

NIKA EA V1.27

# Re-Entry

## Comprehensive Technical Documentation

COMPREHENSIVE ANALYSIS

MetaTrader 5 Expert Advisor  
Signal Engine • Risk Management • Trade Automation

**Copyright © 2026 NIKA. All Rights Reserved.**

February 2026 • Publication-Quality Reference

Prepared by Perplexity Computer

# Table of Contents

<b>Table of Contents</b>	<b>2</b>
<b>Executive Summary</b>	<b>8</b>
<b>System Architecture</b>	<b>10</b>
Three-File System . . . . .	10
Signal Generation Pipeline . . . . .	10
Heiken Ashi Preprocessing . . . . .	11
Linear Regression Deviation Calculation . . . . .	11
Moving Average Smoothing . . . . .	11
Entry Type System . . . . .	12
<b>Signal Engine Deep Dive</b>	<b>13</b>
Rule System Architecture . . . . .	13
Signal Combination Logic . . . . .	13
Mode 0: Nika Lines Crossover (NikaCross) . . . . .	14
Mode 1: Nika Signal Line Level Cross (NikaSigLevCross) . . . . .	14
Mode 2: Histogram Level Cross (HistLevCross) . . . . .	14
Mode 3: Histogram Moving Up/Down (HistUD) . . . . .	15
Mode 4: MA Crossover (MACross) . . . . .	15
Mode 5: Two Nika Signal Lines Crossover (2SigCross) . . . . .	15
Mode 6: Two Nika Signals Cross Average (2SigAvgCross) . . . . .	16

Mode 7: Signal Crosses Two Averages (Sig2AvgCross) . . . . . 16

Mode 8: Signal Line Moving Up/Down (SigUD) . . . . . 16

Mode 9: Average Line Moving Up/Down (AvgUD) . . . . . 16

Mode 10: Signal & Average Moving Up/Down (SigAvgUD) . . . . . 17

Mode 11: MA Price Cross (MAPriceCross) . . . . . 17

Multi-Timeframe Capabilities . . . . . 17

**Moving Average Library 18**

T3 Volume Factor . . . . . 19

Applied Price Types . . . . . 19

Choosing the Right MA Type . . . . . 19

**Trade Management System 21**

Entry Mode . . . . . 21

Direction Filter . . . . . 21

Position Sizing . . . . . 21

Maximum Entry Controls . . . . . 22

Hedge Mode . . . . . 22

Close on Opposite Signal . . . . . 22

Spread and Commission Compensation . . . . . 22

**Stoploss Methods 24**

Method 0: Fixed Pips . . . . . 24

Method 1: ATR Multiplier . . . . . 24

Method 2: Average Price Promille . . . . . 24

Stoploss Steps and Breakeven . . . . . 25

**Take Profit System 26**

TP Level Configuration . . . . . 26

How Partial Closes Work . . . . . 26

TP Interaction with Trailing . . . . . 27

Suggested TP Configurations . . . . . 27

**Trailing Stop System 28**

Mode 0: m-aslt2 (RRR-Based Stepping) . . . . . 28

Mode 1: m-pppt (Percentage Pullback) . . . . . 28

Choosing Between Trailing Modes . . . . . 29

**Re-Entry System 30**

How Re-Entries Work . . . . . 30

Series Counting . . . . . 30

Infinite Mode Safety Considerations . . . . . 30

Cooldown Interaction . . . . . 31

**Scale-In System 32**

10 Trigger Levels . . . . . 32

Positive vs. Negative Triggers . . . . . 32

Range Variation . . . . . 32

Scale-In Example Configuration . . . . . 32

**Risk Management Framework** **34**

Complete Risk Layer Overview . . . . . 34

Layer 1: Max Entries per Direction . . . . . 35

Layer 2: Max Entries per Day . . . . . 35

Layer 3: Re-Entry Series Limit . . . . . 35

Layer 4: Max Open Lot . . . . . 35

Layers 5-8: Daily/Weekly P&L; Limits . . . . . 35

Layer 9: Daily Account Trailing . . . . . 36

Layer 10: Hours Filter . . . . . 36

Layer 11: News Filter . . . . . 36

Layer 12: Trade Cooldown . . . . . 36

Layer 13: Profit Scale-Out . . . . . 36

Layers 14-15: Direction Filter & Spread Compensation . . . . . 37

Filter Evaluation Order . . . . . 37

**Hours & Session Management** **38**

Time Window Configuration . . . . . 38

Timezone Handling . . . . . 38

Friday-Only Close Feature . . . . . 38

Wrap-Around Sessions . . . . . 39

**News Filter** **40**

Data Source . . . . . 40

Impact Levels . . . . .	40
Pause Mechanism . . . . .	40
Auto-Close Functionality . . . . .	41
Visual News Lines . . . . .	41
<b>Dashboard System</b>	<b>42</b>
Dashboard Features . . . . .	42
Configuration . . . . .	42
<b>Strategy Tester Integration</b>	<b>43</b>
Five OnTester Fitness Functions . . . . .	43
Fitness Function Comparison . . . . .	44
Optimization Approach . . . . .	44
<b>Complete Parameter Reference</b>	<b>45</b>
Alerts Settings . . . . .	45
Global Settings . . . . .	45
Trade Settings . . . . .	45
Re-Entry Settings . . . . .	46
Stoploss Settings . . . . .	46
Max Open Lot Settings . . . . .	47
Daily Trailing Settings . . . . .	47
Profit Scaling-Out Settings . . . . .	47
Targets Settings (Take Profit) . . . . .	47

Scale-In Entry Settings . . . . . 48

Trade Cool Down Filter Settings . . . . . 49

Trailing Stop Settings . . . . . 49

Max Daily/Weekly P&L; Settings . . . . . 50

Hours Filter Settings . . . . . 51

News Filter Settings . . . . . 51

Signal Rule Parameters (Per Slot) . . . . . 51

Tester Settings . . . . . 52

Dashboard Settings . . . . . 52

**SET File Format Guide 53**

File Encoding . . . . . 53

Line Format . . . . . 53

Naming Conventions . . . . . 53

Loading in MT5 . . . . . 54

**Optimization Suggestions 55**

What to Optimize First . . . . . 55

Suggested Parameter Ranges . . . . . 55

Walk-Forward Testing Approach . . . . . 56

Multi-Symbol Considerations . . . . . 56

Common Optimization Pitfalls . . . . . 57

---

## Chapter 1

# Executive Summary

---

The Nika EA V1.27 Re-Entry is a professional-grade algorithmic trading system designed for the MetaTrader 5 platform. It represents a comprehensive, highly configurable expert advisor (EA) that combines advanced signal generation, sophisticated trade management, and a multi-layered risk management framework into a single, unified system.

## Target Audience

This EA is built for serious algorithmic traders, quantitative researchers, and professional trading operations that require institutional-level configurability without sacrificing usability. Whether managing a single account or deploying across multiple symbols and timeframes, V1.27 provides the controls necessary for disciplined, rules-based trading.

## Key Capabilities

- **12 Signal Modes:** From classic moving average crossovers to proprietary Nika signal line analysis, each mode can operate across 3 independent rule slots for layered signal confirmation.
- **11 Moving Average Types:** SMA, EMA, WMA, VWMA, RMA, DEMA, TEMA, ZLEMA, HMA, Donchian, and T3 provide a comprehensive toolkit for smoothing and trend detection.
- **Multi-Timeframe Analysis:** Each signal rule slot can be assigned to any MT5 timeframe, enabling cross-timeframe confirmation and filtering.
- **3 Entry Types:** Main signal-driven entries, automated re-entries after losses, and scale-in entries at configurable profit/loss levels.
- **15 Risk Management Layers:** Including daily/weekly P&L limits, max entry controls, news filtering, session management, and more — evaluated in a defined sequence before every trade.
- **10-Level Partial Close System:** Each take-profit level uses a multiplier of the stoploss distance with configurable partial close percentages.
- **Advanced Trailing Stops:** Two trailing modes — RRR-based stepping (m-aslt2) and percentage pullback (m-pppt) — provide flexible profit protection.
- **News Filter Integration:** Automatic trade pausing around high-impact economic events using live data from FairEconomy.
- **5 Optimization Fitness Functions:** Custom OnTester metrics enable systematic walk-forward optimization in the MT5 Strategy Tester.
- **Real-Time Dashboard:** On-chart overlay displaying live P&L, entry state, risk layer status, and news countdown.

## System Overview



At its core, the Nika EA operates through a sophisticated signal generation pipeline: raw price data passes through optional Heiken Ashi preprocessing, then a linear regression deviation calculation, followed by moving average smoothing. The resulting signal and average lines generate trade entries based on crossovers, directional momentum, or level crossings. Positions are then managed through a comprehensive take-profit ladder, trailing stop system, and multi-layered risk controls that operate continuously throughout each position's lifecycle.

The system consists of three files: the main EA (~6,900 lines), a universal indicator (~1,081 lines), and a moving average library (~332 lines). This modular architecture separates signal computation from trade execution, enabling clean maintenance and independent optimization of each component.

Chapter 2

# System Architecture

## Three-File System

The Nika EA V1.27 employs a modular three-file architecture that cleanly separates concerns between trade execution, signal computation, and mathematical foundations. This separation provides maintainability, independent optimization, and clear functional boundaries.

File	Size	Role	Responsibilities
Nika-EA-V1.27-Re-Entry.mq5	~6,900 lines	Main Expert Advisor	Trade execution, order management, risk filters, re-entry logic, scale-in, dashboard, all trade lifecycle management
Universal-Nika-Trading.mq5	~1,081 lines	Universal Indicator	Signal generation pipeline, Heiken Ashi preprocessing, linear regression deviation, MA smoothing, signal/average line computation
All_MA.mq5	~332 lines	Moving Average Library	All 11 MA type implementations (SMA, EMA, WMA, VWMA, RMA, DEMA, TEMA, ZLEMA, HMA, Donchian, T3)

The main EA file communicates with the indicator through MT5's iCustom() interface. Each signal rule slot creates its own indicator instance with the appropriate parameters, timeframe, and configuration. The indicator in turn calls the MA library for all smoothing operations. This three-tier structure means that adding a new MA type requires only a change to the library file, while new signal modes are added to the indicator, and trade logic changes are confined to the EA.

## Signal Generation Pipeline

The signal generation pipeline is the intellectual core of the Nika EA. Understanding its stages is essential for configuring optimal trading strategies.

Stage	Name	Description
1	Price Data Input	Raw OHLCV data from the selected timeframe is loaded from MT5's price history.
2	Heiken Ashi Preprocessing	When enabled per rule, standard candlestick data is transformed into Heiken Ashi candles, smoothing price action and reducing noise.

Stage	Name	Description
3	Linear Regression Deviation	The deviation of price from a midpoint is calculated using linear regression over a configurable period (the 'length' parameter). This produces a centered oscillator-like value.
4	MA Smoothing	The raw deviation values are smoothed using one of 11 available MA types with a configurable period. This produces the 'Signal Line'.
5	Average Line Generation	A second MA smoothing pass (the 'nika_ma_period' parameter) creates the 'Average Line', providing a reference for crossover detection.
6	Signal Detection	Based on the selected signal mode, buy/sell signals are generated from the relationship between Signal Line, Average Line, histogram, price, and user-defined levels.

## Heiken Ashi Preprocessing

Heiken Ashi (HA) candles are a modified candlestick representation that smooths price action by averaging open and close values with the prior candle. The HA Close is the average of the current bar's Open, High, Low, and Close. The HA Open is the average of the previous HA Open and HA Close. High and Low are the maximum and minimum of the current bar's High, Low, and the HA Open/Close values.

This preprocessing step filters out much of the intra-bar noise, producing cleaner trend signals. Each signal rule slot has an independent 'enable\_HA' toggle, allowing some rules to use HA data while others operate on raw price — a powerful multi-view capability.

## Linear Regression Deviation Calculation

The linear regression deviation component measures how far the current price deviates from its statistical midpoint over a lookback period defined by the 'length' parameter. A linear regression line is fitted to the price series, and the deviation (distance from the regression line) forms the raw signal value.

When the deviation is positive and increasing, price is accelerating above its trend. When negative and decreasing, price is accelerating below trend. The crossover of this deviation through zero or through its own smoothed average provides the foundation for most of the EA's signal modes. The 'length' parameter directly controls the lookback period — shorter values create more responsive but noisier signals, while longer values produce smoother but more lagging signals.

## Moving Average Smoothing

After the linear regression deviation is calculated, it passes through one of 11 available moving average types for smoothing. The choice of MA type significantly affects signal characteristics — HMA provides minimal lag, TEMA offers strong smoothing with moderate responsiveness, and T3 with its configurable volume factor allows fine-tuning between lag and smoothness.

The 'ma\_type' parameter selects the smoothing algorithm, 'ma\_price' selects the applied price (Close, Open, High, Low, Median, Typical, Weighted), and the T3 factor (when T3 is selected) controls the aggressiveness of the smoothing. The smoothed output becomes the Signal Line, and a second smoothing pass with 'nika\_ma\_period' creates the Average Line for crossover-based signals.

## Entry Type System

The Nika EA manages three distinct entry types, each identified by a unique magic number for independent tracking and management.

Entry Type	Magic Number	Description
<b>Main Entry</b>	1000 (default)	Signal-driven entries generated by the configured signal rules. These are the primary trades based on the signal engine analysis.
<b>Re-Entry</b>	1001 (default)	Automatically re-enters the market in the same direction after a main entry closes (by stoploss, trailing, or other exit). Designed to recapture trends after temporary adverse moves.
<b>Scale-In</b>	1002 (default)	Adds to an existing position when price reaches configured trigger levels. Supports both scaling into profit (positive triggers) and scaling into drawdown (negative triggers).

Magic numbers are configurable but must remain unique to avoid conflicts with other EAs running on the same account. The separation by magic number allows the EA to independently manage position sizes, trailing behavior, and risk calculations for each entry type.

Chapter 3

# Signal Engine Deep Dive

The signal engine is the decision-making core of the Nika EA. It provides 12 distinct signal generation modes, each available in 3 independent rule slots, creating up to 36 rule instances that can be combined using AND/OR logic. This chapter examines each signal mode in detail.

## Rule System Architecture

Each of the 12 signal modes is available in 3 slots (e.g., NikaCross\_1, NikaCross\_2, NikaCross\_3). Every rule slot has the following common parameters:

- **Enable:** Toggle this rule slot on or off.
- **Timeframe:** Which MT5 timeframe to evaluate this rule on (e.g., M15, H1, H4). Allows multi-timeframe strategies.
- **Role:** Either 'Entry' (can trigger a new trade) or 'Filter' (must agree with the direction but cannot initiate a trade by itself).

The Role system is particularly important. An 'Entry' rule actively generates trade signals. A 'Filter' rule only confirms or blocks trades proposed by Entry rules. This allows, for example, a fast NikaCross on M15 to generate entries, while a higher-timeframe MA crossover on H4 acts as a directional filter.

## Signal Combination Logic

Two parameters control how multiple enabled rules interact:

Parameter	Options	Behavior
_entry_control	ALL / ANY	ALL: Every enabled rule must agree on direction for entry. ANY: At least one enabled Entry rule must signal, and all Filter rules must agree.
_exit_control	ALL / ANY	ALL: Every enabled rule must generate an opposite signal to close. ANY: At least one opposite signal triggers a close.

In ALL mode, the system requires consensus — every enabled rule must agree on direction before a trade is opened. In ANY mode, a single Entry rule can trigger a trade, but all Filter rules must still confirm the direction. This provides two distinct strategy paradigms: high-confidence confluence trading (ALL) versus opportunistic diversified signal capture (ANY).

## Mode 0: Nika Lines Crossover (NikaCross)

The flagship signal mode. NikaCross generates buy and sell signals when the Signal Line crosses above or below the Average Line. This is analogous to a MACD crossover but operates on the Nika signal generation pipeline with configurable Heiken Ashi preprocessing and 11 MA smoothing options.

**Buy Signal:** Signal Line crosses above Average Line from below.

**Sell Signal:** Signal Line crosses below Average Line from above.

Parameter	Description	Notes
ma_type	MA Type (0-10)	Smoothing algorithm for the signal line
ma_price	Applied Price	Which price to use (Close, Open, High, Low, Median, Typical, Weighted)
t3_factor	T3 Factor	Volume factor when T3 MA is selected (0.0-1.0)
enable_HA	Enable Heiken Ashi	Use HA-preprocessed price data
length	Regression Length	Lookback period for linear regression deviation
nika_ma_period	Nika MA Period	Period for the Average Line smoothing
shift	Shift	Bar shift for signal evaluation

*TIP: Best for trend-following strategies. Use longer regression lengths (20-50) with EMA or HMA smoothing for cleaner signals on H1-H4 timeframes.*

## Mode 1: Nika Signal Line Level Cross (NikaSigLevCross)

This mode generates signals when the Nika Signal Line crosses above or below a user-defined level. Unlike NikaCross which uses the Average Line as a dynamic reference, this mode uses a fixed horizontal level, making it useful for detecting when the signal enters overbought/oversold territory.

**Buy Signal:** Signal Line crosses above the specified level from below.

**Sell Signal:** Signal Line crosses below the specified level from above.

The level parameter is user-configurable, defaulting to 0 (zero-line cross).

*TIP: Useful as a mean-reversion entry when the level is set near zero, or as a momentum filter when set at higher absolute values.*

## Mode 2: Histogram Level Cross (HistLevCross)

Generates signals based on the histogram (the difference between Signal Line and Average Line) crossing a user-defined level. The histogram provides a visual and numerical representation of the gap between signal and average, and its crossing of levels can indicate momentum shifts. This mode includes its own HA toggle and smoothing period.

**Buy Signal:** Histogram crosses above the level.

**Sell Signal:** Histogram crosses below the level (or the negative of the level for symmetrical setups).

*TIP: Effective as a momentum confirmation filter. Set level > 0 to require meaningful momentum before entry.*

## Mode 3: Histogram Moving Up/Down (HistUD)

A pure momentum signal that detects when the histogram is increasing or decreasing in value, regardless of its absolute level. Unlike Mode 2 which requires a level crossing, this mode only checks the direction of histogram movement.

**Buy Signal:** Histogram value is increasing (current bar > previous bar).

**Sell Signal:** Histogram value is decreasing (current bar < previous bar).

*TIP: Best as a filter rather than a standalone entry. Confirms that momentum is accelerating in the direction of a trade.*

## Mode 4: MA Crossover (MACross)

A traditional fast/slow moving average crossover. Two independent moving averages of any of the 11 types are computed, and signals are generated on their crossover. This is the most classic technical analysis signal, made powerful by the variety of MA types available.

**Buy Signal:** Fast MA crosses above Slow MA.

**Sell Signal:** Fast MA crosses below Slow MA.

Parameters include separate MA type, period, and applied price for both fast and slow averages. This enables combinations like HMA(10) crossing EMA(50), or TEMA(5) crossing VWMA(20) — hybrid configurations not available in standard MA crossover indicators.

*TIP: Versatile for any market condition. Use HMA for fast MA to reduce lag. Popular configurations: HMA(8)/EMA(21), TEMA(13)/SMA(50).*

## Mode 5: Two Nika Signal Lines Crossover (2SigCross)

This advanced mode creates two independent Nika signal line instances (A and B) with separate configuration for each — different lengths, HA toggles, and shift values. The signal is generated when Line A crosses Line B, functioning as a fast/slow Nika signal crossover.

**Buy Signal:** Nika Signal A crosses above Nika Signal B.

**Sell Signal:** Nika Signal A crosses below Nika Signal B.

*TIP: Use shorter length for Signal A and longer for Signal B. Enable HA on the slower signal for smoother reference.*

## Mode 6: Two Nika Signals Cross Average (2SigAvgCross)

Creates two Nika signal lines that must both cross the same Average Line. This provides stronger confirmation than a single crossover — both fast and slow signal perspectives must agree on the direction relative to the average.

**Buy Signal:** Both Signal A and Signal B are above the Average Line.

**Sell Signal:** Both Signal A and Signal B are below the Average Line.

*TIP: Strong confluence signal. Use for high-confidence entries where both fast and slow perspectives must agree.*

## Mode 7: Signal Crosses Two Averages (Sig2AvgCross)

The inverse of Mode 6 — a single signal line must cross above or below two different average lines. This creates a zone-based signal where the signal must break through both resistance/support levels defined by the two averages.

**Buy Signal:** Signal Line is above both Average Line 1 and Average Line 2.

**Sell Signal:** Signal Line is below both Average Line 1 and Average Line 2.

*TIP: Reduces false signals in ranging markets. The two averages create a band that must be fully penetrated.*

## Mode 8: Signal Line Moving Up/Down (SigUD)

A directional momentum signal based purely on whether the Signal Line is increasing or decreasing. No crossover or level is required — the signal simply detects the direction of movement.

**Buy Signal:** Signal Line value is rising (current > previous).

**Sell Signal:** Signal Line value is falling (current < previous).

*TIP: Best as a filter to confirm momentum direction. Pairs well with crossover-based Entry rules.*

## Mode 9: Average Line Moving Up/Down (AvgUD)

Similar to Mode 8 but tracks the Average Line instead. Since the Average Line is a smoothed version of the Signal Line, its directional changes are slower and represent more established trend shifts.

**Buy Signal:** Average Line value is rising.

**Sell Signal:** Average Line value is falling.

*TIP: Excellent as a trend-direction filter on higher timeframes. Prevents counter-trend entries.*



## Mode 10: Signal & Average Moving Up/Down (SigAvgUD)

The most restrictive directional mode — both the Signal Line AND the Average Line must be moving in the same direction. This ensures that both the fast and slow components of the Nika analysis agree on the current directional bias.

**Buy Signal:** Both Signal Line and Average Line are rising.

**Sell Signal:** Both Signal Line and Average Line are falling.

*TIP: Highest-confidence directional filter. Use on the primary timeframe to prevent entries during trend transitions.*

## Mode 11: MA Price Cross (MAPriceCross)

The simplest and most traditional signal mode — price crossing above or below a single moving average. Despite its simplicity, the availability of 11 MA types and configurable applied price makes this a flexible signal.

**Buy Signal:** Close price crosses above the selected MA.

**Sell Signal:** Close price crosses below the selected MA.

*TIP: Classic trend filter. Use on higher timeframes (H4, D1) as a Filter role with HMA or EMA for responsive trend direction.*

## Multi-Timeframe Capabilities

One of the EA's most powerful features is the ability to assign different timeframes to each rule slot. This enables genuine multi-timeframe analysis within a single EA instance. For example:

- **Entry Rule (NikaCross\_1):** M15 timeframe — fast entry signals
- **Filter Rule (MACross\_1):** H4 timeframe — trend direction filter
- **Filter Rule (AvgUD\_1):** D1 timeframe — macro trend confirmation

This configuration would only enter trades on M15 NikaCross signals that align with both the H4 MA trend and the daily Nika average direction — a three-timeframe confluence system from a single EA.

Chapter 4

# Moving Average Library

The All\_MA library provides 11 distinct moving average implementations, each with different mathematical properties affecting lag, smoothing, and responsiveness. The choice of MA type profoundly impacts signal quality and timing.

ID	Code	Full Name	Description	Lag	Smoothing	Responsiveness
0	SMA	Simple Moving Average	Equal-weighted arithmetic mean over N periods. Maximum smoothing but highest lag. Baseline reference for all other types.	High	High	Low
1	EMA	Exponential Moving Average	Exponentially-weighted average giving more weight to recent prices. The most widely used MA in trading. Good balance of smoothing and responsiveness.	Medium	Medium	Medium
2	WMA	Weighted Moving Average	Linearly-weighted average where the most recent bar has weight N, the second N-1, etc. More responsive than SMA but smoother than EMA.	Medium	Medium	Medium
3	VWMA	Volume Weighted MA	Weighted by volume — periods with higher volume have more influence. Reflects 'real' price where most trading occurred.	Medium	Medium	Medium
4	RMA	Running Moving Average	Also known as Wilder's smoothing. Uses a $1/N$ smoothing factor (vs $2/(N+1)$ for EMA), producing a smoother, more lagging result.	High	High	Low
5	DEMA	Double Exponential MA	$2 \times \text{EMA}(N)$ minus $\text{EMA}(\text{EMA}(N))$ . Significantly reduces lag compared to EMA while maintaining smoothness.	Low-Med	Medium	High
6	TEMA	Triple Exponential MA	$3 \times \text{EMA} - 3 \times \text{EMA}(\text{EMA}) + \text{EMA}(\text{EMA}(\text{EMA}))$ . Further lag reduction over DEMA. Very responsive to price changes.	Low	Medium	Very High
7	ZLEMA	Zero-Lag EMA	Applies EMA to de-lagged data by subtracting the value from $(N-1)/2$ bars ago. Aims to eliminate systematic lag.	Very Low	Low-Med	Very High
8	HMA	Hull Moving Average	$\text{WMA}(2 \times \text{WMA}(N/2) - \text{WMA}(N))$ with period $\sqrt{N}$ . Extremely low lag while maintaining smoothness. Popular for responsive signals.	Very Low	Medium	Very High
9	DONCHIAN	Donchian Channel Mid	Midpoint of the highest high and lowest low over N periods. Price-range based rather than weighted average.	High	Low	Low

ID	Code	Full Name	Description	Lag	Smoothing	Responsiveness
10	T3	T3 Moving Average	Six-pass EMA with a volume factor (0-1) controlling smoothing aggressiveness. Configurable lag-smoothness tradeoff.	Configurable	High	Configurable

## T3 Volume Factor

The T3 moving average is unique in having a user-configurable 'volume factor' (t3\_factor) ranging from 0.0 to 1.0. This parameter controls the aggressiveness of the smoothing:

Factor	Behavior
0.0	Maximum smoothing, behaves like a heavily smoothed EMA. Highest lag but least noise.
0.5	Balanced between smoothing and responsiveness. Good general-purpose setting.
0.7	Tim Tillson's original recommendation. Good balance for most trading applications.
1.0	Maximum responsiveness, minimum additional smoothing. Approaches raw EMA behavior with some overshoot.

## Applied Price Types

All MA calculations support multiple applied price types, allowing traders to choose which aspect of price action feeds the moving average:

Price Type	Description
Close	Default. Most commonly used. Represents the final consensus price of each period.
Open	The opening price. Can be useful for gap analysis.
High	The highest price. Useful for resistance-focused calculations.
Low	The lowest price. Useful for support-focused calculations.
Median	$(\text{High} + \text{Low}) / 2$ . Represents the midpoint of the period's range.
Typical	$(\text{High} + \text{Low} + \text{Close}) / 3$ . Gives weight to the close while including the range.
Weighted	$(\text{High} + \text{Low} + \text{Close} + \text{Close}) / 4$ . Double-weights the close price.

## Choosing the Right MA Type

The choice of MA type should be guided by the signal mode and trading strategy:

- **For trend-following entries (NikaCross, MACross):** HMA or TEMA provide fast response; EMA or T3(0.7) offer a good balance.

- **For trend-direction filters (AvgUD, SigAvgUD):** SMA or RMA provide stable, reliable direction readings with minimal whipsaw.
- **For mean-reversion signals (NikaSigLevCross):** EMA or DEMA balance responsiveness with smoothing to avoid premature signals.
- **For volatile markets:** T3 with higher factor or RMA to absorb spikes without overreacting.
- **For fast-moving markets:** ZLEMA or HMA for near-zero lag response to price changes.

Chapter 5

# Trade Management System

The trade management system governs how the EA opens, sizes, and manages positions. It encompasses entry mode selection, direction filtering, position sizing, hedge mode support, and spread/commission handling.

## Entry Mode

The `_initial_entry_mode` parameter controls whether the EA requires a signal to open its first trade:

Value	Mode	Description
0	Wait For Signal	The EA waits for a valid signal from the configured rules before opening any position. Standard mode for signal-driven strategies.
1	Immediate Entry	The EA opens a position immediately on the next tick after being attached to the chart, without waiting for a signal. Useful for manual direction selection or when combined with Direction Filter.

Immediate Entry is particularly useful when combined with the Direction Filter (Long Only or Short Only) and re-entry system. The trader sets the direction, the EA enters immediately, and re-entries continue in that direction automatically.

## Direction Filter

The `_trade_direction` parameter restricts which side the EA can trade:

Value	Direction	Behavior
0	Long & Short	No restriction. The EA trades both buy and sell signals based on signal rules.
1	Long Only	Only buy signals are acted upon. Sell signals are ignored for entry purposes.
2	Short Only	Only sell signals are acted upon. Buy signals are ignored for entry purposes.

This parameter is part of the V1.27 feature set. It provides a clean way to restrict trading to one side — for example, trading only long on a bullish asset, or only short during a confirmed downtrend.

## Position Sizing

Position sizing is calculated dynamically based on the `_lot_risk_perc` parameter, which defines the percentage of account balance risked per trade. The EA calculates the lot size such that if the stoploss is hit, the loss equals `_lot_risk_perc` percent of the current account balance.

The calculation takes into account the current stoploss distance (whether from Fixed Pips, ATR Multiplier, or Promille method), the symbol's tick value, and the account balance. This ensures consistent risk across different symbols and volatility conditions.

For example, with `_lot_risk_perc` = 0.5 and a \$10,000 account, each trade risks \$50. If the stoploss is 30 pips on EURUSD, the lot size is calculated so that 30 pips of adverse movement equals \$50.

## Maximum Entry Controls

Two parameters control entry frequency and position count:

Parameter	Type	Default	Description
<code>_max_entries</code>	int	1	Maximum number of main entry positions open simultaneously per direction. 1 = one long and one short at most.
<code>_max_day_entries</code>	int	10	Maximum total new entries allowed per trading day across all types. 0 = OFF (unlimited). Resets at the start of each new trading day.

## Hedge Mode

When `_allow_hedge` is enabled, the EA permits simultaneous long and short positions on the same symbol. In hedge mode, a new buy signal can open a long position even if a short position is already active, and vice versa. This is possible on MT5 accounts configured for hedging (as opposed to netting accounts).

Hedge mode is particularly useful when using multiple timeframes — a short-term signal might flip direction while a longer-term position is still running. Without hedge mode, the EA would need to close the existing position first.

## Close on Opposite Signal

When `_close_on_opposite` is enabled, the EA closes existing positions when an opposite-direction signal is generated. The exit signal evaluation uses the `_exit_control` parameter (ALL/ANY) independently from entry control. This creates a 'stop-and-reverse' behavior where the EA always holds a position in the signal direction.

## Spread and Commission Compensation

Two features account for trading costs:

Parameter	Type	Default	Description
<b>_enable_spread_distance_compensation</b>	bool	true	Adjusts entry and exit calculations to account for the current spread. Prevents entries when spread is abnormally wide.
<b>_enable_commission_compensation</b>	bool	false	Adds the estimated commission to the stoploss distance calculation, ensuring that the risk percentage accounts for total trading costs.
<b>_commission</b>	double	0	Commission value per lot used in compensation calculations when enabled.

Chapter 6

# Stoploss Methods

The stoploss system provides three distinct methods for calculating initial stoploss placement. The method is selected via the `_sl_method` parameter and applies to all entry types (Main, Re-Entry, Scale-In).

## Method 0: Fixed Pips

The simplest method — places the stoploss at a fixed number of pips from the entry price. The distance is specified by the `_sl_pips` parameter.

Parameter	Type	Default	Description
<code>_sl_pips</code>	double	50	Distance in pips from entry to stoploss.

Fixed pips is predictable and easy to optimize but does not adapt to market volatility. A 50-pip stop may be appropriate for EURUSD on H1 but too tight for GBPJPY or too wide for a scalping strategy.

## Method 1: ATR Multiplier

The ATR (Average True Range) method dynamically adjusts stoploss distance based on current market volatility. The stoploss is placed at  $ATR \times \text{\_atr\_multiplier}$  distance from entry.

Parameter	Type	Default	Description
<code>_atr_frame</code>	Timeframe	PERIOD_CURRENT	Timeframe for ATR calculation. Use higher timeframes for more stable readings.
<code>_atr_period</code>	int	14	Number of bars for ATR calculation. Standard is 14.
<code>_atr_multiplier</code>	double	1.5	Multiplier applied to ATR value. Higher = wider stop.

ATR-based stops automatically widen during volatile periods and tighten during quiet markets. A multiplier of 1.5 means the stop is 1.5x the average range — giving the trade room to breathe while limiting risk. Typical ranges: 1.0-2.0 for intraday, 2.0-3.0 for swing trading.

## Method 2: Average Price Promille

The Promille method calculates the stoploss as a fraction (in parts per thousand) of the average price over a specified period. This approach normalizes the stoploss across instruments of different price levels.



Parameter	Type	Default	Description
<code>_promille</code>	double	10	Parts per thousand of average price. 10 promille = 1% of average price.
<code>_promille_average_period</code>	int	14	Number of bars used to calculate the average price.

The Promille method is particularly useful for multi-symbol deployment. A 10-promille stop on EURUSD (price ~1.0800) is approximately 10.8 pips, while the same setting on USDJPY (price ~150.00) is approximately 15 pips — automatically scaling to the instrument's price level.

## Stoploss Steps and Breakeven

Two additional features modify the stoploss after entry:

Parameter	Type	Default	Description
<code>_allow_breakeven</code>	bool	true	When enabled, moves the stoploss to breakeven (entry price) once price reaches the first take-profit level.
<code>_allow_sl_steps</code>	bool	true	When enabled, progressively moves the stoploss to the price level of each reached TP level, locking in partial profits.

SL Steps work in conjunction with the take-profit system. As each TP level is reached and partial closes occur, the stoploss advances to lock in profits. For example, when TP2 is hit, the stoploss moves to the TP1 price level. This creates a ratcheting mechanism that prevents the remaining position from turning into a loss.

## Chapter 7

# Take Profit System

The Nika EA implements a sophisticated 10-level partial close system for take profit management. Each level is defined by a multiplier of the stoploss distance and a percentage of the position to close.

## TP Level Configuration

Each of the 10 TP levels has two parameters:

- **Multiplier ( `_tpN_multiplier` ):** The distance from entry, expressed as a multiple of the stoploss distance. For example, if SL is 30 pips and TP1 multiplier is 1.0, TP1 is placed at 30 pips from entry.
- **Partial Close % ( `_tpN_partial_perc` ):** The percentage of the remaining position to close when this level is reached.

Level	Multiplier Param	Close % Param	Default Mult	Default %	Notes
TP1	<code>_tp1_multiplier</code>	<code>_tp1_partial_perc</code>	1.0	0	First target. Often used for breakeven move rather than close.
TP2	<code>_tp2_multiplier</code>	<code>_tp2_partial_perc</code>	0	0	Set multiplier > 0 to enable. Close partial at this level.
TP3	<code>_tp3_multiplier</code>	<code>_tp3_partial_perc</code>	0	0	Third target level.
TP4	<code>_tp4_multiplier</code>	<code>_tp4_partial_perc</code>	0	0	Fourth target level.
TP5	<code>_tp5_multiplier</code>	<code>_tp5_partial_perc</code>	0	0	Fifth target level.
TP6	<code>_tp6_multiplier</code>	<code>_tp6_partial_perc</code>	0	0	Sixth target level.
TP7	<code>_tp7_multiplier</code>	<code>_tp7_partial_perc</code>	0	0	Seventh target level.
TP8	<code>_tp8_multiplier</code>	<code>_tp8_partial_perc</code>	0	0	Eighth target level.
TP9	<code>_tp9_multiplier</code>	<code>_tp9_partial_perc</code>	0	0	Ninth target level.
TP10	<code>_tp10_multiplier</code>	<code>_tp10_partial_perc</code>	0	0	Final target — typically closes remaining position.

## How Partial Closes Work

When price reaches a TP level, the specified percentage of the **remaining** position (not the original size) is closed. This is an important distinction — the percentages are applied sequentially to the current position size.

Example with a 1.0 lot position, SL at 30 pips, and the following configuration:

Level	Distance	Close %	Lots Closed	Remaining
TP1	1.0x (30 pips)	20%	0.20 lots	0.80 lots
TP2	2.0x (60 pips)	25%	0.20 lots	0.60 lots
TP3	3.0x (90 pips)	50%	0.30 lots	0.30 lots
TP4	5.0x (150 pips)	100%	0.30 lots	0.00 lots

Setting a TP multiplier to 0 disables that level. Typically, traders configure 3-5 active TP levels rather than using all 10, unless running very wide-range strategies.

### TP Interaction with Trailing

The take-profit system works alongside the trailing stop system. When both are active:

- TP levels execute partial closes as price advances.
- The trailing stop protects the remaining position from reversal.
- SL Steps move the stoploss to previous TP levels, providing additional protection.
- If the trailing stop is hit before the next TP level, the remaining position is closed at the trailed price rather than the full stoploss distance.

This combination creates a robust profit management approach — securing partial profits at defined levels while using trailing to capture extended moves beyond the planned TP ladder.

### Suggested TP Configurations

Different market conditions and strategies benefit from different TP structures:

Strategy	Configuration	Rationale
Scalping	TP1: 1.0x @ 50%, TP2: 1.5x @ 100%	Quick partial, rapid full close
Day Trading	TP1: 1.0x @ 25%, TP2: 2.0x @ 25%, TP3: 3.0x @ 50%	Gradual scaling with runner
Swing Trading	TP1: 1.5x @ 20%, TP2: 3.0x @ 30%, TP3: 5.0x @ 50%	Wide targets, patient scaling
Trend Following	TP1: 1.0x @ 10%, then trailing	Minimal TP, let trailing manage

## Chapter 8

# Trailing Stop System

The trailing stop system provides dynamic profit protection once a position moves into profit. Two distinct trailing modes are available, selected via `_trail_method`.

## Mode 0: m-aslt2 (RRR-Based Stepping)

The m-aslt2 (Modified Adaptive Stop Loss Trailing v2) mode uses Risk-Reward Ratio (RRR) based stepping. The trailing stop activates when the position's profit reaches a specified RRR level, then steps forward in fixed RRR increments.

Parameter	Type	Default	Description
<code>_trailing_start</code>	double	0.5	RRR level at which trailing begins. 0.5 = trailing starts when profit equals 50% of SL distance.
<code>_trailing_step</code>	double	0.2	RRR increment for each step. 0.2 = trail advances by 20% of SL distance each step.
<code>_trailing_stop</code>	double	1.0	RRR distance behind price. 1.0 = trailing stop is 1× SL distance behind the highest profit point.

Example: With SL = 30 pips, `_trailing_start` = 0.5, `_trailing_step` = 0.2, `_trailing_stop` = 1.0:

- Trailing activates at 15 pips profit ( $0.5 \times 30$ )
- Trail steps forward every 6 pips of additional profit ( $0.2 \times 30$ )
- The trailing stop is placed 30 pips behind peak profit ( $1.0 \times 30$ )

The stepping mechanism prevents the trailing stop from moving backward while advancing it in discrete steps, creating a staircase pattern of protection that locks in profit as price advances.

## Mode 1: m-pppt (Percentage Pullback)

The m-pppt (Modified Percentage Pullback Protection Trailing) mode uses a percentage-based approach. Instead of fixed RRR steps, it monitors how far price has pulled back from its peak profit and closes the position if the pullback exceeds a threshold.

Parameter	Type	Default	Description
<code>_allow_min_pips</code>	bool	true	Enable minimum profit threshold (in RRR) before trailing activates.

Parameter	Type	Default	Description
<code>_pullback_start</code>	double	0.5	Minimum RRR of profit before the pullback monitor becomes active.
<code>_pullback_perc</code>	double	30	Close the position if price pulls back this percentage from peak profit. 30 = close if 30% of maximum floating profit is lost.

Example: With `_pullback_start` = 0.5 (SL = 30 pips, so trailing after 15 pips profit) and `_pullback_perc` = 30:

- Price moves to +40 pips profit (peak).
- 30% of 40 = 12 pips pullback threshold.
- Position closes if profit drops to +28 pips (40 - 12).

The percentage pullback mode is particularly effective for volatile markets where fixed-distance trailing can be too easily triggered. By using a percentage of the peak profit, the trailing automatically adapts to the size of the move — larger moves get more breathing room.

### Choosing Between Trailing Modes

Mode	Mechanism	Strength	Weakness	Best For
<code>m-aslt2</code>	Discrete stepping	Predictable, systematic	Can leave gaps between steps	Trending markets with steady progress
<code>m-pppt</code>	Percentage-based	Adaptive to move size	May close during normal retracements	Volatile markets with extended moves

Chapter 9

# Re-Entry System

The Re-Entry system automatically re-opens positions in the same direction after a trade closes. This is designed to recapture trends after temporary adverse moves that trigger a stoploss or trailing exit.

## How Re-Entries Work

When a Main Entry position closes (by stoploss, trailing stop, or other mechanism), the EA evaluates whether to re-enter:

- The previous position's direction is recorded (Buy or Sell).
- The re-entry counter for that direction is incremented.
- If the counter is below `_max_re_entry_orders`, a new position is opened in the same direction.
- The re-entry uses the same risk parameters but receives a Re-Entry magic number (default 1001).
- The counter resets when a successful trade occurs or when the series limit is reached.

## Series Counting

The re-entry system tracks entries in a 'series' — a sequence of consecutive re-entries in the same direction. The `_max_re_entry_orders` parameter controls how many re-entries are allowed per series:

Value	Behavior
0	OFF — No re-entries. Standard single-trade mode.
1-N	Allow up to N re-entries per series. After N consecutive re-entries, the series stops.
-1	INFINITE — Unlimited re-entries. The EA continues re-entering in the same direction indefinitely until a profitable exit or manual intervention.

## Infinite Mode Safety Considerations

**IMPORTANT: Setting `_max_re_entry_orders` to -1 (INFINITE) creates unlimited re-entries. This means the EA will continue opening positions in the same direction regardless of how many consecutive losses occur. This mode **MUST** be used in combination with other risk controls:**

- **Daily Loss Limit:** Set `_daily_loss_value` to cap total daily losses.
- **Weekly Loss Limit:** Set `_weekly_loss_value` as a secondary backstop.
- **Max Open Lot:** Set `_max_open_lot` to prevent position size escalation if combined with scale-in.

- **Max Day Entries:** Set `_max_day_entries` to limit the number of re-entries per day.

## Cooldown Interaction

The Trade Cooldown filter interacts with re-entries through the `_waitBars_re_entry` parameter. When set to a value  $> 0$ , the EA waits the specified number of bars (on the `_cooldown_frame` timeframe) before executing a re-entry. This prevents rapid-fire re-entries during choppy conditions.

The cooldown applies independently to each entry type: `_waitBars_main` for main entries, `_waitBars_re_entry` for re-entries, and `_waitBars_scale` for scale-in entries. This allows, for example, a 5-bar cooldown on re-entries while main entries have no cooldown.

Chapter 10

# Scale-In System

The Scale-In system allows adding to an existing position at up to 10 configurable trigger levels. Each level specifies a Risk-Reward Ratio (RRR) trigger point and a position size as a percentage of the main entry.

## 10 Trigger Levels

Scale-in entries are triggered when the position's floating P&L reaches specific RRR levels relative to the stoploss distance. Each level has two parameters:

- **\_scale\_trigger\_N**: The RRR at which this scale-in triggers. 0 = disabled for this level.
- **\_scale\_size\_N**: The lot size as a percentage of the main entry's lot size. 50 = half the main position size.

## Positive vs. Negative Triggers

V1.27 supports both positive and negative trigger values, enabling two distinct scale-in strategies:

Trigger	Strategy	Description	Best For
<b>Positive (+)</b>	Scale into Profit	Trigger when position is in profit. +1.0 = scale in when profit reaches 1× SL distance. Used to add to winners — a pyramiding strategy.	Trend-following, momentum
<b>Negative (-)</b>	Scale into Drawdown	Trigger when position is in drawdown. -1.0 = scale in when loss reaches 1× SL distance. Used to average into positions — a DCA-like approach.	Mean-reversion, grid strategies

## Range Variation

The `_scale_range_variation` parameter (default 0.05) adds randomness to trigger levels to avoid clustering. With a variation of 0.05, a trigger level of 1.0 RRR could fire anywhere between 0.95 and 1.05 RRR. This prevents multiple EAs on the same account from triggering identical scale-ins simultaneously.

## Scale-In Example Configuration

Level	Trigger (RRR)	Size (% of Main)	Description
<b>Level 1</b>	+1.0	50%	Add half position at 1:1 RRR in profit



Level	Trigger (RRR)	Size (% of Main)	Description
Level 2	+2.0	30%	Add 30% at 2:1 RRR
Level 3	+3.0	20%	Add 20% at 3:1 RRR
Level 4	-0.5	25%	Add 25% at 50% drawdown to SL
Levels 5-10	0	—	Disabled

This hybrid configuration adds to winning positions (levels 1-3) while also averaging in slightly during drawdowns (level 4). The decreasing size percentages for profit levels prevent over-exposure as the trend extends.

***IMPORTANT: Negative triggers (scaling into drawdown) increase risk significantly. Always use in combination with Max Open Lot limits and daily loss controls.***

## Chapter 11

# Risk Management Framework

The Nika EA implements 15 independent risk management layers that are evaluated in sequence before every trade decision. This defense-in-depth approach ensures that no single point of failure can lead to uncontrolled losses.

## Complete Risk Layer Overview

#	Layer	Key Parameter	Description
1	Max Entries per Direction	_max_entries	Limits simultaneous positions per direction (buy/sell). Default: 1.
2	Max Entries per Day	_max_day_entries	Total new entries allowed per trading day. 0 = OFF. Default: 10.
3	Re-Entry Series Limit	_max_re_entry_orders	Maximum consecutive re-entries. 0 = OFF, -1 = INFINITE.
4	Max Open Lot	_max_open_lot	Maximum total lot size. Can apply per symbol or all symbols. 0 = OFF.
5	Daily Loss Limit	_daily_loss_value	Maximum allowed loss per day (money or %). 0 = OFF.
6	Daily Profit Limit	_daily_profit_value	Stops trading after reaching daily profit target. 0 = OFF.
7	Weekly Loss Limit	_weekly_loss_value	Maximum allowed loss per week. 0 = OFF.
8	Weekly Profit Limit	_weekly_profit_value	Stops trading after reaching weekly profit target. 0 = OFF.
9	Daily Account Trailing	_allow_daily_trailing	Trailing stop on daily P&L.; Locks in portion of daily profits.
10	Hours Filter	_allow_trading_time	Restricts trading to specified hours. Supports Friday-only close.
11	News Filter	_allow_news_filter	Pauses trading around economic news events by impact level.
12	Trade Cooldown	_waitBars_main/re/scale	Minimum bars between entries per type.
13	Profit Scale-Out	_allow_profit_scale_out	Partial close when floating profit reaches target.
14	Direction Filter	_trade_direction	Restricts to Long Only, Short Only, or Both.
15	Spread Compensation	_enable_spread_distance_compensation	Accounts for spread in entry/exit calculations.

## Layer 1: Max Entries per Direction

The `_max_entries` parameter (default: 1) limits the number of simultaneously open Main Entry positions per direction. With `_max_entries` = 1, the EA can hold at most one long and one short position at any time. Increasing this value allows multiple entries in the same direction, which can be useful for strategies that enter at different signal points within the same trend. This limit applies only to Main Entry positions — Re-Entries and Scale-Ins have their own controls.

## Layer 2: Max Entries per Day

The `_max_day_entries` parameter (default: 10) caps the total number of new entries (across all types) per trading day. This prevents excessive trading during volatile or choppy conditions. The counter resets at the beginning of each new trading day. Setting this to 0 disables the limit (unlimited daily entries).

## Layer 3: Re-Entry Series Limit

As detailed in the Re-Entry chapter, `_max_re_entry_orders` controls consecutive re-entries. This layer specifically limits the 'series' of losses — if a trade direction keeps hitting stoplosses, the series limit stops the EA from continuing to bleed in that direction.

## Layer 4: Max Open Lot

The `_max_open_lot` parameter sets an absolute ceiling on total open lot size. The `_max_open_lot_symbol` parameter determines scope:

Scope	Behavior
0 (Current Symbol)	Limits total lots on the EA's attached symbol only.
1 (All Symbols)	Limits total lots across ALL symbols on the account. Cross-EA protection.

## Layers 5-8: Daily/Weekly P&L; Limits

Four parameters create a box around allowable profit and loss on daily and weekly timeframes:

Parameter	Default	Description
<code>_daily_loss_value</code>	0	Maximum loss per day. When reached, all positions close and no new trades until resume time.
<code>_daily_profit_value</code>	0	Maximum profit per day. When reached, trading stops until resume time. Protects against giving back gains.
<code>_weekly_loss_value</code>	0	Maximum loss per week. Broader protection than daily limits.
<code>_weekly_profit_value</code>	0	Maximum profit per week. Weekly profit ceiling.

The `_max_pl_type` parameter determines whether limits are in absolute money (0) or account percentage (1). Resume parameters (`_daily_resume_hour`, `_weekly_resume_day`, etc.) control when trading resumes after a limit is hit.

## Layer 9: Daily Account Trailing

When `_allow_daily_trailing` is enabled, the EA applies a trailing stop mechanism to the day's total P&L. This locks in a portion of daily profits while allowing continued trading. Three parameters control the behavior:

Parameter	Name	Description
<code>trailStartInput</code>	Trail Start %	P&L; percentage at which daily trailing activates.
<code>trailStepInput</code>	Trail Step %	Step increment for advancing the daily trail.
<code>trailAmountInput</code>	Trail %	How much of daily profit to protect. When daily P&L; retraces this percentage from peak, all positions close.

## Layer 10: Hours Filter

The hours filter restricts trading to specified time windows. Detailed in the Hours & Session Management chapter.

## Layer 11: News Filter

The news filter pauses trading around economic events. Detailed in the News Filter chapter.

## Layer 12: Trade Cooldown

The cooldown filter enforces a minimum number of bars between entries, applied independently per entry type. The `_cooldown_frame` parameter sets the timeframe for counting bars.

Parameter	Type	Default	Description
<code>_wait_bars_main</code>	int	0	Bars to wait between Main Entries. 0 = no cooldown.
<code>_wait_bars_re_entry</code>	int	0	Bars to wait between Re-Entries. 0 = no cooldown.
<code>_wait_bars_scale</code>	int	0	Bars to wait between Scale-In entries. 0 = no cooldown.

## Layer 13: Profit Scale-Out

When `_allow_profit_scale_out` is enabled, the EA partially closes positions when floating profit reaches a trigger value. This is different from the TP system — it operates on total P&L (not per-position RRR) and is designed

for account-level profit protection.

Parameter	Type	Default	Description
_profit_scale_out_calc_type	enum	0	0 = Balance Percent, 1 = Fixed Amount
_profit_scale_out_trigger_value	double	3	Trigger value (% or money amount)
_profit_scale_out_close_percent	double	50	Percentage of position to close when triggered

## Layers 14-15: Direction Filter & Spread Compensation

The Direction Filter (`_trade_direction`) is covered in the Trade Management chapter. Spread Compensation (`_enable_spread_distance_compensation`) adjusts all distance calculations to include the current spread, preventing entries when spread is abnormally wide and ensuring stoploss distances account for the cost of crossing the spread.

## Filter Evaluation Order

Risk filters are evaluated in a specific sequence before any trade is placed. If any filter blocks the trade, subsequent filters are not evaluated. The general order is:

1. Direction Filter — Is the signal direction allowed?
2. Hours Filter — Are we within the allowed trading window?
3. News Filter — Is there upcoming high-impact news?
4. Daily/Weekly P&L Limits — Have any limits been reached?
5. Max Entries per Direction — Are we at the position limit?
6. Max Entries per Day — Have we reached the daily limit?
7. Max Open Lot — Would this trade exceed the lot ceiling?
8. Trade Cooldown — Have enough bars passed since the last entry?
9. Spread Compensation — Is the spread acceptable?

This ordering prioritizes fast checks (direction, time) before more expensive calculations (P&L sums, lot counting). It also ensures that fundamental blockers (wrong direction, outside hours) are caught before resource-intensive checks.

---

## Chapter 12

# Hours & Session Management

---

The hours filter provides precise control over when the EA is allowed to trade, supporting timezone conversion, daylight savings adjustment, and a unique Friday-only close feature.

## Time Window Configuration

Parameter	Type	Default	Description
<code>_allow_trading_time</code>	bool	false	Master toggle for the hours filter.
<code>_trade_start_hour</code>	int	—	Trading window start hour (0-23).
<code>_trade_start_minute</code>	int	—	Trading window start minute (0-59).
<code>_trade_end_hour</code>	int	—	Trading window end hour (0-23).
<code>_trade_end_minute</code>	int	—	Trading window end minute (0-59).

## Timezone Handling

The EA can operate in either server time or a fixed UTC offset:

Parameter	Type	Default	Description
<code>_use_server_time_zone</code>	bool	true	When true, uses the broker server's timezone for all time calculations.
<code>_timezone</code>	int	-4	UTC offset when not using server time. -4 = UTC-4 (Eastern US).
<code>_adjust_daylight</code>	bool	false	Automatically adjusts the UTC offset for daylight saving time transitions.

Using a fixed timezone with DST adjustment is recommended for consistent behavior across broker server changes. The DST adjustment follows US daylight saving rules (second Sunday in March, first Sunday in November).

## Friday-Only Close Feature

The `_close_on_time_filter` parameter controls what happens to open positions when the time window ends:

Value	Mode	Behavior
0	Close	All open positions are closed immediately when the time filter window expires, regardless of the day of the week.
1	Close On Friday Only	Positions are kept open through the week when the time window expires (Monday-Thursday). Positions are only force-closed on Fridays, preventing weekend gap exposure.

The Friday-only close option is particularly valuable for swing trading strategies that want to avoid weekend gaps while allowing positions to run overnight on weekdays.

## Wrap-Around Sessions

The hours filter supports wrap-around sessions where the start time is later than the end time. For example, setting start to 22:00 and end to 06:00 creates an overnight trading window from 10 PM to 6 AM. The EA automatically handles the day boundary crossing.

---

## Chapter 13

# News Filter

---

The news filter automatically pauses trading around high-impact economic events to avoid unexpected volatility. It sources event data from FairEconomy's CSV feed and supports configurable pause durations by impact level.

### Data Source

News events are downloaded from the FairEconomy economic calendar CSV feed. The data includes event name, currency, impact level, and scheduled time. The EA caches this data and refreshes periodically to stay current.

### Impact Levels

Impact	Description
<b>Holiday</b>	Market holidays. Trading is paused entirely for affected currencies.
<b>Low</b>	Low-impact events. Optional filter — typically minor data releases.
<b>Medium</b>	Medium-impact events. Moderate market reaction expected.
<b>High</b>	High-impact events. Major data releases (GDP, CPI, interest rate decisions).
<b>NFP</b>	Non-Farm Payrolls and similar high-impact US employment data. Most volatile events.

### Pause Mechanism

Each impact level has a configurable pause duration in seconds:

Parameter	Type	Default	Description
<code>_high_pause</code>	int	60	Seconds to pause before and after high-impact events.
<code>_medium_pause</code>	int	60	Seconds to pause before and after medium-impact events.
<code>_low_pause</code>	int	60	Seconds to pause before and after low-impact events.



The pause is applied symmetrically — if `_high_pause` is 60 seconds, trading is blocked from 60 seconds before to 60 seconds after the event. During the pause window, no new entries are allowed.

## Auto-Close Functionality

When `_auto_close_on_news` is enabled (default: true), the EA not only prevents new entries but also closes all open positions before the news event begins. This provides complete protection against news-driven volatility spikes.

## Visual News Lines

When `_draw` is enabled, the EA draws vertical lines on the chart at news event times, color-coded by impact level. This provides visual confirmation of upcoming events and the pause windows. Font, colors, and line styles are configurable through the visual settings parameters.

Chapter 14

# Dashboard System

The Nika EA includes a real-time on-chart dashboard overlay that provides at-a-glance monitoring of the EA's state, active positions, and risk layer status.

## Dashboard Features

- **Real-Time P&L Display:** Current floating profit/loss for all open positions, updated on every tick.
- **Entry State Monitoring:** Shows which entry types are active, pending, or blocked by risk filters.
- **Position Summary:** Number of open positions by type (Main, Re-Entry, Scale-In) and direction.
- **Risk Layer Status:** Visual indicators for active risk filters (hours, news, daily limits, etc.).
- **News Countdown:** Time remaining until the next scheduled news event and its impact level.
- **Trade History:** Recent trade actions (entries, exits, partial closes) limited by `_max_dashboard_actions`.

## Configuration

Parameter	Type	Default	Description
<code>_show_dashboard</code>	bool	true	Master toggle for the dashboard overlay.
<code>_max_dashboard_actions</code>	int	10	Maximum number of recent actions shown in the trade history panel.

Additional visual parameters control font size, colors, and positioning. The dashboard is designed to be informative without obscuring chart analysis. During optimization in the Strategy Tester, the dashboard is automatically suppressed for performance.

---

## Chapter 15

# Strategy Tester Integration

---

The Nika EA provides deep integration with the MT5 Strategy Tester through five custom fitness functions for the OnTester callback. These functions enable systematic optimization using composite performance metrics.

## Five OnTester Fitness Functions

The `_drawdown_type` parameter selects which formula is used to compute the custom OnTester result:

### Formula 0: $\text{Result\%} / (\text{MaxDD\%} + 2)$

The simplest formula. Divides the net profit percentage by the maximum drawdown percentage plus 2 (the +2 prevents division by zero and penalizes very small drawdowns less aggressively). Favors strategies with high returns relative to drawdown.

**Best for:** Initial broad parameter sweeps where you want to quickly identify parameter regions with favorable return-to-risk ratios.

### Formula 1: $\text{Result\%} \times \text{RecoveryFactor}$

Multiplies the net profit percentage by the Recovery Factor (net profit / max drawdown). This quadratically rewards high returns while penalizing drawdowns. Recovery Factor alone can favor strategies with very small drawdowns regardless of total return — multiplying by Result% ensures that absolute performance matters.

**Best for:** Identifying strategies that combine high returns with strong recovery characteristics.

### Formula 2: $(\text{Result\%} / \text{MaxDD\%} + 2) \times \text{RecoveryFactor}$

Combines Formulas 0 and 1 — the return-to-drawdown ratio is multiplied by the Recovery Factor. This creates a highly selective metric that rewards consistency above all else.

**Best for:** Final optimization passes where you want the most robust, consistent parameter sets.

### Formula 3: $(\text{Result\%} / \text{MaxDD\%}) \times \text{RecoveryFactor}$

Similar to Formula 2 but without the +2 offset in the denominator. This version is more aggressive — strategies with very low drawdowns receive disproportionately high scores. Can be useful for finding ultra-conservative parameter sets.

**Best for:** Risk-averse optimization where minimizing drawdown is the primary concern.

Formula 4: (Result% / MaxDD%) x Sharpe

Replaces Recovery Factor with the Sharpe Ratio, which measures risk-adjusted return per unit of volatility. This formula values not just total return and drawdown, but the consistency of returns period-to-period.

**Best for:** Strategies targeting smooth equity curves. The Sharpe component heavily penalizes volatile returns even if total profit is high.

Fitness Function Comparison

ID	Formula	Focus	When to Use
0	Result% / (MaxDD% + 2)	General purpose	Safe baseline for initial sweeps
1	Result% x RecoveryFactor	Return-focused	Favors high absolute returns with recovery
2	(Result%/(MaxDD%+2)) x RF	Consistency-focus ed	Most selective, rewards robustness
3	(Result%/MaxDD%) x RF	Risk-averse	Aggressive drawdown penalty
4	(Result%/MaxDD%) x Sharpe	Smooth equity	Penalizes return volatility

Optimization Approach

The Strategy Tester integration works with MT5's built-in genetic algorithm and grid search optimizers. When running custom optimization (as opposed to built-in metrics like Max Profit), set the optimization criterion to 'Custom' and the selected `_drawdown_type` formula will be used as the fitness value.

Additional tester parameters:

Parameter	Type	Default	Description
<code>_show_indicators</code>	bool	—	Whether to render indicator overlays during visual testing.
<code>_nika_name</code>	string	—	Custom name for the Nika indicator file (if renamed).
<code>_ma_name</code>	string	—	Custom name for the MA library file (if renamed).

## Chapter 16

# Complete Parameter Reference

This chapter provides a comprehensive reference for every input parameter in the Nika EA V1.27 Re-Entry, organized by input group exactly as they appear in the EA's settings panel.

## Alerts Settings

Variable Name	Display Name	Type	Default	Description
<code>allow_sound_alerts</code>	Enable Sound Alerts	bool	true	Play sound on trade events
<code>allow_push_alerts</code>	Enable Push Alerts	bool	false	Send push notifications to mobile MT5 app
<code>allow_email_alerts</code>	Enable Email Alerts	bool	false	Send email notifications via MT5 mail settings

## Global Settings

Variable	Display Name	Type	Default	Modifier	Description
<code>_main_magic</code>	Main Magic Number	int	1000	sinput	Magic number for main entries
<code>_re_entry_magic</code>	Re-Entry Magic	int	1001	sinput	Magic number for re-entries
<code>_scale_magic</code>	Scale Magic	int	1002	sinput	Magic number for scale-in entries
<code>_deviation_permille</code>	Deviation Per mille	double	1	—	Slippage tolerance in permille

## Trade Settings

Variable	Display	Type	Default	Description / Options
<code>_initial_entry_mode</code>	Entry Mode	enum	0	0=Wait For Signal, 1=Immediate Entry
<code>_trade_direction</code>	Direction	enum	0	0=Long&Short, 1=Long Only, 2=Short Only
<code>_max_entries</code>	Max Entries	int	1	Max positions per direction

Variable	Display	Type	Default	Description / Options
<b>_max_day_entries</b>	Max Day Entries	int	10	Max entries per day, 0=OFF
<b>_lot_risk_perc</b>	Risk %	double	0.5	Percentage of balance risked per trade
<b>_close_on_opposite</b>	Close On Opposite	bool	false	Close position on opposite signal
<b>_entry_control</b>	Entry Control	enum	0	0=ALL rules agree, 1=ANY rule triggers
<b>_exit_control</b>	Exit Control	enum	0	0=ALL for exit, 1=ANY for exit
<b>_allow_hedge</b>	Allow Hedge	bool	true	Allow simultaneous long and short positions
<b>_enable_spread_distance_compensation</b>	Spread Comp.	bool	true	Compensate for spread in calculations
<b>_enable_commission_compensation</b>	Commission Comp.	bool	false	Include commission in SL calculation
<b>_commission</b>	Commission	double	0	Commission per lot for compensation calc

## Re-Entry Settings

Variable	Display Name	Type	Default	Description
<b>_max_re_entry_orders</b>	Max Re-Entry Orders	int	0	0=OFF, -1=INFINITE, N=max series count

## Stoploss Settings

Variable	Display	Type	Default	Description
<b>_sl_method</b>	SL Method	enum	0	0=Fixed Pips, 1=ATR Multi, 2=Promille
<b>_sl_pips</b>	SL Pips	double	50	Fixed pips distance (Method 0)
<b>_atr_frame</b>	ATR Timeframe	TF enum	CURRENT	Timeframe for ATR (Method 1)
<b>_atr_period</b>	ATR Period	int	14	ATR lookback period (Method 1)
<b>_atr_multiplier</b>	ATR Multiplier	double	1.5	ATR multiplier (Method 1)
<b>_promille</b>	Promille	double	10	Parts per thousand (Method 2)
<b>_promille_average_period</b>	Avg Period	int	14	Price average period (Method 2)

## Max Open Lot Settings

Variable	Display	Type	Default	Description
<code>_max_open_lot</code>	Max Open Lot	double	0	0=OFF, maximum total open lot size
<code>_max_open_lot_symbol</code>	Scope	enum	0	0=Current Symbol, 1=All Symbols

## Daily Trailing Settings

Variable	Display	Type	Default	Description
<code>_allow_daily_trailing</code>	Enable	bool	false	Toggle daily trailing
<code>trailStartInput</code>	Trail Start %	double	0.0	P&L % to start trailing
<code>trailStepInput</code>	Trail Step %	double	0.0	Step increment %
<code>trailAmountInput</code>	Trail %	double	0.0	Protection percentage
<code>DailyResumeHour</code>	Resume Hour	enum	h9	Hour to resume after daily trail hit
<code>DailyResumeMinute</code>	Resume Minute	int	0	Minute to resume

## Profit Scaling-Out Settings

Variable	Display	Type	Default	Description
<code>_allow_profit_scale_out</code>	Enable	bool	false	Toggle profit scale-out
<code>_profit_scale_out_calc_type</code>	Calc Type	enum	0	0=Balance %, 1=Fixed Amount
<code>_profit_scale_out_trigger_value</code>	Trigger	double	3	Trigger value (% or amount)
<code>_profit_scale_out_close_percent</code>	Close %	double	50	% of position to close

## Targets Settings (Take Profit)

Variable	Display	Type	Default	Description
_allow_breakeven	Breakeven	bool	true	Move SL to entry at TP1
_allow_sl_steps	SL Steps	bool	true	Step SL to previous TP levels
_tp1_multiplier	TP1 Mult	double	1.0	TP1: SL distance multiplier, 0=OFF
_tp1_partial_perc	TP1 Close%	int	0	TP1: % of remaining position to close
_tp2_multiplier	TP2 Mult	double	0	TP2: SL distance multiplier, 0=OFF
_tp2_partial_perc	TP2 Close%	int	0	TP2: % of remaining position to close
_tp3_multiplier	TP3 Mult	double	0	TP3: SL distance multiplier, 0=OFF
_tp3_partial_perc	TP3 Close%	int	0	TP3: % of remaining position to close
_tp4_multiplier	TP4 Mult	double	0	TP4: SL distance multiplier, 0=OFF
_tp4_partial_perc	TP4 Close%	int	0	TP4: % of remaining position to close
_tp5_multiplier	TP5 Mult	double	0	TP5: SL distance multiplier, 0=OFF
_tp5_partial_perc	TP5 Close%	int	0	TP5: % of remaining position to close
_tp6_multiplier	TP6 Mult	double	0	TP6: SL distance multiplier, 0=OFF
_tp6_partial_perc	TP6 Close%	int	0	TP6: % of remaining position to close
_tp7_multiplier	TP7 Mult	double	0	TP7: SL distance multiplier, 0=OFF
_tp7_partial_perc	TP7 Close%	int	0	TP7: % of remaining position to close
_tp8_multiplier	TP8 Mult	double	0	TP8: SL distance multiplier, 0=OFF
_tp8_partial_perc	TP8 Close%	int	0	TP8: % of remaining position to close
_tp9_multiplier	TP9 Mult	double	0	TP9: SL distance multiplier, 0=OFF
_tp9_partial_perc	TP9 Close%	int	0	TP9: % of remaining position to close
_tp10_multiplier	TP10 Mult	double	0	TP10: SL distance multiplier, 0=OFF
_tp10_partial_perc	TP10 Close%	int	0	TP10: % of remaining position to close

## Scale-In Entry Settings

Variable	Display	Type	Default	Description
_scale_range_variation	Range Variation	double	0.05	Random variation +/- % on trigger levels
_scale_trigger_1	Trigger 1	double	0	RRR trigger level 1, 0=OFF, +/- supported
_scale_size_1	Size 1	double	0	Lot size as % of main position
_scale_trigger_2	Trigger 2	double	0	RRR trigger level 2, 0=OFF, +/- supported



Variable	Display	Type	Default	Description
_scale_size_2	Size 2	double	0	Lot size as % of main position
_scale_trigger_3	Trigger 3	double	0	RRR trigger level 3, 0=OFF, +/- supported
_scale_size_3	Size 3	double	0	Lot size as % of main position
_scale_trigger_4	Trigger 4	double	0	RRR trigger level 4, 0=OFF, +/- supported
_scale_size_4	Size 4	double	0	Lot size as % of main position
_scale_trigger_5	Trigger 5	double	0	RRR trigger level 5, 0=OFF, +/- supported
_scale_size_5	Size 5	double	0	Lot size as % of main position
_scale_trigger_6	Trigger 6	double	0	RRR trigger level 6, 0=OFF, +/- supported
_scale_size_6	Size 6	double	0	Lot size as % of main position
_scale_trigger_7	Trigger 7	double	0	RRR trigger level 7, 0=OFF, +/- supported
_scale_size_7	Size 7	double	0	Lot size as % of main position
_scale_trigger_8	Trigger 8	double	0	RRR trigger level 8, 0=OFF, +/- supported
_scale_size_8	Size 8	double	0	Lot size as % of main position
_scale_trigger_9	Trigger 9	double	0	RRR trigger level 9, 0=OFF, +/- supported
_scale_size_9	Size 9	double	0	Lot size as % of main position
_scale_trigger_10	Trigger 10	double	0	RRR trigger level 10, 0=OFF, +/- supported
_scale_size_10	Size 10	double	0	Lot size as % of main position

## Trade Cool Down Filter Settings

Variable	Display	Type	Default	Description
_cooldown_frame	Cooldown TF	TF enum	CURRENT	Timeframe for bar counting
_wait_bars_main	Main Bars	int	0	Bars between main entries, 0=OFF
_wait_bars_re_entry	Re-Entry Bars	int	0	Bars between re-entries, 0=OFF
_wait_bars_scale	Scale Bars	int	0	Bars between scale-ins, 0=OFF

## Trailing Stop Settings

Variable	Display	Type	Default	Description
<b>_allow_trailing</b>	Enable Trailing	bool	false	Master toggle
<b>_trail_method</b>	Trail Method	enum	0	0=m-aslt2, 1=m-pppt
<b>_trailing_start</b>	Trail Start	double	0.5	m-aslt2: RRR to start trailing
<b>_trailing_step</b>	Trail Step	double	0.2	m-aslt2: RRR step increment
<b>_trailing_stop</b>	Trail Stop	double	1.0	m-aslt2: RRR distance behind price
<b>_allow_min_pips</b>	Allow Min RRR	bool	true	m-pppt: Enable min profit threshold
<b>_pullback_start</b>	Pullback Start	double	0.5	m-pppt: Min RRR before active
<b>_pullback_perc</b>	Pullback %	double	30	m-pppt: Close at this % pullback

## Max Daily/Weekly P&L; Settings

Variable	Display	Type	Default	Description
<b>_max_pl_type</b>	P&L Type	enum	0	0=Value in money, 1=Value in %
<b>_daily_loss_value</b>	Daily Loss	double	0	Max daily loss, 0=OFF
<b>_daily_profit_value</b>	Daily Profit	double	0	Max daily profit, 0=OFF
<b>_daily_resume_hour</b>	Daily Resume Hr	int	—	Hour to resume after daily limit
<b>_daily_resume_minute</b>	Daily Resume Min	int	—	Minute to resume
<b>_weekly_loss_value</b>	Weekly Loss	double	0	Max weekly loss, 0=OFF
<b>_weekly_profit_value</b>	Weekly Profit	double	0	Max weekly profit, 0=OFF
<b>_weekly_resume_day</b>	Resume Day	enum	—	Day of week to resume
<b>_resume_hour_week</b>	Weekly Resume Hr	int	—	Hour to resume weekly
<b>_resume_minute_week</b>	Weekly Resume Min	int	—	Minute to resume weekly

## Hours Filter Settings

Variable	Display	Type	Default	Description
<code>_allow_trading_time</code>	Enable	bool	false	Toggle hours filter
<code>_trade_start_hour</code>	Start Hour	int	—	Trading start hour (0-23)
<code>_trade_start_minute</code>	Start Minute	int	—	Trading start minute (0-59)
<code>_trade_end_hour</code>	End Hour	int	—	Trading end hour (0-23)
<code>_trade_end_minute</code>	End Minute	int	—	Trading end minute (0-59)
<code>_use_server_time_zone</code>	Server Time	bool	true	Use broker server timezone
<code>_timezone</code>	UTC Offset	int	-4	Manual UTC offset
<code>_adjust_daylight</code>	DST Adjust	bool	false	Auto-adjust for DST
<code>_close_on_time_filter</code>	Close Mode	enum	0	0=Close, 1=Close On Friday Only

## News Filter Settings

Variable	Display	Type	Default	Description
<code>_allow_news_filter</code>	Enable	bool	false	Toggle news filter
<code>_auto_close_on_news</code>	Auto Close	bool	true	Close positions before news (sinput)
<code>_high</code>	High Impact	bool	—	Filter high-impact events (sinput)
<code>_medium</code>	Medium Impact	bool	—	Filter medium-impact events (sinput)
<code>_low</code>	Low Impact	bool	—	Filter low-impact events (sinput)
<code>_high_pause</code>	High Pause	int	60	Pause seconds for high impact
<code>_medium_pause</code>	Medium Pause	int	60	Pause seconds for medium impact
<code>_low_pause</code>	Low Pause	int	60	Pause seconds for low impact

Additional visual parameters (`_draw`, colors, font, line style) control the news line rendering on the chart.

## Signal Rule Parameters (Per Slot)

Each of the 12 signal modes has 3 slots. The naming convention is: `_[ModeName]_[param]_[slot#]`. Common parameters across all modes:

Pattern	Display	Type	Default	Description
<b>_[Mode]_allow_N</b>	Enable	bool	false	Enable this rule slot
<b>_[Mode]_frame_N</b>	Timeframe	TF enum	CURRENT	Timeframe for evaluation
<b>_[Mode]_role_N</b>	Role	enum	0	0=Entry, 1=Filter

Signal-specific parameters vary by mode. NikaCross adds `ma_type`, `ma_price`, `t3_factor`, `enable_HA`, `length`, `nika_ma_period`, `shift`. MACross adds fast/slow MA configurations. 2SigCross adds separate A/B instance parameters. HistLevCross/HistUD add HA toggle, `length`, and smoothing period.

## Tester Settings

Variable	Display	Type	Default	Description
<b>_drawdown_type</b>	Fitness Function	enum	0	0-4: Five OnTester formulas
<b>_show_indicators</b>	Show Indicators	bool	—	Render indicators in visual test
<b>_nika_name</b>	Indicator Name	string	—	Custom indicator filename
<b>_ma_name</b>	MA Library Name	string	—	Custom MA library filename

## Dashboard Settings

Variable	Display	Type	Default	Description
<b>_show_dashboard</b>	Show Dashboard	bool	true	Toggle chart dashboard
<b>_max_dashboard_actions</b>	Max Actions	int	10	Recent actions shown

Additional dashboard parameters control font, colors, and positioning of the overlay elements.

Chapter 17

# SET File Format Guide

SET files store EA parameter configurations for MT5. Understanding the format enables automated parameter management, batch optimization, and programmatic strategy deployment.

## File Encoding

V1.27 SET files use **UTF-16 Little Endian** encoding with a Byte Order Mark (BOM). This is the standard encoding for MT5 SET files. The BOM bytes are FF FE at the start of the file. Any tool or script reading/writing SET files must handle this encoding correctly — using UTF-8 or ASCII will corrupt the file.

## Line Format

Each parameter is stored on a single line with the following format:

```
paramname=value | start | step | stop | Y/N
```

Field	Description
paramname	Parameter variable name (e.g., _sl_pips, _max_entries)
value	Current parameter value
start	Optimization start value (for Strategy Tester optimization)
step	Optimization step increment
stop	Optimization end value
Y/N	Y = optimize this parameter, N = fixed (not optimized)

Example lines:

```
_sl_pips=50 | 30 | 5 | 100 | N
_atr_multiplier=1.5 | 1.0 | 0.1 | 3.0 | Y
_NikaCross_allow_1=true | 0 | 0 | 0 | N
```

## Naming Conventions

V1.27 parameter names follow specific patterns:

- **Standard parameters:** Underscore prefix (e.g., `_sl_pips`, `_max_entries`, `_trade_direction`).
- **Alert parameters:** No underscore prefix (`allow_sound_alerts`, `allow_push_alerts`, `allow_email_alerts`).
- **Daily trailing:** camelCase (`trailStartInput`, `trailStepInput`, `trailAmountInput`, `DailyResumeHour`, `DailyResumeMinute`).
- **Signal rules:** `_[ModeName]_[param]_[slot#]` (e.g., `_NikaCross_allow_1`, `_MACross_frame_2`).
- **News sinput params:** Underscore prefix (`_auto_close_on_news`, `_high`, `_medium`, `_low`).

## Loading in MT5

To load a SET file in MetaTrader 5: Open the EA properties dialog, click the 'Load' button, navigate to the SET file, and select it. MT5 will populate all parameters from the file. Parameters not present in the SET file will retain their current values.

Chapter 18

Optimization Suggestions

Effective optimization of the Nika EA requires a systematic approach. With over 200 configurable parameters, exhaustive grid search is impractical. This chapter provides a structured optimization methodology.

What to Optimize First

Parameters should be optimized in order of impact, starting with the signal engine and working outward to risk controls:

Priorit y	Category	Parameters	Rationale
1	Signal Mode Selection	Which signal mode(s) to use and on which timeframes	Highest impact — determines the fundamental strategy
2	Signal Parameters	Length, MA type, MA period per rule slot	Defines signal sensitivity and lag
3	Stoploss Method & Distance	SL method, pips/ATR multiplier/promille	Directly affects risk-reward and win rate
4	Take Profit Levels	TP multipliers and partial close percentages	Determines profit capture behavior
5	Trailing Stop	Mode, start, step, stop/pullback parameters	Affects how much of a move is captured
6	Re-Entry Settings	Max re-entries, cooldown bars	Controls loss recovery behavior
7	Risk Filters	Daily/weekly limits, hours, news	Fine-tuning — reduce exposure to known risks

Suggested Parameter Ranges

The following ranges provide reasonable starting points for optimization. Adjust based on the target timeframe and instrument.

Parameter	Range	Step	Notes
Signal length	5-100	5	Shorter for scalping, longer for swing
Nika MA period	3-30	1	Controls average line smoothing
SL Pips	10-200	10	Symbol and timeframe dependent

Parameter	Range	Step	Notes
ATR Multiplier	0.5-4.0	0.25	1.0-2.0 most common range
ATR Period	7-28	7	14 is standard, test around it
TP1 Multiplier	0.5-3.0	0.25	First target distance
TP2 Multiplier	1.0-5.0	0.5	Second target distance
Partial Close %	10-50	10	Per TP level
Trailing Start (RRR)	0.2-2.0	0.1	When trailing activates
Trailing Step (RRR)	0.1-1.0	0.1	Step increment
Risk %	0.1-2.0	0.1	Position sizing

## Walk-Forward Testing Approach

Walk-forward analysis (WFA) is essential for validating optimization results and detecting overfitting. The recommended approach:

- **Step 1 — In-Sample Optimization:** Optimize on the first 70% of available history (e.g., 2 years of 3 years total).
- **Step 2 — Out-of-Sample Validation:** Test the optimized parameters on the remaining 30% (the most recent period).
- **Step 3 — Rolling Window:** Shift the window forward by 3-6 months and repeat. This produces multiple in-sample/out-of-sample periods.
- **Step 4 — Consistency Check:** Parameters should be similar across rolling windows. Large variation indicates overfitting.
- **Step 5 — Final Validation:** Use the most recent out-of-sample period as the final decision point.

Use Fitness Function 2 or 4 for walk-forward optimization — they reward consistency over raw returns, which produces parameters more likely to perform out-of-sample.

## Multi-Symbol Considerations

When deploying the EA across multiple symbols:

- **Universal vs. Symbol-Specific:** Some parameter sets work across multiple pairs (universal), while others are tuned per symbol. Test both approaches.
- **Correlation:** Avoid deploying highly correlated pairs (e.g., EURUSD and USDCHF short) with the same strategy — losses will compound.
- **Max Open Lot:** Use `_max_open_lot` with 'All Symbols' scope to cap total account exposure across all EA instances.
- **Magic Numbers:** Ensure unique magic numbers per symbol instance to prevent cross-interference.
- **Stoploss Method:** ATR or Promille methods auto-adjust to each symbol's volatility, making them better for multi-symbol deployment than Fixed Pips.



## Common Optimization Pitfalls

- **Curve Fitting:** Over-optimizing on historical data produces parameters that perform perfectly in-sample but fail live. Always validate out-of-sample.
- **Too Many Parameters:** Optimizing more than 3-4 parameters simultaneously exponentially increases the search space. Optimize in stages.
- **Insufficient History:** Use at least 1-2 years of data for daily strategies, 3-6 months for intraday. More is better.
- **Ignoring Spread:** Backtest results can be misleading if spread is set too low. Use realistic spread settings.
- **Selection Bias:** Don't cherry-pick the best result — look for parameter regions (clusters) where many nearby settings perform well.