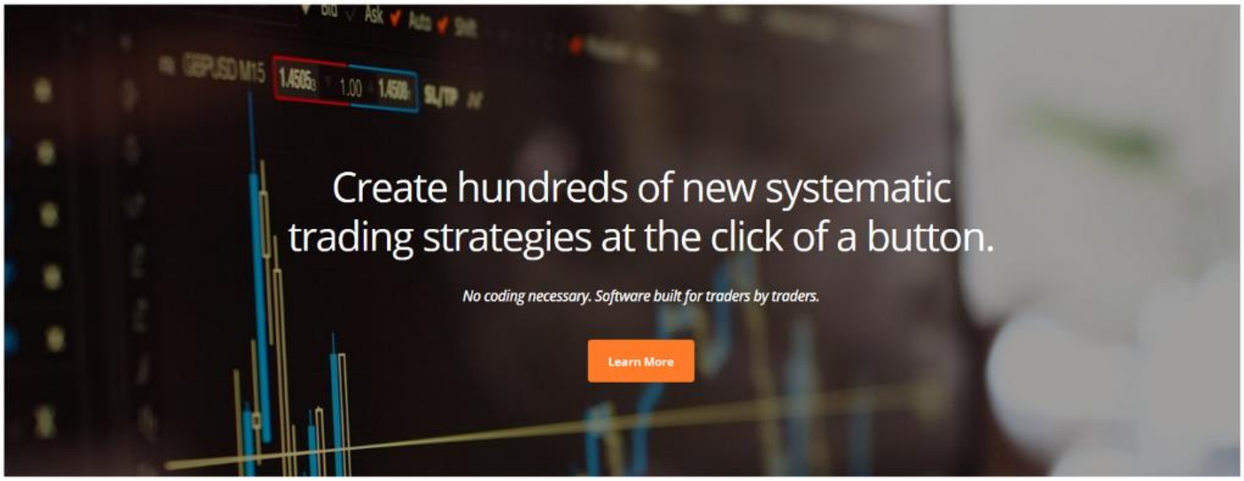
2022

[redacted]



Create hundreds of new systematic trading strategies at the click of a button.

No coding necessary. Software built for traders by traders.

Axiom Futures, LLC

Build Alpha

Risk Disclosure

FUTURES AND FOREX TRADING CONTAINS SUBSTANTIAL RISK AND IS NOT FOR EVERY INVESTOR. AN INVESTOR COULD POTENTIALLY LOSE ALL OR MORE THAN THE INITIAL INVESTMENT. RISK CAPITAL IS MONEY THAT CAN BE LOST WITHOUT JEOPARDIZING ONES FINANCIAL SECURITY OR LIFESTYLE. ONLY RISK CAPITAL SHOULD BE USED FOR TRADING AND ONLY THOSE WITH SUFFICIENT RISK CAPITAL SHOULD CONSIDER TRADING. PAST PERFORMANCE IS NOT NECESSARILY INDICATIVE OF FUTURE RESULTS.

Hypothetical Performance Disclaimer

HYPOTHETICAL PERFORMANCE RESULTS HAVE MANY INHERENT LIMITATIONS, SOME OF WHICH ARE DESCRIBED BELOW. NO REPRESENTATION IS BEING MADE THAT ANY ACCOUNT WILL OR IS LIKELY TO ACHIEVE PROFITS OR LOSSES SIMILAR TO THOSE SHOWN; IN FACT, THERE ARE FREQUENTLY SHARP DIFFERENCES BETWEEN HYPOTHETICAL PERFORMANCE RESULTS AND THE ACTUAL RESULTS SUBSEQUENTLY ACHIEVED BY ANY PARTICULAR TRADING PROGRAM. ONE OF THE LIMITATIONS OF HYPOTHETICAL PERFORMANCE RESULTS IS THAT THEY ARE GENERALLY PREPARED WITH THE BENEFIT OF HINDSIGHT. IN ADDITION, HYPOTHETICAL TRADING DOES NOT INVOLVE FINANCIAL RISK, AND NO HYPOTHETICAL TRADING RECORD CAN COMPLETELY ACCOUNT FOR THE IMPACT OF FINANCIAL RISK OF ACTUAL TRADING. FOR EXAMPLE, THE ABILITY TO WITHSTAND LOSSES OR TO ADHERE TO A PARTICULAR TRADING PROGRAM IN SPITE OF TRADING LOSSES ARE MATERIAL POINTS WHICH CAN ALSO ADVERSELY AFFECT ACTUAL TRADING RESULTS. THERE ARE NUMEROUS OTHER FACTORS RELATED TO THE MARKETS IN GENERAL OR TO THE IMPLEMENTATION OF ANY SPECIFIC TRADING PROGRAM WHICH CANNOT BE FULLY ACCOUNTED FOR IN THE PREPARATION OF HYPOTHETICAL PERFORMANCE RESULTS AND ALL WHICH CAN ADVERSELY AFFECT TRADING RESULTS.

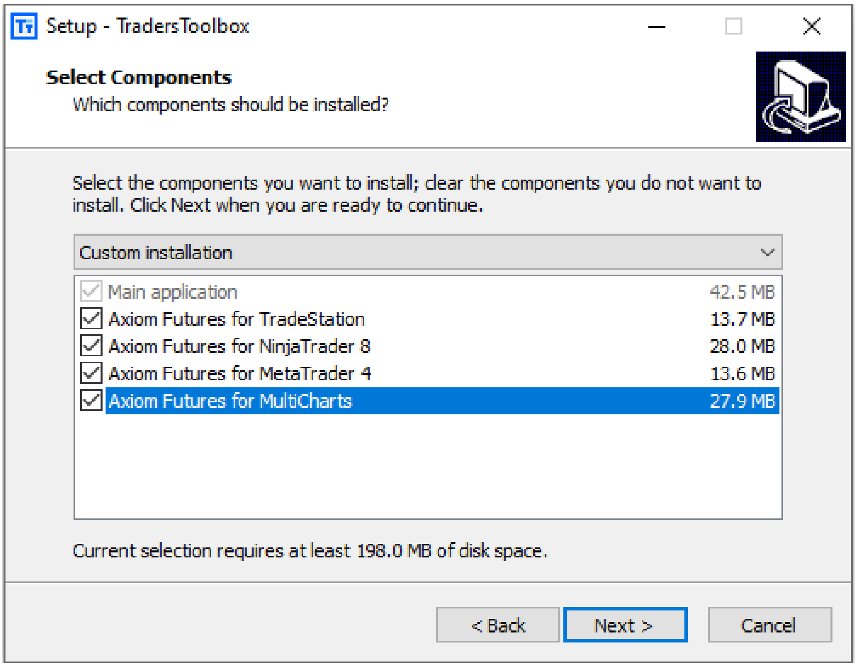
Navigating the Signal Glossary/User Guide

Looking for something specific? Hit CTRL + F at the same time to bring up a search bar. Then type in what you are looking for and hit enter.

For example, CTRL + F then type “Importing into TradeStation” or even simpler “Import” and you can find all the instances of the word Import in the Guide. This will make things easy to use.

Why Install one of the DLLs?

The DLLs (and support files provided in the install email) are essential to make sure every unique custom signal, indicator and rule inside Build Alpha works in each of the supported trading platforms. If you only wish to use BA to test ideas and not to execute any strategies, then you do not need to install any DLL or support file. It is best to choose which ones to install or to just install all when you first install Build Alpha. If you selected multiple platforms, please **DO NOT** restart your computer until **ALL** have installed. To do so, please select all (or those that apply).



Select Components

Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

Custom installation

☑ Main application	42.5 MB
☑ Axiom Futures for TradeStation	13.7 MB
☑ Axiom Futures for NinjaTrader 8	28.0 MB
☑ Axiom Futures for MetaTrader 4	13.6 MB
☑ Axiom Futures for MultiCharts	27.9 MB

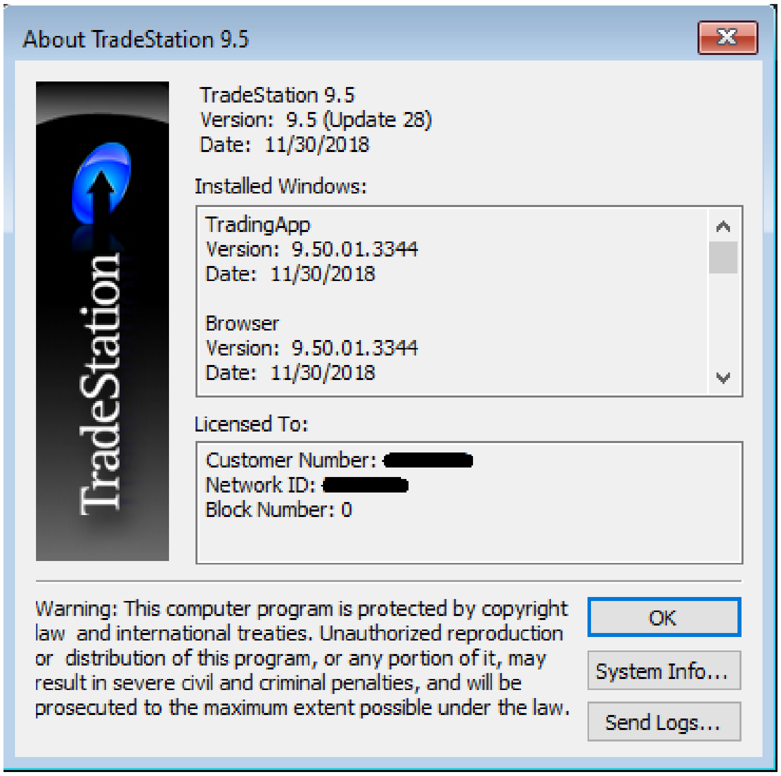
Current selection requires at least 198.0 MB of disk space.

Setting Up DLL in TradeStation

Please run the appropriate installer sent in the install email. Here it is again:

https://www.dropbox.com/s/awhuj1jdl8kldcf/Release_AxiomFutures_TradeStation_setupv1.exe?dl=0

Inside TradeStation please send me your customer ID number so I can grant your account access to the Build Alpha data, DLL and functions. To find your customer ID inside the TradeStation platform please go to Help -> About. Then please send me the code.



About TradeStation 9.5

TradeStation 9.5
Version: 9.5 (Update 28)
Date: 11/30/2018

Installed Windows:

TradingApp
Version: 9.50.01.3344
Date: 11/30/2018

Browser
Version: 9.50.01.3344
Date: 11/30/2018

Licensed To:

Customer Number: [redacted]
Network ID: [redacted]
Block Number: 0

Warning: This computer program is protected by copyright law and international treaties. Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under the law.

Note:

Please also install the ELD support file. File -> Import Easy Language Document -> ELD -> Select this file.

https://www.dropbox.com/s/k41pi622xxya1ie/BuildAlpha.ELD?dl=0

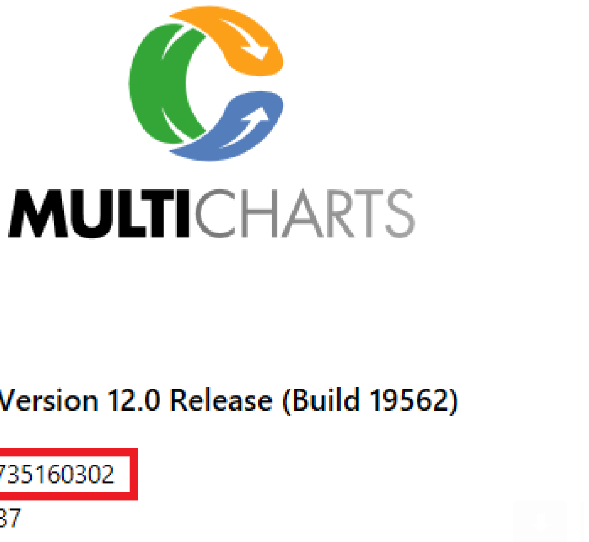
Setting Up DLL in MultiCharts

Please run the appropriate installer sent in the install email. Here it is again:

https://www.dropbox.com/s/73m9w31v5h6tre1/Release_AxiomFutures_MultiCharts_setupx.exe?dl=0

Inside MultiCharts, please find your Licensed to number. Select Help -> About MultiCharts

If yours does not say Licensed to but says ‘Evaluation Copy’ then please send me your User ID instead.



MultiCharts64 Version 12.0 Release (Build 19562)

Licensed to: MC_7735160302

User ID: 2274787337

Note

Please also install the support ELD file. Open the Power Language Editor, File -> Import -> Select this ELD

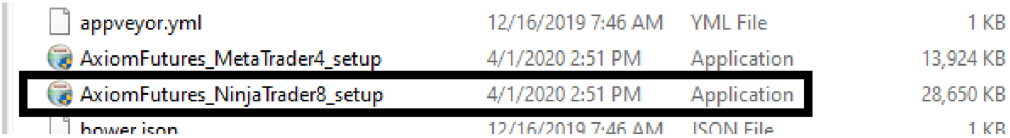
https://www.dropbox.com/s/k41pi622xxya1ie/BuildAlpha.ELD?dl=0

Setting up DLL in Ninjatrader8

If you did not select NT8 during initial Build Alpha installation, then please download and install the appropriate installer sent with the install email. Here it is again:

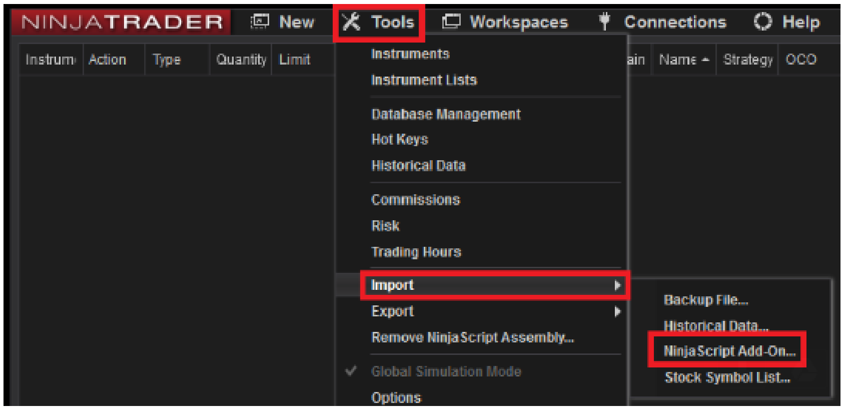
https://www.dropbox.com/s/1wzzyrwccfo1kir/Release_AxiomFutures_NinjaTrader8_setup.exe?dl=0

If you did, please go to C:/Users/{name}/TradersToolbox and run the NT8 installer:

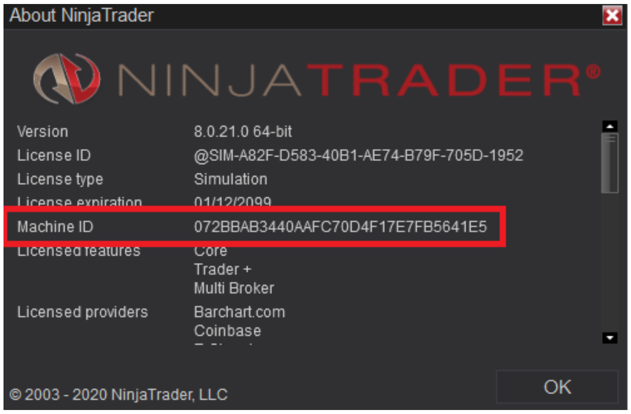


appveyor.yml	12/16/2019 7:46 AM	YML File	1 KB
AxiomFutures_MetaTrader4_setup	4/1/2020 2:51 PM	Application	13,924 KB
AxiomFutures_NinjaTrader8_setup	4/1/2020 2:51 PM	Application	28,650 KB
bower.json	12/16/2019 7:46 AM	JSON File	1 KB

Inside Ninjatrader8, select Tools -> Import -> Ninjascript Add On… and then select the zip file in your Documents/Axiom Futures/NinjaTrader/ folder. It will be AxiomFutures.zip.



Then please go to Help->About and send the Machine ID to me so I can activate your NT8 for full access



About NinjaTrader

Version	8.0.21.0 64-bit
License ID	@SIM-A82F-D583-40B1-AE74-B79F-705D-1952
License type	Simulation
License expiration	01/12/2099
Machine ID	072BBAB3440AAFC70D4F17E7FB5641E5
Licensed features	Core Trader + Multi Broker
Licensed providers	Barchart.com Coinbase

© 2003 - 2020 NinjaTrader, LLC

Note: You will also need to install the BuildAlphaNT8.zip. Tools->Import->Addons->Select the zip file

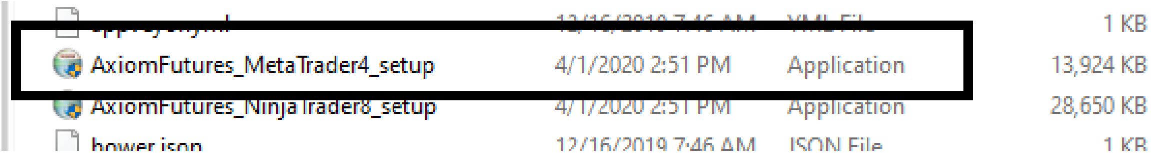
https://www.dropbox.com/s/qrnk7yrt8wqu8ei/BuildAlphaNT8.zip?dl=0

Setting up DLL in Metatrader4

If you did not select MT4 during initial Build Alpha installation, then please download and install the appropriate installer sent with the install email. Here it is:

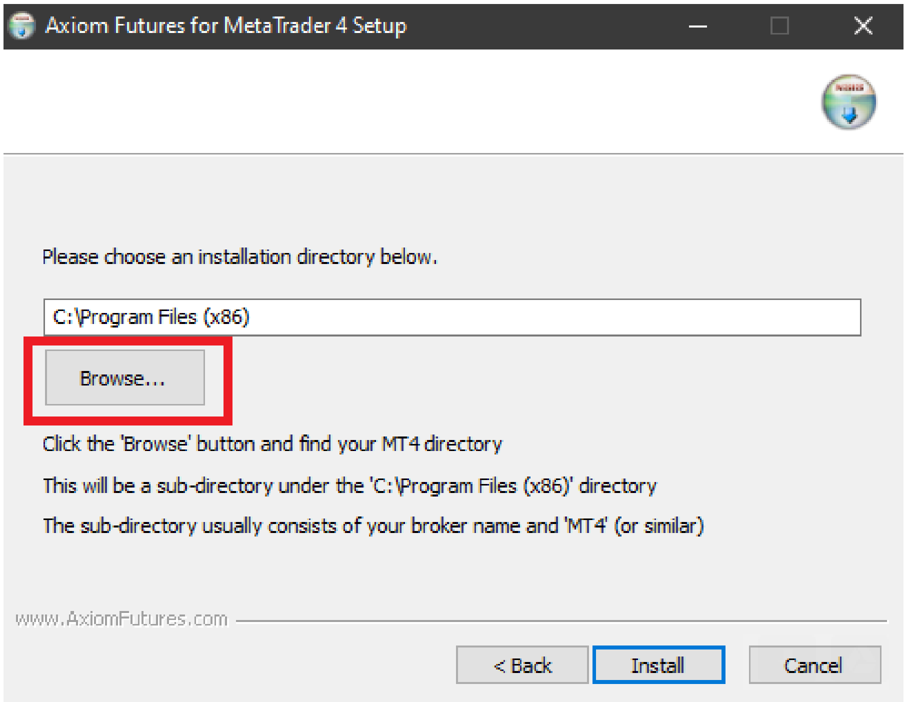
https://www.dropbox.com/s/wxxgaotsz180klc/Release_AxiomFutures_MetaTrader4_setupv1.exe?dl=0

If so, please go to Users/{name}/TradersToolbox/TradersToolbox and run the MT4 installer



appveyor.yml	12/16/2019 7:46 AM	YML File	1 KB
AxiomFutures_MetaTrader4_setup	4/1/2020 2:51 PM	Application	13,924 KB
AxiomFutures_NinjaTrader8_setup	4/1/2020 2:51 PM	Application	28,650 KB
bower.json	12/16/2019 7:46 AM	JSON File	1 KB

When prompted, please user browse to find your MT4 installation. They're almost always in the Program Files (x86) folder.



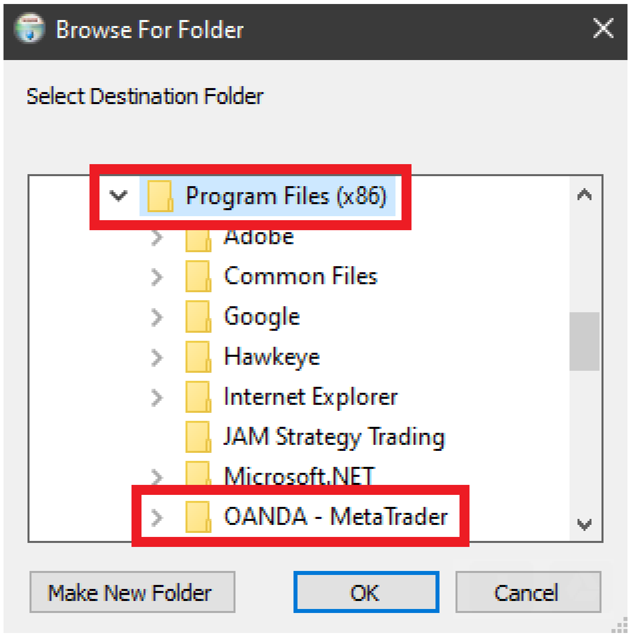
Please choose an installation directory below.

C:\Program Files (x86)

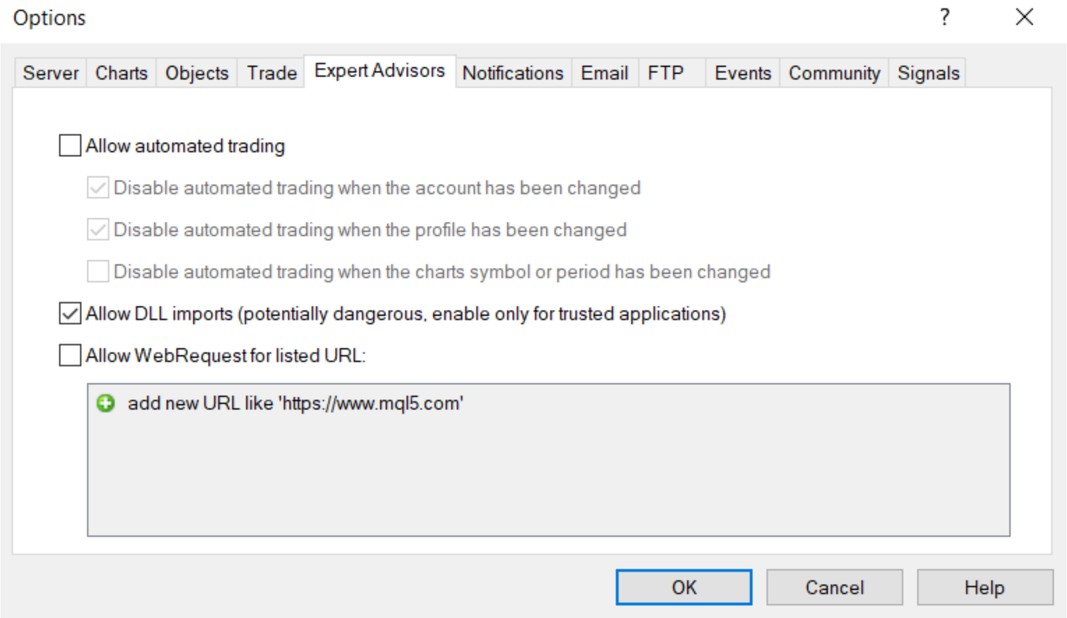
Click the 'Browse' button and find your MT4 directory

This will be a sub-directory under the 'C:\Program Files (x86)' directory

The sub-directory usually consists of your broker name and 'MT4' (or similar)



Then inside MT4, please go to Tools -> Options -> Expert Advisors and enable DLLs



□ Allow automated trading

☑ Disable automated trading when the account has been changed

☑ Disable automated trading when the profile has been changed

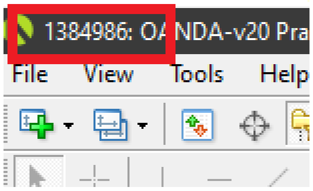
□ Disable automated trading when the charts symbol or period has been changed

☑ Allow DLL imports (potentially dangerous, enable only for trusted applications)

□ Allow WebRequest for listed URL:

add new URL like 'https://www.mql5.com'

Finally, please send me your MT4 ID so I can activate.



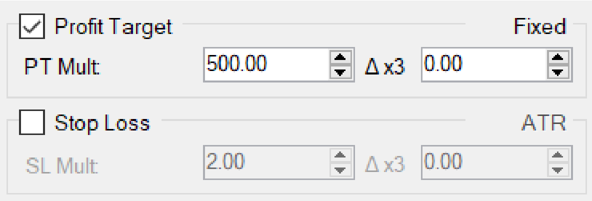
Note: You will also need to download the include file and put in your MT4 Include folder.

Standard Exits:

Max Holding Time: This is the maximum number of bars a strategy will hold a position. If set to 3, for example, then the strategy will exit after being in the trade for 3 bars. If you are using the built-in daily data, then 3 would be 3 days. If using 15-minute custom intraday data, then 3 would equate to 45 minutes. To turn this feature "off" just set the value extremely high that will never evaluate as true. For example, 999999.

Profitable Closes: This is the number of profitable closes that would trigger the strategy to exit a position. For example, set to 2 would require two closes to be greater than a long entry price before an exit would trigger. For example, enter on Monday at 20.50. Tuesday closes at 20.65 and profitable closes is now 1. Wednesday closes at 20.61 which is still greater than Monday's entry of 20.50 so profitable closes is now 2. The strategy would then exit. To turn this feature "off" just set the value extremely high that will never evaluate as true. For example, 999999. Also, please note profitable closes set greater than the max holding time can never evaluate to true. You cannot have 6 profitable closes with a max holding time of 4 bars, for example.

Profit Target: Can be turned on and off. It can also be changed between ATR based and Dollar based. If on and set to ATR, the PT Mult is a multiple of the 20 period Average True Range with 20 being configurable in the settings window. An ATR setting of 2 would mean we want to set a profit target at 2 ATR units above our entry (for longs). Using ATR/volatility targets allows system to be dynamic and evolve with the market as opposed to using static values like 10 points or 3%. You can adjust the length of the Average True Range in the settings menu. (Services-> Settings-> ATR length). If you click on the "ATR" it will change to "Fixed". When set to fixed, please enter the dollar amount you wish to exit with a profit at. For example, this will set a profit target of $500



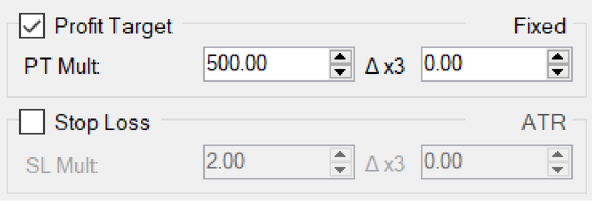
☑ Profit Target Fixed
PT Mult 500.00 Δ x3 0.00
□ Stop Loss ATR
SL Mult 2.00 Δ x3 0.00

Stop Loss: Same as Profit Target.

Trailing Stop Loss: Both an ATR and Fixed option. Configurable the same as Profit Target and Stop Loss. Updated based on the maximum value reached while in the trade (for longs).

Highest High: This acts as a price-based target for long strategies and a stop for short strategies. A long strategy would exit at the highest high of the past N days. N can be configured by setting HH Look. A HH Look set to 5 would mean a strategy would exit at the highest high of the past 5 rolling bars.

Lowest Low: Same as Highest High



☑ Profit Target Fixed
PT Mult 500.00 Δ x3 0.00
□ Stop Loss ATR
SL Mult 2.00 Δ x3 0.00

Update: You can now test multiple exit combinations in one simulation using Delta x3 feature. Each exit option has two input values. The one on the left is the starting point and the input on the right is the increment or step amount. For example, setting max time to 2 and 1 would test max times of 2, 3, 4. Setting PT to 0 and 1.5 would test OFF, 1.5, 3.0. Setting PT to 1 and 1.5 would test 1.00, 2.50, 4.00. Leaving the second value to 0 will only check the starting value and will make for a faster simulation. In general, the more you select the more time it will take to process.

Fitness Functions:

Fitness Functions are the performance metric Build Alpha will attempt to create strategies that excel in. For example, you can create strategies that try to optimize total profit, least drawdown, best profit to drawdown ratio, or even smoother equity curve. Below is a definition and calculation (if permitting) of each of the fitness functions.

PNL: Total profit. Build Alpha will create and save strategies that make the most money in the test period.

DD: Drawdown. Build Alpha will create and save strategies that have the lowest drawdown over the test period.

PNLDD: Profit to Drawdown ratio. Build Alpha will create and save strategies that have the highest profit to drawdown ratio

Ratio WL: Average winning trade divided by average losing trade.

Profit Factor: Gross Money Made divided by Gross Money Lost.

Sharpe: Expected return given expected amount of volatility. A measure of risk-adjusted returns.

Sortino: Expected return given an expected amount of negative volatility. Similar to Sharpe but only focuses on volatility of losing trades.

Average Trade: Average Trade in Dollars.

Win Percentage: Number of winning trades divided by number of trades.

CPC: Win percentage multiplied by ratio WL multiplied by profit factor. Generally, values under 1.2 should be discarded.

CorrCoef: Correlation Coefficient is a measure of how linear an equity curve is. If it jumps around or is non-linear in shape then this value will be low, but if it progress with a slope close to 1 then the value shall be close to 1. The higher this value the more linear the account growth or equity curve is.

T-Test: T Statistic. A measure of how statistically valid the average trade is. That is, is the average trade actually greater than 0. A higher value means a higher chance the average trade is a true amount and not something achieved by luck or data mining.

CAGR: Compound Annual Growth Rate. Simply defined as (((End/Begin)^(1/years)) – 1) * 100.

SQM:
trade count = if more than 100 trades use 10 else use sqrt(trade count)
(Average Trade / (Average Losing Trade + Stdev Losing Trade)) * trade count

SQN: SquareRoot(Trade Count) * (Average Trade / Stdev Trade)

E-Ratio: This searches for the highest edge-ratio in the first ten bars. Please refer to buildalpha.com/eratio for detailed explanation of edge ratio.

K-Ratio: Searches for the highest K-Ratio.

$$\text{k-ratio (2013)} = \frac{\text{Slope of Least-Squares Regression Trendline}}{\text{Standard Error of Slope}} \times \frac{\text{Square Root (NPY)}}{\text{N}}$$

Expectancy Score: (Average win * prob(W) + Average Loss * prob(L)) * (No. trades / No. Bars in Backtest)

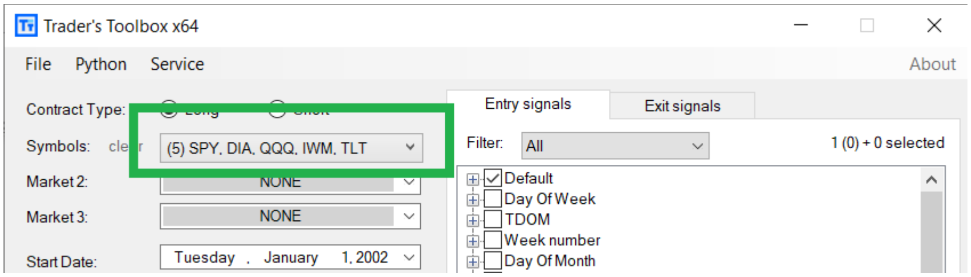
Perfect Profit Percentage: This assumes perfect forward information. We find the absolute BEST equity curve then find strategies that are the closest percentage to this perfect (unattainable) equity curve.

Perfect Profit Correlation: Similar to above but looking for correlation instead of percentage.

Robustness Index: Average trade from in-sample period divided by average trade from out-of-sample period. This is not yet added to the Fitness Function dropdown but is available in the Out of Sample output results.

Symbol Selection

You can either choose to develop a strategy on one market of interest or across a basket of selected symbols. Please note that any symbol selected in the 'Symbol' dropdown will be used to generate the best strategies. The following example uses 5 markets in a basket to find/create the best strategies (SPY, DIA, QQQ, IWM, TLT).



Contract Type: ○ Long ○ Short

Symbols: clear (5) SPY, DIA, QQQ, IWM, TLT

Market 2: NONE

Market 3: NONE

Start Date: Tuesday , January 1, 2002

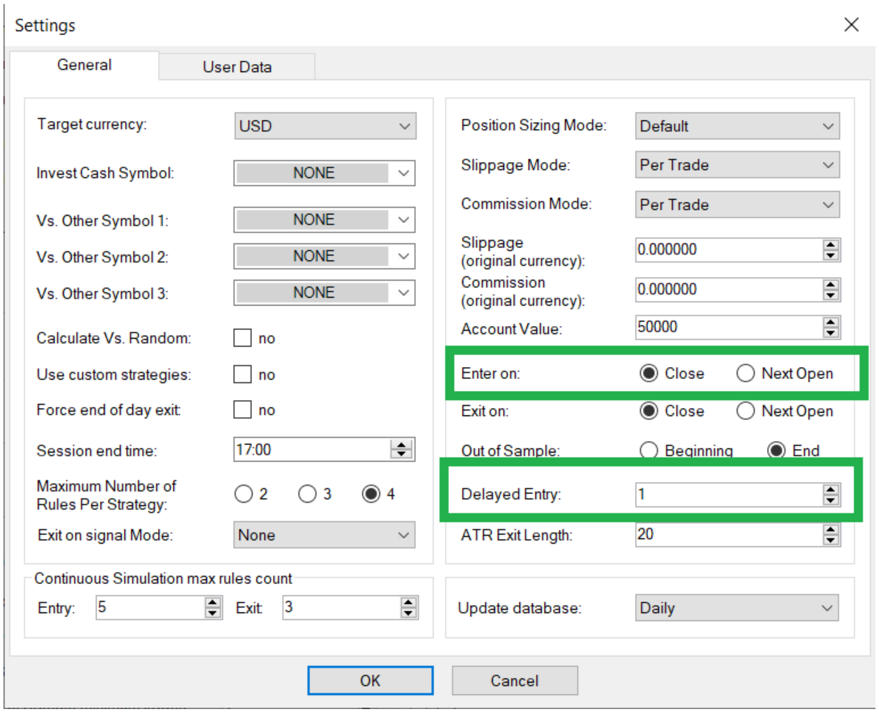
Filter: All

1 (0) + 0 selected

☑ Default
□ Day Of Week
□ TDOM
□ Week number
□ Day Of Month

Delay Entries

Some of you require a one (or more day) lag from when the signal occurs and when you can execute the order. You can automatically add this lag to the code by setting the delay to a value greater than 0 in Settings menu. An example of this would be... entry is true on Monday's close but because delay is set to 1 the entry will occur on Tuesday's close (please note this also means entry is set to 'this bar close'). If Enter On was set to Next Open and the condition for entry was True on Monday's close, then the entry would happen on Wednesday's open (assuming delayed entry set to 1).



Target currency: USD
Invest Cash Symbol: NONE
Vs. Other Symbol 1: NONE
Vs. Other Symbol 2: NONE
Vs. Other Symbol 3: NONE
Calculate Vs. Random: □ no
Use custom strategies: □ no
Force end of day exit: □ no
Session end time: 17:00
Maximum Number of Rules Per Strategy: ○ 2 ○ 3 ● 4
Exit on signal Mode: None
Continuous Simulation max rules count
Entry: 5 Exit: 3

Position Sizing Mode: Default
Slippage Mode: Per Trade
Commission Mode: Per Trade
Slippage (original currency): 0.000000
Commission (original currency): 0.000000
Account Value: 50000
Enter on: ● Close ○ Next Open
Exit on: ● Close ○ Next Open
Out of Sample: ○ Beginning ● End
Delayed Entry: 1
ATR Exit Length: 20
Update database: Daily

SIGNAL DESCRIPTIONS

Day of the Week

dayOfWeek(date) = 1 – translates – if the day of the week is Monday (the first day of the week).

dayOfWeek(date)<>4 – translates – if the day of the week is NOT Thursday (the fourth day of the week).

Signal Count: 10

Week Number

weekNumber = 1 – translates – it is the first trading week of the month. If Friday is the first of the month then this would only be true on that Friday. The following Monday would begin the second trading week of the month.

weekNumber <> 4 – translates – it is NOT the fourth trading week of the month.

Signal Count: 10

Day of Month

Dayofmonth(date) >= 20 – translates – the calendar day of the month is greater than or equal to 20. March 22 is true and May 19 is false.

Signal Count: 7

Time

Time signals use time format HHMMSS so 15:30:00 would be displayed as 153000. To set time equal to 9:30 you would do 93000.

To create a time signal like Time is less than 3 AM or greater than 7 AM you can create a custom comparative signal that pairs Time >= 00000 and Time <= 30000 and combine that with Time >= 70000 and Time <= 235900.

Signal Count: 2

Odd/Even Day

oddday = 1 – translates – today’s calendar day is an odd number. March 7 is true. August 8 is false.

Signal Count: 2

Month and Quarter

Month(date) = 1 – translates – January.

Month(date) > 3 and Month(date) <= 6 – translates – second quarter of the year.

Signal Count: 33

Highest and Lowest

Open >= highest(open,5) – translates The current bar’s open is greater than the previous 5 bar’s opening prices.

Signal Count: 49

OHLC - Open, High, Low, Close

open[0] > high[3] – translates – the current bar’s open is greater than the high of the bar three bars ago. The square bracketed number references the number of bars ago with 0 being the most recent bar.

There are over 1,400+ OHLC comparison signals to choose from. They reference as many as 10 bars ago.

Signal Count: 1,443

OHLC – Manipulations

open[0] < SqrtHL[0] – translates – the current bar’s open is less than the square root of the current bar’s high multiplied by the current bar’s low.

There are a variety of manipulations so all are listed.

HLMedian is the median price between the current bar’s high and current bar’s low.

BodyMedian is the median price between the current bar’s open and the current bar’s close.

HLC is the average price of the current bar’s high, low, and close.

CubeHLC is the cube root of the product of the current bar's high, low and close.

OHLC is the average price of the current bar's open, high, low, and close.

Signal Count: 85

OHLC vs. Simple Moving Average

open[0] > average(close,200) – translates – the current bar's open is greater than the simple moving average of the most recent 200 bars.

Most only check where the closing price is in relation to the moving average, but Build Alpha offers signals that check where the open, high, low, and close are in relation to multiple moving averages.

Build Alpha also includes signals comparing the simple moving average to the simple moving average calculated a few bars ago.

average(close,50)[0] > average(close,50)[2] – translates – the 50 period SMA is greater than the 50 period SMA calculated 2 bars ago.

Signal Count: 157

OHLC vs. Exponential Moving Average

See above but replace average with xaverage.

Signal Count: 157

Consecutive OHLC

consecutive(close,2,0) = 1 – translates – two consecutive lower closes. The first input into the "consecutive" function is the price you would like to check; in this case, we are using close. The second input to the "consecutive" function is the amount of consecutive occurrences you would like to check; in this case, we are looking for two consecutive closes. The third input to the "consecutive" function is 1 for higher and 0 for lower; in this case, we are checking for 2 consecutive lower closes.

consecutive(open,4,1) = 1 – translates – four consecutive higher opens.

Signal Count: 48

Donchian Breakouts/Channel Breakouts

high[0] >= highest(high,50)[0] – translates – this bar is the highest high of the last 50 bars. This bar has made a new 50 bar high.

low[0] <= lowest(low,10)[0] – translates – this bar is the lowest low of the last 10 bars. This bar has made a new 10 bar low.

Build Alpha offers comparisons for opens and closes, as well. Build Alpha also offers OHLC in both directions: highest and lowest.

Signal Count: 48

Percent Change

percentChange >= 2 – translates – this bar's percent change is greater than 2%. Percent Change is calculated close over close. That means, (current bar's close – previous bar's close)/previous bar's close.

Signal Count: 10

Base: 1 Max: 3 Step 0.5 would test 1%, 1.5%, 2.0%, 2.5%, 3.0%

Performance To Date

MTD, QTD and YTD are month to date, quarter to date and year to date performance (in percent) of the underlying asset.

MTD[0] < 5.0 – translates – the current bar's month to date performance is less than 5%

Signal Count: 102

Base: 1 Max: 3 Step 0.5 would test 1.0%, 1.5%, 2.0%, 2.5%, 3.0%

NR7 and WR7

range <= lowest(range,7) – translates – this bar's range is the lowest range over the last 7 bars. This would qualify as the popular "NR7" or narrow range 7 often talked about in technical analysis and candlestick literature.

range >= highest(range,4) – translates – this bar's range is the greatest range over the last 4 bars. This would qualify as a WR4 or wide range 4 bar that is often talked about in technical analysis and candlestick literature.

Signal Count: 10

Japanese Candlesticks

All candle sticks can be selected based on their respective name. For example, hammer, inverted hammer, doji.

InvertedHammer – both the open and the close are below the midpoint of the bar.

Doji – the open and close difference are less than or equal to 10% of the entire bar's range.

DownWick would be an open or close contained within the previous bar’s low and the minimum of the previous close and previous open. The current open is contained within the lower wick of the previous candle.

Signal Count: 20

Day or Bar Type

There are four major day types that all deal with the range comparisons. They are:

1. Inside range
2. Outside range
3. Up Range (high is above comparison high, low is above comparison low)
4. Down Range (high is below comparison high, low is below comparison low)

Each of these 4 range types can be paired with either a positive or negative close. This leaves us with 8 possible day types.

Future revisions will display day types as:

InsideRange(4,1) – translates – when comparing the current bar’s range to the bar 4 bars ago, the current bar’s range is inside the 4 bars ago’s range. The current bar also has a positive close when compared to the close 4 bars ago. The first input is for the lookback comparison and the second input is for positive close or negative close comparison.

UpRange(3,0) – translates – current bar’s high is above the high 3 bars ago and the current bar’s low is above the low from 3 bars ago. The second input of 0 signifies that current bar closed below the close from 3 bars ago.

Signal Count: 40

Bar Path

These signals represent the closing pattern over the past three bars. There can be 8 possible outcomes of how the last three bars closed in comparison to each bar’s previous bar.

barPath = 1 – translates – close[0] < close[1] and close[1] < close[2] and close[2] < close[3]

barPath = 2 – translates – close[0] < close[1] and close[1] < close[2] and close[2] > close[3]

barPath = 3 – translates – close[0] < close[1] and close[1] > close[2] and close[2] < close[3]

barPath = 4 – translates – close[0] < close[1] and close[1] > close[2] and close[2] > close[3]

barPath = 5 – translates – close[0] > close[1] and close[1] < close[2] and close[2] < close[3]

barPath = 6 – translates – close[0] > close[1] and close[1] < close[2] and close[2] > close[3]

barPath = 7 – translates – close[0] > close[1] and close[1] > close[2] and close[2] < close[3]

barPath = 8 – translates – close[0] > close[1] and close[1] > close[2] and close[2] > close[3]

Signal Count: 8

Pivot Points and R1 R2 S1 S2

Pivotpoint[0] > pivotpoint[3] – translates – the current bar's pivot point is greater than the pivot point calculated as of three bar's ago.

Pivot point is calculated as: (high + low + close) / 3.0

close[0] > R1 – translates – this bar's close is greater than the first resistance level.

Pivot Points, Support levels and Resistance levels have been calculated according to this standard: http://stockcharts.com/school/doku.php?id=chart_school:technical_indicators:pivot_points

Signal Count: 25

Inter Bar Rank or IBR

This is also known as interbar strength or IBS.

Ibr >= .85 – translates – this bar's close closed in the top 85% of this bar's range.

ibr <= .25 – translates – this bar's close closed in the bottom 25% of this bar's range.

Calculated (close – low) / range

Signal Count: 12

Please note the parametric signals are expressed as a %. So parametric IBR signals will translate as follows

IBR > 25 would mean IBR > 25%

Base: 25   Max: 85   Step: 15 would test 25,40,55,70,85

Average True Range

atr[0] > atr[2] – translates – the 20 period average true range calculated on this bar is greater than the 20 period average true range calculated two bars ago.

AverageTrueRange(10) > AverageTrueRange(50) – translates the 10 period average true range is greater than the 50 period average true range. Or in other words the short-term volatility is increasing compared to the longer term average.

These can be used to define if range or volatility are contracting/expanding.

Signal Count: 91

Volatility Breakouts

high[0] > close[1] + 1.5 * atr[1] – translates – this bar's high is greater than 1.5 average true range units than the previous bar's close. This would be a "breakout" of ATR volatility bands.

Signal Count: 16

AutoCorrelation

autocor(close,5,20) <= -.1 – translates – the autocorrelation between this bar's close and the close of 5 bar's ago calculated for the previous 20 bars is <= -0.1.

Signal Count: 2

Please note parametric Autocorrelation signals are expressed as percentages. So .1 is 10. Autocor(5,20) > 10 is the same as the built-in signal Autocor(5,20) > .1

Base: -10   Max: 30   Step: 10 would test -10, 0, 10, 20, 30

Bollinger Bands

close[0] >= BollingerBand(c,20,2) – translates – this bar's close closed above the 2 standard deviation Bollinger Band of the 20 period simple moving average.

BollingBand(c,20,2)[0] > BollingerBand(c,20,2)[3] – translates – this bar's upper Bollinger Band closed higher than the upper Bollinger Band calculated three bars ago.

Signal Count: 10

Commodity Channel Index CCI

close[0] < CCI(20)[0] – translates – this bar's close is less than the commodity channel index calculated over the previous 20 bars.

Signal Count: 10

Directional Movement Index

Signal Count: 4

Hurst Exponent

A measure of how mean reverting or trending a market has been over the last n bars.

Hurst(20)[0] > .65 – translates – the hurst exponent is greater than .65.

Hurst(20)[0] > Hurst(20)[2] – translates – the hurst exponent calculated this bar is greater than the hurst exponent calculated two bars ago.

Signal Count: 22

Please note parametric Hurst signals are expressed as percentages. The built-in signal Hurst(20)[0] > .65 would be the equivalent of the parametric Hurst(N)[0] > 65

Base: 35 Max: 65 Step: 15 would test 35, 50, 65

Keltner Channel

A measure of volatility surrounding most recent price action. Similar to the more widely used Bollinger Bands.

Close[0] cross above KeltnerChannel(c,20,1.5) – translates – the current bar's close crossed above its upper Keltner Channel calculated on the closing prices of the past 20 bars using 1.5 multiple for Keltner calculation.

Signal Count: 10

Kaufman Efficiency Ratio

Signal Count: 39

Please note parametric signals are expressed as a percentage. Kaufman(10)[0] > 0.50 is the equivalent of the parametric Kaufman(10)[0] > 50

Base: 10 Max: 50 Step: 10 would test 10, 20, 30, 40, 50

Moving Average Convergence Divergence MACD

Macd(c,12,26)[0] > 0 – translates – the MACD line is greater than 0.

Macdhist(c,12,26)[0] > macdhist(c,12,26)[4] – translates – the MACD histogram (MACD line – MACD signal line) calculated this bar is greater than the MACD histogram calculated four bars ago. A signal of increasing momentum.

Signal Count: 14

Relative Strength Index RSI

RSI contains filters and signals. For example,

Rsi(c,14) >= 90 – translates – this bar's 14 period RSI is greater than 90.

RSI(c,14) crosses above 90 – translates – this bar's 14 period RSI crossed above 90 (on this specific bar both of these prior examples will be true; however, this signal will not be true on the next bar as it "crossed over" 90 on this bar).

RSI(c,2)[0] <= RSI(c,2)[3] – translates – this bar's 2 period RSI is less than or equal to the 2 period RSI calculated 3 bars ago.

Signal Count: 139

Stochastics

Stochastics contains filters and signals. For example,

Stochastics(14) <= 20 – translates – the 14 period stochastics calculated on this bar is below 20.

Stochastics(14) crosses above 80 – translates – the 14 period stochastics calculated on this bar crosses above 80.

Signal Count: 58

Rate of Change ROC

Rate of Change is calculated as (close – close of lookback)/close of lookback.

rateOfChange(close,3) – translates – the difference of this bar's close minus the close of 3 bars ago divided by the close of 3 bars ago.

Signal Count: 36

Please note parametric signals are expressed as a percentage. RateOfChange(Close,3)[0] > 6 means greater than 6%

Momentum

See above. However, momentum is raw value not converted to percentage. Simply (Close – Close of Lookback).

Signal Count: 36

TDOM

Trading day of the month. For example, if September 3rd is a Monday then it would be the first trading day of the month. However, next year when September 3rd is a Tuesday then it would be the second trading day of the month.

TDOM[0] == 15 – translates – today is the fifteenth trading session of the month.

Signal Count: 23

TDLM

Trading days left in the month. For example, on September 30, 2019 there are no trading days left in the month and this function would return 0.

TDLM[0] == 4 – translates – there are 4 trading days left in the current month.

Signal Count: 24

Value Charts

These are an attempt to quantify price action profile. First, detrend the data by subtracting the rolling length median. Then dived by the product of 1 / length and rolling length range.

Open – average((h+l)/2,length) / (1/length * average(range,length))

This moves all values from a scale of -12 to +12 and closely represents a normal distribution. A negative extreme is 'oversold' and a positive extreme is 'overbought'.

Reference: https://www.dropbox.com/s/zr9nb0iystpk05v/Value_Chart_Overview.pdf?dl=0

Signal Count: 584

VIX

Vix[0] > 35 – translates – the vix's on this bar is greater than 35.

Vix[0] > Vix[2] – translates – the vix's close on this bar is greater than the vix's close of 2 bars ago.

Build Alpha will automatically match dates in the background. If you import custom data, Build Alpha will synchronize the Vix signals behind the scenes. If no dates match then no Vix signals will appear.

Signal Count: 44

Volume

Volume[0] >= average(volume,10) – translates – this bar's volume is greater than the average volume calculated over the previous 10 bars.

Volume[0] <= volume[4] – translates – the volume of this bar is less than or equal to the volume of four bars ago.

Signal Count: 13

Wins Last

WinsLast(20) <= 10 – translates – the number of up closes in the last 20 bars is equal to or less than 10.

This can be used to identify periods where the market may be "overdue", but a reminder about gambler's fallacy.

Signal Count: 49

Composites

Composite signals are simply averages of an indicator over a variety of lookbacks.

CompositeRSI(2,24) – translates – the average RSI over RSI2 + RSI3 + RSI4 + … RSI23 + RSI24 / (24 – 2 + 1)

Signal Count: 152

Super Smoother

A zero lag (less lag) exponential moving average.

Signal Count: 19

Economic Numbers

You can test around or on the date of the economic release. For example, today is a GDP release date then buy. You can also test the numeric value of the release. GDP > 2.5% then buy. Quite a bit of flexibility here, but I have made notes about the 'scale' of the numbers in each release as many have different units.

Most are in the scale they're reported in. For example, Treasury auctions are quoted in percent. GDP is also quoted in percent. So is Durable Goods, PPI, CPI, FOMC, Redbook, Industrial Production, Export Price Index, Factory Orders

Jobless Claims are quoted in the thousand. 282K is displayed as 282 in our data. The historic print on March 26, 2020 of 3283K or 3,283,000 is quoted as 3283

Nonfarm Payrolls is also reported in thousands.

Housing Starts, Existing Home Sales, EIA Petroleum Inventories and JOLTS job openings are quoted in millions. 1.599M or 1,599,000 is quoted as 1.599 in our data

International Trade (Trade Balance) is quoted in billions. -48.60B would be -48.60

Consumer Confidence, Consumer Sentiment, ISM, Empire State Mfg Survey, Chicago PMI are all in their raw values as reported.

Market Breadth

Percentage of Stocks Up – this is the number of SP500 stocks expressed as a percent that closed higher each day (0 to 100)

Percentage of Stocks Down – this is the number of SP500 stocks expressed as a percent that closed lower each day (0 to 100)

NYSE Adv-Decl is the total number of advancing NYSE issues or declining issues expressed as a percentage of total number of NYSE issues (0 to 100). ADD 74% means 74% of stocks are advancing

TRIN – market breadth signal ranging from 0 to 3 https://www.investopedia.com/terms/a/arms.asp

TICK – market breadth signal ranging from -2000 to 2000 https://www.investopedia.com/terms/t/tickindex.asp

No. 52Wk Highs – count of the stocks making a 52 week high each day (values 0 to 100+)

No. 52Wk Lows – count of the stocks making a 52 week low each day (values 0 to 100+)

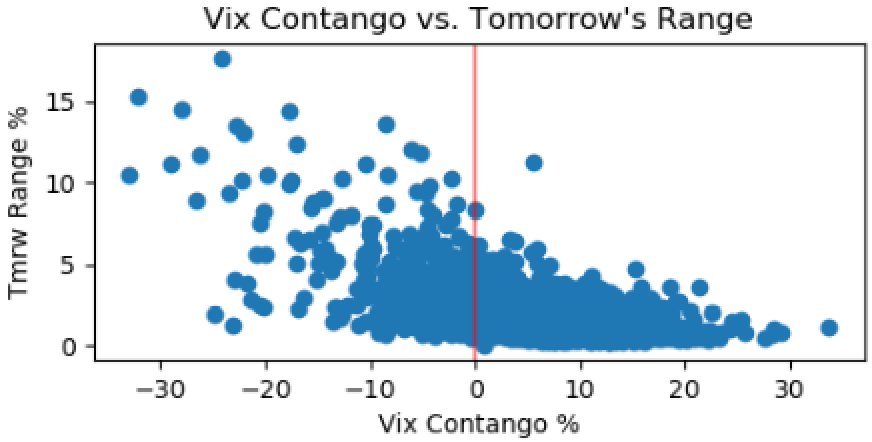
Percent SP500 Abv XXMA – Percentage of stocks in the SP500 above their XX moving average (0 to 100)

Percent SP500 Bel XXMA – Percentage of stocks in the SP500 below their XX moving average (0 to 100)

Vix Term Structure

VX1 refers to the front month Vix futures contract. VX4 refers to the fourth month Vix futures contract.

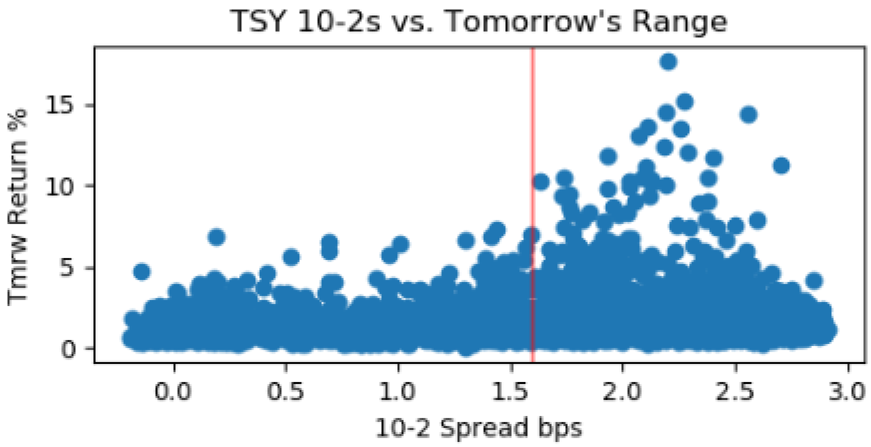
Contango1-2 is the percent difference between the front month and the second month vix futures contract. Contango1-2 of -10 means the second contract is more than 10% below the first contract. You can see a Vix in contango of 10% starts to show significant more volatility in the SP500 in the next day's trading. These signals can be great filters for volatility regimes but do your own testing!



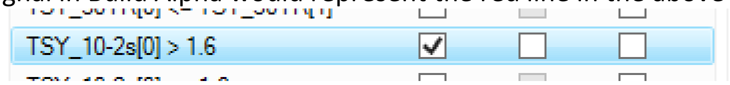
Treasury Yields

TSY_1MO means the yield on the 1-month duration US treasury bill.

TSY_10-2s means the 10YR yield minus the 2YR yield. It is a popular quantified approach to summarize the yield curve. It too can signal different volatility regimes. This value is also expressed as a percent.


TSY 10-2s vs. Tomorrow's Range

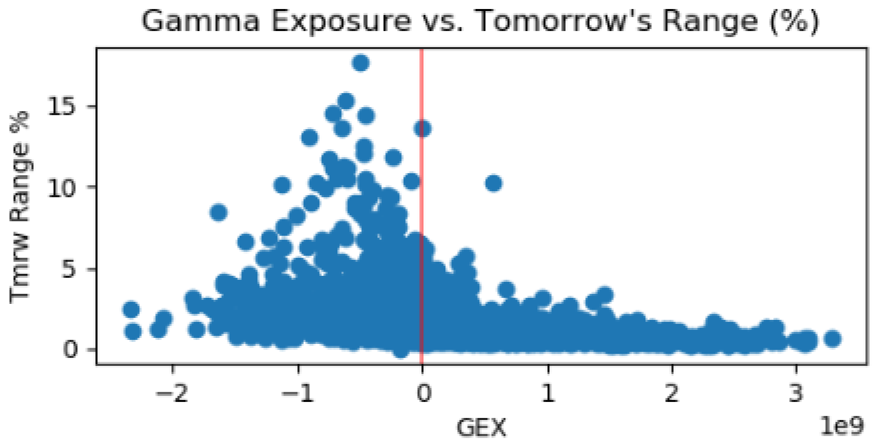
This signal in Build Alpha would represent the red line in the above graph


TSY_10-2s[0] > 1.6 ☑ □ □

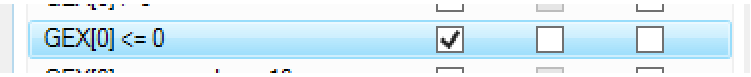
Option Data

GEX is Gamma Exposure of SPX Index. GEX500 is the Gamma Exposure of the ~500 SPX components. To read more gamma exposure and market maker positioning please check this white paper here: https://squeezemetrics.com/download/white_paper.pdf

And this thread: https://www.reddit.com/r/wallstreetbets/comments/e6ptjr/understanding_gamma_exposure_in_the_market_an/


Gamma Exposure vs. Tomorrow's Range (%)

This would test the red line in Build Alpha, e.g.



GEX[0] <= 0 ☑ □ □

These values range from -10,000,000,000 to +10,000,000,000

DIX

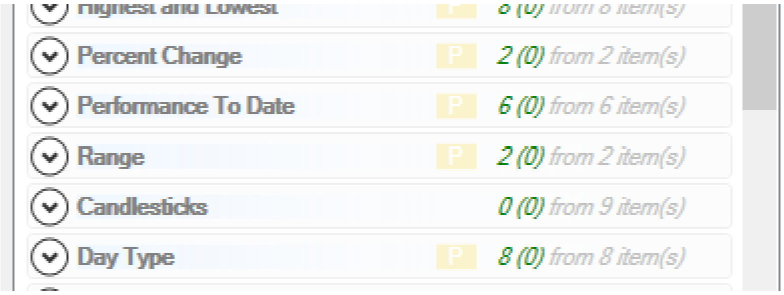
DIX is Dark Pool Index and is another metric taken from the aggregated stock issues. The Dark Pool Index aggregates the short sale volume on all 500 S&P stocks. This is a dollar weighted ratio of dark pool buying to dark pool selling.

Values range from 30% to ~60%

DIX > 35 would mean DIX > 35%

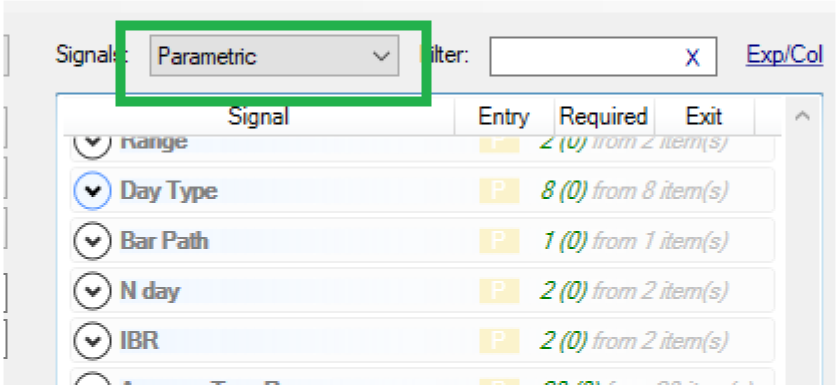
Changing Parameters

Signals with a ‘P’ next to them indicate they are parametric signals and can be adjusted to the user’s considerations. You can also access all parametric signals by changing the Signals Filters Dropdown.



Highest and Lowest	P	8 (0) from 8 item(s)
Percent Change	P	2 (0) from 2 item(s)
Performance To Date	P	6 (0) from 6 item(s)
Range	P	2 (0) from 2 item(s)
Candlesticks		0 (0) from 9 item(s)
Day Type	P	8 (0) from 8 item(s)

Candlesticks are non-parametric and do not have the P



Signals: Parametric

Range	P	2 (0) from 2 item(s)
Day Type	P	8 (0) from 8 item(s)
Bar Path	P	1 (0) from 1 item(s)
N day	P	2 (0) from 2 item(s)
IBR	P	2 (0) from 2 item(s)

To change parameters, please click and expand a group. Click on the signal of choice. Please note the following syntax.

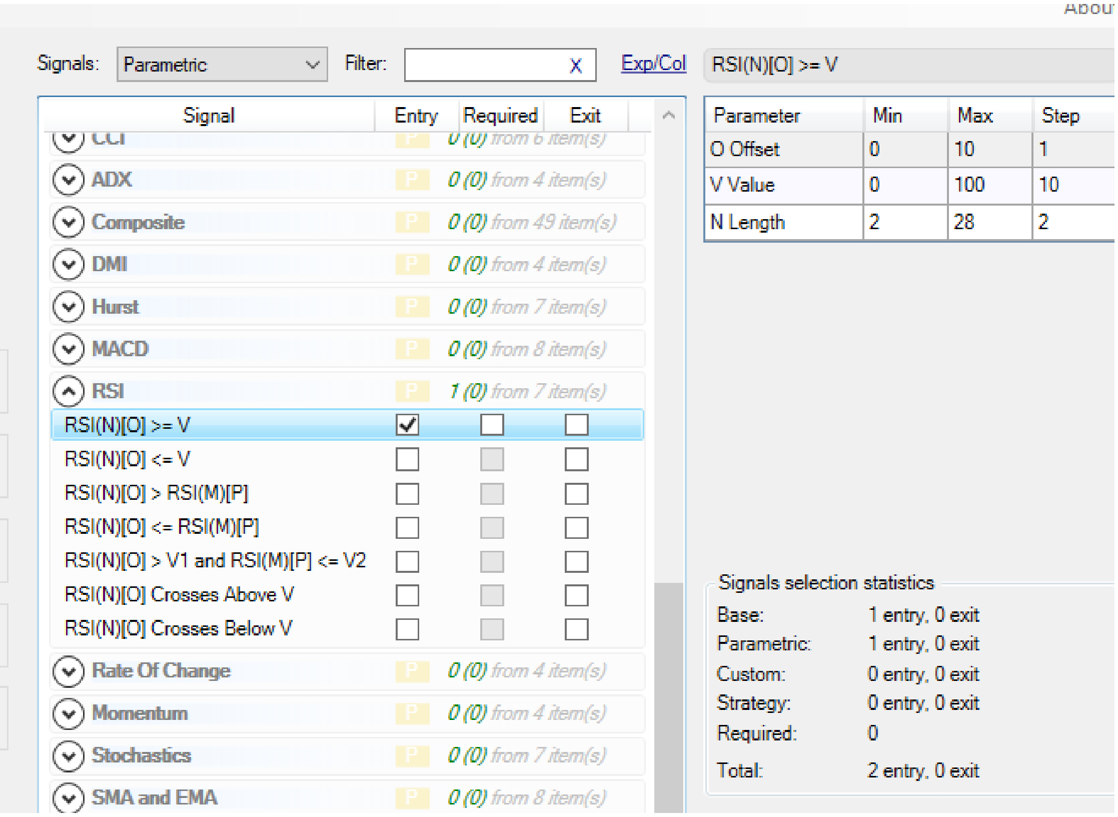
N = length

V = value

O = offset

RSI(N)[O] > V   can also be read as RSI(2)[0] > 75 which in plain English is the current bar’s two period RSI is greater than 75.

To create permutations of this rule here is how we can set it up in Build Alpha’s main GUI



RSI(N)[O] >= V

Signal	Entry	Required	Exit
CCI	P	0 (0) from 6 item(s)	
ADX	P	0 (0) from 4 item(s)	
Composite	P	0 (0) from 49 item(s)	
DMI	P	0 (0) from 4 item(s)	
Hurst	P	0 (0) from 7 item(s)	
MACD	P	0 (0) from 8 item(s)	
RSI	P	1 (0) from 7 item(s)	
RSI(N)[O] >= V	☑	□	□
RSI(N)[O] <= V	□	□	□
RSI(N)[O] > RSI(M)[P]	□	□	□
RSI(N)[O] <= RSI(M)[P]	□	□	□
RSI(N)[O] > V1 and RSI(M)[P] <= V2	□	□	□
RSI(N)[O] Crosses Above V	□	□	□
RSI(N)[O] Crosses Below V	□	□	□
Rate Of Change	P	0 (0) from 4 item(s)	
Momentum	P	0 (0) from 4 item(s)	
Stochastics	P	0 (0) from 7 item(s)	
SMA and EMA	P	0 (0) from 8 item(s)	

Parameter	Min	Max	Step
O Offset	0	10	1
V Value	0	100	10
N Length	2	28	2

Signals selection statistics

Base:	1 entry, 0 exit
Parametric:	1 entry, 0 exit
Custom:	0 entry, 0 exit
Strategy:	0 entry, 0 exit
Required:	0
Total:	2 entry, 0 exit

Notice the base rule in the top right of the image. We have selected (via check mark) this base rule as an entry. We have created Offset values of 0,1,2,3,4,5,6,7,8,9,10.

We have created values of 0,10,20,30,40,50,60,70,80,90,100 as we have set V Value to 0 to 100 by 10.

We have created N Length of 2,4,6,8,10,12,14,16,18,20,22,24,26,28 as we have set 2 to 28 by 2.

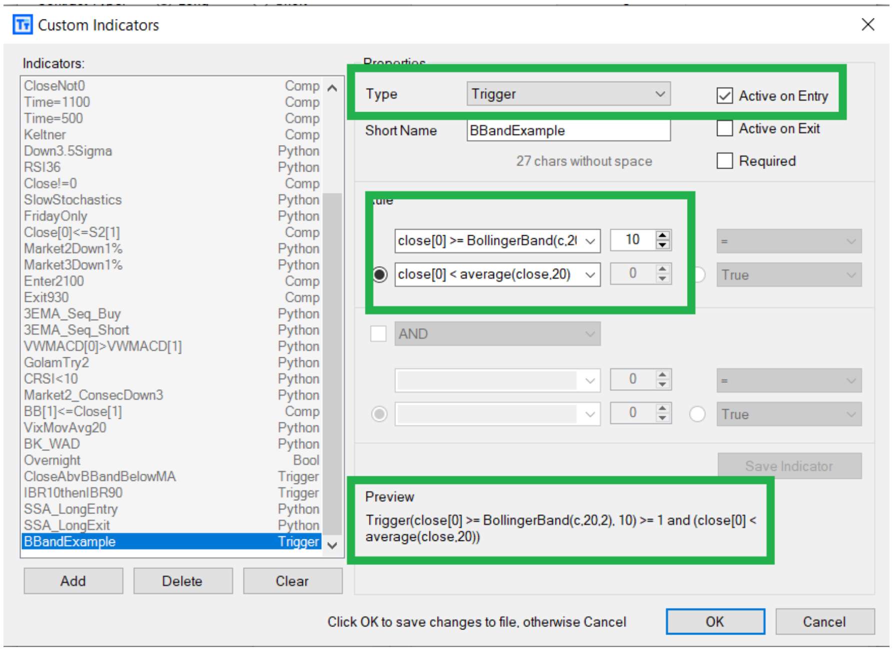
Even though Build Alpha only has 1 parametric entry selected in the Signals selection statistics please note that when you hit simulate Build Alpha will be able to create ALL the possible combinations of your inputs of O,V and N.

Custom Signals

To create custom signals, you can hit File->Custom Indicators or hit F4 from the main interface. This will bring up a new window in which you can create from drop down lists or using Python. For Python signals, please visit buildalpha.com/python or watch the related video at the video training page.

You can create Boolean (true false signals), comparative signals, cross over signals or 'trigger' signals.

A trigger signal is a combination of two signals that occur within N bars of each other. A simple example is we close above the upper Bollinger Band and then within 10 days we close below the middle moving average of the Bollinger Band. To create this signal and a visual example you could do…



Custom Indicators

Indicators:

CloseNot0	Comp
Time=1100	Comp
Time=500	Comp
Keltner	Comp
Down3.5Sigma	Python
RSI36	Python
Close!=0	Comp
SlowStochastics	Python
FridayOnly	Python
Close[0]<=S2[1]	Comp
Market2Down1%	Python
Market3Down1%	Python
Enter2100	Comp
Exit930	Comp
3EMA_Seq_Buy	Python
3EMA_Seq_Short	Python
VWMACD[0]>VWMACD[1]	Python
GolamTry2	Python
CRSI<10	Python
Market2_ConsecDown3	Python
BB[1]<=Close[1]	Comp
VixMovAvg20	Python
BK_WAD	Python
Overnight	Bool
CloseAbvBBandBelowMA	Trigger
IBR10thenIBR90	Trigger
SSA_LongEntry	Python
SSA_LongExit	Python
BBandExample	Trigger

Properties

Type: Trigger
☑ Active on Entry
Short Name: BBandExample
☐ Active on Exit
27 chars without space
☐ Required

Rule

close[0] >= BollingerBand(c,2(— 10 — =
● close[0] < average(close,20) — 0 — ○ True

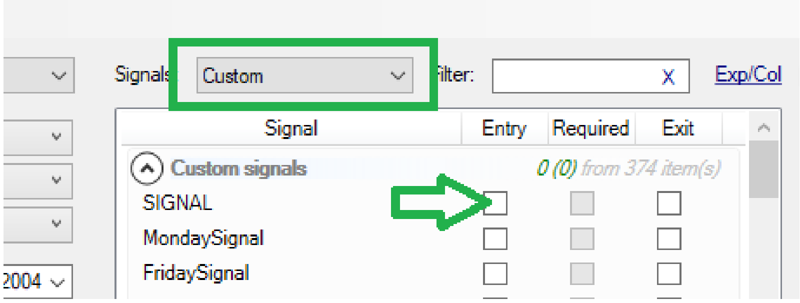
☐ AND

— 0 — =
○ — 0 — ○ True

Preview
Trigger(close[0] >= BollingerBand(c,20,2), 10) >= 1 and (close[0] < average(close,20))

Edit to Custom Signals

You can now create custom signals in the same way but control their active-ness from the main GUI. Please use the dropdown menu and select 'Custom'. Then you can turn any custom indicator on as an entry, require it or as an exit.



Inter Market Signals

Similar to Vix, intermarket signals allow the user to select a second and third market to help find profitable trading opportunities. Intermarket signals will calculate OHLC comparisons, OHLC vs SMA, OHLC vs EMA, and RSI for secondary and tertiary markets. For example, intermarket signals can allow the creation of a system that says if the S&P500 is below its 200 period SMA then buy Gold if Gold's volume is above its average 10 bar volume.

Build Alpha always uses $Vix.x as data2 in TradeStation Easy Language code. Build Alpha always displays Easy Language code so market2 symbols will show 'data3' and market3 symbols will show 'data4'. For further information please read the next section.

Importing Code into TradeStation and Multicharts:

Build Alpha always includes $Vix.x (Volatility Index) as the second data stream. When exporting strategies using intermarket signals please always make $Vix.x the second data stream on your MultiCharts/TradeStation charts. The symbol selected as Market 2 on Build Alpha's input screen will need to be data stream 3 in MultiCharts/TradeStation. The symbol selected as Market 3 on Build Alpha's input screen will need to be data stream 4 in MultiCharts/TradeStation. NinjaTrader/MT4 do this automatically in the exportable code. Thanks.

The Quantopian Python code uses slightly different indexing. For example, in Build Alpha a signal like Close[0] < Close[1] signifies the current bar's close is less than the previous bar's close. Quantopian/Python takes the value inside the square brackets and multiplies it by negative one and then subtracts one from it. The conversion would be C[(0 * -1) -1] < C[(1*-1)-1] or simply translated to C[-1] < C[-2]. Build Alpha's automatic code generator takes care of this conversion for you, but it is important I point it out before you think the generate code feature is not working properly!

Note: TradeStation currently does not like our weeknumber of CompositeRSI functions. I do not want to make an adjustment to our code generator just for these two functions. If you have a strategy -that generates a signal that uses either o these functions MORE THAN ONCE then please edit the code to this after copying to TradeStation.

Condition1 = weeknumber <> 3 and weeknumber <> 1 and close[0] > close[2];

-TO-

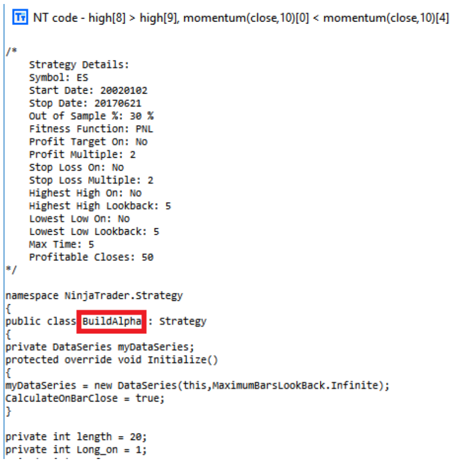
Value55 = weeknumber;

Condition1 = value55 <> 3 and value55 <> 1 and close[0] > close[2];

We are trying to avoid calling these two functions twice in the same code until we can find a better solution.

Import Build Alpha Strategies into Ninja Trader

1. If you plan on using the exportable code Build Alpha generates for NinjaTrader7 or NinjaTrader8 then please realize that every single strategy Build Alpha produces for NinjaTrader will be named “Build Alpha”. When you create a new strategy inside NinjaTrader you will give it a name of your choosing. You must edit the Build Alpha generated code by replacing the words “Build Alpha” with the name you chose. For example, if I create a new strategy in NinjaTrader and name it StrategyNo1 then I would replace Build Alpha in the code below and write StrategyNo1 before compiling it.



NT code - high[8] > high[9], momentum(close,10)[0] < momentum(close,10)[4]

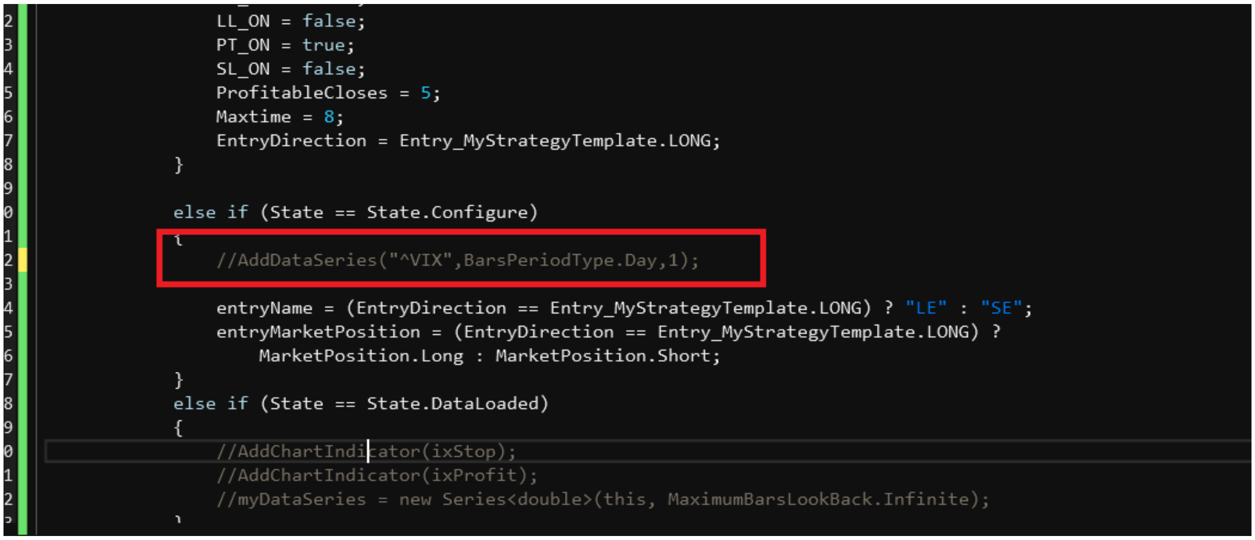
```
/*
    Strategy Details:
    Symbol: ES
    Start Date: 20020102
    Stop Date: 20170621
    Out of Sample %: 30 %
    Fitness Function: PNL
    Profit Target On: No
    Profit Multiple: 2
    Stop Loss On: No
    Stop Loss Multiple: 2
    Highest High On: No
    Highest High Lookback: 5
    Lowest Low On: No
    Lowest Low Lookback: 5
    Max Time: 5
    Profitable Closes: 50
*/

namespace NinjaTrader.Strategy
{
public class BuildAlpha : Strategy
{
private DataSeries myDataSeries;
protected override void Initialize()
{
myDataSeries = new DataSeries(this,MaximumBarsLookBack.Infinite);
CalculateOnBarClose = true;
}

private int length = 20;
private int Long_on = 1;
```

2. Intermarket strategies in NinjaTrader require us to add one line of code for the markets used into the code. I can not do this through Build Alpha because some providers have different symbols for the same markets. For example, if we create a strategy inside Build Alpha that traded GLD and used SPY as Build Alpha’s market 2 we would generate NT8 (or 7) code that had this line:

// AddDataSeries(“^Vix”,BarsPeriodType.Day,1);



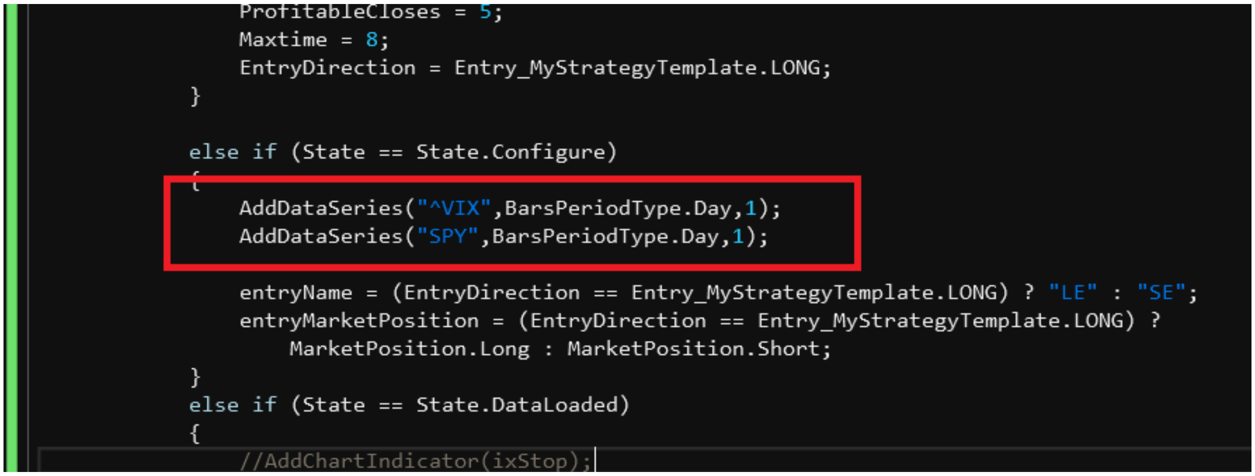
```
            LL_ON = false;
            PT_ON = true;
            SL_ON = false;
            ProfitableCloses = 5;
            Maxtime = 8;
            EntryDirection = Entry_MyStrategyTemplate.LONG;
        }

        else if (State == State.Configure)
        {
            //AddDataSeries("^VIX",BarsPeriodType.Day,1);

            entryName = (EntryDirection == Entry_MyStrategyTemplate.LONG) ? "LE" : "SE";
            entryMarketPosition = (EntryDirection == Entry_MyStrategyTemplate.LONG) ?
                MarketPosition.Long : MarketPosition.Short;
        }
        else if (State == State.DataLoaded)
        {
            //AddChartIndicator(ixStop);
            //AddChartIndicator(ixProfit);
            //myDataSeries = new Series<double>(this, MaximumBarsLookBack.Infinite);
```

We will need to remove the two forward slashes at the beginning of the line. Then add a new line directly below it for our second (and if necessary third) markets from Build Alpha. Even if your strategy does not have any Vix signals still include this line at the top! Below is the finished code for adding SPY daily data.

AddDataSeries(“^VIX”,BarsPeriodType.Day,1);
AddDataSeries(“SPY”,BarsPeriodType.Day,1);



```
            ProfitableCloses = 5;
            Maxtime = 8;
            EntryDirection = Entry_MyStrategyTemplate.LONG;
        }

        else if (State == State.Configure)
        {
            AddDataSeries("^VIX",BarsPeriodType.Day,1);
            AddDataSeries("SPY",BarsPeriodType.Day,1);

            entryName = (EntryDirection == Entry_MyStrategyTemplate.LONG) ? "LE" : "SE";
            entryMarketPosition = (EntryDirection == Entry_MyStrategyTemplate.LONG) ?
                MarketPosition.Long : MarketPosition.Short;
        }
        else if (State == State.DataLoaded)
        {
            //AddChartIndicator(ixStop);
```

If you cannot find this line in your code please hit ctrl+f to bring up the ‘Find’ feature and type in ‘Vix’ then hit enter and Ninjatrader will move your cursor to the correct line.

Long story short, make sure the symbols are correct for your data provider if using intermarket signals in a strategy.

Import Build Alpha Strategies into MetaTrader4

Please put both the include files in the Include folder. The include files were linked in the install email.

This pertains to intermarket strategies that use market's 2 and/or 3. There is a line of code in the generated MT4 code produced by Build Alpha that has the strategy's performance and other information. It will look similar to this:

```
"H4<O9 L>L3 Nop Nop ENDENTRY ENDEXIT 25635.972 3561.993 137 0.606 618.964 476.630 568.502 481.943
1.996 1.299 1.570 0 0 0 0 0 2.000 2.000 5 50 1 GBPUSD 1440 0000 2359 1700 9999999 -9999999 999999"
```

The section highlighted and bolded in red is Build Alpha's market 2 and the timeframe it should be. 1440 is used for daily. If you use custom data or it says GBPUSD 0, for example, then please change 0 to 60 if you used 60 minute charts or change the 0 to 30 if you used 30 minute charts, etc.

If you use both market 2 and market 3 then please change both. Here is an example below:

```
"O>O3 sprd1hlcHLC3 Nop Nop 24959.989 2367.986 130 0.577 667.066 455.817 646.855 384.918 1.996 1.463
1.685 1.77640097421236E-14 0 0 0 0 2.000 2.000 5 50 1 GBPUSD 0 EURAUD 0 0000 2359 1700 9999999 -
9999999 999999"
```

For example, let's assume this strategy was built using GBPUSD 60 minute and EURAUD 120 minute data so please change the 0's in this line of the generated code to 60 and 120 respectively when you copy and paste it into MT4.

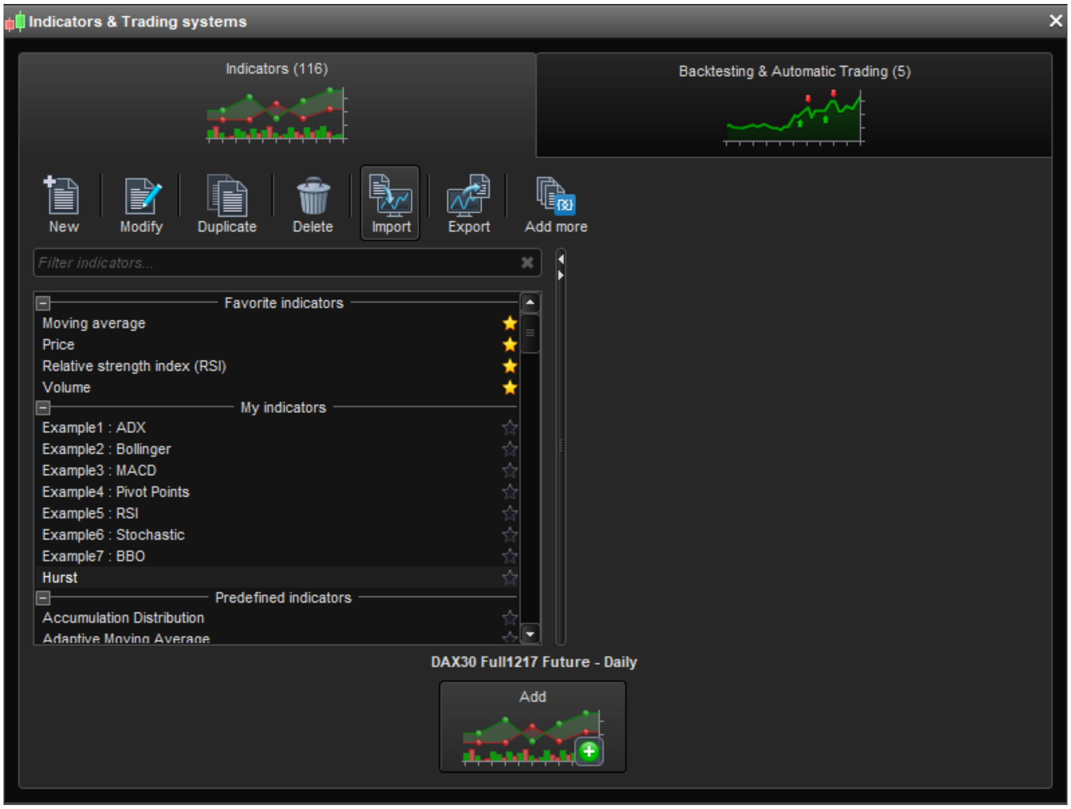
Import Build Alpha Strategies into Pro Real Time

There are several custom functions that need to be imported into Pro Real Time in order to make some of the Build Alpha signals work inside the PRT platform. If the strategy you've created uses one of these signals, the Build Alpha code will display the signal as "myHurst", for example. The "my" in the PRT code in front of the signal name signifies you must import the strategy AND add a function call to the code.

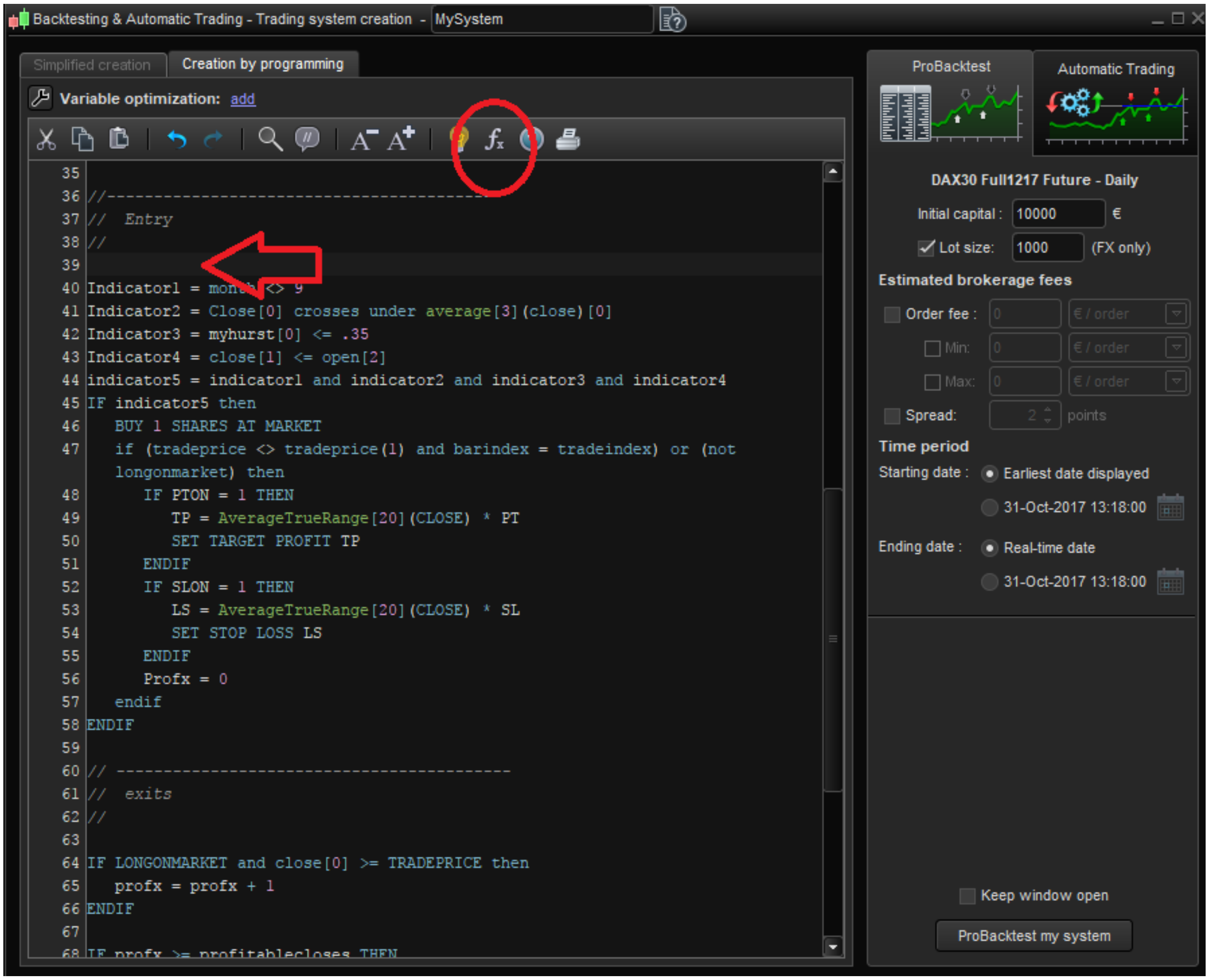
Here is an example of the Build Alpha generated Pro Real Time code that uses a custom signal called myHurst.

```
Indicator1 = month <> 9
Indicator2 = Close[0] crosses under average[3](close)[0]
Indicator3 = myhurst[0] <= .35
Indicator4 = close[1] <= open[2]
indicator5 = indicator1 and indicator2 and indicator3 and indicator4
```

Now we must import our custom hurst function first. To do so open the code editor and select import -> desired function.



Once that is finished, we can copy and paste our Build Alpha code into a new strategy in PRT. Once we've done so we need to add our "function CALL method" to make this custom ("my") function work. Click above indicator one so your cursor is above indicator one. Then hit the F(x) button above…



Backtesting & Automatic Trading - Trading system creation - MySystem
Simplified creation
Creation by programming
Variable optimization: add
ProBacktest
Automatic Trading

```
//------------------------------------------
//  Entry
//

Indicator1 = month <> 9
Indicator2 = Close[0] crosses under average[3](close)[0]
Indicator3 = myhurst[0] <= .35
Indicator4 = close[1] <= open[2]
indicator5 = indicator1 and indicator2 and indicator3 and indicator4
IF indicator5 then
   BUY 1 SHARES AT MARKET
   if (tradeprice <> tradeprice(1) and barindex = tradeindex) or (not
   longonmarket) then
      IF PTON = 1 THEN
         TP = AverageTrueRange[20](CLOSE) * PT
         SET TARGET PROFIT TP
      ENDIF
      IF SLON = 1 THEN
         LS = AverageTrueRange[20](CLOSE) * SL
         SET STOP LOSS LS
      ENDIF
      Profx = 0
   endif
ENDIF

// ------------------------------------------
//  exits
//

IF LONGONMARKET and close[0] >= TRADEPRICE then
   profx = profx + 1
ENDIF

IF profx >= profitablecloses THEN
```

DAX30 Full1217 Future - Daily

Initial capital : 10000 €

☑ Lot size: 1000 (FX only)

Estimated brokerage fees

☐ Order fee : 0 € / order

☐ Min: 0 € / order

☐ Max: 0 € / order

☐ Spread: 2 points

Time period

Starting date : ● Earliest date displayed

○ 31-Oct-2017 13:18:00

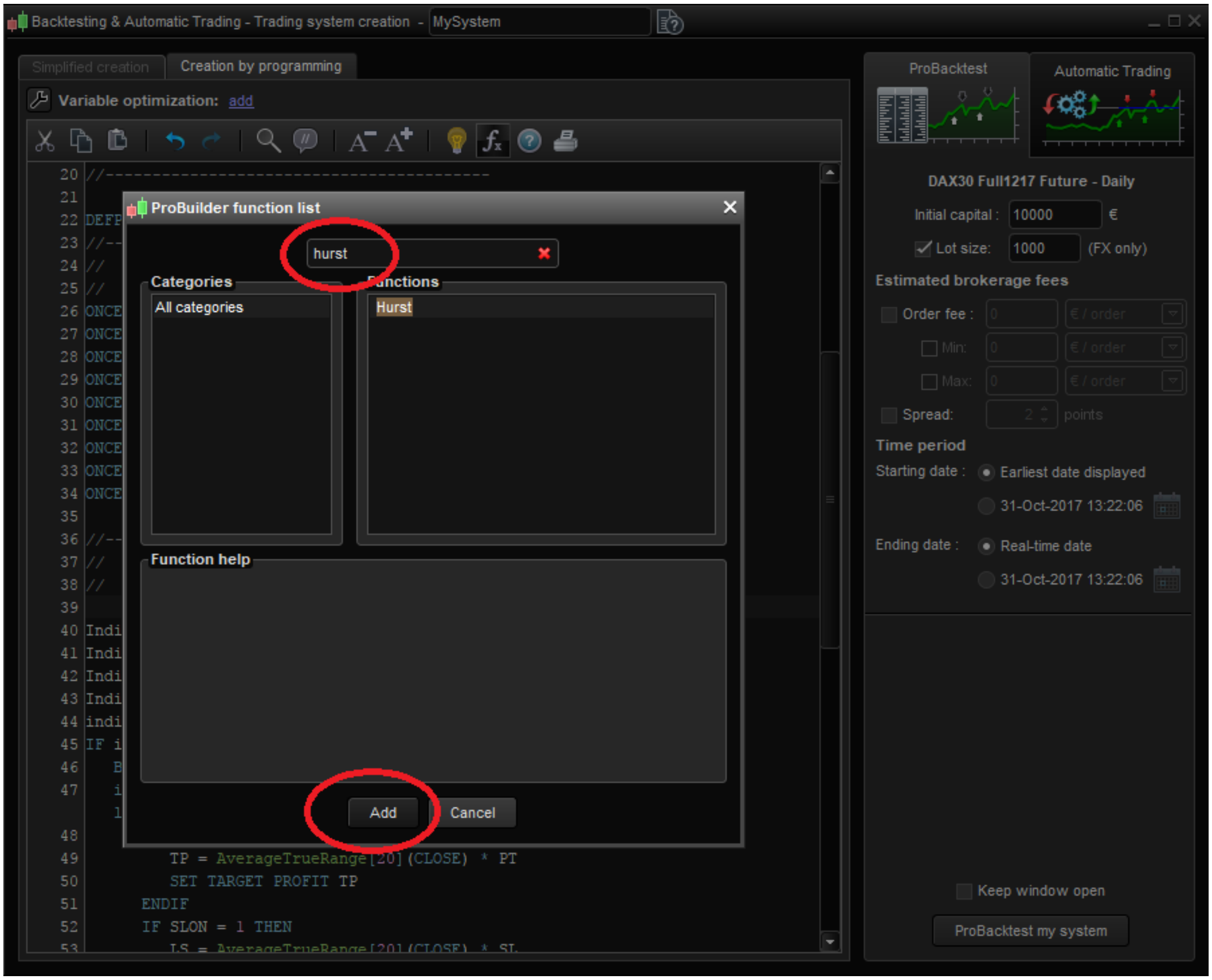
Ending date : ● Real-time date

○ 31-Oct-2017 13:18:00

☐ Keep window open

ProBacktest my system

Then type in the function you need. It should be whatever the displayed signal is minus the “my”. In this case, myhurst becomes hurst.



hurst

All categories

Hurst

```
DEFP
ONCE
Indi
indi
IF i
    TP = AverageTrueRange[20](CLOSE) * PT
    SET TARGET PROFIT TP
  ENDIF
  IF SLON = 1 THEN
    LS = AverageTrueRange[20](CLOSE) * SL
```

DAX30 Full1217 Future - Daily

Initial capital : 10000 €

☑ Lot size: 1000 (FX only)

Estimated brokerage fees

□ Order fee : 0 € / order

□ Min: 0 € / order

□ Max: 0 € / order

□ Spread: 2 points

Time period

Starting date : ● Earliest date displayed

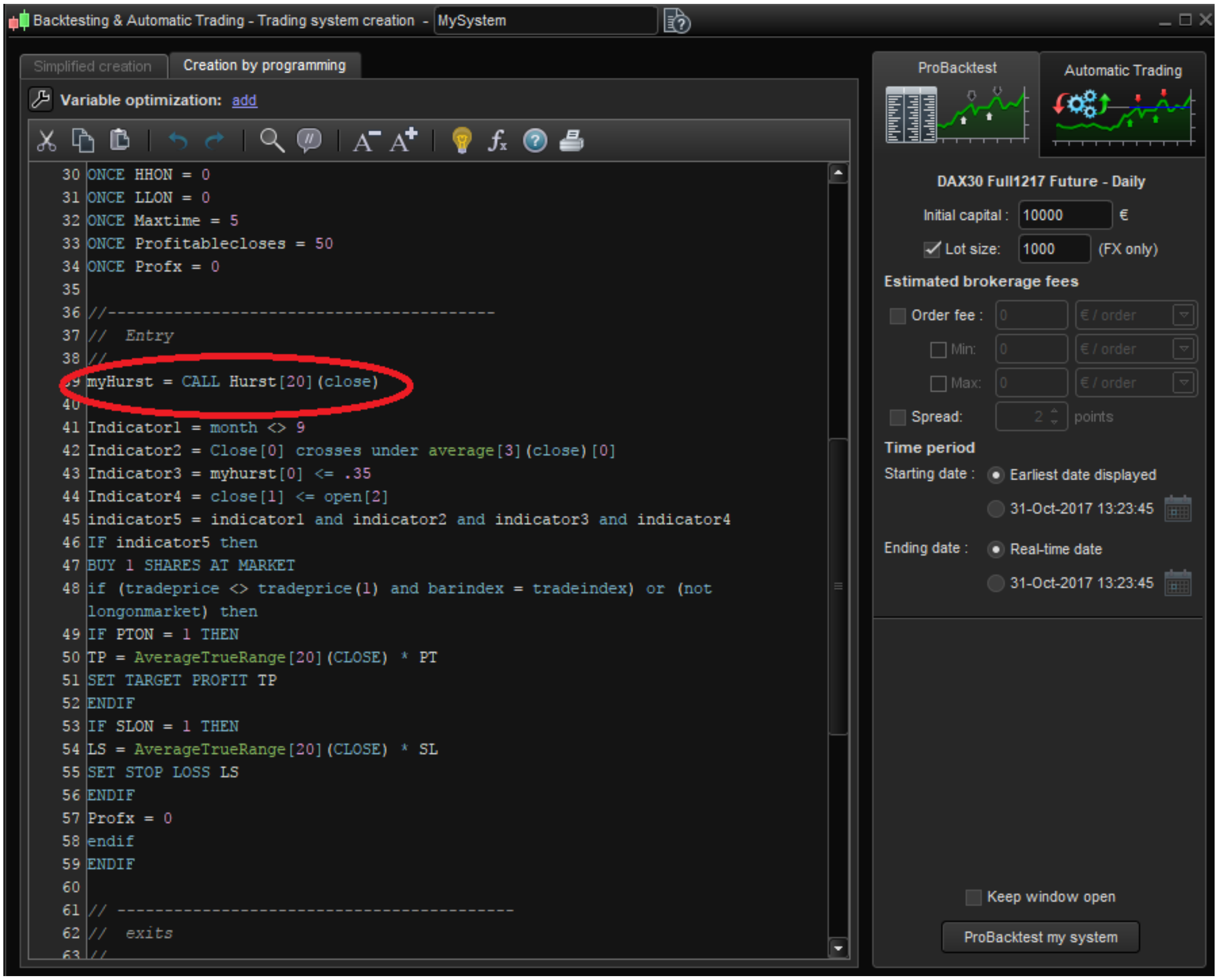
○ 31-Oct-2017 13:22:06

Ending date : ● Real-time date

○ 31-Oct-2017 13:22:06

□ Keep window open

Finally, PRT will automatically add this line of code to our Build Alpha code when we select the "Add" button. It will look like this in the code afterwards – and at this point we're good to hit the backtest button in the bottom right.


Backtesting & Automatic Trading - Trading system creation - MySystem
Simplified creation
Creation by programming
Variable optimization: add
ProBacktest
Automatic Trading
ProBacktest my system

```
ONCE HHON = 0
ONCE LLON = 0
ONCE Maxtime = 5
ONCE Profitablecloses = 50
ONCE Profx = 0

//-------------------------------------------
//  Entry
//
myHurst = CALL Hurst[20](close)

Indicator1 = month <> 9
Indicator2 = Close[0] crosses under average[3](close)[0]
Indicator3 = myhurst[0] <= .35
Indicator4 = close[1] <= open[2]
indicator5 = indicator1 and indicator2 and indicator3 and indicator4
IF indicator5 then
BUY 1 SHARES AT MARKET
if (tradeprice <> tradeprice(1) and barindex = tradeindex) or (not longonmarket) then
IF PTON = 1 THEN
TP = AverageTrueRange[20](CLOSE) * PT
SET TARGET PROFIT TP
ENDIF
IF SLON = 1 THEN
LS = AverageTrueRange[20](CLOSE) * SL
SET STOP LOSS LS
ENDIF
Profx = 0
endif
ENDIF

// ------------------------------------------
//  exits
//
```

DAX30 Full1217 Future - Daily

Initial capital : 10000 €

☑ Lot size: 1000 (FX only)

Estimated brokerage fees

□ Order fee : 0 € / order

□ Min: 0 € / order

□ Max: 0 € / order

□ Spread: 2 points

Time period

Starting date : ● Earliest date displayed
○ 31-Oct-2017 13:23:45

Ending date : ● Real-time date
○ 31-Oct-2017 13:23:45

□ Keep window open

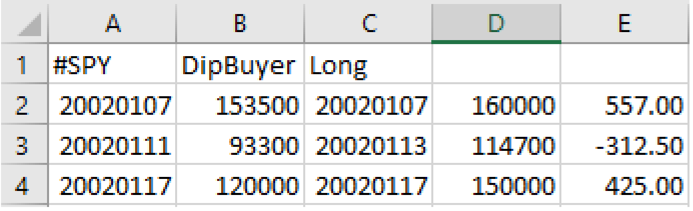
Importing Custom Strategies

Build Alpha allows the importation of strategies built outside of Build Alpha for analysis inside Build Alpha. You can also add your imported custom strategies to portfolios to test on portfolios containing non-BA and BA strategies.

To create a custom strategy import file please use the following format.

#BASymbol, Strategyname, Long
Yyyymmdd,hhmmss,yyyymmdd,hhmmss,result
Yyyymmdd,hhmmss,yyyymmdd,hhmmss,result
Yyyymmdd,hhmmss,yyyymmdd,hhmmss,result

The first date and time are the entry timestamp. The second are the exit timestamp. Here is an image of a correctly formatted file.

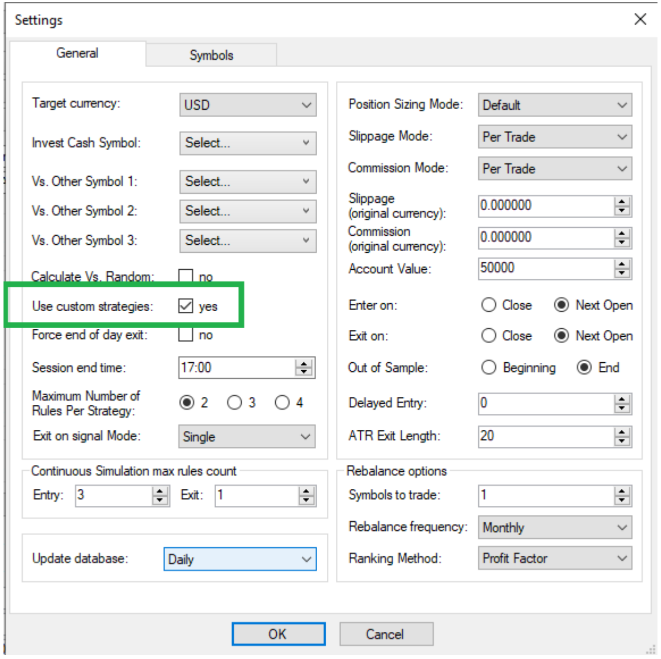


	A	B	C	D	E
1	#SPY	DipBuyer	Long		
2	20020107	153500	20020107	160000	557.00
3	20020111	93300	20020113	114700	-312.50
4	20020117	120000	20020117	150000	425.00

This would read in whenever a Long SPY simulation is run

Strategyname is arbitrary and what will show in the BA results window. Please avoid spaces and special characters.

Then make sure the Use Custom Strategy setting in the Services->Settings menu is set to Yes.

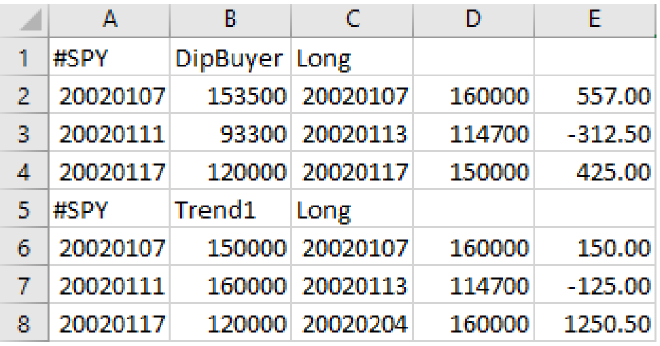


Target currency: USD
Invest Cash Symbol: Select...
Vs. Other Symbol 1: Select...
Vs. Other Symbol 2: Select...
Vs. Other Symbol 3: Select...
Calculate Vs. Random: □ no
Use custom strategies: ☑ yes
Force end of day exit: □ no
Session end time: 17:00
Maximum Number of Rules Per Strategy: ● 2 ○ 3 ○ 4
Exit on signal Mode: Single
Continuous Simulation max rules count
Entry: 3 Exit: 1
Update database: Daily

Position Sizing Mode: Default
Slippage Mode: Per Trade
Commission Mode: Per Trade
Slippage (original currency): 0.000000
Commission (original currency): 0.000000
Account Value: 50000
Enter on: ○ Close ● Next Open
Exit on: ○ Close ● Next Open
Out of Sample: ○ Beginning ● End
Delayed Entry: 0
ATR Exit Length: 20
Rebalance options
Symbols to trade: 1
Rebalance frequency: Monthly
Ranking Method: Profit Factor

Then File->Custom Strategies->Add-> Select your file. Then the strategy will be imported after you run a simulation for the symbol and direction in the first line of your custom strategy file.

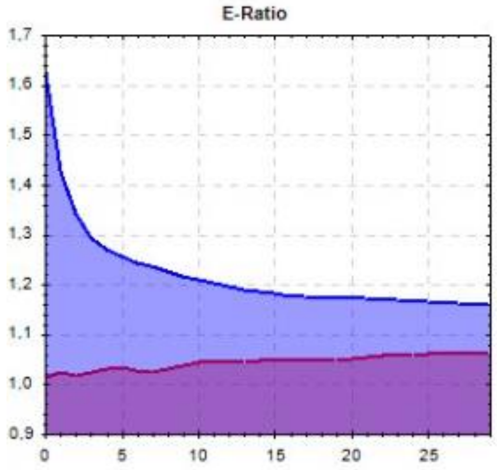
Note: You can have more than one strategy in one file as long as they are separated by unique header rows (first line above).



	A	B	C	D	E
1	#SPY	DipBuyer	Long		
2	20020107	153500	20020107	160000	557.00
3	20020111	93300	20020113	114700	-312.50
4	20020117	120000	20020117	150000	425.00
5	#SPY	Trend1	Long		
6	20020107	150000	20020107	160000	150.00
7	20020111	160000	20020113	114700	-125.00
8	20020117	120000	20020204	160000	1250.50

ERATIO

There are a few videos and blog posts regarding E-Ratio. The important thing to recognize is that both long and shorts are favorable above 1.00. One of the videos says shorts are desirable when they are under 1.00 but please realize this is wrong and has been updated. BOTH long and short E-Ratio should be desired above 1.00 now.



Multiple Time Frame strategies

To create multiple time frame strategies, you will need to select a second or third market. Build Alpha will automatically match their dates and times as well as the use the appropriate "active" signal from each market. If you want BA to be forced to use a signal from a specific time, make sure you make it bold by right clicking and requiring the specified signal!

Remember when exporting multi timeframe strategies you need to make sure your charts in which ever platform you're using for execution are set up properly. For example, see the first note of this guide regarding Vix data.

Force End of Day Exit

You can force Build Alpha to create strategies that exit at the same time every day. This is useful in creating pure day trading strategies that avoid overnight risk, if desired. You can turn this on in the settings menu and specify a time of day to exit. If you turn this on it will be added to the generated code automatically.

Please note that if this is on and you exit at say 17:00, for example. If there is another bar at 17:45 and your entry signal is true, then the software will enter again! If you want to create a system that only trades from 9:00 to 17:00 then you can right click on that time signal to require the software to use this rule and then also select force end of day exit at 1700.

Portfolio Mode

When adding strategies to portfolio mode you may notice “different” results. Strategies added to portfolio mode are saved using daily marked to market results as opposed to all the individual trades. This is typically industry standard to view a portfolio of strategies in this format and also allows the software to save on some space/memory consumption.

The P&L and drawdowns will be the same, but for example an individual strategy with a winning percentage of 65% might have a different winning percent in portfolio mode. This is because instead of looking at the trades individually we are looking at the number of profitable days!

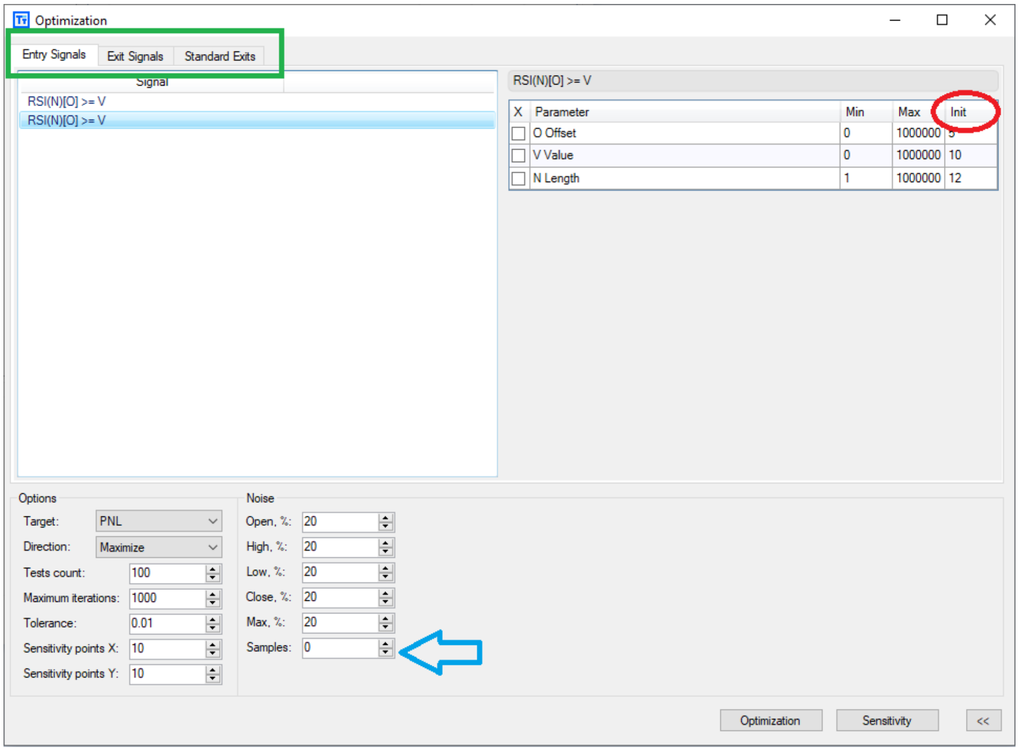
For example,

Day 1 has 3 trades. +50, +75, -35. The individual strategy will show a winning percentage of 67% (2 wins / 3 total trades)

When we save this to the portfolio mode we will see

Day 1 +90 and if we only traded on this day the winning percentage when viewed in portfolio mode would show 100% because 100% of the days were profitable. Again, this is done to save space, reduce memory consumption, and combine strategies of multiple timeframes/frequencies.

Optimization



Optimization

Signal
RSI(N)[O] >= V
RSI(N)[O] >= V

RSI(N)[O] >= V

X	Parameter	Min	Max	Init
□	O Offset	0	1000000	5
□	V Value	0	1000000	10
□	N Length	1	1000000	12

Options

Target: PNL
Direction: Maximize
Tests count: 100
Maximum iterations: 1000
Tolerance: 0.01
Sensitivity points X: 10
Sensitivity points Y: 10

Noise

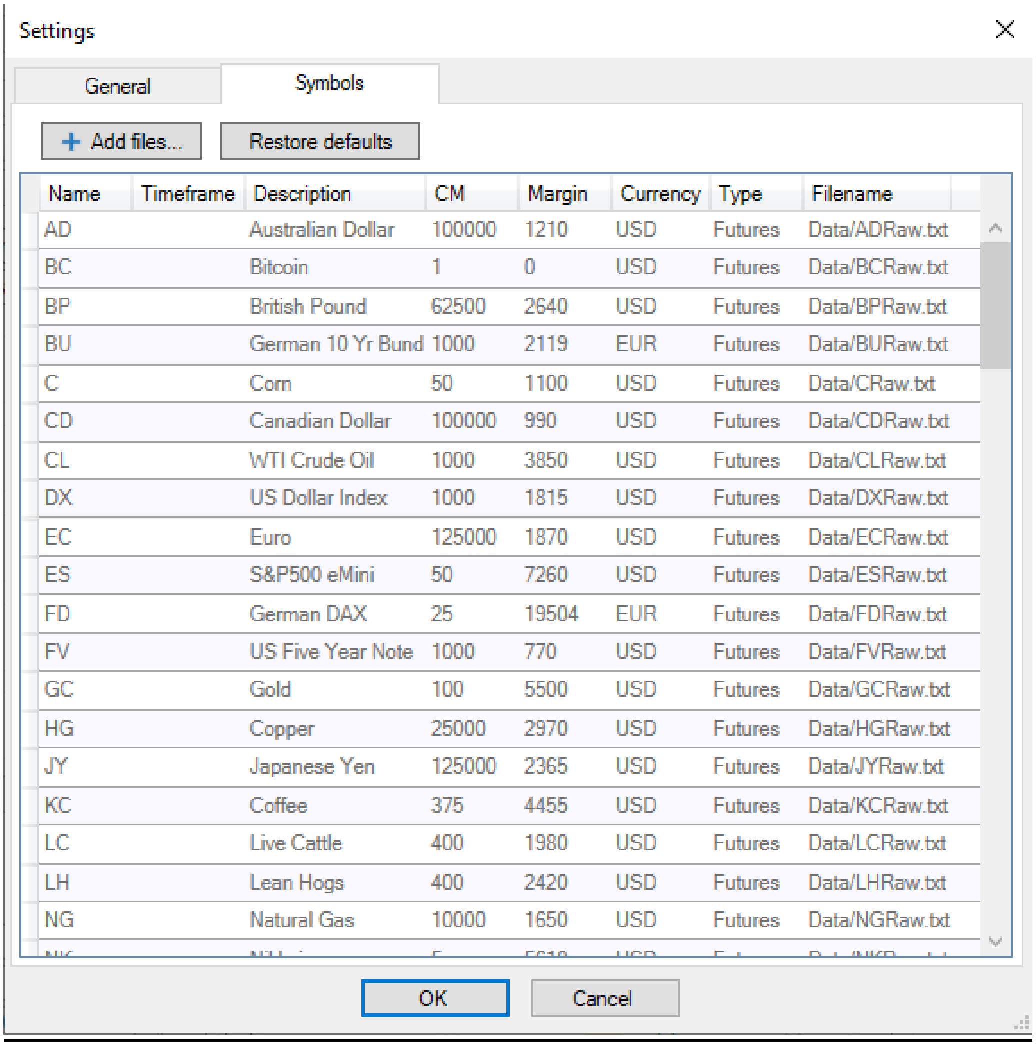
Open, %: 20
High, %: 20
Low, %: 20
Close, %: 20
Max, %: 20
Samples: 0

The green box shows how you can change from entries, exits and standard exits. Select one and it will be highlighted in blue. You can then edit the possibilities in the right table.

Notice the red circle. This is the initial value where the optimization algorithm will begin searching. It is NOT a step value like the main Build Alpha interface.

You can also optimize across noise adjusted samples. Notice the blue arrow. In this example, we will add 0 samples to our original price data to optimize across.

Importing Custom Data



Name	Timeframe	Description	CM	Margin	Currency	Type	Filename
AD		Australian Dollar	100000	1210	USD	Futures	Data/ADRaw.txt
BC		Bitcoin	1	0	USD	Futures	Data/BCRaw.txt
BP		British Pound	62500	2640	USD	Futures	Data/BPRaw.txt
BU		German 10 Yr Bund	1000	2119	EUR	Futures	Data/BURaw.txt
C		Corn	50	1100	USD	Futures	Data/CRaw.txt
CD		Canadian Dollar	100000	990	USD	Futures	Data/CDRaw.txt
CL		WTI Crude Oil	1000	3850	USD	Futures	Data/CLRaw.txt
DX		US Dollar Index	1000	1815	USD	Futures	Data/DXRaw.txt
EC		Euro	125000	1870	USD	Futures	Data/ECRaw.txt
ES		S&P500 eMini	50	7260	USD	Futures	Data/ESRaw.txt
FD		German DAX	25	19504	EUR	Futures	Data/FDRaw.txt
FV		US Five Year Note	1000	770	USD	Futures	Data/FVRaw.txt
GC		Gold	100	5500	USD	Futures	Data/GCRaw.txt
HG		Copper	25000	2970	USD	Futures	Data/HGRaw.txt
JY		Japanese Yen	125000	2365	USD	Futures	Data/JYRaw.txt
KC		Coffee	375	4455	USD	Futures	Data/KCRaw.txt
LC		Live Cattle	400	1980	USD	Futures	Data/LCRaw.txt
LH		Lean Hogs	400	2420	USD	Futures	Data/LHRaw.txt
NG		Natural Gas	10000	1650	USD	Futures	Data/NGRaw.txt

Name: When possible please set name to the same symbol as the built-in data when possible. This is how Build Alpha knows what tick increment and rounding to use.

Timeframe: Not really used but how you can make repeated symbols unique. This can be any type of alphanumeric text.

Example: If importing ES 15 minute data then you'd enter the Name as ES and timeframe as 15 or fifteen or xlyxlu15 (doesn't matter).

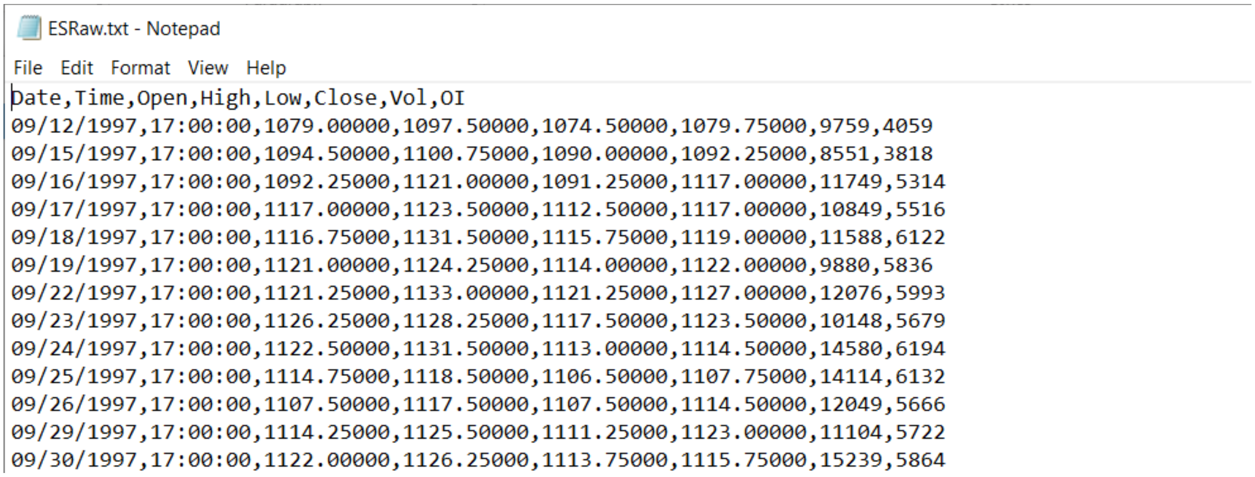
The CM is contract multiplier and is simply the big point value of the futures contract. For crypto, FX, stocks and ETFs you can leave this at 1. If using default position sizing, you can set multiplier to be your desired size. Here is a reference for big point values for futures markets:

https://www.barchart.com/futures/contract-specifications

Margin is the initial margin for futures market and is used in the position sizing calculations (explained below). There is a margins.csv file in the data folder you can reference but it is wise to set this based on what your broker sets. For example, ES margin is currently 6380 to trade 1 contract. Note: this margin is different than forex margin. For FX, ETFs, Crypto, or other you can leave this at 1.

Type: Please choose the appropriate type between Futures, ETFs, Forex, Crypto (fractional sizing allowed), and other.

Here is a built-in file showing the correct/current format for custom data. Please note the open interest column. If your data does not have open interest data, please add a column of all 0s or some dummy value (99999).

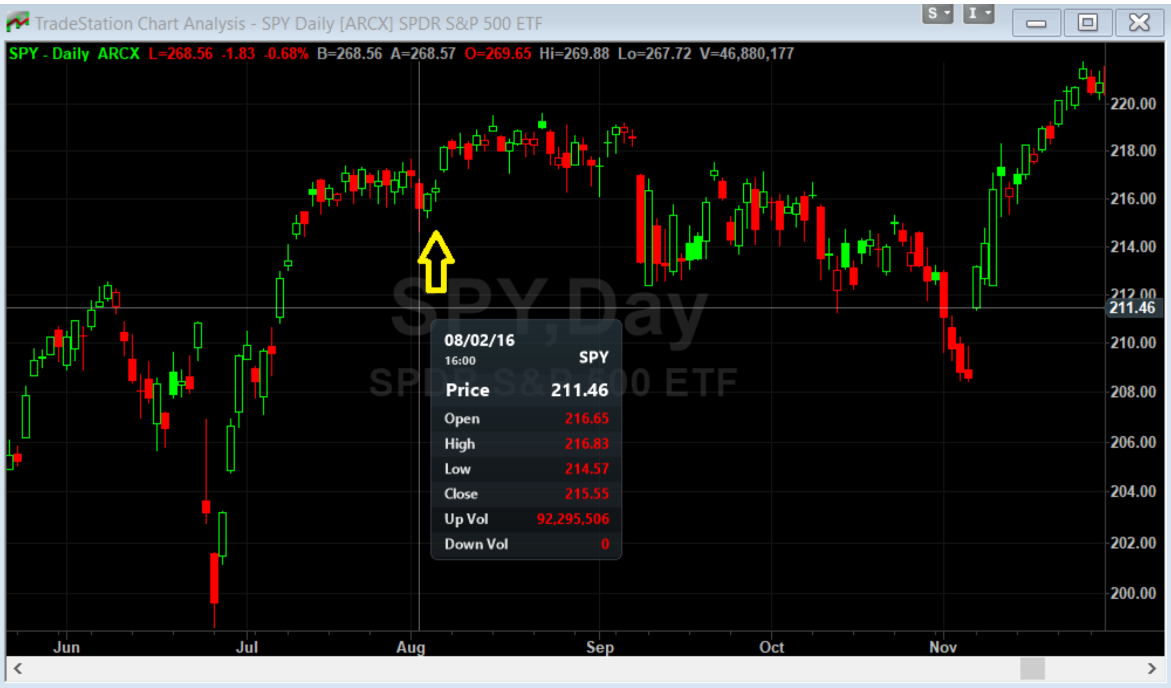


```
Date,Time,Open,High,Low,Close,Vol,OI
09/12/1997,17:00:00,1079.00000,1097.50000,1074.50000,1079.75000,9759,4059
09/15/1997,17:00:00,1094.50000,1100.75000,1090.00000,1092.25000,8551,3818
09/16/1997,17:00:00,1092.25000,1121.00000,1091.25000,1117.00000,11749,5314
09/17/1997,17:00:00,1117.00000,1123.50000,1112.50000,1117.00000,10849,5516
09/18/1997,17:00:00,1116.75000,1131.50000,1115.75000,1119.00000,11588,6122
09/19/1997,17:00:00,1121.00000,1124.25000,1114.00000,1122.00000,9880,5836
09/22/1997,17:00:00,1121.25000,1133.00000,1121.25000,1127.00000,12076,5993
09/23/1997,17:00:00,1126.25000,1128.25000,1117.50000,1123.50000,10148,5679
09/24/1997,17:00:00,1122.50000,1131.50000,1113.00000,1114.50000,14580,6194
09/25/1997,17:00:00,1114.75000,1118.50000,1106.50000,1107.75000,14114,6132
09/26/1997,17:00:00,1107.50000,1117.50000,1107.50000,1114.50000,12049,5666
09/29/1997,17:00:00,1114.25000,1125.50000,1111.25000,1123.00000,11104,5722
09/30/1997,17:00:00,1122.00000,1126.25000,1113.75000,1115.75000,15239,5864
```

PLEASE WATCH THE VIDEO "Adding Custom Data" on the buildalpha.com/demo page for more instruction.

Data Differences:

Some platforms report data differently. This can be very dangerous when backtesting IF you are unaware. When trying to replicate results across platform please ensure your data is matching. Build Alpha's data will purposely attempt to replicate TradeStation's data as the most of Build Alpha user's use TradeStation. For example, look at this example from Aug 04, 2016 (yellow arrows) comparing TradeStation data to NinjaTrader8 data. You can see TradeStation does not make a lower low on Aug 4. However, NinjaTrader does. So when concerned about mismatching results please always, always check for data differences!!

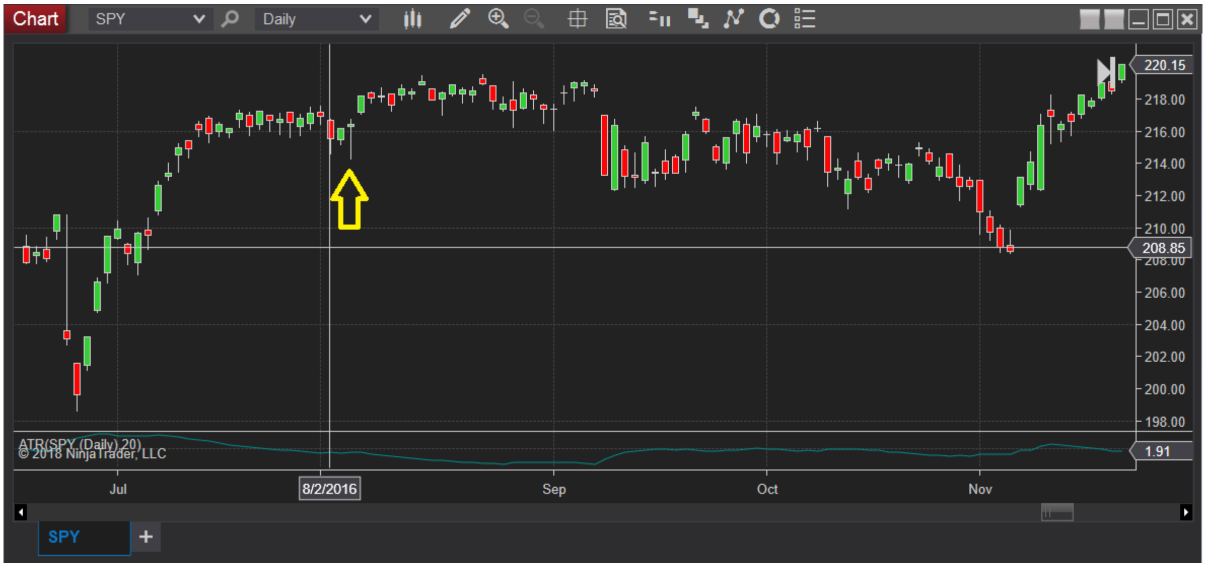


TradeStation Chart Analysis - SPY Daily [ARCX] SPDR S&P 500 ETF

SPY - Daily ARCX L=268.56 -1.83 -0.68% B=268.56 A=268.57 O=269.65 Hi=269.88 Lo=267.72 V=46,880,177

08/02/16
16:00
SPY

Price	211.46
Open	216.65
High	216.83
Low	214.57
Close	215.55
Up Vol	92,295,506
Down Vol	0



Filtering Trades vs. Including the ‘extra’ rule from the beginning:

Build Alpha allows us to look at our results and then right click on any strategy to ‘add an additional rule’. This can be helpful when you notice a strategy does very poorly in the month of November. It might be smart to add a rule to eliminate trading in November (month(date) <> 11). However, Build Alpha is not resimulating this strategy but simply removing trades that meet this ‘added’ condition. There is a big difference between resimulating and filtering. Here is an example below. This difference can cause strategy code with ‘additional added rules’ to look different when you export the code into platforms (which would be ‘resimulated’ in the other platforms vs. Bas filtering). However, going forward both will produce the same results on new data (regardless of filtered or resimulated) but there can be differences in backtests because of this nuance.

For example, imagine a strategy that below that has entry signals on bar 1,3,4,5,7 and holds for 2 days.

Trades would be

Entry bar 1
Exit bar 3

Entry bar 3
Exit bar 5

Entry bar 5
Exit bar 7

Entry bar 7
Exit bar 9

Now imagine if we just "filter" it to not trade on the 3rd bar. Filtered trades (from our existing list – what BA does with ‘add a rule’ feature) will now be

Entry bar 1
Exit bar 3

Entry bar 5
Entry bar 7

Entry bar 7
Exit bar 9

Now imagine if we include the "exclude bar 3" rule from the start (and re-simulate the entire strategy) we'd have trades

Entry bar 1
Exit bar 3

Entry bar 4
Exit bar 6

Entry bar 7
Exit bar 9

See how it can be different depending on when you add a rule (from the start or add as filter after simulation)

Position Sizing

Currently, there are three position sizing methods in Build Alpha. They are all configurable through the setting menu.

1. Default
2. Fixed Dollar
3. Volatility Based

Default method will buy 1 futures contract, 100 shares of ETFs or stocks, and 100000 FX (1 lot).

Fixed dollar simply buys a user specified dollar amount each trade. The benefit of this is to make apples to apples comparisons of trades that happen from an asset that has gone a large change in price. For example, buying 100 shares of stock XYZ when it was $10 is not the same thing as buying 100 shares when it trades at $200.

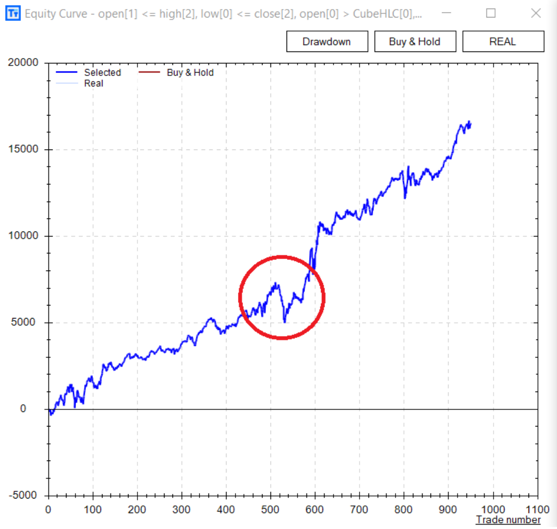
Users can set the fixed dollar amount by setting the account value in the settings menu. Setting the account value to $1000 would purchase 100 shares of the $10 stock and 5 shares of the $200 stock. Both positions would control $1000 worth of stock.

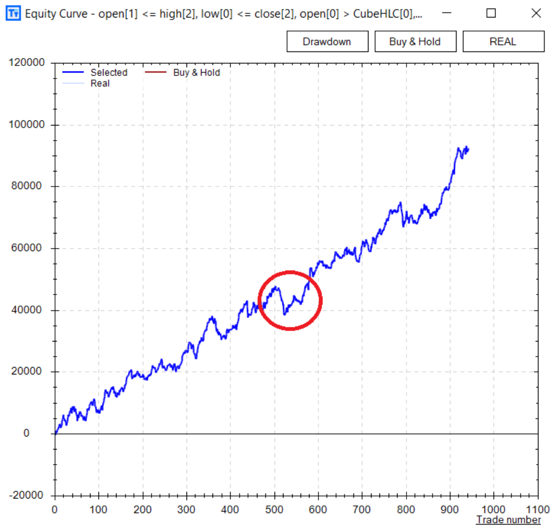
Volatility Based sizing allows position sizes to vary inversely to market risk (volatility). That is, as volatility increases, we dynamically reduce our exposure. The formula for the equation is below. The 1% is arbitrary and will be expanded in future updates.

Account value * .01 / ATR

Please note that systems with a stop loss turned on (e.g., 2 ATR stop) will multiply ATR by the SL multiple in the above equation to further reduce the risk. If system does not use a stop then a default of 1 ATR will be used.

In the example below, the same system is shown. The left-hand side uses the default setting whilst the right hand side uses the volatility method. The volatility method has reduced the drop from the peak equity by almost 40% from 31.98% to 19.48%.





Note: Any position size calculated to be less than 1 will be traded as a 1. For example, calculate position size to be .94 contracts then we will trade 1 instead of skipping the trade. This has been debated and settled as best practice between a group of respected beta users. If you have objections or want this configurable please EMAIL me. Thank you.

<u>Position Sizing in Easy Language</u>

The function to dynamically calculate position sizing for non FX instruments is position_size(). It contains multiple inputs described below for this example position_size(“Fut”,”ATR”,50000,4800,atr,sl_on*2).

Position_size(market,mode,account,margin,atr,sl_mult)

Market: “Fut”, “Other”, “FX”

Mode: “ATR”,”Fixed”,”Default”

Account: Your account size. Set by BA setting

Margin: Margin for the contract. Set by BA setting

ATR: atr calculated

SL_Mult: this is pre set by BA and will default in the code if you did not use SL in BA.

Position Sizing in Easy Language FX ONLY

For FX using position sizing in TradeStation, MultiCharts please note you need to put 5 instruments on your chart. This is because the default second data stream is always $VIX.X. Then data streams 3 and 4 are reserved for Build Alpha's intermarket signals (whether or not you have used them on this strategy). Then the 5 data stream is the currency pair to convert the traded pair to USD. Please reference the "Importing Code to TradeStation/MultiCharts" section above.

What symbol to use for data5 with FX in Easy Language?

Fixed method: XXX + account currency where XXX is the first pair in the traded pair.

Example: Account is USD and trading EURGBP. The 5th data stream needs to be EURUSD

Volatility method: account + YYY where YYY is the second pair in the traded pair

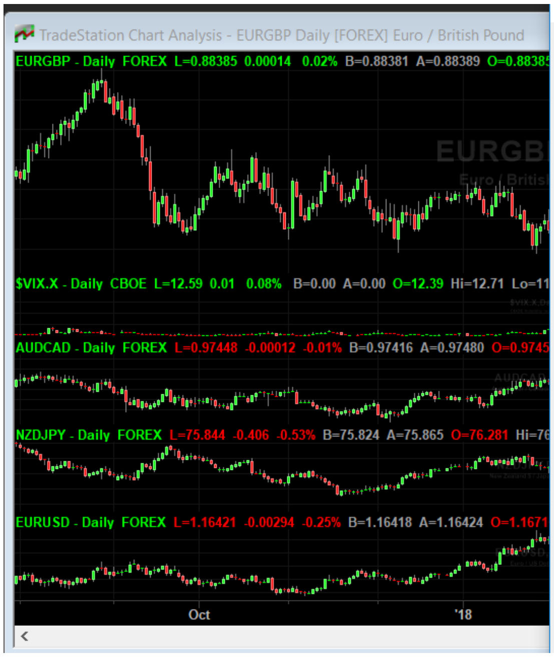
Example: Account is USD and trading EURGBP. The 5th data stream needs to be USDGBP. Since most do not have that please insert GBPUSD and the code will handle the rest.

Here is a photo of 5 data series on one chart to make FX sizing work with Easy Language. Notice $VIX.X is second data stream. It is always included in the BA pre-built signals. If you do not USE vix signals in your strategy or do NOT have vix data from your provider, please put a dummy symbol in its place like 'AAPL' or the same as the primary symbol (in this case EURGBP).

Build Alpha's market2 is TradeStation's data 3 and in this case is AUDCAD

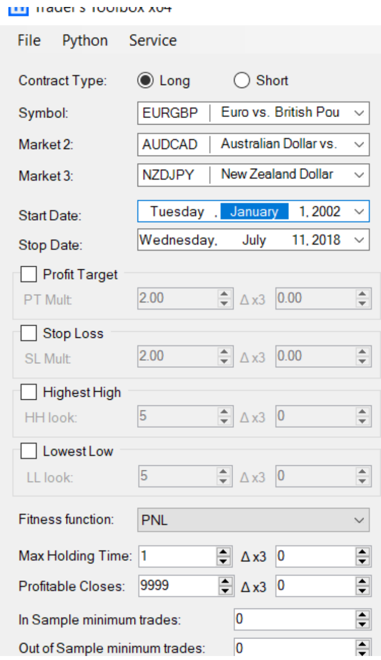
Build Alpha's market3 is TradeStation's data 4 and in this case is NZDJPY

Then data 5 is based on the conversion for 'Fixed' method above. If this strategy used volatility sizing then data 5 would be GBPUSD.



TradeStation Chart Analysis - EURGBP Daily [FOREX] Euro / British Pound

```
EURGBP - Daily  FOREX  L=0.88385  0.00014  0.02%  B=0.88381  A=0.88389  O=0.88385
$VIX.X - Daily  CBOE  L=12.59  0.01  0.08%  B=0.00  A=0.00  O=12.39  Hi=12.71  Lo=11
AUDCAD - Daily  FOREX  L=0.97448  -0.00012  -0.01%  B=0.97416  A=0.97480  O=0.9745
NZDJPY - Daily  FOREX  L=75.844  -0.406  -0.53%  B=75.824  A=75.865  O=76.281  Hi=76
EURUSD - Daily  FOREX  L=1.16421  -0.00294  -0.25%  B=1.16418  A=1.16424  O=1.1671
```



Trader's Toolbox x64

Contract Type: ● Long ○ Short

Symbol: EURGBP | Euro vs. British Pou

Market 2: AUDCAD | Australian Dollar vs.

Market 3: NZDJPY | New Zealand Dollar

Start Date: Tuesday , January 1, 2002

Stop Date: Wednesday, July 11, 2018

□ Profit Target

PT Mult: 2.00 Δ x3 0.00

□ Stop Loss

SL Mult: 2.00 Δ x3 0.00

□ Highest High

HH look: 5 Δ x3 0

□ Lowest Low

LL look: 5 Δ x3 0

Fitness function: PNL

Max Holding Time: 1 Δ x3 0

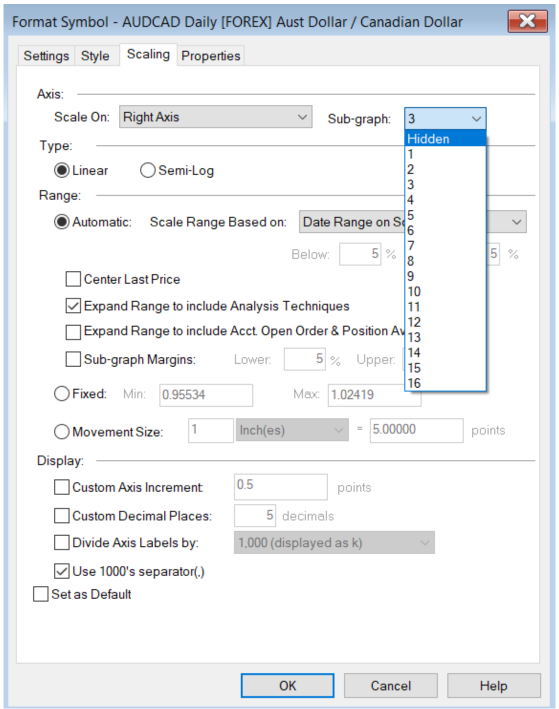
Profitable Closes: 9999 Δ x3 0

In Sample minimum trades: 0

Out of Sample minimum trades: 0

Hiding Price Series in TradeStation/MultiCharts

It can be annoying to have so many price series visible. In order to have them on the chart but not visible please double click on any price series you want to hide, go to scaling tab, and set Sub-graph to “Hidden”.



Format Symbol - AUDCAD Daily [FOREX] Aust Dollar / Canadian Dollar

Axis:

Scale On: Right Axis   Sub-graph: 3

Type:

● Linear ○ Semi-Log

Range:

● Automatic: Scale Range Based on: Date Range on S

Below: 5 %   5 %

☐ Center Last Price

☑ Expand Range to include Analysis Techniques

☐ Expand Range to include Acct. Open Order & Position Av

☐ Sub-graph Margins: Lower: 5 % Upper:

○ Fixed: Min: 0.95534 Max: 1.02419

○ Movement Size: 1 Inch(es) = 5.00000 points

Display:

☐ Custom Axis Increment: 0.5 points

☐ Custom Decimal Places: 5 decimals

☐ Divide Axis Labels by: 1,000 (displayed as k)

☑ Use 1000's separator(,)

☐ Set as Default

Position Sizing for FX ONLY in NinjaTrader

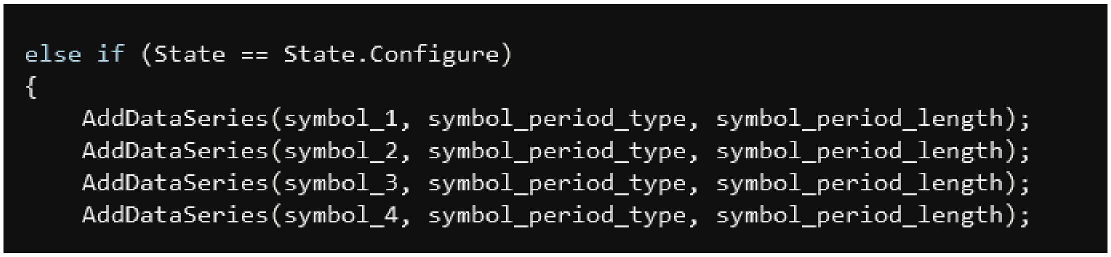
Ninjatrader FX we need to handle the currency conversions on our own. Similar to the other platforms. So, what symbol to use to convert based on symbol traded and position sizing method selected?

Fixed method: XXX + account currency where XXX is the first pair in the traded pair.
Example: Account is USD and trading EURGBP. The pair we need for conversion is EURUSD

Volatility Method: account currency + YYY where YYY is the second pair in the traded pair
Example: Account is USD and trading EURGBP. The pair we need for conversion is USDGBP. Which does not exist for most platforms. So, we just use GBPUSD and it will convert in the code on its own.

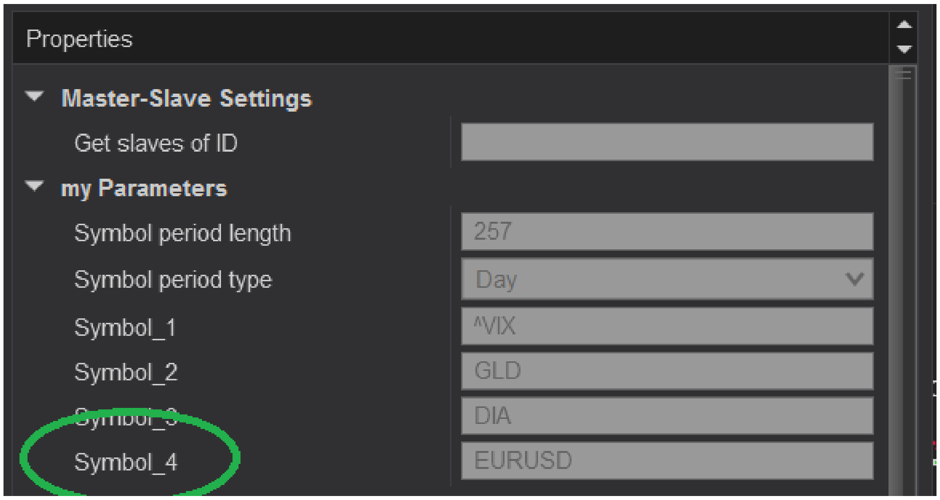
Once we have the pair, we need for proper conversion we must add it to the code. Build Alpha should set it on its own but if not please follow below. Let’s say we need to add EURUSD to our EURGBP strategy using Fixed method of sizing. Here is how with photos…

Step 1. Uncomment additional data streams. If your strategy does not use intermarket signals or multiple timeframes please uncomment these anyways to all symbol_4 to be used for currency conversion.



```
else if (State == State.Configure)
{
    AddDataSeries(symbol_1, symbol_period_type, symbol_period_length);
    AddDataSeries(symbol_2, symbol_period_type, symbol_period_length);
    AddDataSeries(symbol_3, symbol_period_type, symbol_period_length);
    AddDataSeries(symbol_4, symbol_period_type, symbol_period_length);
```

Step 2. Put the appropriate conversion symbol as symbol4 in the strategy settings. If it is not already set by Build Alpha.



Properties

Master-Slave Settings

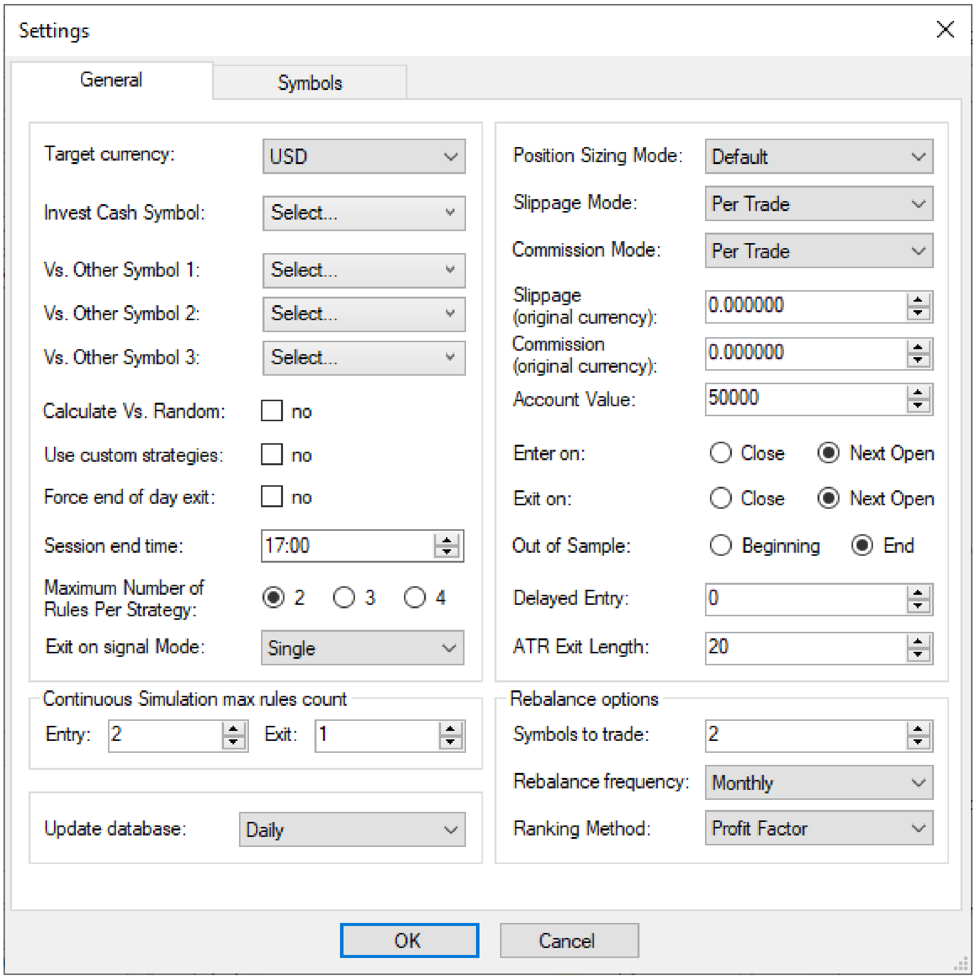
Get slaves of ID	

my Parameters

Symbol period length	257
Symbol period type	Day
Symbol_1	^VIX
Symbol_2	GLD
Symbol_3	DIA
Symbol_4	EURUSD

Rebalance Strategies

These are strategies that rank a basket of securities based on a user defined metric at a user defined period and then apply the trading strategy to only the top (or bottom) ranked N symbols in the basket. You can configure these metrics in the Services -> Settings Menu



Target currency: USD
Invest Cash Symbol: Select...
Vs. Other Symbol 1: Select...
Vs. Other Symbol 2: Select...
Vs. Other Symbol 3: Select...
Calculate Vs. Random: □ no
Use custom strategies: □ no
Force end of day exit: □ no
Session end time: 17:00
Maximum Number of Rules Per Strategy: ● 2 ○ 3 ○ 4
Exit on signal Mode: Single

Continuous Simulation max rules count
Entry: 2 Exit: 1

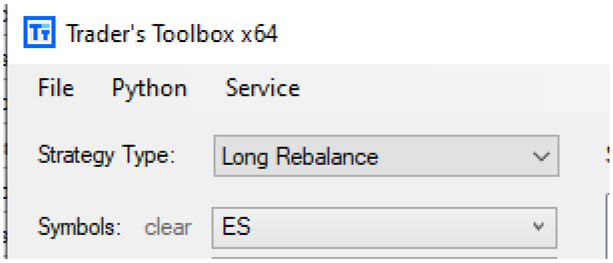
Update database: Daily

Position Sizing Mode: Default
Slippage Mode: Per Trade
Commission Mode: Per Trade
Slippage (original currency): 0.000000
Commission (original currency): 0.000000
Account Value: 50000
Enter on: ○ Close ● Next Open
Exit on: ○ Close ● Next Open
Out of Sample: ○ Beginning ● End
Delayed Entry: 0
ATR Exit Length: 20

Rebalance options
Symbols to trade: 2
Rebalance frequency: Monthly
Ranking Method: Profit Factor

The above example would trade the top 2 symbols based on a ranking of monthly profit factor. Let's say we have 4 symbols: SPY, TLT, GLD, USO. Each month we calculate each's profit factor. In the next month, we would only apply our strategy to the two symbols with the highest profit factor. Next month we will re-rank and do the same.

You can change the strategy type in the upper left of the main Build Alpha interface.

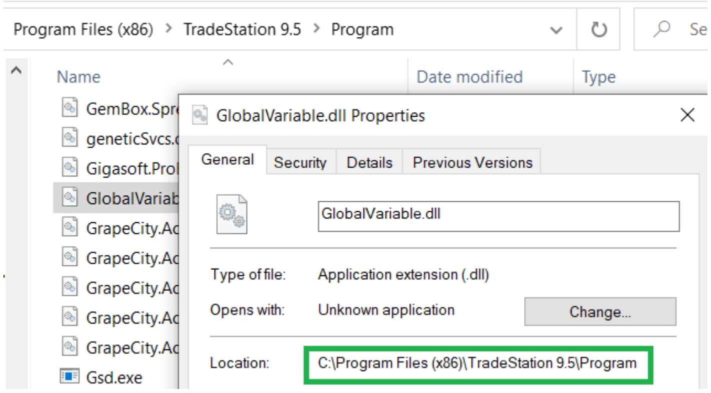


Strategy Type: Long Rebalance
Symbols: ES

Setting Up Rebalance in TradeStation

Please download GlobalVariables.dll from this link:
https://www.dropbox.com/s/91nuiqrq9bn7vu2/GlobalVariable.dll?dl=0

Then place this file in your C:/Program Files (x86)/TradeStation 9.5/Program/ folder



GlobalVariable.dll

Type of file: Application extension (.dll)

Opens with: Unknown application

Location: C:\Program Files (x86)\TradeStation 9.5\Program

Then please watch the video over at buildalpha.com/video with the correct password

Please note these strategies cannot be backtested in either TradeStation or MultiCharts. They can only be live traded.

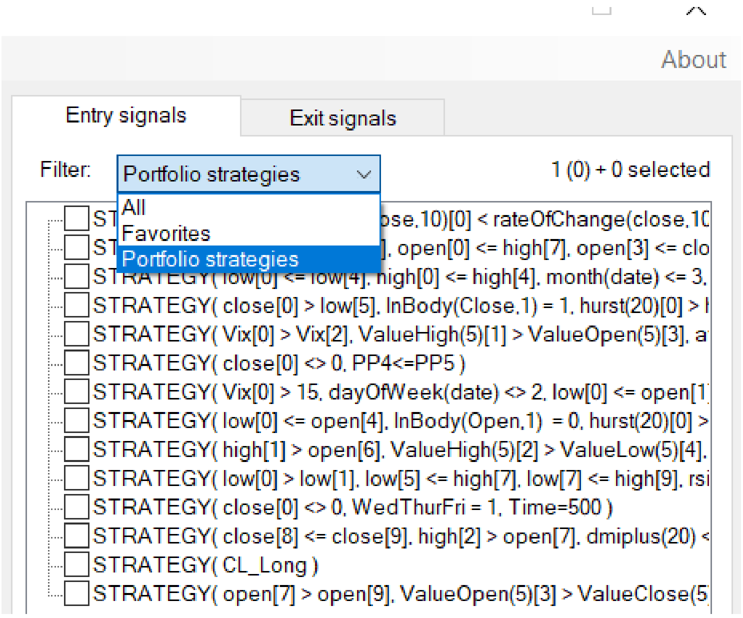
Setting Up Rebalance in Ninajtrader8 and MetaTrader4

Each symbol needs its own strategy. In the NT8,MT4 code the additional symbols are already added. Please watch the respective videos at buildalpha.com/video

MT4 cannot backtest these strategies and can only trade them live.

Ensemble Strategies

These are strategies that use other strategies as input signals. Think of this as creating a voting system. Strategy A, B and C each get one vote. If two of them say long, then the Ensemble strategy will go long. To use strategies as input to new strategies please use the dropdown menu above entry signals and select which strategies to include in the simulation (remember you can name the strategies in portfolio mode by double clicking on the name)



Filter: Portfolio strategies

1 (0) + 0 selected

□ ST ose,10)[0] < rateOfChange(close,1C
□ ST], open[0] <= high[7], open[3] <= clo
□ STRATEGY(low[0] <= low[4], high[0] <= high[4], month(date) <= 3,
□ STRATEGY(close[0] > low[5], InBody(Close,1) = 1, hurst(20)[0] > I
□ STRATEGY(Vix[0] > Vix[2], ValueHigh(5)[1] > ValueOpen(5)[3], a
□ STRATEGY(close[0] <> 0, PP4<=PP5)
□ STRATEGY(Vix[0] > 15, dayOfWeek(date) <> 2, low[0] <= open[1
□ STRATEGY(low[0] <= open[4], InBody(Open,1) = 0, hurst(20)[0] >
□ STRATEGY(high[1] > open[6], ValueHigh(5)[2] > ValueLow(5)[4],
□ STRATEGY(low[0] > low[1], low[5] <= high[7], low[7] <= high[9], rsi
□ STRATEGY(close[0] <> 0, WedThurFri = 1, Time=500)
□ STRATEGY(close[8] <= close[9], high[2] > open[7], dmiplus(20) <
□ STRATEGY(CL_Long)
□ STRATEGY(open[7] > open[9], ValueOpen(5)[3] > ValueClose(5

To see how to set up Ensemble strategies in each platform please watch the appropriate video at BuildAlpha.com/video

Before you can run ensemble strategies in TradeStation please:

1. Run the installer https://www.dropbox.com/s/u0s699njlpydhxr/TS_ADE-ELC_Installer2.01.exe?dl=0
2. Import this ELD: https://www.dropbox.com/s/2edmd76exlza97b/AllDataEverywhere.ELD?dl=0

If you have trouble, try this installer version instead: DATA/TS_ADE-ELC_Installer2.01.exe

Before you can run ensemble strategies in MultiCharts please:

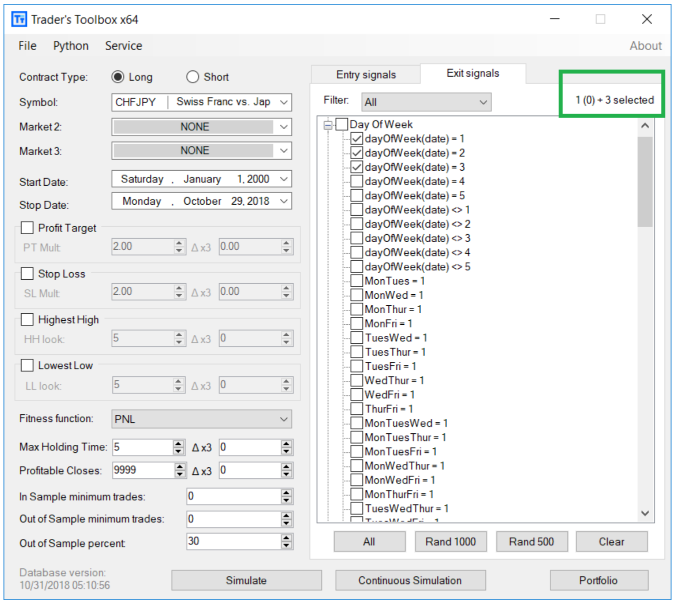
1. Import this ELD: https://www.dropbox.com/s/2edmd76exlza97b/AllDataEverywhere.ELD?dl=0
2. If using 32 bit Multicharts, paste this file into your MultiCharts folder: https://www.dropbox.com/s/ulxsjr6ysvil111/ELCollections.dll?dl=0
3. If using 64 bit Multicharts, paste this file into your MultiCharts folder: https://www.dropbox.com/s/hlzdetxjr5pkfql/ELCollections_64.dll?dl=0
   After pasting this into your folder please rename it to ELCollections.dll

   MC Folder is probably under the path: C:/Program Files/TS Support/MultiCharts/

note: at this time Pro Real Time cannot support Ensemble strategies. MT4 and Ninjatrader have slight difficulties with backtesting these strategies but do not require any additional files/documents for set up

Exit Signals

Exit signals can be selected by switching over to the Exit Signals tab (white when selected). Once exit signals are selected the count will appear to the right of the plus sign (note the green box).



Contract Type: ● Long ○ Short

Symbol: CHFJPY | Swiss Franc vs. Jap

Market 2: NONE

Market 3: NONE

Start Date: Saturday , January 1, 2000

Stop Date: Monday , October 29, 2018

□ Profit Target

PT Mult: 2.00 Δ x3 0.00

□ Stop Loss

SL Mult: 2.00 Δ x3 0.00

□ Highest High

HH look: 5 Δ x3 0

□ Lowest Low

LL look: 5 Δ x3 0

Fitness function: PNL

Max Holding Time: 5 Δ x3 0

Profitable Closes: 9999 Δ x3 0

In Sample minimum trades: 0

Out of Sample minimum trades: 0

Out of Sample percent: 30

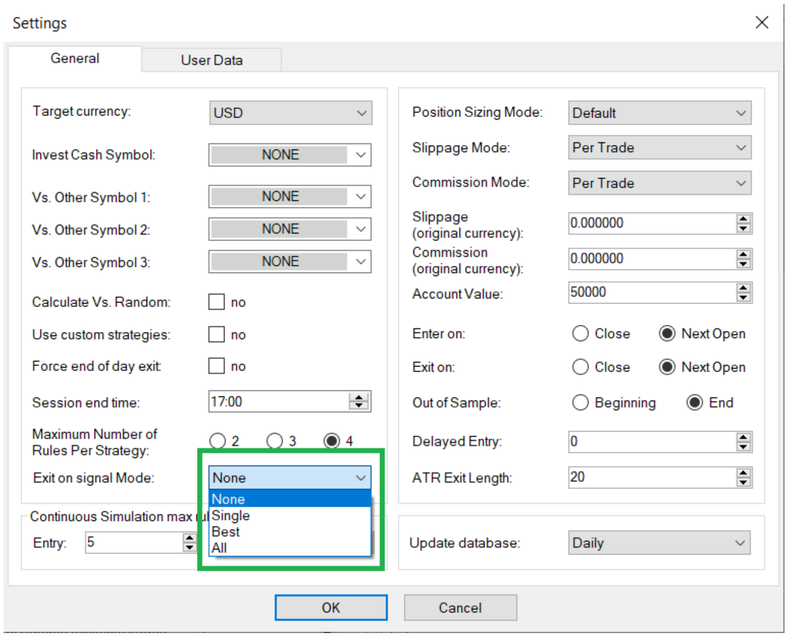
Database version: 10/31/2018 05:10:56

Filter: All

1 (0) + 3 selected

- □ Day Of Week
  - ☑ dayOfWeek(date) = 1
  - ☑ dayOfWeek(date) = 2
  - ☑ dayOfWeek(date) = 3
  - □ dayOfWeek(date) = 4
  - □ dayOfWeek(date) = 5
  - □ dayOfWeek(date) <> 1
  - □ dayOfWeek(date) <> 2
  - □ dayOfWeek(date) <> 3
  - □ dayOfWeek(date) <> 4
  - □ dayOfWeek(date) <> 5
  - □ MonTues = 1
  - □ MonWed = 1
  - □ MonThur = 1
  - □ MonFri = 1
  - □ TuesWed = 1
  - □ TuesThur = 1
  - □ TuesFri = 1
  - □ WedThur = 1
  - □ WedFri = 1
  - □ ThurFri = 1
  - □ MonTuesWed = 1
  - □ MonTuesThur = 1
  - □ MonTuesFri = 1
  - □ MonWedThur = 1
  - □ MonWedFri = 1
  - □ MonThurFri = 1
  - □ TuesWedThur = 1

In the settings menu there are 4 options: None, Single, Best, All



Target currency: USD

Invest Cash Symbol: NONE

Vs. Other Symbol 1: NONE

Vs. Other Symbol 2: NONE

Vs. Other Symbol 3: NONE

Calculate Vs. Random: □ no

Use custom strategies: □ no

Force end of day exit: □ no

Session end time: 17:00

Maximum Number of Rules Per Strategy: ○ 2 ○ 3 ● 4

Exit on signal Mode: None

Continuous Simulation max ru

Entry: 5

Position Sizing Mode: Default

Slippage Mode: Per Trade

Commission Mode: Per Trade

Slippage (original currency): 0.000000

Commission (original currency): 0.000000

Account Value: 50000

Enter on: ○ Close ● Next Open

Exit on: ○ Close ● Next Open

Out of Sample: ○ Beginning ● End

Delayed Entry: 0

ATR Exit Length: 20

Update database: Daily

None: This will not use any exit signals in simulation mode even if some are selected

Single: This will use the BA strategy engine and only apply one exit signal per strategy. It will still use other exits if configured (PT, SL, HH, LL, Max time and profitable close). It only applies one 'signal' exit per strategy.

Best: This will use the BA strategy engine and apply the best combination of exits per strategy. This can use any number of exit signals per strategy. Note this also has the LONGEST processing time as the search space is significantly larger!

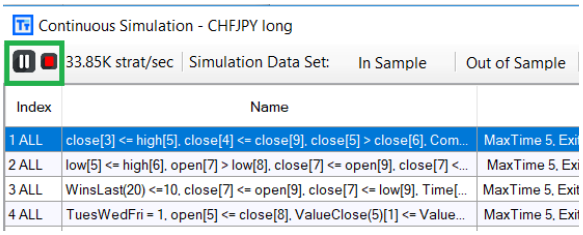
All: This will use the BA strategy engine and apply all selected signal exits to each strategy. So every exit selected must be true in order to force an exit. This is more for testing specific ideas and hypothesis testing as opposed to rapid prototyping.

Simulation vs. Continuous Simulation

The normal simulation button (left) runs the original Build Alpha strategy engine which intelligently reduces the search space in attempt to find the best strategies with a [more] strict convergence point – point in which no better strategies can be created within reasonably avoiding excessive chance of overfitting.

The continuous simulation button (right) runs non-stop until the user chooses to pause or stop the simulation. BA can be left endlessly, and Build Alpha will continue to generate strategies (only showing best by fitness). It will not stop once convergence (point no better strategies can be found reasonably) is met as some believe it is worth to keep pushing for better and better evolution. The user can monitor, view and run other Build Alpha features while continuous simulation is running, paused or stopped.

To pause or stop a continuous simulation please notice the two options in the top left of the output window:

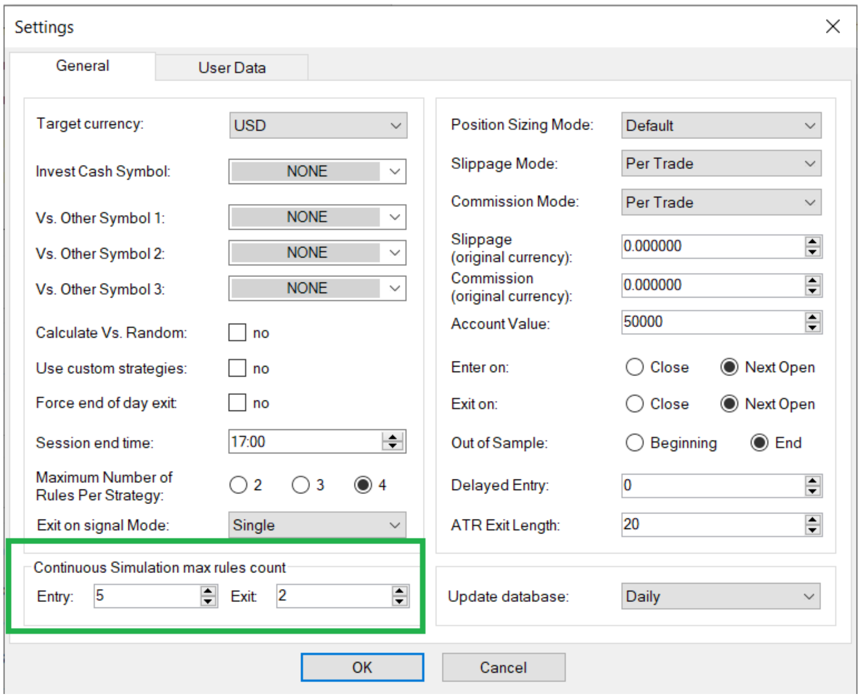


Continuous Simulation - CHFJPY long

33.85K strat/sec | Simulation Data Set: In Sample | Out of Sample

Index	Name	
1 ALL	close[3] <= high[5], close[4] <= close[9], close[5] > close[6], Com...	MaxTime 5, Exit
2 ALL	low[5] <= high[6], open[7] > low[8], close[7] <= open[9], close[7] <...	MaxTime 5, Exi
3 ALL	WinsLast(20) <=10, close[7] <= open[9], close[7] <= low[9], Time[...	MaxTime 5, Exit
4 ALL	TuesWedFri = 1, open[5] <= close[8], ValueClose(5)[1] <= Value...	MaxTime 5, Exit

Please also note you can configure maximum entry rules and maximum exit rules to be used during continuous simulation with a separate setting in setting menu:



Target currency: USD
Invest Cash Symbol: NONE
Vs. Other Symbol 1: NONE
Vs. Other Symbol 2: NONE
Vs. Other Symbol 3: NONE
Calculate Vs. Random: □ no
Use custom strategies: □ no
Force end of day exit: □ no
Session end time: 17:00
Maximum Number of Rules Per Strategy: ○ 2 ○ 3 ● 4
Exit on signal Mode: Single

Continuous Simulation max rules count
Entry: 5 Exit: 2

Position Sizing Mode: Default
Slippage Mode: Per Trade
Commission Mode: Per Trade
Slippage (original currency): 0.000000
Commission (original currency): 0.000000
Account Value: 50000
Enter on: ○ Close ● Next Open
Exit on: ○ Close ● Next Open
Out of Sample: ○ Beginning ● End
Delayed Entry: 0
ATR Exit Length: 20

Update database: Daily

The above allows BA continuous simulation to build strategies with UP TO 5 entry rules and UP to 2 exit signals (plus other exits SL, PT, Max time, Profitable closes, HH, LL).

The “Maximum Number of Rules Per Strategy” setting above is for the original Build Alpha strategy engine which is utilized by hitting the ‘Simulate’ button as opposed to the Continuous Simulation button.

For more information or clarification send me an email: **David@buildalpha.com**

All Links

TradeStation DLL & Support File:

https://www.dropbox.com/s/awhuj1jdl8kldcf/Release_AxiomFutures_TradeStation_setupv1.exe?dl=0

https://www.dropbox.com/s/k41pi622xxya1ie/BuildAlpha.ELD?dl=0

MultiCharts DLL & Support File:

https://www.dropbox.com/s/73m9w31v5h6tre1/Release_AxiomFutures_MultiCharts_setupx.exe?dl=0

https://www.dropbox.com/s/k41pi622xxya1ie/BuildAlpha.ELD?dl=0

Ninjatrader8 DLL & Support File:

https://www.dropbox.com/s/1wzzyrwccfo1kir/Release_AxiomFutures_NinjaTrader8_setup.exe?dl=0

https://www.dropbox.com/s/qrnk7yrt8wqu8ei/BuildAlphaNT8.zip?dl=0

Metatrader4 DLL & Support File:

https://www.dropbox.com/s/wxxgaotsz180klc/Release_AxiomFutures_MetaTrader4_setupv1.exe?dl=0

https://www.dropbox.com/s/o47hfnwhk0vi0me/CStrategy_v2-15.mqh?dl=0

Pro Real Time Support File:

https://www.dropbox.com/s/5q7ho2bj5fwwc9z/PRT.zip?dl=0

TradeStation Global Variable DLL:

https://www.dropbox.com/s/91nuiqrq9bn7vu2/GlobalVariable.dll?dl=0

ADE Installer TradeStation:

https://www.dropbox.com/s/u0s699njlpydhxr/TS_ADE-ELC_Installer2.01.exe?dl=0

https://www.dropbox.com/s/2edmd76exlza97b/AllDataEverywhere.ELD?dl=0

DATA/TS_ADE-ELC_Installer2.01.exe

ADE Installer MultiCharts:

https://www.dropbox.com/s/ulxsjr6ysvil111/ELCollections.dll?dl=0

List of Private Videos

- New Interface Overview
- Saving and Loading Configurations
- Introduction
- Performance Metrics and Fitness Functions
- Sample Size and Fitness Functions Part2
- InterMarket Signals
- InterMarket Signals Part2
- Selecting Exit Criteria
- Exits Part 2
- Signal Exits
- Test Configurations
- Continuous Simulation Mode
- Familiarizing Yourself with the Output Window
- Individual Equity Curves, Drawdowns, Benchmarks
- Vs. Random Protecting Against Data Mining Bias

- Vs. Random and viewing simulation distributions
- E-Ratio (Edge Ratio)
- Monte Carlo Bands and VaR
- Monte Carlo Drawdown Analysis
- Monte Carlo Tests
- Variance Testing and Forward Simulating
- Variance Testing Part2
- Robustness Testing
- Robustness Testing Part 2
- Noise Test
- Randomized Out of Sample and Seasonality Breakdown
- Rerun Strategy with Adjustments
- Add a Rule
- Generating Tradeable Code
- Portfolio Features
- Importing Custom Strategies
- Custom Rule/Indicator Builder
- Outlier Strategies
- Monte Carlo Resample
- Intraday Checks Test Part One
- Intraday Checks Test Part Two
- Intraday Edge Explained
- Intraday Edge In TradeStation
- Pro Real Time - Custom Signals
- Pro Real Time - Import Strategy Code
- Python Development Environment
- Python Basics
- Python Example: Technical Analysis Library
- Python Example: Gaps
- Python Example: Sigma Scores
- Python Example: Machine Learning with Sci-kit
- Dukascopy FREE data download
- Read In Custom Strategies From TradeStation
- NinjaTrader8 Import Code
- Ensemble Set Up
- Ensembles in TradeStation/MultiCharts