

Holy Graily Bob 'n Dotty User Guide

Introduction

This guide contains a brief description of the purpose of all the inputs in the EA. The use of 'criminal' in my forum to describe the brokers, is only semi-humorous. The behaviour some of these cowboys is shocking. Use of 'criminal' or 'crim' in this document refers to these people. Note that it **does not** apply to Global Prime, who are scrupulously honest.

Most of us use HGBnD as a basket trader, hoping to close out when the overall position reaches our basket take profit. It is possible to trade her purely as a single trade EA by setting the inputs appropriately, but anyone wanting to do this is probably better off with the dedicated EA at [Holy Graily Bob](#).

Remember to leave StopLossPips and TakeProfitPips at zero if you are basket trading.

The Holy Grail Indicator

Download this from Bob's thread at [Nanningbob's Holy Grail Indicator forum](#). This is an indicator developed with breathtaking brilliance by milanese and elixe at Bob's request. It tells manual traders when to take a trade and when to close it. HGI gives a variety of signals; here we are concerned with:

- Blue wavy lines. These are strong trend indicators. Market is above the line: close sell and enter buy trades. Market is below the line: close buy and enter sell trades.
- Large green up arrows. Market is trending up: close sell and enter buy trades.
- Small diagonal green up arrows known as RAD arrows. Re-enter an existing up trend: close sell and enter buy trades.
- Large red down arrows. Market is trending down: close buy and enter sell trades.
- Small diagonal red down arrows known as RAD arrows. Re-enter an existing down trend: close buy and enter sell trades.
- Yellow wavy lines: market has entered a range, so close all trend trades.

You will also need the hgi_lib.ex4 file from the first post in Bob's thread at [Nanningbob's Holy Grail Indicator forum](#); you can save it in either the Experts or Libraries folder. HGBnD needs this to run and will unload herself if the library is missing.

There is a downside to HGI. It is only of use in the here-and-now. It may generate a signal and the conditions that generated the signal may vanish but the signal will remain on the chart until you change chart time frames or restart the platform. The signal will only remain on the chart if the conditions that created it remain at the close of the chart. Looking back through history makes it look as though HGI is pinpoint accurate; it isn't. It *is* very, very good. Also bear in mind that the position of the signal has nothing to do with the market price at the time HGI generated the signal; the market was a long way away from it and there is no way of telling what the price was.

HGI stores signal data in one of seven buffers that an EA can interrogate. If the conditions that generated the signal change, the buffer empties. HGB'nG interrogates the buffers and will show no signal when this happens, even though the signal graphic is still on the chart. It will look as though HGB'nD has taken a wrong decision; it hasn't.

The only price data each candle on our chart retains is: high, low, open and close prices. The 'ticks' you see when running a visual back test in Strategy Tester are guesses based on where these four prices ended at the close of the candle. It is impossible to write an indi that would accurately show when HGI generated a signal that subsequently disappeared.

How HGBnD uses HGI

HGI has three functions:

1. Define the trend by looking for the most recent signal on a higher time frame. By default, she looks back over 20 candles on the H4. No signal in those 20 candles means the market has been moving in one direction for a long time; a 'correction' is due, so we are better off staying away.
 - Long trends are determined by the large green arrows and blue squiggly lines.
 - Short trends are determined by the large red arrows and blue squiggly lines.
2. Providing the trade trigger in the direction of the defined trend. By default, this is the M5:
 1. HGBnD looks at the previous candle as soon as it is closed, and sends a trade if there is an appropriate signal. There is no way of telling at which point in the candle the signal was generated, but being still in place at the close of the candle means that the conditions that generated the signal still apply. They might disappear at the next tick, but that is Forex for you.
3. Creating and managing some of the hedging functions.

The HGBnD core library

This contains the core functions of all the HGBnD family. You need it for any of the family to run. Download it from [My HGBnD shell code thread](#). It goes in your /Platform/Include folder.

Grid trading and offsetting

Assuming you have enabled gridding, hedging and offsetting enabled (this is the default), here is what will happen following a new trade signal when there are no trades already open:

- Either send an immediate market or pending trade in the direction indicated by the trend and the trade signal.
- Send a grid of buy and sell stops on either side of this initial trade.

Imagine we are in a long trend and there has just been a buy trade trigger. The trades have been placed. Here are the favourable scenarios:

1. The market screams off into the stratosphere, our basket take profit is hit, HGBnD closes all the trades and deletes the outstanding pendings. Party time.
2. The market rises and fills a buy or few then reverses and fills a sell or few. Then it rises again and carries on its merry way. More buys fill and eventually overwhelm the losing sells, closing at our hedged basket take profit.
3. The market rises and fills a buy or few then reverses and fills a sell or few. Then it carries on its merry way downwards. More sells fill and eventually overwhelm the losing buys, closing at our hedged basket take profit.

The markets can range between two quite far distant prices and we can end up with a hedged position with umpteen buys and sells, none of which are ever closing. This is where offsetting comes in. I have reproduced this in the inputs section as well:

- **Offsetting:** this involves using profitable trade or trades from one part of the grid to close in conjunction with a loser elsewhere. Here is an example of the most simple form of offsetting:
 - you have four buy trades open and the distance between them is 10 pips:

- trade 4 has just opened and so is the spread in the red.
- trade 3 is +10.
- trade 2 is +20
- trade 1 is +30.
- the market starts to reverse. It goes all the way down past trade 1; what was once a beautifully profitable position is now in a stonking loss. Instead of this, offsetting kicks in then the market has fallen by one grid size; 10 pips in this example.
- So now:
 - trade 4 is -10.
 - trade 3 is at breakeven.
 - trade 2 is +10
 - trade 1 is +20.
- we close trade 4 and trade 1 at an overall profit of +10 and DD is reduced considerably if the market continues to fall.

Full hedging

This comes into play when we end up with ludicrously unbalanced positions. Those of us who have been around since the start of development from RnR through Dottybot in her various guises to here are familiar with these:

- We have both buys and sells open. Market movement has not allowed any basket take profit to hit but has closed a lot of trades in one direction.
- Imagine we have totals of 0.03 buy lots and 0.15 sell lots open and HGI finds a change of trend to long. HGBnD will:
 - Close as many of the sells as possible. This will be all of them if the total cash value of the sell trades is positive or at break even. It will close the profitable ones if not, so they do not turn into losers if the market does rise.
 - Calculate whether adding a full hedge trade is necessary. If so, she will send a buy trade with a lot size that means the total of buy lots = the total of sell lots.
 - Continue to monitor the position and look for opportunities to close trades – especially the full hedge trade following another trend change:
 - Having closed a full hedge trade profitably, HGBnD can use these profits to close down the equivalent value of losing trades on the opposite side of the hedge i.e. sells if the full hedge trade was a buy.
 - Having a full hedge trade open turns off offsetting. I am not sure if this is the correct approach yet.

Inputs

Enter all pip inputs as pips. Forget the x digit crap so beloved by the crims; my EA's all convert your pips into the points required by your crim. We are indebted to Lifesys for the code.

- **General Inputs**

-
- **Lot:** your chosen lot size. Make sure your lot size is acceptable to your criminal.
- **RiskPercent:** this tells an EA to calculate the lot size as a risk percentage of your account balance. It uses the pips count in StopLossPips.
 - I do not see how this is compatible with the full hedging functionality.
- **LotsPerDollopOfCash:** you can have your lot size automatically calculated this lot size per amount of cash in the account balance or equity. The default settings would deliver 0.01 lots per \$1,000. A zero input turns this feature off.
- **SizeOfDollop:** the cash increments used.
 - I do not see how this is compatible with the full hedging functionality.
- **UseBalance:** use the account balance for the calculation.
- **UseEquity:** use the account equity for the calculation.
 - Example of use, choosing the equity:
 - equity = \$2133.56
 - LotPerDollopOfCash = 0.01.
 - SizeOfDollop = \$1,000.
 - Calculated lot size is 0.02.
- **StopTrading, TradeLong and TradeShort:** these allow you to control the trading direction or even stop trading altogether if all you want this EA to do is manage an open trade.
- **TakeProfitPips:** your take profit.
- **StopLossPips:** your stop loss.
- **Magic number and trade comment:** leave these alone unless you know what you are doing.
- **IsGlobalPrimeOrECNCriminal:** set this to true if your criminal insists on two-stage order-sending. This is irrelevant if you do not use StopLoss and TakeProfit. Remember that IBFX are ECN even though they do their best to hide this fact and stop all their 'valued clients' actually sending any trades; the EA has code to detect IBFX as the crim and set this input accordingly. It will also detect Global Prime accounts.
- **MaxSlippagePips:** 'slippage' is the price changing in between you sending off your trade and it being accepted by the market maker on the other side of your trade. The trade is cancelled if slippage exceeds this figure.

- **HGI inputs**

- **UseTrendHGI:** turns the trend filter on or off.
- **HgiTrendTimeFrame:** the higher time frame you want to use to determine the trend.
- **HgiTrendTimeFrameCandlesLookBack:** the number of candles to search for the latest trend change. No signal over this period means the trend has been in place for a long time and could be due a reversal, so we are best off staying out of the market.
- **UseTradingTimeFrameHGI:** tells the EA to use the trading time frame HGI as part of the trade trigger decision.

- The next three inputs tell HGBnG which HGI signals to use as trade triggers:
 - **HgiTrendTradingAllowed:** the large green and red arrows. Recommended.
 - **HgiRadTradingAllowed:** the small, diagonal red and green arrows. These indicate re-entry opportunities and I do not recommend using them.
 - **HgiWaveTradingAllowed:** the blue squiggly trend lines. Recommended.
- The next three inputs tell HGBnD how to behave following a change of trend:
 - **HgiCloseOnYellowRangeWave:** the yellow squiggly lines represent a range start or end and tells HGBnD to close trades when one appears. I do not think anybody enables this if they are basket trading.
 - **HgiCloseOnOppositeSignalOrTrendChange:** tells the bot to close buys when the trend changes to short, and sells when it changes to long.
 - **HgiOnlyCloseProfitablePositions:** this works in conjunction with the previous two inputs. It tells HGBnG that trade closures must not result in a cash loss. She may still close winning trades to avoid them turning into losers, depending on circumstances.
- The next six inputs are concerned with adding and monitoring a full hedge trade:
 - **UseFullHedging:** turns this feature on/off.
 - **BuysAndSellsUnbalancedAtLots:** this is the point at which a position is considered to be 'unbalanced'. For example: the default is 0.1; there are 0.03 buy lots open and 0.18 sell lots open; the difference between the two groups is >0.1, and so is 'unbalanced'. HGBnD will send full hedge buy trade of 0.15 lots following an HGI buy signal.
 - **OnlyCloseFullHedgeWhenInProfit:** this tells HGBnD only to close a full hedge trade if it is in profit.
 - **OffsetOppositeLosersAgainstProfit:** if enabled and HGBnD has just closed a profitable full hedge trade, she will use the profit to close as many of the losers as possible, on the other side of the hedge.
 - **FullHedgeComment:** the order comment that labels the trade as a full hedge trade. HGBnG uses this to identify a full hedge, so you must not change this input once you have a full hedge in place.
 - **CloseProfitableFullHedgeOnYellowWave:** tells HGBnD to close profitable full hedge trades when there is a yellow range wave signal. She will also offset opposite direction trades if OffsetOppositeLosersAgainstProfit is set to 'true'.
- **Trading style**
 - **ImmediateMarketOrders:** tells HGBnD to take an immediate market trade before sending grids.
 - **Associated details:**
 - **StackByCandleDirection:** this is for those of you not using the grid trading features described later on. Imagine you have blue dots and HGBnD has bought. He will buy again when a candle closes red – 'buying the dip'.
 - **MaxTradesAllowed:** the maximum number of stacked trades you will allow.
- **The next two groups of inputs** uses the principle of 'buy low, sell high' - often stated but never before explained properly. CJ and DFC does this perfectly:
 - identify the highest high and lowest low on the chart.
 - the market is rising if the lowest low is later than the highest high, so only take buy trades. Only buy from the bottom and do not trade the middle of the chart.
 - the market is falling if the highest high is later than the lowest low, so only take sell trades. Only sell from the top and do not trade the middle of the chart.
 - buying low and selling high should be done with the maximum amount of chart

space available so:

- close all windows on the left of your chart - Navigator, Market Watch etc.
- HGBnD adjusts your right hand margin to its minimum size = 10% of the chart.
- HGBnD draws solid lines to mark the highest and lowest prices the market made. She ignores spikes, so switch to line chart if you want to see why the lines are drawn where they are.
- **UseSixths:**
 - this adopts Bob's Sixths principal that we played with a few years ago:
 - divide the chart into rows of equal sixths:
 - buy only from the bottom sixth and close towards the middle.
 - sell only from the top sixth and close towards the middle.
 - HGBnD adds a dotted line to show the 'trading zone'.
 - ChartDivisor: the input that the bot uses to calculate the size of the 'sixths'. A value of 5 here would divide the chart into fifths. A value of 4 would divide it into four quarters. Playing with the value of this input is the only way to work out which is best.
- **Inputs common to both Sixths and Buy Low Sell High:**
 - NoOfBarsOnChart: this allows you to set the number of candles over which the peaks are calculated. Use the "Show number of bars on the chart script" attached to the core library post.
 - ShowTradingArea: this tells the EA to draw a dotted line that defines the trading areas most likely to be successful at the peak high (sell) and peak low (buy).
- **UseBuyLowSellHigh:** this tells HGBnD to buy only when the lowest low is later than the highest high. Vice versa for sells. There are four time frames that you can use in combination to rule the direction in which the bot can trade. Each has a boolean such as UseHighestTimeFrame to turn it on/off, a time frame input and line colour/thickness choices.
 - There are inputs to control the placing of the information labels showing you what is happening on your chosen time frames. Play with them until you get your display suiting you.:
 - BlshDisplayX is the position in pixels on the X (horizontal) axis.
 - BlshDisplayY is the position in pixels on the Y (vertical) axis.
 - You have a choice of font and font size.
- **Hedging:** 'hedging' means having simultaneous buy and sell trades open on the same chart.
 - **UseHedgingWithGrid:** tells HGBnD to send a grid of stop orders in both directions when there is a new nonlagdot signal. There are two profit targets for a hedged position. The can both be set and the first one to be hit will make HGBnD close the position. A zero value disables them:
 - **HedgeProfitPips:** the number of pips the profit needs to reach to trigger a position close.
 - **HedgeProfitCash:** as above, but for cash.
- **Grid and hedge trading:** these can be linked as I do, or you can use grid trading on its own. 'Grid' trading involves sending a grid of pending trades in the desired direction. These pending trades become market orders when the market reaches the pending price. The upside is that profits accrue rapidly when the market cooperates. The downside is that several pendings might fill then the market reverses, leading to eye-watering draw down.

- **Grid**
 - **UseGrid:** tells HGBnD to send a grid of buy stop orders following a buy signal, and sell stop orders following a sell signal.
 - **GridSize:** the number of stop orders to send.
 - **DistanceBetweenTradesPips:** the distance in pips between each stop order. This can be calculated dynamically using ATR – there is a section about this further down this guide.
 - **DeletePendingsDuringWideSpread:** the spread can widen dramatically during news events or when something spooks the markets. This tells HGBnD to delete unfilled stop orders during a wide spread event, and resend them when the spread returns to normal. The spread filter is a part of all my EA's and there is a section describing how it works further down.
 - **DeletePendingsOnNewSignal:** tells HGBnD to delete unfilled stop orders when the trend changes – only applies to unhedged positions and positions with no market trades.
- **Offsetting:** this involves using profitable trade or trades from one part of the grid to close in conjunction with a loser elsewhere. Here is an example of the most simple form of offsetting:
 - you have four buy trades open and the distance between them is 10 pips:
 - trade 4 has just opened and so is the spread in the red.
 - trade 3 is +10.
 - trade 2 is +20
 - trade 1 is +30.
 - the market starts to reverse. It goes all the way down past trade 1; what was once a beautifully profitable position is now in a stonking loss. Instead of this, offsetting kicks in then the market has fallen by one grid size; 10 pips in this example.
 - So now:
 - trade 4 is -10.
 - trade 3 is at breakeven.
 - trade 2 is +10
 - trade 1 is +20.
 - we close trade 4 and trade 1 at an overall profit of +10 and DD is reduced considerably if the market continues to fall.
- **UseOffsetting:** turns this feature on and off.
- **AllowComplexSingleSidedOffsets:** this kicks in when there are a lot more trades open. It allows for more than one trade to close at the profitable end of the position if that is required to close the loser at the other end.
- **MinOpenTradesToStartOffset:** the minimum number of market trades that must be open before offsetting is allowed.
- **FillInGaps:** offsetting leaves gaps in between groups of market trades when the market yoyos.
 - imagine this scaenario:
 - HGBnD has just responded to a buy signal, sent a market buy, 10 buy stops and 10 sell stops.
 - 4 of the buys fill, then the market reverses right down and fills 4 sells;
 - offsetting closes 2 of the buys on the way down, leaving 2 open.
 - The market rises and offsetting closes two of the sells.
 - This leaves perfectly hedged position of two buys and two sells with a large gap in between them. There is also a gap in between the top

buy and the next buy stop, and the bottom sell and the next sell stop. The market has to travel a long way in either direction to fill the next stop order and unbalance the hedge, and then a lot further still to reach a hedged TP. We can end up with several gaps and a number of groups of hedged trades a loooooong way from one another if the market yoyos badly.

- Here is how HGBnD fills in the gaps. Imagine the scenario I described of the simple offset of one loser at the top of the grid by one winner at the bottom:
 - he makes a note of the price of the losing trade being closed, and replaces it with a fresh buy stop in case the market reverses again and continues to rise – the gap is filled to take advantage of this.
 - Suppose instead the market falls and the first sell stop fills. The market is now (DistanceBetweenTradesPips x 2) pips away from the lowest buy trade – a gap. HGBnD fills the gap by adding a new buy stop at the lowest buy trade's opening price minus DistanceBetweenTradesPips – again the gap is filled.
- **ATR for grid trading:** “Average True Range” is an oscillator that calculate the average pips movement over a specific time frame and number of candles. Find it in Platform/Navigator/Indicators/Oscillators. It calculates DistanceBetweenTradesPips as (ATR / GridSize).
 - **UseAtrForGrid:** enables dynamic calculation of DistanceBetweenTradesPips and will override you manual setting.
 - **GridAtrTimeFrame:** the time frame you want for the calculation.
 - **GridAtrPeriod:** the number of candles to involve in the calculation. This and the GridAtrTimeFrame give a default of 4 weeks of trading.
 - **GridAtrMultiplier:** traders tend to use multiples of ATR in their trading. This input allows you to do the same.
 - **MinimumDistanceBetweenTradesPips:** some pairs have small ATR values and dividing them by GridSize can result in a <0 result if you are careless when loading up the bot. HGBnD will force DistanceBetweenTradesPips to adopt this value should the calculation result in <0.
- **Basket take profit:** grid traders tend to regard the trades in the grid as a 'basket' and so have global take profits and stop losses:
 - **BasketTakeProfitPips:** a 'hard' take profit figure. Close the position when the pips profit reaches this figure.
 - **UseAtrForBasketTP:** this tells HGBnD to calculate the pips TP dynamically using ATR. This overrides the BasketTakeProfitPips input.
 - **TpPercentOfAtrToUse:** the percentage of ATR to use for the take profit. For example:
 - at the moment, GJ D1 ATR is roughly 430 pips a day.
 - My TpPercentOfAtrToUse input is 50, so I want a basket TP of 50% of ATR, so:
 - my GJ basket TP is 215 pips.
 - **RestockGridAfterTP:** this tells HGBnD to resend a grid after taking profit. Hedged grids will be sent automatically. Single-sided grids will only be sent if the dot is the correct colour – I think.
 - **UseDydynamicClosure:** this is a really imaginative way of lowering drawdown by not allowing winners to turn into losers. This is a brilliant contribution by dydynamic and takes care of situations where price keeps rallying or dropping with little or no retracement and suddenly drops but cannot be offset even if some of the trades are in profit and greater than the

BasketTakeProfit value:

- Let's assume the following:
- Buy trades Opened = 50;
- Sell trades Opened = 5;
- 40 out of the 50 buy trades are loss trades and below the 5 sell trades that are in profit as the market goes down.....
- price turns around and 25 out of the 50 buy trades are now in profit but still way below the 5 sell trades that are still in profit.....
- If the 25 buy trades in profit are greater than our BasketTakeProfit, Dottybot should close the 25 buy trades leaving the remaining 25 buy trades that are still loss trades....

- **Safety features**

- **MinMinutesBetweenTrades:** is the post close sleep period. The EA scans the order history tab for the latest closed trade, and will not attempt to trade again until MinMinutesBetweenTrades has passed.
- **MinMinutesBetweenTradeOpenClose:** occasionally, a strange concatenation of coding bug, poor logic or criminal behaviour by your market maker will cause trades to open then close immediately for the loss of the spread. This can occur hundreds of times, draining your account. The EA scans your platform's History tab and triggers a 'rogue trade' event if it finds a trade that has opened and closed within MinMinutesBetweenTradeOpenClose. You will hear an alert and the EA will show a flashing message that says, " ***** ROBOT SUSPENDED. POSSIBLE ROGUE TRADING ACTIVITY. REMOVE THIS EA IMMEDIATELY ***** "
- **'Stealth technology'** The EA can hide your real stop loss and take profit from your criminal:
 - **HiddenPips:** the EA sends a 'hard' stop loss and take profit with the trade. Your inputs have HiddenPips added to them. The EA draws sl and tp lines on your chart at the correct values, and closes the trade when one of the lines is crossed. It will respond appropriately if you move the lines manually.
- **CSS:** Currency Slope Strength measures the strength of each individual currency in the pair. This utterly stunning indicator was coded by Baluda and is a breathtaking addition to our trading armoury. It offers us the facility to buy a pair only when the first currency in the pair is getting stronger and the second weaker. Vice-versa for a sell. Go to Bob's thread at <http://www.stevehopwoodforex.com/phpBB3/viewtopic.php?f=38&t=1481> and read "10.4 CSS info page.pdf" to read details of how this indi works, but to summarise, the indi gives you this information about each of the two individual currencies that make up the pair:
 - > 0 and the angle of the representative line is rising: currency is strengthening and the rate of strengthening is accelerating.
 - > 0 and the angle of the representative line is falling: currency is still strengthening but the rate of strengthening is slowing.
 - < 0 and the angle of the representative line is falling: currency is weakening and the rate of weakening is accelerating.
 - < 0 and the angle of the representative line is rising: currency is weakening but the rate of weakening is slowing.
 - **Inputs** – these need to be the same as those you use with the indi:
 - UseCSS: turns this filter on/off.
 - AutoSymbols: tells the indi to detect the symbols offered by your criminal automatically and include them all in the strength calculation..

- symbolsToWeigh: the list of symbols to use in the strength calculation if AutoSymbols is 'false'.
- MaxBars: cpu processor saver. Maximum no of candles over which to make the calculations.
- AddSundayToMonday: for those of you with a Sunday candle, converts this to represent the open of the Monday candle.
- TimeFrame: explained in the inputs instructions.
- IgnoreFuture: not a clue what this is about, so leave it alone.
- **Trading hours.** This is fantastic functionality provided by Baluda.
 - Sort these out for yourself, using the information here. *Never* ask questions about it in my threads. You will regret doing so should you ignore this warning.
 - use the 24 hour clock format.
 - Enter trading periods in your own local time. Forget broker server time and GMT offsets.
 - Enter as many trading times as you want:
 - precede trading start times with a '+' e.g. +08.00
 - precede trading stop times with a '-' e.g. -13.15
 - separate each value with a comma. **Do not leave spaces.**
 - An example. Imagine you want to trade between 7 and 11 am and 1 and 5.30 pm, your input will look like this:
 - +07.00,-11.00,+13.00,-17.30
 - The next four inputs offer some extra controls over individual days. These you need to enter in your broker's server time, 24 hour clock:
 - **SundayStartTradingTime:** for those of us still on Sunday night when Sydney opens after the weekend.
 - **MondayStartTradingTime:**
 - **FridayStopTradingHour:** HGBnD will send no more new-signal trades after this time is reached. The idea is to have as few trades open over the weekend as possible.
 - **SaturdayStopTradingTime:** for those of you in upside down land.
- **Rollover time:** 'rollover' is the time that brokers update your swap costs/rewards. This happens at 22.00 London time. The spreads typically widen a lot and can cause a lot of difficulty for an EA managing trades, or trying to send/close/delete them. It is best to have an ea disabled during this period.
 - **DisableDottyDuringRollover:** turns EA disabling on/off. Defaults to on.
 - **RollOverStarts:** disable HGBnD at this time.
 - **RollOverEnds:** restart HGBnD at this time.
- **Trade balance filters:** these help you to avoid entering trades that could prove detrimental if the market turns against you for a particular currency, and to avoid entering trades at news-release times.
 - **UseZeljko:** named after Zeljko who corrected the code for this filter and made it work. This filter ensures 'balanced' trading. The easiest way to describe it is to use hypothetical trades. Imagine that:
 1. this EA buys GBPUSD.
 2. this EA buys GBPJPY. You are now heavily exposed to GBP.
 3. Something unexpected happens (and something unexpected *a/ways* happens in Forex) and the pound plummets, dropping like a stone into the abyss. Both your trades scream into huge drawdown.
 4. To avoid this, having taken the GBPUSD trade, this EA will not buy another GBPxxx pair. Imagine instead that a Sell GBPJPY trade arises,

just before the market plummets. This time, your GU trade is screaming into the abyss, but your GJ trade is going stratospheric. One trade 'balances' the other in the event of something dramatic happening.

- **OnlyTradeCurrencyTwice:** works in conjunction with UseZeljko. Again, imagine the above scenario nos 1 & 4. Now you have a GU Buy and a GJ sell open – perfectly balanced trades. If you now further trade any pair involving GBP, you will unbalance your trading again, leaving you exposed to unexpected events. This filter prevents a third trade being opened involving GBP.
- Note: both UseZeljco and OTCT work more deeply than the example I have just given. For example, now you have a GU buy open, balanced trading does not allow a further buy xxxUSD trade to open, only a sell xxxUSD – then OTCT kicks in again..... Got a headache yet?
- **Swap filter:** some pairs have dreadful swap in one direction. This filter allows you to avoid trading pairs in the direction that would cost a fortune in swap. How much relevance this has to a system that could follow a trend for hundreds of pips is open to debate, but once you have seen one of these adverse-swap pairs hang around going nowhere for a couple of weeks, you will see why a lot of us want nothing to do with them.
- **Margin checks:** these help avoid over-trading by limiting the number of trades that can be opened. this EA will make the calculations before sending a trade and abort if there is insufficient margin to allow further trading. There are two to chose from; the default indicates my preference.
 - **Scoobs** check: scooby-doo is a former pro trader with the big banks; we have benefited hugely from his advice. This filter compares the current account margin with the free margin divided by 100, and aborts the trade if the margin is greater than the result of this calculation.
 - **ForexKiwi** check. ForexKiwi contributed this filter. It looks at the margin percent figure and aborts the trade if yours is less than the figure you specify in FkMinimumMarginPercent.
- **Average spread inputs:** We do not want an EA trading during a stop hunt. To learn about the crim's stop hunting tactics, go to CJ's thread at <http://www.stevehopwoodforex.com/phpBB3/viewtopic.php?f=59&t=1572>. There is code to calculate the average spread and store it for retrieval every time you restart the bot.
 - The first time you run the ea it will take time to calculate the average spread. The screen will display a message telling you how far into the process it is.
 - You may want to recalculate the average spread. The easiest way to force a recalculation is to delete the Global Variable that stores the average. Perform this when you know there are no major news events coming up.
 - Press the F3 key to open the Global Variables window.
 - The relevant GV's begin with the chart symbol then " average spread" e.g. "EURUSD average spread". Highlight the GV and delete it. Close down and restart your platform and RRTnP will detect the need for a recount.
 - **TicksToCount:** the number of ticks to use as the averaging period.
 - **MultiplierToDetectStopHunt:** this multiplies the average spread and pauses the bot if the spread exceeds this.
 - The EA keeps a running tally of the spread and recalculates the average every 500 ticks. The chart includes a display of the average, along with a notification of the widest spread since the EA was last started/restarted.
- **Chart snapshots after opening and closing trades:**
 - The EA can take a picture of your chart when it opens or closes a trade. Use

this to take pictures if you suspect that the EA has not behaved as expected. There is a section of inputs just underneath the minimum pips section, with inputs to turn this feature on/ off and to adjust the height and width of the snap. Files are saved with the name "ChartScreenShot" and the ticket number, when the EA opens or closes a trade. Navigate to the files via File/Open Data Folder/MQL4/Files. Remember to delete these files from time to time to avoid clogging up your disk drive.

- **Email thingies:** these features need enabling via Tools|Options on your platform.
 - **EmailTradeNotification:** tells the EA to email you an alert when it has sent a trade.
 - **SendAlertNotTrade:** tells the EA not to send a trade when it discovers a trigger. Instead it will:
 - sound an alert on your platform.
 - Send you an alert via email.
 - **AlertPush:** sends either/both the above alerts and sends a 'push' to the latest mobile devices such as iPad etc.

Trade Management

The remaining inputs are all about individual trade management. Management is a cut-down module from Multi-purpose trade management EA available from <http://www.forexfactory.com/showthread.php?t=89371>. The full management EA has a wide range of extra features, and there is a User Guide to describe it.

Features included here:

- **Break even settings:** set a break even stop loss after the price reaches the setting in BreakEvenPips. You can use this in conjunction with the Part-closure routine (details later), as well as a stand-alone routine.
 - **BreakEven:** set to true to enable this facility.
 - **BreakEvenPips:** the number of pips you want the market to move in your favour before setting the stop loss to the order entry price, ensuring the trade cannot turn into a loss.
 - **BreakEvenProfit:** will add this to the stop loss to a buy order, subtract it for a sell order. My default of 2 pips means the sl is set to break even + 2, ensuring a minimum of 2 pips profit. Set it to 0 if you do not want this feature.
 - **HalfCloseEnabled:** will close half the trade when the market reaches your breakeven point. It is up to you to ensure that your lot size allows this.
- **Jumping stop loss settings:** this will jump the stop loss by JumpingStopPips when the price moves in your favour by that number of pips. Many traders consider this to be a better option than a straight trailing stop. The first time this option is triggered by the market price, it will set the stop loss to break even. After that, it will increment the sl by JumpingStopPips every time the market moves sufficiently in your favour.
 - **JumpingStop:** set to true to enable this facility.
 - **JumpingStopPips:** the number of pips to jump. For example, my default of 300 works like this:
 - Market price hits order open price + 300: moves sl to break even.
 - Market price hits order open price + 600: moves sl to + 300.
 - Market price hits order open price + 900: moves sl to + 600.
 - **AddBEP:** adds BreakEvenPips to the break even if set to 'true'.

- **Candlestick jumping stop.** This jumps the stop at the close of a candle:
 - **UseCandlestickTrailingStop:** turns this on/off.
 - **CstTimeFrame:** this allows you to use a different time frame to that of the chart. Use integer values to correspond with your chosen time frame I.e. 1 for M1, 240 for H4, 1440 for D1 etc.
 - **CstTrailCandles:** the number of candles ago to use as the trail. For example, you are in a buy trade and want the stop loss to trail the default of 1, then the EA will set the stop at the low of the previous candle, so long as this is higher than the current stop loss.
 - **TrailMustLockInProfit:** tells the EA to start moving the stop loss only when it will be moved to > break even.
- **Trailing stop loss settings:** works like the conventional trailing stop you can enter into the MT4 platform.
 - **TrailingStop:** turns this on/off.
 - **TrailingStopPips:** your trail distance.

Chart feedback display.

The shells place information about the EA's inputs on the chart. You have the option to use the general Comment text, which you cannot personalise, or the text function provided by Paul Bachelor (lifesys) at SHF.

- **DisplayAsText:** tells the EA to use Paul's display function. This puts text into labels drawn on the chart; these sometimes split words and so sometimes appears a little strange.
- **KeepTextOnTop:** stops the chart candles from obscuring Paul's text.
- The remaining inputs control the start point, font and colour of the text in Paul's labels. Play with these to personalise your feedback.

Disclaimer and Risk Disclosure:

Trading foreign exchange on margin carries a high level of risk, and may not be suitable for all investors. The high degree of leverage can work with as well as against you. Before deciding to invest in foreign exchange you should carefully consider your investment objectives, level of experience, and risk appetite. The possibility exists that you could sustain a loss of some or all of your initial investment and therefore you should not invest money you cannot afford to lose. You should be aware of all the risks associated with foreign exchange trading, and seek advice from an independent financial advisor should you have any doubts.

I will put this a tad more bluntly:

Most Forex traders lose all their money.

- Using this EA in trading Forex does not guarantee success.
- Trading with this EA could lead to serious financial loss.
- Trading this EA without understanding its underlying trading strategies *guarantees* traders will lose their money.

Good luck. Have fun.