data vendor lists the price for the BP contract in full dollars, i.e. like Vendor A above 1.4886, since there are 62,500 pounds in a contract. The Big Point Value is 62,500. If the vendor lists the price in cents like Vendor B above at 148.86, the Big Point Value is 625.

Using Vendor A's Pricing the price of a British Pound goes from 1.4886 to 2.4886, this will result in a gain of \$62,500 since there are 62,500 British Pounds in a contract. Using Vendor B's pricing, this corresponds to a price movement of 100, from 148.86 to 248.86 which will also result in a gain of \$62,500. Thus the Big Point Value for Vendor A is 100 times the size of the Big Point Value for Vendor B because the price units are 1/100th as large.

For a given price, the contract value should be the same. The contract value can be determined by multiplying the price by the Big Point Value for the market. Since the Big Point Value adjusts for unitary differences the price for a data vendor multiplied by the Big Point Value for that vendor should be the same as that of any other data vendor. Thus:

$$148.86 \times 625 = 1.4886 \times 62500$$

The Big Point Value should be *expressed directly in the currency in which the futures contract is priced*. For example, the SOFFEX Swiss Government bond contract is priced in Swiss Francs (ISO currency code CHF) so its Big Point Value should be expressed directly in terms of CHF.

Edit Time: 2/19/2022 11:26:11 AM

Topic ID#: 126

Section 6 – Error Log and Output File

The Error Log (also stored in the Results folder), logs any errors that occur during order generation.

In most cases, it will generally be empty. If a serious error occurs, an error dialog message will pop up. So it should not be necessary to regularly inspect this file, except for troubleshooting.

The Output file will be normally be empty. If you have a support issue, you may be asked by Technical Support to open it.

You can safely ignore both of these files, unless you suspect something is correct after orders are created, or a system test doesn't look correct.

Edit Time: 2/19/2022 11:28:01 AM

Topic ID#: 135

Section 7 – Folders

The main Trading Blox folder "TradingBlox" contains the following sub folders:

Folder	Contains
Backups	Contains the backup zip files that are created once a day or on demand from the File/Backup menu item.
Blox	Contains the blox for the systems displayed.
Data	Contains the Futures, Forex, stock data and Dictionaries folder.
Export	Systems and suites are exported here.
Extensions	Added abilities in the form of ".dll" files are stored here.
Forex Rates & Forex Set Files	Location of all ISO Rates.txt files for the currency lend and borrow rates. The Forex Portfolios (or Forex Set files).
Futures Sets	The Futures Portfolios (or Futures Set files).
Images	Images used for HTML-based reporting.
Import	This folder will contain any of the ".tbz" and suite zip files imported.
Log Files	Normal.log files are here.
Orders	Order reports both HTML and csv. Old reports are deleted based on preference setting.
Results	The test results and reports. Contains information from every test run. Old results are deleted based on preference setting.
Service Reports	Created service reports are here.
Simulation Suites	The Simulation Suite files.
Sounds	The sound files used by Trading Blox.
Stock Sets	The Stock Portfolios (or Stock Set files).
stunnel Files	Dll capability files.
Systems	Contains the system files.

Edit Time: 3/5/2022 10:55:37 AM

Topic ID#: 141

Section 8 – Forex Carry Calculations

NOTE: This section assumes a familiarity with Forex trading. It is not intended as a tutorial. See https://fxtrade.oanda.com/currency_trading/intro_currency_exchange.shtml for an excellent tutorial of Forex concepts.

Here is an interest rate carry cost calculator:

<https://www.oanda.com/products/fxmath/interest.shtml>

Trading Blox accounts for the costs of [Forex Carry](#) by performing the following tasks each day after the close for each Forex position held:

1. Determine the quantity of both Base and Quote currency based on the price at entry. For long trades one buys the base currency and sells the quote currency, so the carry cost is determined by lending the base currency and borrowing the quote, and for shorts it lends the quote and borrows the base.
2. Determine the applicable historical lend and borrow interest rates for each currency for the day. The file contains the lend, then the borrow, rates. The calculations then assumes you borrow (pay) interest at the borrow interest rate from the history file, and that you lend (receive) interest at the lend interest rate from the file. The rate used for each day is from the current day.
3. Calculate the current days interest based on the quantity of each currency, the applicable rate for the currency, and the position direction (long or short). The base currency bought/sold is the Forex Trade Size from the global parameters. The quote currency bought/sold is this trade size times the current closing price of the forex pair.
4. Convert each of the resulting interest payments back into dollars at the then current exchange rates for the Base and Quote currencies. Rates used are current day rates.
5. Subtract what you need to pay from what you receive to get the net interest -- lending income minus borrowing payments.
6. Divide the total calculated yearly interest by 365.25 to determine the daily rate.
7. Interest is accrued/paid starting on the entry date, and includes weekends and holidays. Interest is not accrued/paid on the exit date.

Required Data Files for Forex Testing

To test a given forex instrument, you need:

- Price history for the instrument.
- If the forex pair is a cross, a USD to Base or USD to Quote price history so the entry and exit prices can be converted to dollars.
- If you turn on carry computation you also need an interest rate history file and USD conversion file for both the Base and Quote currencies. These can be sparsely populated to include only days when the interest rates change. For example, a USD history could be done containing only those days when the Fed changes the rates.

Here is a good source of interest rate data:

<https://fx1.oanda.com/user/interestrate.html>

More reading on the carry cost subject available here:

https://fxtrade.oanda.com/fxtrade/interest_calculation.shtml#

Edit Time: 2/19/2022 11:50:44 AM

Topic ID#: 142

Section 9 – Forex Dictionary Format

The Forex Dictionary is stored on disk as an ASCII Text file called "ForexInfo.txt". This file contains information about the name and location of the data file, the format, the pip size, and the spread. It is stored in the Forex Data Folder (see below).

DON'T EDIT THIS FILE BY HAND - PLEASE USE THE FOREX DICTIONARY EDITOR

Forex Data Folder

In Trading Blox, the folders containing forex price data files are set using the [Data and Folder Preferences](#) section of the [Preferences](#) editor.

The default folder setting for forex data is:

C:\Trading Blox\TradingBlox\Data\Forex\

ForexInfo.txt

ForexInfo.txt is a very important file stored in the Forex Data Folder, and it contains all the pertinent information about a particular futures market. An example line in this file might be:

EURUSD,"EEC Euro / U.S. Dollar",ASCII,EURUSD.TXT,0.0001,2.00

Moving left to right, Trading Blox knows that:

1. the trading symbol for this market is EURUSD
2. the forex market name is EEC Euro / U.S. Dollar
3. the file is in the ASCII format
4. the folder\filename is "EURUSD.TXT" which means that it's full pathname is:
C:\Trading Blox\TradingBlox\Data\Forex\EURUSD.TXT
5. the pip size is 0.0001
6. the spread is 2.0 pips

Position 3 and 4 will likely be different from data vendor to data vendor because each generally has a different naming convention for forex data.

The symbol defined in the first column must be identical to the symbol used in Portfolio Files. Portfolio Files use the *symbol* to lookup markets in the Forex Dictionary.

Section 10 – Futures Dictionary Format

The Futures Dictionary is stored on disk as an ASCII Text file called "FuturesInfo.txt". This file contains information about the exact name and location of the data file, the description, contract size, as well as number of digits to display for graphs and trade listings. It must be placed in the top level of the futures data folder you are using. Please [contact technical support](#) to see if your vendor's data is supported, or for assistance with your data vendor.

DON'T EDIT THIS FILE BY HAND - PLEASE USE THE FUTURES DICTIONARY EDITOR

Futures Data Folder

In Trading Blox, the folders containing futures price data files are set using the [Data and Folder Preferences](#) section of the [Preferences](#) editor.

The default folder setting for futures data is:

C:\Trading Blox\TradingBlox\Data\Futures\

FuturesInfo.txt

FuturesInfo.txt is a very important file stored in the Futures Data Folder, and it contains all the pertinent information about a particular futures market. An example line in this file might be:

```
GC,Gold,TechTools,GC__53A.TTD,COMEX,GJMQVZ,"100 troy  
ounces",USD,100,2025,HG:,SI:,1,0.1,1,820
```

Moving left to right, Trading Blox knows that:

1. The trading symbol for this market is GC
2. The futures market name is Gold
3. The file is in the TechTools format
4. The folder\filename is "GC\GC_53A.TTD" which means that it's full pathname is: C:\Trading Blox\TradingBlox\Data\Futures\GC\GC_53A.TTD
5. The exchange is COMEX
6. The months traded are GJMQVZ
7. The trading unit is 100 troy ounces
8. The price is quoted in USD
9. The big point value is 100
10. The current exchange mandated initial margin requirement is \$2025
11. Gold is closely correlated with SI (silver)
12. Gold is loosely correlated with HG (high grade copper)
13. Gold should display with one digit of accuracy
14. The minimum tick unit is in tenths or 0.1
15. The minimum tick is one tenth or 1 * 0.1
16. The sort for order generation is 820.

Position 3 and 4 will likely be different from data vendor to data vendor because each generally has a different naming convention for futures data. This is especially for back-adjusted continuous contracts. CSI even names their files differently depending on the exact back-adjusting algorithm used to join the various individual contracts.

The colons at the end of the mapping separate the markets within a loosely correlated or closely correlated group. The correlated groups themselves are delimited by commas. For instance, Soybean Meal is loosely correlated to Corn, Oats, and Wheat, and is closely correlated to Bean Oil and Soybeans:

```
SM,"Soybean Meal",TechTools,SM___53A.TTD,CBOT,FHKNQUVZ,"100  
tons",USD,100,675,C:O:W:,BO:S:,2,0.1,1,1030
```

The symbol defined in the first column must be identical to the symbol used in Portfolio Files. Portfolio Files use the *symbol* to lookup markets in the Futures Dictionary.

Edit Time: 2/19/2022 12:04:32 PM

Topic ID#: 145

Section 11 – Glossary of Terms

This is not a complete glossary because *some knowledge of trading concepts is assumed*. This Glossary contains the most important definitions used in Trading Blox.

Average True Range (ATR)

The Average True Range is a measure of volatility which indicates the average range of price movement in a single day. This measure was developed by J. Welles Wilder and introduced in his book, *New Concepts in Technical Trading Systems* (1978).

The average of the *True Range* where True Range is the range including opening gaps, or the greatest of:

- The current high minus the current low
- The current high minus the previous close
- The previous close minus the current low

The last two entries account for price gaps where the entire day's prices do not overlap the previous day's close.

Trading Blox allows you to set the number of days to use when creating an Average True Range Indicator. In the Trading Blox supplied systems and optional blox can use either an [Exponential Moving Average](#) method or a Simple Average method to calculate this indicator. In the Builder version of Trading Blox other averaging methods can be scripted and used as well. For our purposes here it is important to note that the [Exponential Moving Average](#) is a different formula than the [Wilder Moving Average Formula](#) used by Wells Wilder in his book "New Concepts in Technical Trading Systems." As an example, the 20 day Wilder ATR is the equivalent to a 39 day Exponential ATR.

$$\text{Exponential MA Days} = (\text{Wilder MA Days} \times 2) - 1$$

Autocorrelation

A term referring to the correlation of a time series with a lagged version of itself. Sometimes referred to as serial correlation or lagged correlation.

Breakout

A chart pattern used to indicate a rise in a stock's price above its resistance level or drop below its support level. It is typically measured in terms of days. An N-Day breakout is a breakout of the highest high of the last N Days or the lowest low of the last N Days, where N is the number of days for the breakout. Thus a 20-Day breakout of the highs would occur when the price exceeded the highest high of the last 20 days.

Breakouts are used in trend-following systems to indicate a potential start of a trend.

Bollinger Band

A Bollinger Band is a pricing channel which consists of a top and bottom price which are defined in terms of a [Simple Moving Average](#) and the price a fixed number of [Standard Deviations](#) above that moving average, and below that moving average.

Closed Equity

Closed Equity is a measure of equity that take the starting balance and adjusts for the profits of *closed out trades only*. It does not take into account the value of open positions.

Confidence Level

A Confidence Level is used during Monte Carlo Simulations to indicate that a certain percentage of the simulated equity curves demonstrated results for a particular measure that were greater than the confidence level. For example, a 95% Confidence Level MAR Ratio of 0.8 indicates that 95% of the simulation iterations showed a MAR Ratio of 0.8 or better.

Core Equity

Core Equity is a measure of equity that includes [Closed Equity](#) plus that portion of [Open Equity](#) that would be realized if all the current positions were exited at their current stop values (i.e. [Locked-In Profits](#)).

Core Equity = Closed Equity + Locked-In Profits

Expectation:

(TotalWinPercent - TotalLossPercent) / TotalRiskPercent

Total Win Percent = sum of (gain as percent of trade entry equity) of all winning trades, including washes

Total Loss Percent = sum of (loss as percent of trade entry equity) of all losing trades

Total Risk Percent = sum of (percent of trade entry equity risked) of all trades

Exponential Moving Average

Similar to a simple moving average (SMA), except that more weight is given to the latest data. Also known as an exponentially weighted moving average, this type of moving average reacts more quickly to recent price changes than a [Simple Moving Average](#). Our Exponential Moving Averages start with a simple moving average of the number of days in the moving average, and then use the Exponential formula thereafter.

An *Exponential Moving Average* is computed according to the following formula:

$$\text{Today's MA} = \text{Yesterday's MA} + \frac{2}{N+1} (\text{Current Price} - \text{Yesterday's MA})$$

Exponential MA Days = (Wilder MA Days X 2) - 1

Forex Carry

An adjustment for the costs of borrowing and lending the two currencies involved in a Forex transaction.

Locked-In Profits

That portion of [Open Equity](#) that would be realized if all the current positions were exited at their current stop values.

MAR Ratio This statistic, developed by the Managed Accounts Report newsletter, is a simple ratio that divides the **CAGR** (Compound annual growth rate) by the test periods biggest draw

down percentage. For example, when a test shows a **CAGR** = 19.08% and it has its Largest Drawdown = 29.1% then results are: $(19.08 / 29.1) = 0.655670$

Moving Average

An average measure of prices that is computed using a fixed number of days. For example, a 20 day moving average is computed using the last 20 days. There are two common types of moving averages: [Simple Moving Average](#), and [Exponential Moving Average](#).

N

A measure of volatility, N is a concept used by the Turtles (student's of Richard Dennis and Bill Eckhardt) to represent the underlying volatility of a particular market.

N is simply a 20-day [Average True Range](#). Conceptually, N represents the average range in price movement that a particular market makes in a single day, accounting for opening gaps. N was measured in the same points as the underlying contract.

See the [Original Turtle Rules Document](#) for more info on N as used by the Turtles.

Open Equity

A measure of the value of positions that have been entered but not closed out. In Trading Blox, *Open Equity* is the current value of each position measured at the close of the previous day.

For a given long position:

Position Open Equity = (Current **Price** - Entry **Price**) X Contracts **or**
Shares X Dollars per Point

For a given short position:

Position Open Equity = (Entry **Price** - Current **Price**) X Contracts **or**
Shares X Dollars per Point

Percent Profit Factor

A measure of the relative profitability of a system. Like Profit Factor below except it is calculated using:

Percent Profit Factor = Average Win Percent / Average Loss Percent

Profit Factor

A measure of the relative profitability of a system. Profit Factor is calculated using:

Profit Factor = Total Winning Trade Profits / Total Losing Trade Losses

R-Multiple™

The concept of an R-Multiple™ was pioneered in 1993 by private trader Chuck Branscomb. R-Multiple™ is short for "Risk Multiple".

An R-Multiple™ is simply the profit or loss for a given trade divided by the entry risk. The entry risk is defined as the difference between the entry price and the stop price at the time of the entry.

```
' R-Multiple Calculation Example
' Profit / ( ( entry fill price - entry protective stop ) * quantity )
2243.68 / ( ( 8.27 - 8.25 ) * 1516 ) = 74
```

To learn about the origin of R-Multiple™s, see the forum topic [Improve Risk/ Reward by using Multiple Timeframes](#) on the Trader's Roundtable forum.

R Squared

A statistical measure of how close an equity curve is to a straight line plotted on a logarithmic graph. A value of 0 indicates a jagged line and a value of 1.0 represents a straight line. A fixed percentage investment that compounded and paid daily would have a straight line equity curve and an R Squared value of 1.0.

Set File

A Set file is an ASCII Text file which defines a portfolio which will appear in the Portfolio manager section of each system. See: [Portfolio Editor](#) for more information.

Simple Moving Average

A simple, or arithmetic moving average is calculated by adding the closing price of the security or futures market for a number of time periods, then dividing this total by the number of time periods. In other words, it's the average stock price over a period of time. Unlike the [Exponential Moving Average](#), this type of moving average gives equal weight to each daily price.

Standard Deviation

In trading, a Standard Deviation of price data is often used as a measure of volatility. In this context it is usually measured over a fixed number of days rather than over the entire data set.

More generally, the standard deviation (SD) of a given data set is derived from its variance, i.e. the arithmetic mean of all the squared distances between the data values and their arithmetic mean:

$$\text{Variance} = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x}_{ari})^2$$

The standard deviation is defined as the square root of the variance:

$$\text{Standard Deviation} = \sqrt{\text{Variance}}$$

For more info see: <https://mathworld.wolfram.com/StandardDeviation.html>

Total Equity

Total Equity is a measure of equity that takes into account all closed out trades and open position equity (or Open Equity).

$$\text{Total Equity} = \text{Closed Equity} + \text{Open Equity}$$

Wilder Moving Average Formula

Welles Wilder's indicators, including Average True Range do not use the standard exponential moving average formula. The Wilder formula for computing an N-Day Exponential Moving Average is:

$$\text{Today's MA} = \text{Today's Price} \times \frac{1}{N} + \text{Yesterday's MA} \times \frac{N-1}{N}$$

You can determine the equivalent non-Wilder [*Exponential Moving Average*](#) using the following formula:

$$\text{Exponential MA Days} = (\text{Wilder MA Days} \times 2) - 1$$

Edit Time: 2/19/2022 12:03:34 PM

Topic ID#: 150

Section 12 – Moving the License

Installation on a new computer

The license agreement and authentication process allows for one user to install on one or two computers initially (depending on the edition), plus additional purchased computers, for non simultaneous use. The first computers you install the application on will be accepted by the authentication server. You can always re-install as many times as you need to on these authenticated computers. If you try to install on a new computer, it will not authenticate. You must be connected to the internet to authenticate your installation for the first time on a new computer. The authentication process uses both the computer username and the computer identification, so two users logged into the same computer server would be seen as two separate installations.

In the case where you get a new computer and would like to MOVE an install from an old computer to this new computer, you must use the Remove License menu option to first remove the license from your old computer. This will only remove the license, and not the Trading Blox folder. You will then be able to install the application on the new computer, or if you want to save all your changes you can just move the entire Trading Blox folder from the old computer to the new computer. You may need to update your data folder locations and such. Do not just uninstall the application from the old computer, as that will not remove the license from that computer.

If you have a hard drive failure or some other catastrophe that makes it impossible to use the Remove License process, you can reset your license account from the Customer Login section of the website. Be sure to keep your maintenance current so that this online process will be immediate. If your maintenance is not current, you will need to request a manual reset. You can do all of this from the Customer Login section of the website.

Edit Time: 9/12/2020 9:50:00 AM

Topic ID#: 178

Section 13 – Portfolio File Format

There is one Portfolio File per Portfolio. These portfolios are stored in ASCII Text files with a ".set" extension in folders inside the main Trading Blox installation folder. There is a "Futures Sets" folder for futures, a "Forex Sets" folder, and a "Stock Sets" folder for stocks.

C:\Trading Blox\TradingBlox\Futures Sets
C:\Trading Blox\TradingBlox\Forex Sets
C:\Trading Blox\TradingBlox\Stock Sets

DON'T EDIT PORTFOLIO FILES BY HAND - PLEASE USE THE [PORTFOLIO EDITOR](#)

The only contents of a Portfolio File are the list of instrument symbols with one symbol per line. Each line has a line-feed and carriage return character following the symbol.

Trading Blox automatically detects all portfolio files present in each and places them as an option in the Portfolio Manager pull down menu.

Forex and Stock Portfolio File Order

With the exception of Futures Markets, the order in which markets are stored in a *Portfolio* file will likely affect performance. While not at all obvious, the reasoning is simple. Each trading day, Trading Blox processes the list of markets exactly as they are ordered in the *Portfolio* file. If markets are stored alphabetically, and multiple trades are signalled in a single day in closely correlated markets, then it is possible, that trades would be taken in AD, BP, and CD, but rejected in JY due to the effect of Max Closely Correlated Units. Under the same set of circumstances, if the markets had been stored in reverse alphabetical order, then JY, CD, and BP trades would be taken, and AD would be rejected.

Since the Portfolio Editor orders the files alphabetically by symbol, this is the order that is written to the set files and therefore the order that trade orders are processed by Trading Blox.

This is one of the inherent limitations of testing with daily data. In the example above, it is impossible to discern which trade actually fired off in which order without the aid of intra-day data, so Trading Blox must take trades based upon the order of instruments in the *Portfolio* file.

This is analogous to the issue that arises with attempting to track consecutive wins/losses in a multi-commodity systems during back testing on daily data. Consider, for example, a system that enters and exits on the open: As of last night, the system had closed out a total of ten consecutive losses over the prior few weeks. Today, 2 positions (in different markets) are to be exited on the open, one at a profit, and one at a loss. If the profitable trade is counted first, then the losing string is broken at 10. Alternately, if the losing trade is counted first, then the string of consecutive losses is extended to 11, and is broken by the profitable trade.

Which is correct? Actually, it depends on which pit or exchange opened first. But what if both trades were executed simultaneously? The answer is that this is one of the inevitable limitations of testing with daily data, and it is a function of the order in which the markets in the portfolio are presented for processing by the back testing software.

Futures Market Order

The orders for Futures markets are processed in a special field *Order Sort* (see: [Futures Dictionary](#) for details) which is part of the Futures Dictionary entry for each market. Trading Blox recommends setting this value to the opening time of the market.

This will increase the accuracy of the simulation because markets that open first are more likely to have positions before markets that open later.

The Order Sort value also makes it easier to handle actual orders because they appear in the same order that the markets will open when generating orders using the Suite | Generate Orders menu item.

Edit Time: 2/19/2022 12:51:14 PM

Topic ID#: 185

Section 14 – Risk and Return

The ideal investment is one that offers the greatest return, with the least amount of risk.

Thus, given the choice between two investments with identical returns, the one with the least fluctuation, or variability of returns, is generally deemed the superior investment. But as Jack Schwager points out in *Managed Trading, Myths & Truths*, "Too many investors take the mistake of focusing solely on return...It is also critical to incorporate some measure of risk as part of the evaluation process."

Adjusting for Risk

Opportunity involves risk. It is important, therefore, to understand the risks we take, in order to evaluate whether we are being justly rewarded. Measures of risk-adjusted return (such as the Sharpe and the MAR Ratios, do just that: They literally divide an investment's return by the risk required to achieve it.

They are all, in one form or another, ratios of gain-to- pain.

Between risk and return, return is by far the more tractable concept. Looking at an arithmetic average or a compounded rate of return will tell you just about everything you need to know about reward.

The evaluation of risk, on the other hand-trying to determine what risk is, and how to measure it-is a more daunting task, and one that is complicated by the fact that the human perception of risk, and our tolerance for it, varies greatly with circumstances such as age, health, and wealth (to name just a few), along with a host of psychological and emotional factors.

If you manage money for others, for instance, your tolerance for risk will be influenced strongly by your clients' risk preferences, and will be entirely different from that of an individual trader whose goal may be to shoot for the highest possible returns without blowing out.

Drawdown

Drawdown is measured as a percentage retracement from a previous equity peak (or account balance high). This downside risk statistic is sampled peak-to-valley, and typically uses marked-to-the-market daily or monthly data points.

Maximum Drawdown is the largest such negative excursion during the life of the simulation, and conveys the maximum pain an investor would have had to endure in order to achieve the resulting return. Maximum Total Equity Drawdown is the denominator of the MAR Ratio (which uses daily data points), and the Calmar Ratio (monthly data points).

Duration of the Longest Drawdown, like Maximum Drawdown, is a one-time event, and is measured from previous equity peak, to new equity peak. It is also an important performance statistic psychologically not only for individual traders, but for fund managers, who are prone to losing clients during extended underwater periods.

One reason why various measure of drawdown and its duration are such important statistics to many traders is that even good systems typically spend far more time in periods of drawdown than they do making new equity highs.

But while Maximum Drawdown and Period of Longest Drawdown represent unique, extreme downside risk events, they convey almost no information about overall volatility.

Standard Deviation

The overall variability of investment returns is best described by Standard Deviation. It is typically calculated based on monthly, or annual percentage changes (rather than dollar fluctuations of the equity curve series itself).

Standard Deviation measures both upside and downside volatility, and is literally the square root of variance, which describes the dispersion of data points around an average. Standard deviation is the denominator of the Sharpe Ratio, a classic measure of risk vs. reward.

Standard Deviation is an easy concept to convey without resorting to mathematical formulae, and the following example demonstrates why averages with large standard deviations can be misleading, and why two investments with similar returns can have dramatically different risk characteristics:

The U.S. Virgin Islands have an average annual temperature of approximately 80°F. This is paradise, and it is habitable year-round.

Let's compare this with an actual location in Southern California, not too far inland from the Pacific Ocean. It has a year-round average temperature of 76°F, which is a little cooler than paradise. So far, so good.

Research shows that the average daily high throughout the year is 90°F, and the average daily low is 62°F. Still habitable, but it's beginning to sound a little hot. Maybe not, though; the Virgin Islands has an average daily high/low of 86° / 74°F (<https://www.orcasailing.com/bvi-details.html>)

Our pleasant California locale also has very mild winters. More research, though, shows that while the average daily low for the summer is 77°F, the average daily summer high is...105°F. So it's apparent that, even though its average annual temperature is slightly lower than the Virgin Islands, the range, or variance of its temperature is significantly higher, and may even be cause for alarm, where habitability is concerned.

Further digging turns up the fact that 1996 was the hottest summer on record: There were 40 days over 120°F, and 103 days over 110°F. The summer of 1974 was no more hospitable, setting a record of 134 consecutive days with a maximum temperature of over 100°F. In 1913, the temperature reached 129°F or above, five days in a row (a world record at the time), and the hottest temperature ever recorded was 134°F on July 10 of that same year. (<https://wrgis.wr.usgs.gov/docs/parks/deva/weather.html>)

Our pleasant-sounding climate, with a year-round average temperature lower than the Virgin Islands, now sounds more like an inferno. And it is: It's Furnace Creek in Death Valley, CA, the hottest, driest place in North America.

So the vast difference in habitability between these two locales-which share similar year-round average temperatures-lies solely in the standard deviation of their respective temperatures.

There is no doubt that, by treating upside and downside volatility the same, Standard Deviation does a very good job of describing the overall variability of investment returns. However, there is a school of thought which maintains that Standard Deviation is an inappropriate risk measure under certain circumstances because it unfairly penalizes the high upside volatility often experienced by trend-following Commodity Trading Advisors.

Trend-following CTAs typically employ rigid stop-loss strategies that produce a return profile characterized by a small number of well-contained losses, and an even smaller number of large winning trades. So the argument that standard deviation unfairly penalizes their risk-adjusted returns may well have merit.

In response, some risk-adjusted metrics-such as the Sortino Ratio-look only at downside Standard Deviation (also called semi-deviation). While similar to the Sharpe Ratio in general form, the denominator of the Sortino Ratio calculates the standard deviation of the negative data points only, against the mean of the population (of all data points).

Determining Your Own Measures

The various risk-adjusted return statistics all look at the same picture from slightly different angles, and there simply isn't a single metric that will satisfy every investor, under all circumstances. But understanding the components of these various statistics will allow you to determine which combination is right for you, and Trading Blox offers a number of risk-adjusted metrics that help you to do just that.

As a parting thought, it is worth noting that the distribution of returns from the world's financial markets are non-normally distributed, which means simply that extreme events are likely to occur more frequently than randomly (normally) distributed returns would dictate. In addition, there is evidence that the volatility of financial markets increases steadily with time. (Bernstein, Peter L. - Capital Ideas; The Improbable Origins of Modern Wall Street, pps. 21-22. The Free Press, 1992.)

Therefore, it is imprudent to think that the maximum risk statistics derived from any amount of historical testing will not be exceeded in the future. They most likely will be. Good traders count on it, and plan for it.

Section 15 – System Component Table

System Component:	Corresponding Block Type:
What to Trade	Portfolio Manager
When to Trade	Entry Blox
Whether to Trade	Risk Manager
How Much to Trade	Money Manager
When to get out	Exit Blox
Any New Idea Not Related to Above	Auxiliary

Edit Time: 2/19/2022 11:51:29 AM

Topic ID#: 215

Section 16 – Technical Support

Trading Blox has a technical support team dedicated to helping you resolve any technical or functional issues you may experience. Technical support personnel can be reached by email at any time, or via our Internet discussion forum.

If your issue can be resolved on the spot, it will be. If necessary, someone from our support team will investigate the issue, and contact you directly with an answer as soon as one is available.

Current email, phone, and address here: <https://www.tradingblox.com/tradingblox/contact.htm>

Forum: <https://www.tradingblox.com/forum/viewforum.php?f=58>

If you can't access the above forum, please be sure you have registered and are logged onto the forum. If you still can't access this forum, you will need to update your customer profile in the Customer Login <https://www.tradingblox.com/tradingblox/updateLogin.php> section of the website. Here when you update your forum username, it will add you to the customer group. Then try the above link again.

Be sure to keep your customer profile update, so we have your latest address, phone, and email in case we need to contact you about new versions and updates.

PDF Manual Files

Please visit the [Documentation](#) webpage for the most recent PDF, Browser based, and Help File based Documentation for both Trading Blox and the Blox Builder.

Create Service Report

In many cases it is helpful for us to have a copy of your environment. We have a utility called Create Service Report, that you can access from the Help menu. When you have created the report, email us the output and we will be able to trouble shoot your issue more easily and quickly.

To create, view, and send the service report:

- Use the Help / Create Service Report menu item. This will create a zip file of your blox and systems. It will open the folder containing this file.
- Email this file to customer support. Check the website for the current customer support email address.

Index

- . -

.tbz 350

- A -

Always on Tick 102
 Annual Sortino Ratio 386, 388
 Annual Sortino Ratio: 386
 Appendices 522
 Authentication 555
 Average Loss Percent 386
 Average Risk Percent 386
 Average Trade Percent 386
 Average Win Percent 386

- B -

Backups 524
 Batch Job 527
 Batch-File 527
 Batch-Script Files 527
 Big Point Value Calculations 541
 Blox Basic Editor 82
 Blox Editor Colors 213
 BPV 341
 Broker Positions 102

- C -

Command Line Switches 527
 Create Service Report 562

- D -

Data Folders 183
 Data Options 183
 Data Providers 540

- E -

Earn Interest 102
 Entry Day 102
 Entry Day Retracement 102
 Equity to Trade 102
 Expectation 386
 Export folder 259
 Exporting Suites 353

- F -

F7 7
 Fixed Length 102
 Folders 544
 Forex Carry Calculations 545
 Futures Dictionary Editor 183

- G -

General 193
 General Reporting Settings 193
 Generate Orders 7
 Global 133, 481
 Global Suite Selected 131
 Global Suite System 259
 Global System 133, 481
 Glossary 550
 Goodness 133, 481
 GSS 131, 259

- H -

Historical Data

- Currency File 276
- Dictionaries 276
- Exchange File 276
- Forex Dictionary Editor 291
- Forex Interest Rates 276
- Futures Dictionary Editor 297
- Price Data 276
- Stock Dictionary Configuration 310

Historical Testing 329

- Order Filter 336
- Parameter Stepping 331
- Setting Test Dates 335

- I -

Ignore All Test Positions 102

Importing Suites 350

Included Systems and Blox 416

- ADX System 438
- ATR Channel Breakout System 440
- Bollinger Breakout System 443
- Bollinger Counter-Trend System 446
- Chandelier Exit 436
- Donchian System 451
- Dual Moving Average System 453
- Gap Against Exit 437
- MACD System 456
- Money Managers 417
- Portfolio Managers 434
- Risk Managers 419
- RSI Trend Catcher System 458
- Stats for Multi Parameter Charts 492
- Stochastic System 461
- Triple Moving Average System 466
- Turtle System 470
- Walk Forward 481

Installation 10, 562

Interest 102

- L -

Large Simulation Reporting 205

Log Files 395

- Daily Equity Log 399

- Monthly Equity Log 400

- PrintOutput.csv 401

- Trade Log 396

Losing Months 386

Losses 386

- M -

Margin to Equity Ratio 386

Max Margin 102

Max Margin/Equity to Trade 102

Max Position Units 102

Minimum 102

Minimum Slippage 102

Monte Carlo 199

Move Computer 555

Move License 555

Moving License 555

Multi-Parameter Goodness 102

- N -

NotePad++ 221

- O -

OOS 481

Order Generation 496

- Broker Positions 499

- Order Generation Preferences 499

- Preferences 499

Overview 7

- P -

Parameters

Stepping 331

Percent 102

Portfolio Editor 318

Portfolio Manager Editor 183

Portfolios

Editing Portfolios 318

Sample Portfolios 325

Preferences

Blox Basic Colors 213

Blox Editor Size 213

Data Folders and Options 183

General Reporting 193

Report Content Specification 339

Trade Chart Colors 208

Trade Details 210

Preferences Editor 181

Profit Factor 386

Profit Factor Ratio 386

- R -

Reducing Optimization Time 193

Remove License 555

Retracement 102

Risk and Return 558

- S -

Service Report 562

Single Test Reporting 201

Slippage 102

Slippage Percent 102

Small Simulation Reporting 203

Smart Fill 102

Smart Fill Exit 102

specific keywords 527

Start Date 102

Start Date Stepping 102

Summary 7

System Area Section Limitations and a blox name ordering. 250, 251

System Editor 82, 133, 259, 481

System Editor Information 23

System Requirements 10

Systems

Working with 259

- T -

Technical Support 562

Test End 102

Test End Type 102

Test Reporting Sections 193

Test Reporting Types 193

Test Results 361

Equity Graph (Linear) 370

Equity Graph (Logarithmic) 369

Instrument Performance Summary 384

Monthly Returns 372

Monthly Returns Distribution 373

Multi-parameter Surface Chart 365

Parameter Stepping Graph 367

R-Multiple™ Contribution 377

R-Multiple™ Distribution 376

Stepped Parameter Performance 363

System Parameter Settings 380

Total Risk Profile 374

Trading Performance and Statistics Report 386

Yearly Performance Summary Report 378

Test Start Date 102

Thread 102

Thread Count 102

Total Loss Dollars 386

Total Trades 386

Total Win Dollars 386

Trade Always on Tick 102
Trade Chart Colors 208
Trade Details 210
Trading Blox Installation 221
TradingBlox.ini 216, 221, 365

- U -

Units 102
USD_Rates.txt 295
Use All Data 102
Use Broker Positions 102
Use Date 102
Use Start Date Stepping 102
Using Trading Blox 14
 Editing Parameters 67
 Test Results 356
 Test Trade Report 86

- V -

VB-Script 527
VB-Script Files 527

- W -

Walk Forward 481
Walk Forward Stats 2 133, 481
Walk-Forward Suite 131
Winning Months 386
Wins 386