



**Omnesys Technologies**



**NEST Strategy Script  
(Conversion Reversal Bidding Strategy)**

May 14<sup>th</sup>, 2014

**Version 2.0.0.1**

## Document Information

DOCUMENT CONTROL INFORMATION	
DOCUMENT	Nest Strategy Script.
VERSION	2.0.0.1
AUTHOR	RUSHABH DOSHI
REVIEWER	BRIJESH DAMODAR / ATHUL KUDVA
REVIEW NOTES	
VERSION NOTES 2.0.0.1	This document explains the Conversion Reversal Bidding Strategy functionality in BFO segment.
KEYWORDS	

### **Proprietary Notice**

This document contains confidential information of Omnesys Technologies Pvt Ltd, which is provided for the sole purpose of permitting the recipient to evaluate the deployment document submitted herewith. In consideration of receipt of this document, the recipient agrees to maintain such information in confidence and to not reproduce or otherwise disclose this information to any person outside the group directly responsible for evaluation of its contents, except that there is no obligation to maintain the confidentiality of any information which was known to the recipient prior to receipt of such information from Omnesys Technologies Pvt Ltd, or becomes publicly known through no fault of recipient, or is received without obligation of confidentiality from a third party owing no obligation of confidentiality to Omnesys Technologies Pvt Ltd.

## Conversion Reversal Bidding Strategy (BFO):

This Conversion Reversal strategy allows trading in BFO Options (One Call and One Put of same strike) and future segments (One Futures).

### Nest Strategy Client:

**Portfolio Tree:** Each portfolio comprises of 3 tokens (i.e call, put for same strike, same expiry of a particular symbol and the future (of same or different expiry) for the same symbol).

### Input Parameters:

**Pro/Cli :** It gives user an option to do in PRO or in CLI. When cli is selected will give an option to select Account id and Participant id.

**Exch-Seg:** User needs to populate relevant exchange. For this strategy, its BFO.

**Symbol-Opt:** User needs to select relevant symbol for that particular exchange for the option token.

**Expiry Date Opt:** User needs to select the relevant expiry date for the options (both for call and for put).

**Strike:** User needs to select the relevant strike for the options both for call and for put).

**Symbol-FUT** User needs to select relevant symbol for that particular exchange for the futures token.

**Expiry Date-FUT:** User needs to select the relevant expiry date for the future token for the selected symbol. It can be same or different from option expiry date.

**Type:** It has two options: Conversion or Reversal. If Conversion is selected, then Sell call, buy put at same strike and same expiry, and buy future of same or different expiry will take place. If Reversal is selected then buy call, sell put at same strike and same expiry, and sell future of same or different expiry will take place.

**Bid Type:** This parameter is exclusively for the first leg. It has three options: CALL, PUT & FUT. It is this parameter that decides which leg will be the first leg (bidding leg). Price of the bidding leg is derived based on the strike rate, and the price of the other two legs. It also depends on whether the user has selected type as Conversion or Reversal.

**First Leg Type:** There are two options in the order type: Best Bid/Ask and Sweep/Stand.

- a. **Best Bid/Ask:** If user selects Best Bid/Ask, for placing first leg, our system will calculate the first leg price depending on the limit specified by the user, the strike of the option as well as the rate of second and third leg to check whether rate available in the market is better than the derived price. After first leg is completed, the system will place the remaining leg depending on the option selected for the second leg type. For example, If the user is buying the first leg and the calculated first leg price is greater than the market rate of the first leg, then it will modify the current bid rate

by 1 tick size to become best buyer (bid rate + 1 tick size). If calculated first leg price is less than the market rate of the first leg price, it will stand at that price and will not bid to become best buyer. Similarly, if the user is selling first leg and calculated first leg price is less than the market rate of the first leg, it will modify the current first leg price (ask rate) by 1 tick size to become best seller (ask rate – 1 tick size). If calculated first leg price is more than the market rate of the first leg, then it will stand at that price and will not try to become a best seller. **It should be noted that bidding to become best buyer and best seller will continue till the calculated first leg price (derived price). So calculated option price becomes a floor/ceiling for bidding.**

- b. **Sweep/Stand:** If user selects Sweep/Stand Price, then the strategy will place a single day order for the first leg price depending on the derived price calculated based on the limit specified by the user, strike price of the option and the second & third leg market price. After first leg is completed, the system will place the remaining leg depending on the option selected for the second leg type. For placing the first leg, our system will calculate the price and place the order in the market at the calculated price but **will not try** become best buyer or best seller.

**Second Leg Type:** This parameter is exclusively for the remaining legs. Once the first leg gets completed, how should the remaining legs (second leg and third leg) be placed depends on this parameter. It has four options to place the second leg and are mentioned below:

1. **LTP Based MPP%:** Once the first leg is completed, second leg and third are placed based on the LTP +/- MPP%. In other words, let say, user is selling first leg, and second leg is the buy order, it will take the LTP of that scrip add x% specified in the MPP% parameter and then place the order. If for any reason it is not completed, and there is a change in LTP, it will change the order accordingly. Similarly, let say, user is buying first leg, and second leg is the sell order, it will take the LTP of that scrip subtract x% specified in the MPP% parameter and then place the order. If for any reason it is not completed, and there is a change in LTP, it will change the order accordingly. For any reason, if this option is selected, and user stops the strategy, the pending second leg order and/or the pending third leg order will be canceled.
2. **Opposite:** Once the first leg is completed, second and third leg is placed as limit orders with the Opposite rate. For example, if one is buying, then ask rate will be taken as price for limit order and if one is selling, then bid rate will be taken as price for the limit order.
3. **Same:** Once the first leg is completed, second and third leg is placed as limit orders with the Same rate. For example, if one is buying, then bid rate will be taken as price for limit order and if one is selling then ask rate will be taken as price for the limit order.
4. **Stored Price:** Once the first leg is completed, second and third leg is placed as a limit order at the price that was used to calculate the first leg price.

**Timer:** This is exclusively for the remaining legs (second leg and third leg). If the Leg type for the 'Second Leg Type' is selected as Opposite, Same, or Stored Price, timer will become

applicable. After timer expiration, any open orders for second and/or third leg will be converted to limit order whose price is based on LTP Based MPP%. Timer starts as soon as the second and third leg is placed by system.

**MPP%:** This parameter is applicable for the Second leg.

For Second/Third Leg **Buy** Orders:

$$\text{Placed Second/Third Leg Buy Price} = \text{LTP} + (\text{LTP} * \text{MPP}\%)$$

For Second/Third Leg **Sell** Orders:

$$\text{Placed Second/Third Leg Sell Price} = \text{LTP} - (\text{LTP} * \text{MPP}\%)$$

For call, put and futures, mpp% parameter is provided for each leg. However, based on the selection of first leg type, corresponding mpp% parameter will be disabled accordingly.

**Limit (Rs.):** It is the desired spread that needs to be specified by the user. Based on current market price of the second leg, and third leg, along with the limit specified by the user, the price for the first leg is calculated and will enter the market. As soon as the price of the either or both legs changes, the price of the first leg (bidding leg) will change accordingly. For example:

Let say user is doing a conversion with call as the first leg with limit specified by the user, so the price for the first leg will be calculated as:

$$\text{Call Option Rate} = \text{Limit} - \text{Strike Rate} + \text{Future Ask Rate} + \text{Put Option Ask Rate}$$

If the user was doing a reversal with call as the first leg with limit specified by the user so the price for the first leg call option would be calculated as:

$$\text{Call Option Rate} = \text{Future Bid Rate} + \text{Put Option Bid Rate} - \text{Strike Rate} - \text{Limit}$$

Similarly, depending on whether one is doing conversion or reversal and the first leg selected, the above formula can be tweaked to calculate the price of the first leg (bidding).

**Quote Threshold (in Rs):** The order is placed only when the market spread is more than or equal to Quote threshold. The idea is to start bidding when the Quote threshold is breached. It is only applicable during placement of orders and not considered during modification of orders.

**Market Protection Type:** is useful to prevent user from manual/typing errors. It has three options:

1. **In Percentage:** is useful to prevent user from manual/typing errors. For example, let's say user is doing a conversion and the current market conversion rate comes out to be 10, and user has entered market protection as 30% and if user has entered the limit as 6, the system will not allow the orders to be executed (placed) below Rs. 7. Similarly, if the calculated market rate comes out to be -10, and user has entered market protection as 30%, and entered limit as -14, then the system will not allow

the orders to be executed (placed) below Rs. -13. In other words, it will prevent the user from the downside risk by as much as 30% during the initial placement of orders.

2. **Absolute:** The orders will not be generated if the set limit is beyond the market protection limit. Based on the value entered, it will create a band of Current difference +/- Market protection limit. If the set limit is beyond this band it will not allow any orders to be placed.
3. **No (Market Protection):** If this option is selected, then the Market protection will not be considered before placing the orders.

**Market Protection:** This is where the value needs to be entered for market protection after selecting market protection type. If market protection type is selected as no, this parameter field will be disabled. In order to apply the market protection, one needs to calculate the current market conversion rate and/or current market reversal rate to compare with the limit specified by the user. Following formulae are used to calculate the market conversion rate and market Reversal Rate:

For Conversion, since we are buying future and buying put while selling the call, we consider ask rate for future and put option while bid rate for the call option to calculate the current conversion value:

$$\text{Calculated Market Conversion Value} = \text{Strike Rate} + \text{Call Option Bid Rate} - \text{Future Ask Rate} - \text{Put Option Ask Rate.}$$

For Reversal, since we are selling future, buying call and Selling put, we consider bid rate for future and put and ask rate for Call to calculate the current reversal value:

$$\text{Calculated Market Reversal Value} = \text{Future Bid Rate} + \text{Put Option Bid Rate} - \text{Strike Rate} - \text{Call Option Ask Rate.}$$

**Order Lots:** This is the quantity (in lots) to be placed per opportunity and can be multiple lots but it has to be a multiple of total order lots.

**Total Lots:** This is the total trade quantity (in lots) that the user intends to trade.

**Tick Size (Rs.):** It is a minimum tick size for the selected token. It is separate for each token (Call, Put and Futures).

**Tick Mod (Rs.):** is considered for bidding and is used to decrease the number of modifications. If the difference between current order price and new bidding price is greater than or equal to the Tick Mod then only order is modified with the new bidding price. It is applicable for first leg only.

**Threshold Qty (%):** It is used to check whether a mentioned percentage of the order lot is available for the second leg & third leg for the specified market Depth. If it is available then only system will place/bid for the first leg. If the specified threshold quantity is not available in either or both legs for the specified depth, no first leg orders will be placed. Even during modification of first leg, quantity checks are done for both legs. If any time

during the modification, if the quantity becomes unavailable, the existing first leg open order will be cancelled.

**Depth:** The threshold quantity % will be checked for the second leg & third leg for the specified market depth. It specifies till how much depth one must check for the order lot quantity. It should be noted that although depth is 5, but if the quantity is available at lower depth (say 3<sup>rd</sup> depth) itself, it will take weighted average accordingly and then calculate the first leg price. In other words, the strategy will not calculate the weighted average price for the particular leg till the depth specified, but will calculate the weighted average price based on the quantity available and price at which it is available.

**Nest Strategy Engine:** The execution of the script happens in the Nest Strategy engine running at backend.

**Execution:**

1. Based on the type selected (conversion or reversal), bid type selected (call, put or Fut), First Leg Type selected and the limit specified by the user, the user entered rate is compared with the market calculated rate. If the market protection condition is satisfied based on the selected market protection type (Percentage, Absolute, or no Market Protection), the system will place the first leg as limit order in the market. Before placing the order it will also check whether sufficient threshold qty % for both second leg and third leg is available in the user-specified market depth. It will also check whether the quote threshold is breached during initial placement of orders. If yes, then system will place the limit order for the first leg.
2. Based on the change in the second and/or third leg price, the price of the first leg will change accordingly, taking the user entered limit into consideration (using above-mentioned formulae) and subject to the tick mod condition being satisfied. During modification too, quantity check is done. At any given point of time, during modification, if the quantity specified in the order lot (taking into consideration threshold qty% and market depth) is not available, system will cancel the first leg open order.
3. Once the first leg order is completed, system will place the second leg and third leg orders depending on the 'Second Leg Type' selected. If the leg type selected is other than LTP Based MPP% then system will place 2 limit orders for the second and third leg depending on the option selected. As soon as the limit orders are placed, timer for both the remaining legs starts. After expiration of timer, any of the open limit orders are converted to limit order whose price is based LTP Based MPP%. For Second leg type as LTP Based MPP%, MPP% parameter field is enabled for second and third leg, and when the first leg is completed, the second leg is placed based on the LTP Based MPP%. For second/third leg buy order, it will add MPP% to the LTP and place it and for second/third leg sell order, it will subtract MPP% to the LTP. If for any reason, the orders are still open and LTP changes, the placed price will be modified accordingly. For LTP Based MPP%, any open second leg/third leg order will get cancelled, when the strategy is stopped.

**Portfolio and Parameters Front-end Screenshot:**

Conversion Reversal Bidding Strategy Version 2.0.0.1

PortFolio Name	Exch-Seg	Symbol	Expiry Date Opt	Strike Price	Expiry Date Fut	Type
						CONVERSION

Bid Type	FirstLegType	SecondLegType	Mkt.Prot.Type	Mkt. Prot	Order Lot	Total Lot	Account Details		
CALL	BestBid/Ask	Same	Percentage	30	0	0	Pro/Cli	A/C ID	Part ID
							PRO	11365	

Limit (Rs)	Thr. qty	Timer	Depth	Tick Size	Qt Thr	Tick Mod	Trade Lots	Pending Lots	MPP% Fut	MPP% Call	MPP% Put
0	100	1	5		0				0.05		0.05

Save Cancel



## Risk Management

Risk Management System can be configured to have following checks before the orders are released to the exchange. The checks which are defined by exchange with respect to Algorithms are in place in the system. Below mention rules can be configured in the system to control the risk parameter which is defined by exchange.

Sr.No.	Checks	Rules to set	Remarks
1	Price Check	Check Price Range Based on LTP Check Circuit Limit	This rules will create a price range on the basis of Last Traded Priced as per the percentage set in the category window.
2	Quantity Check	Order Quantity including Square off Order Board Lot Quantity including Square off Order	This rules will restrict per order the number of quantity to be placed in market which is defined the category window. The user can define the number of quantity in Weights and in lots for Futures.
3	Order Value Check	Order Value including Square off Order	This rule will restrict per order the order value which can be placed in the market which is defined in category window
4	Trade Price Protection Check	Check Circuit Limit including square off order	This rule does not allow to place the order which has been placed above the Higher Circuit Limit or Lower circuit limit which id defined for contract/scrip by exchange
5	Market Price Protection	Check Price Range Based on LTP	This rule will create a price range on the basis of Last Traded Priced as per the percentage set in the category window.
6	Cumulative Open Order Value check	Pending order value	This rule will restrict the Open Order with the Value set in the category
7	Automated Execution Check	Turnover Order Level and Turnover Order Level Limit	This rule will calculate the value of all executed/ Unexecuted and un confirm orders and if breach the value set in category then further order will get rejected
8	Automatic stoppage in event of Algo execution leading to a loop or a runaway situation.	Order Throttle	If there number of order per seconds breaches the value which is set in Throttle then further order gets rejected by the system.
9	Net Position Vs. available margin	Gross Exposure, Gross Exposure Derivative, Var	User can set the risk parameter based on Exposure and Margin based on which the margin used will be

		Margin Order Level, Span Margin Order Level	calculated on the basis of position taken. If the Margin used is equal to Cash margin then further order will be rejected by the system
10	RBI Violation checks for FII Restricted stocks.	Restricted Basket or RMS Blocking	User need to create a Restricted basket for the scrip and assign to the category of the user / client. Also RMS blocking can be used.
11	MWPL violation check	RMS Ban Symbol or RMS Blocking	The scrip for which market wide position limit is breach then scrip can be blocked or it needs to be in Ban.
12	Position Limit Checks	Scrip Group / Scrip Margin	User can define the quantity scrip wise in which the position cane be taken in scrip group and then it needs to be assign to category at client level
13	Trading Limit Checks	Scrip Group / Scrip Margin	User can define the quantity scrip wise in which the position cane be taken in scrip group and then it needs to be assign to category at branch level
14	Exposure Limit check at individual client level and at overall level	Gross Exposure and Gross Exposure Limit	User can define the Exposure at Branch Level as well as Broker Level
15	Number of orders for the logic	NA	Depends on the user parameter (i.e order qty) set. This can vary from a minimum of a single lot to a maximum of total qty set.
16	Maximum number of scrips / contracts in which the logic will work at a time	NA	At a time, maximum number of scrips/contracts in which logic will work at a time is 3.
17	Number of legs		Two or Three