Kestutis Balciunas

NINJATRADER 8 INDICATOR · NINJASCRIPT, C#

User Guide

Order blocks, fair value gaps, equal levels, liquidity sweeps and previous-period levels, signals on closed bars with an entry, a stop and two targets, a 30-row dashboard with a 24-timeframe scanner, alerts, and series a strategy can read.

PRODUCT	Smart Liquidity Matrix for NinjaTrader 8
VERSION	1.0.0
DATE	November 2026
NINJATRADER	8.1.x Desktop, tested on 8.1.9.0
PACKAGE	`SmartLiquidityMatrix.zip`, compiled NinjaScript Add-On
SHA-256	`f0e11bc685ec44af74c53783d42c3d66157b6fafafdff9a99cbe4662154cc172`
AUTHOR	Kestutis Balciunas, https://kestutisbalciunas.com/
PRODUCT PAGE	https://kestutisbalciunas.com/ninjatrader/smart-liquidity-matrix/
SUPPORT	https://kestutisbalciunas.com/contact/

Trading involves substantial risk. Futures, foreign currency and options trading contains substantial risk and is not for every investor. Past, simulated and backtested performance is not necessarily indicative of future results.

Contents

01 What it is

02 Installation

03 Quick start

04 Calculate mode and bar timing

05 Outputs for strategies and the Market Analyzer

06 Parameters

07 Sessions, time and volume

08 Compatibility

09 Troubleshooting

10 How this edition was verified

11 Licensing

12 Support

13 Risk disclosure

01

What it is

Chart indicator for NinjaTrader 8 Desktop (NinjaScript, C#) for order blocks, fair value gaps, equal highs and lows, liquidity sweeps and previous-period levels. One state machine, evaluated on closed bars, prints BUY and SELL signals with an entry line, a stop and two targets. A 30-row dashboard shows the market, the trend readings, the active zones and the active signal, and a scanner runs the same state machine on twenty-four timeframes of the chart's instrument. It has alerts and eight series a strategy can read.

It only draws and alerts: it never places, modifies or closes orders. A webhook is not part of this edition: the indicator makes no network call of its own.

02

Installation

Requirements

- NinjaTrader 8.1.x Desktop for Windows (tested on 8.1.9.0). Not for NinjaTrader 8.0 or NinjaTrader Web.
- A data connection that delivers bars for the chart and for the other timeframes the indicator requests, daily bars included. The previous-period levels, the higher-timeframe row and the 1D, 1W and 1M scanner rows read NinjaTrader's daily, weekly and monthly series of the instrument, so a gap in those series shows on the chart (see the chapter on sessions, time and volume).
- No Order Flow+ features, no depth, no tick history. Volume is read only by the dashboard's Volume row and by the optional Live Confluence filter.

Import the Add-On package

1. Control Center > Tools > Import > NinjaScript Add-On.
2. Select `SmartLiquidityMatrix.zip` and confirm. NinjaTrader compiles the assembly and reports that the import is complete.
3. No restart is needed. If a chart already had an older version, remove and re-add the indicator (a compile does not reload running instances).

Verify installation

1. Open a chart with at least a few hundred bars (any instrument, a 1 to 60 minute interval is a good first look), press Ctrl+I.
2. In the Available list find **Smart Liquidity Matrix**, click add, click OK. The first load requests 24 more data series of the same instrument, so it takes longer than a single-series indicator.
3. The NinjaScript Output window prints a line such as `Smart Liquidity Matrix v1.0.0 by Kestutis Balciunas: 4924 bars loaded, 24 of 24 secondary series, Calculate OnBarClose`.
4. The chart shows the dashboard, translucent order-block and fair-value-gap boxes, faint grey boxes for zones price has already closed through, `1D High`, `1D Low` and `EQ 50%` level lines near the right edge, and `BUY ***` or `SELL ***` plates where a signal printed. If the lowest dashboard rows are cut off, set **10 Dashboard > Text Size** to **Small**.

Uninstall / update

- Remove: Control Center > Tools > Remove NinjaScript Assembly > Smart Liquidity Matrix.
- Update: import the new zip over the installed version (NinjaTrader replaces the assembly), then remove and re-add the indicator on every chart that uses it, or reload the workspace.

03

Quick start

First chart

1. Add **Smart Liquidity Matrix** to a chart (Ctrl+I). Leave every setting at its default.
2. **On a small price panel, pick a smaller dashboard.** The dashboard has 30 rows. At the default **10 Dashboard > Text Size** of Normal it is about 615 pixels tall at 100 % Windows display scaling, plus an 8-pixel margin, so a 590-pixel price panel cuts off its last scanner rows. Set **Text Size** to **Small** (about 510 pixels). Large is about 780 pixels. The sizes grow with the display scaling.
3. Load enough history. The swings and zones on screen are built from the bars before them, and the indicator does not extend the chart's data: the Data Series setting **Days to load** decides. Aim for at least 5,000 bars where your feed allows it. The other timeframes load their own bars, independent of the chart's setting.
4. Signals are decided on closed bars. The forming bar is not calculated.

Reading the chart

- **Order blocks**: each confirmed swing low creates a bullish block and each confirmed swing high a bearish one. The block is the high to low of the first down candle (bullish block) or up candle (bearish block) found looking back from the swing, across **OB Candle Scan Range** bars; if there is none, the swing bar itself. A tag reads `Bull OB` or `Bear OB`. Cyan is bullish, red is bearish. An active block runs to five bars past the last bar. At most **Bullish OB Count** and **Bearish OB Count** blocks stay active, 3 per side by default, and an older one is removed.
- **Fair value gaps**: a gap between a bar's low and the high two bars back (bullish, green) or a bar's high and the low two bars back (bearish, red). Dashed border. At most 3 per side stay active by default.
- **Faint grey boxes**: a zone a close has gone through, below a bullish zone or above a bearish one. It turns grey, stops at that bar, loses its tag and stays on the chart. It takes no further part in anything. The newest 501 boxes stay, active and grey together.
- **BUY and SELL**: a green `BUY ***` plate under the bar's low or a red `SELL ***` plate over its high, with a small triangle. The stars are always three: the score is fixed in the calculation and does not vary.
- **Trade levels**: with **Show SL/TP Levels** on, a signal draws a white solid ENTRY line, a red dashed SL line and two green dashed lines, TP1 and TP2, each running 30 bars from the signal bar, with a price tag just past the end. Only the newest signal's set exists, and it is removed when the signal state closes.
- **Liquidity sweeps**: a purple `Sweep` plate and a small cross. Upside: the bar's high went above the level and it closed back below. Downside is the mirror image. The level is an equal high or low on that bar, or the previous period's level.
- **Equal highs and lows**: when the last two swing highs (or lows) lie within 0.1 ATR of each other but are not exactly equal, a dotted line and a wide faint line carry the tag `EQH` (or `EQL`). They run from the older swing to five bars past the last bar and exist only while the newest bar's state holds them.
- **Previous-period levels**: `1D High` and `1D Low`, a solid line and a wide faint one from 30 bars back to five bars past the last bar. The tag prints **Previous Period TF** as you typed it. They come from NinjaTrader's own series for that timeframe: if the daily bars have a gap, these levels are the last ones before the gap (see the chapter on sessions, time and volume).
- **Equilibrium**: a grey dotted `EQ 50%` line halfway between the current swing high and swing low.

Tags past the last bar are pinned at the right edge, and step left of the dashboard when it would cover them. The price axis follows the bars, not the drawings, so a far stop or target can lie outside the panel: drag the price axis to see it.

What you will not see

No BoS or CHoCH line or label is ever drawn, and no candle is tinted. The structure-break events exist in the calculation and do not fire, in any edition of this product (0 events on the 4,923 bars of the verification run). So **Show Break of Structure**, **Show Change of Character**, their colours and **Max Structure Lines** have no visible effect, the dashboard's **Chart Trend**

always reads NEUTRAL, and its HTF row reads `Divergent` beside an HTF reading. The signals, zones, sweeps and levels do not depend on them.

Minimum Confluence Score, **Signal Cooldown (bars)**, **Use HTF Trend Filter** and **HTF EMA Length** do not filter a signal by themselves. **Use HTF Trend Filter** selects the dashboard's HTF row, and **HTF EMA Length** sets the average behind that row, which the optional filters also read. The working filters are in **13 Signal Filters**, off by default.

Signal rules in one paragraph

A **BUY** prints when the signal state goes long on a closed bar. That needs EMA(20) above EMA(50), a bar that closed above its open, a low at or below the last 5-bar swing low plus 0.3 x ATR(14), a close above that swing low, and RSI(14) under 70. The state goes flat when the low touches its stop (that swing low minus 0.5 x ATR), the high touches its target (entry plus 2 x risk), or EMA(20) falls below EMA(50). A **SELL** is the mirror image, with RSI above 30. The exit is tested before the entry on each bar.

The drawn levels are not the levels that close the state. The lines are a risk picture at your settings. The state keeps its own:

	DRAWN ON THE CHART	WHAT CLOSES THE STATE
swing used	**Swing Length**	always 5
stop	swing minus **SL Padding** x ATR	swing minus 0.5 x ATR
targets	entry plus risk, and plus risk x **Risk:Reward Ratio**	entry plus 2 x risk

Risk is the distance from the entry to the drawn stop, at least 10 ticks. At the defaults (Swing Length 5, SL Padding 0.2, ratio 2.0) the stops differ, so a drawn line can be crossed while the dashboard still shows the trade, or the trade can close with its line untouched. The indicator does not mark a line as reached or count outcomes on the chart.

The dashboard

ROWS	WHAT THEY READ
header	product, instrument, chart period and version
MARKET	the newest price and its change from the session's daily open; the instrument type and currency
TREND ANALYSIS	Chart Trend (always NEUTRAL, see above); HTF: price against the higher-timeframe EMA; RSI (14); ATR (14) with its share of the price (Low, Normal, HIGH)
CONFLUENCE MAP	active order blocks and fair value gaps per side; equal levels; the newest bar's volume against its 20-bar average
ACTIVE SIGNAL	direction and bars since the signal; the entry with its open result in percent; the drawn stop, TP1 and TP2 with their distance in percent
SIGNAL SCANNER	`BUY`, `SELL` or `--` per timeframe: `1m` to `45m`, `1H` to `12H`, `1D`, `1W`, and `1M`, which is one month

Each scanner cell is the same state machine, run on that timeframe's own closed bars. `--` means flat, or a row that has not delivered yet: the state needs at least 50 closed bars of its timeframe. **Scanner Warm-up Marker** on shows a dot on a row that has delivered nothing yet. The scanner feeds no signal, no drawing and no alert. The footer's `Non-Repainting` refers to the signal path: signals are decided on closed bars. **Position** offers six places on the chart.

Settings to try first

SETTING	WHAT IT CHANGES
Text Size	Small, Normal or Large. See step 2 above.
Swing Length	Chart swings, so the order blocks, the equal levels, the equilibrium and the drawn stop. Not the signals: the state keeps its own 5-bar swing.
Bullish / Bearish OB Count, **Bullish / Bearish FVG Count**	How many zones stay active per side. 0 creates none of that kind.

SETTING	WHAT IT CHANGES
Label Position	Inside (at the block's middle), Outside (at the top of a bullish block, the bottom of a bearish one) or None.
Lite Mode	Thinner, fainter drawing: no wide faint lines, thinner borders.
Show SL/TP Levels	Hides the lines and tags only. The signals are unchanged.
Highlight Liquidity Sweeps	Off stops the sweep events themselves, so their alerts stop too, not only the drawing.
Previous Period TF	Minutes as a number (`60`, `240`), or `1D`, `2D`, `3D`, `1W`, `1M`. A value it does not recognise means one day.
Scanner Rows	Off adds only the series the levels need and leaves every row at `--`. Use it on a slow PC.
13 Signal Filters	A cooldown, a higher-timeframe filter and a score filter. A suppressed signal can print on a later bar while the state still holds. All off by default.

Alerts

Enable Alerts is the master switch and **Alert On** picks the events: `SignalsOnly` by default, `SignalsAndSweeps`, `SignalsAndStructure` or `All`. They are evaluated once per closed bar and only in realtime, never during the history pass, and never for a bar that closed more than two bar periods plus five seconds ago.

The messages:

- A BUY prints: `Smart Liquidity Matrix: BUY signal on MNQ 12-26 5 | Score: 3/5 | Price: 28746.25`
- A SELL prints: `Smart Liquidity Matrix: SELL signal on MNQ 12-26 5 | Score: 3/5 | Price: 28746.25`
- A bar sweeps above a level: `SLM: Upside liquidity sweep on MNQ 12-26 5`
- A bar sweeps below a level: `SLM: Downside liquidity sweep on MNQ 12-26 5`

The message carries the instrument, the chart period (`5`, `60`, `D`, `W`, `M`, `30S`, `500T`) and, for a signal, the bar's close. The structure events are selectable and do not fire (see above). **Webhook JSON Format** changes the Buy and Sell text to JSON with the same values plus the drawn stop and targets; it is the message only.

CHANNEL	DEFAULT	WHAT ARRIVES
Enable Log Alerts	on	one row in the Alerts Log and the same line in the NinjaScript Output window
Enable Sound Alerts	off	the file named in **Sound File** (default `Alert1.wav`): a name inside NinjaTrader's sounds folder or a full path; blank plays nothing
Enable E-mail Alerts	off	an e-mail to **E-mail To**, with **E-mail Subject** and the message as its body, through NinjaTrader's Mail share service (Tools > Options > Share)

A webhook is not part of this edition: the indicator makes no network call of its own.

The final bar of a session

A signal on a session's final bar appears after the reopen. NinjaTrader closes a bar when the next bar's first tick arrives, so on a closed market that last bar is calculated and drawn only at the first tick after the reopen. Its alert is raised only if the reopen comes within two bar periods plus five seconds of the bar's close; otherwise the signal prints with no alert.

Using the values in a strategy or the Market Analyzer

Eight hidden series are exposed: `Signal` (+1, -1, 0), `ActiveDirection` (+1, -1, 0), `ActiveStop`, `ActiveTp1`, `ActiveTp2`, `Sweep` (+1, -1, 0), `PreviousHigh` and `PreviousLow`. Example: `if (SmartLiquidityMatrix(...).Signal[0] > 0)`. With `OnEachTick` or `OnPriceChange` the bar that just closed is `[1]`. Neither use has been tested yet. The indicator adds secondary series, and NinjaTrader expects a calling strategy to add the same series.

What it does not do

It places no orders and sends nothing over a network of its own. It draws what the closed bars show; it does not forecast where price goes next.

04

Calculate mode and bar timing

- Default `Calculate.OnBarClose`: one engine commit per closed bar.
- `OnEachTick` / `OnPriceChange`: the closed bar is committed once, at the first tick of the next bar. Between commits the last-bar block reruns at most once a second: it extends the active zones, recreates the level lines and refreshes the dashboard's price and volume. The forming bar is not fed to the calculation in any mode, so these modes print the same signals on the same bars and do not make a signal earlier.
- A BUY, a SELL, a sweep or a zone's creation is decided on a closed bar and does not change afterwards. What moves on the newest bars by design: an active zone's right edge, which sits five bars past the last bar; the equal-level, previous-period and equilibrium lines, which exist only for the newest state; the dashboard and the scanner.
- A swing is confirmed **Swing Length** bars after its own bar, and the signal state's own swing five bars after. That is a delay, and what it allows can only happen after the confirmation.
- **A signal on a session's final bar appears after the reopen.** NinjaTrader closes a bar when the next bar's first tick arrives, so on a closed market the session's last bar stays uncommitted until the first tick after the reopen, and only then is it calculated and drawn. Its alert is raised only if the reopen comes within two bar periods plus five seconds of that bar's close; otherwise the signal prints with no alert.
- Historical pass: one call per closed bar. A still-forming last bar is not committed. The engine slot is the NinjaTrader bar index and a skipped range is refilled from the series, so the historical to realtime transition can neither double-feed nor skip a bar.
- With `OnEachTick` / `OnPriceChange`, Tick Replay takes the same path as per-tick realtime.
- Secondary series calculate nothing when they update. With `OnBarClose`, the dashboard and the scanner refresh at each chart bar's close and, in realtime, when a secondary bar closes, at most once a second. A 1m scanner row therefore does not wait for a 1H chart bar.
- The indicator keeps calculating while its chart window is minimized (`IsSuspendedWhileInactive = false`), and an alert is raised only for a bar that closed within the last two bar periods plus five seconds, so a catch-up after a suspension or a reconnect does not replay old alerts. On tick, volume, range and other non-time bars the period in that limit is the smallest spacing of the first 300 bars, so on a slow chart of that kind an alert can be withheld.
- A bar's OPEN time is its NinjaTrader close time minus the bar period; on a daily or longer chart, the session end of the bar before it; on non-time bars, the close stamp of the bar before it. The level series are mapped on it.
- The price axis follows the bars, not the drawings (`IsAutoScale = false`), so a far stop or target can lie outside the panel.

05

Outputs for strategies and the Market Analyzer

PLOT / SERIES	VALUES
`Signal` (plot 0, hidden)	+1 on the closed bar of a BUY, -1 on the closed bar of a SELL, 0 otherwise
`ActiveDirection` (plot 1, hidden)	the signal state after the bar: +1 long, -1 short, 0 flat
`ActiveStop` (plot 2, hidden)	the drawn stop of the newest signal; no value before the first signal
`ActiveTp1` (plot 3, hidden)	the drawn TP1 of the newest signal; no value before the first signal
`ActiveTp2` (plot 4, hidden)	the drawn TP2 of the newest signal; no value before the first signal

PLOT / SERIES	VALUES
`Sweep` (plot 5, hidden)	+1 on an upside liquidity sweep bar, -1 on a downside one, 0 otherwise
`PreviousHigh` (plot 6, hidden)	the previous period's high as this bar saw it; no value before the first delivery
`PreviousLow` (plot 7, hidden)	the previous period's low as this bar saw it; no value before the first delivery

The series are not drawn. `ActiveStop`, `ActiveTp1` and `ActiveTp2` keep their last values after the trade closed; `ActiveDirection` says whether one is open. They are the drawn levels, which are not the levels that close the signal state (see the quick start). `Signal` follows the signal state and not the drawing switches, so **Show SL/TP Levels** off leaves it unchanged. `Sweep` is the exception: **Highlight Liquidity Sweeps** off also stops the sweep events themselves, and the series then reads 0.

With `Calculate.OnBarClose`, `[0]` is the bar that just closed. With `OnEachTick` / `OnPriceChange`, the bar that just closed is `[1]` and the forming bar `[0]` has not been written.

Example in a strategy: `var slm = SmartLiquidityMatrix(...); if (slm.Signal[0] > 0) ...` (the accessor is generated by NinjaTrader on compile; every input except the 13 colours is one of its 54 arguments). Calling the indicator from a strategy or a Market Analyzer column has not been tested. It adds secondary series, and NinjaTrader expects a calling strategy to add the same series.

06

Parameters

The inputs keep the titles, order, defaults and ranges of the other editions (the group names carry a number prefix because NinjaTrader sorts groups alphabetically). NinjaTrader additions: E-mail To and the two diagnostic switches in group 16. Sound File defaults to a file in NinjaTrader's sounds folder. The indicator makes no network call, so Webhook JSON Format selects the alert text only.

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
01 Market Structure	Swing Length	int	5	2 to 20	`SwingLen`	Bars on each side of a swing high or low on the chart. The signal state keeps its own 5-bar swing.
01 Market Structure	Show Break of Structure	bool	true		`ShowBoS`	
01 Market Structure	BoS Color	colour	#2962FF		`BosBrush`	
01 Market Structure	Show Change of Character	bool	true		`ShowChoch`	
01 Market Structure	CHoCH Color	colour	#FF6D00		`ChochBrush`	
01 Market Structure	Max Structure Lines	int	5	1 to 20	`MaxStructLines`	
02 Order Blocks	Bullish OB Count	int	3	0 to 10	`ShowBullOB`	
02 Order Blocks	Bullish OB Color	colour	#00BCD4		`BullOBBrush`	
02 Order Blocks	Bearish OB Count	int	3	0 to 10	`ShowBearOB`	
02 Order Blocks	Bearish OB Color	colour	#FF5252		`BearOBBrush`	
02 Order Blocks	OB Candle Scan Range	int	5	1 to 10	`ObScanBars`	
03 Fair Value Gaps	Bullish FVG Count	int	3	0 to 10	`ShowBullFVG`	
03 Fair Value Gaps	Bullish FVG Color	colour	#4CAF50		`BullFVGBrush`	
03 Fair Value Gaps	Bearish FVG Count	int	3	0 to 10	`ShowBearFVG`	
03 Fair Value Gaps	Bearish FVG Color	colour	#F44336		`BearFVGBrush`	
04 Liquidity & Equal Levels	Show Equal Highs/Lows	bool	true		`ShowEqLevels`	
04 Liquidity & Equal Levels	Equal Highs	colour	#FF5252		`EqHighBrush`	
04 Liquidity & Equal Levels	Equal Lows	colour	#00BCD4		`EqLowBrush`	
04 Liquidity & Equal Levels	Highlight Liquidity Sweeps	bool	true		`ShowLiqSweep`	Also switches the sweep events and their alerts on or off, as in the other editions.

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
04 Liquidity & Equal Levels	Sweep Color	colour	#AA00FF		`LiqSweepBrush`	
05 Previous Period Levels	Previous Period TF	string	"1D"		`PpTf`	Timeframe of the previous-period levels: minutes as a number (60, 240), 1D, 1W or 1M. Printed on the level tags as typed.
05 Previous Period Levels	Previous Period High	bool	true		`ShowPph`	
05 Previous Period Levels	PP High Color	colour	#FF5252		`PphBrush`	
05 Previous Period Levels	Previous Period Low	bool	true		`ShowPpl`	
05 Previous Period Levels	PP Low Color	colour	#00BCD4		`PplBrush`	
05 Previous Period Levels	Show Equilibrium Level	bool	true		`ShowEquilibrium`	
06 Signal Engine	Enable Buy/Sell Signals	bool	true		`EnableSignals`	
06 Signal Engine	Minimum Confluence Score	int	2	2 to 5	`MinConfluence`	Has no effect on the signals in any edition. Group 13 holds the filter that does.
06 Signal Engine	Signal Cooldown (bars)	int	20	5 to 100	`SignalCooldown`	Has no effect on the signals in any edition. Group 13 holds the filter that does.
07 Risk Management	Show SL/TP Levels	bool	true		`ShowSlTp`	
07 Risk Management	Risk:Reward Ratio	double	2.0	1.0 to 5.0	`RrRatio`	Scales the drawn TP2. The signal state closes on its own fixed 2.0.
07 Risk Management	SL Padding (x ATR)	double	0.2	0.0 to 2.0	`SlPadding`	Scales the drawn stop. The signal state closes on its own stop, 0.5 x ATR beyond its pivot.
08 Higher Timeframe Filter	Use HTF Trend Filter	bool	true		`UseHtf`	Selects the dashboard's HTF row only. Group 13 holds the filter that acts on signals.
08 Higher Timeframe Filter	Higher Timeframe	string	"1D"		`HtfTf`	Timeframe of the HTF average: minutes as a number (60, 240), 1D, 1W or 1M.
08 Higher Timeframe Filter	HTF EMA Length	int	50	10 to 200	`HtfEmaLen`	
09 Visual Settings	Lite Mode	bool	false		`LiteMode`	
09 Visual Settings	Label Position	NinjaTrader.NinjaScript.SlmLabelPos	NinjaTrader.NinjaScript.SlmLabelPos.Inside		`LabelPos`	
09 Visual Settings	Buy Signal Color	colour	#00E676		`BuySignalBrush`	
09 Visual Settings	Sell Signal Color	colour	#FF1744		`SellSignalBrush`	
10 Dashboard	Show Dashboard	bool	true		`ShowDashboard`	
10 Dashboard	Position	NinjaTrader.NinjaScript.SlmDashPos	NinjaTrader.NinjaScript.SlmDashPos.TopRight		`DashPos`	

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
10 Dashboard	Text Size	NinjaTrader.NinjaScript.SlmDashSize	NinjaTrader.NinjaScript.SlmDashSize.Normal		`DashSize`	Thirty rows at Normal are about 615 pixels tall. On a short chart use Small.
11 Alerts	Enable Alerts	bool	true		`EnableAlerts`	Master switch of the alerts. Group 15 selects the events and channels.
11 Alerts	Webhook JSON Format	bool	false		`WebhookMode`	Signal alert text as JSON instead of a sentence (the message only; no network call is made).
12 Scanner	Scanner Rows	NinjaTrader.NinjaScript.SlmScannerRows	NinjaTrader.NinjaScript.SlmScannerRows.On		`ScannerRows`	On adds 24 data series to the chart. Off adds only those the levels need and leaves the rows at --.
12 Scanner	Scanner Warm-up Bars	int	600	120 to 5000	`ScannerWarmupBars`	Bars each scanner row loads (1W at most 520, 1M at most 240).
12 Scanner	Scanner Warm-up Marker	bool	false		`ScannerWarmupMarker`	Shows a dot instead of -- on a row that has delivered nothing yet.
12 Scanner	Scanner Debug Log	bool	false		`ScannerDebug`	Prints how many rows are warm to the NinjaScript Output window, once per chart bar.
13 Signal Filters	Use Live Cooldown	bool	false		`UseLiveCooldown`	
13 Signal Filters	Live Cooldown (bars)	int	20	1 to 500	`LiveCooldownBars`	
13 Signal Filters	Use Live HTF Filter	bool	false		`UseLiveHtfFilter`	
13 Signal Filters	Use Live Confluence	bool	false		`UseLiveConfluence`	
13 Signal Filters	Live Minimum Score	int	2	2 to 5	`LiveMinScore`	
14 Visuals	Max Chart Objects	int	900	100 to 4000	`MaxChartObjects`	Upper limit on what the zones paint in view, two per zone.
14 Visuals	Label Pool Size	int	256	32 to 512	`LabelPoolSize`	Upper limit on labels painted in view. Trade and level tags come first, signal and sweep history last.
14 Visuals	Marker Half Width (bars)	double	1.0	0.2 to 5.0	`MarkerHalfWidth`	
14 Visuals	Marker Height (% of pane)	double	1.2	0.1 to 10.0	`MarkerHeightPct`	
14 Visuals	Clamp Right-Edge Labels	bool	true		`ClampRightEdgeLabels`	Keeps tags past the last bar readable at the right edge of the chart and beside the dashboard.
15 Alert Channels	Enable Log Alerts	bool	true		`EnableLogAlerts`	The Alerts Log window and the NinjaScript Output window.
15 Alert Channels	Enable Sound Alerts	bool	false		`EnableSoundAlerts`	

GROUP	PARAMETER	TYPE	DEFAULT	RANGE	PROPERTY	NOTES
15 Alert Channels	Sound File	string	"Alert1.wav"		`SoundFile`	A .wav file name in NinjaTrader's sounds folder, or a full path.
15 Alert Channels	Enable E-mail Alerts	bool	false		`EnableEmailAlerts`	Sends through NinjaTrader's Mail share service.
15 Alert Channels	E-mail To	string	""		`EmailTo`	
15 Alert Channels	E-mail Subject	string	"Smart Liquidity Matrix"		`EmailSubject`	
15 Alert Channels	Alert On	NinjaTrader.NinjaScript.SlmAlertOn	NinjaTrader.NinjaScript.SlmAlertOn.SignalsOnly		`AlertOn`	Which events raise an alert.
16 NinjaTrader	Debug Mode	bool	false		`DebugMode`	Writes diagnostics to slm_nt_debug.log in the NinjaTrader 8 folder.
16 NinjaTrader	Dump Bars + Outputs CSV	bool	false		`DebugDumpCsv`	Writes the bars and the engine outputs to slm_dump_<instrument>_<period>.csv and the other timeframes' bars to a folder beside it (parity checks).

67 parameters.

07

Sessions, time and volume

Sessions

The indicator has no session logic of its own for the chart: every bar the chart delivers is fed to the engine in order, so the chart's Trading Hours template decides which bars exist. With an ETH template (for example `CME US Index Futures ETH`) the overnight bars take part in the swings, the zones, the signals and the ATR; with an RTH template they do not. A zone across a session gap simply spans the bars on both sides.

The other timeframes are requested for the chart's instrument, with the template left to NinjaTrader's default for such a request. The verification run used overnight hours. A chart on an RTH template has not been tested, so its levels and scanner rows may follow the instrument's default template rather than the chart's.

The final bar of a session

A signal on a session's final bar appears after the reopen. NinjaTrader closes a bar when the next bar's first tick arrives, and the indicator calculates a bar only once it has closed, so on a closed market the last bar stays uncommitted until the first tick after the reopen. It is then calculated and drawn. Its alert is raised only if the reopen comes within two bar periods plus five seconds of that bar's close; otherwise the signal prints with no alert. The same applies to any later close on a quiet instrument: the bar is committed when the next tick starts a new bar.

History

The swings and zones on screen are built from the bars before them, starting from the first bar the chart loaded. The indicator does not extend the chart's data, so the Data Series setting **Days to load** decides how much history is behind them. Two charts that loaded different amounts of history can differ for a while from where the shorter one starts. The newest 501 boxes, 501 lines and 501 labels stay on the chart, and the oldest are retired first. Grey zones count among the boxes.

Other timeframes

Each secondary series loads its own bars, independent of the chart's setting: each scanner row **Scanner Warm-up Bars** plus 50 (600 plus 50 by default; `1W` at most 520 and `1M` at most 240), and the series behind the levels at least 3,000. You get what your feed holds. On the verification run the daily series held 1,931 bars, the weekly 391 and the monthly 91, reaching back to May 2019. On that 1-minute chart the `1m` row carried the same 4,924 bars as the chart.

A bar of another timeframe reaches the chart only once it has closed, which the indicator takes as the next bar having opened. So the scanner and the levels never read a forming bar's high, low or close.

- **Previous-period levels** read the bar before the one a chart bar opened in: inside today's session they are yesterday's high and low.
- **The HTF row** compares the price with the EMA of closed bars of **Higher Timeframe**. It reads `BEARISH` while the EMA has no value yet.
- **The daily change** in the MARKET row is measured from the open of the session's daily bar.

When the daily bars have a gap

The previous-period levels, the HTF reading and the 1D, 1W and 1M rows are only as current as NinjaTrader's daily series. It is a separate series from the chart's bars, and on some connections it can lag or have holes while the minute bars are complete.

This was measured on the verification PC: the daily series had no bars for the trading dates 26 to 30 October 2026 and jumped from 23 October to the forming bar of 2 November, while the 1-minute chart had every session. Every chart bar from 27 October to 2 November therefore saw the high and low of 23 October as the previous day's, and sweeps were measured against those levels. The indicator raises no message when this happens.

To check it, compare the `1D High` and `1D Low` tags with the previous session on your chart. If they are old, use the chart's right-click menu > Reload All Historical Data on a connection that delivers daily history, or open a daily chart of the instrument to see where its bars stop. **Debug Mode** writes each series' asked and loaded bar count and its last open time to `slm_nt_debug.log`.

Time stamps

NinjaTrader stamps an intraday bar with its CLOSE time in the platform time zone, and the bar's open time is reconstructed as close time minus the bar period. A daily bar carries its trading date at midnight, a weekly bar the Friday of its week and a monthly bar the last day of its month, with no time of day, so the indicator takes such a bar's end from `Bars.GetSessionEndTime`. On the verification run the daily bars ended at 21:00 UTC, which is 16:00 Central. Each other-timeframe bar's open time is stored as the end of the bar before it, so a shortened session bar cannot deliver its predecessor early. Everything drawn is positioned by bar index. Slots past the last bar, where the zones, the level lines and their tags end, are placed at the average bar spacing of the view.

Volume

Volume does not take part in the signals, the zones, the sweeps or the levels at the defaults, which come from open, high, low and close alone. It is read by the dashboard's Volume row, which sets the newest bar's volume against the 20-bar average of closed bars, and by the optional **Use Live Confluence** filter. With `OnBarClose` the newest bar has only just started when the row refreshes, so it usually reads below average at that moment. The CSV dump writes the bar's volume as well.

Contract rollover

A continuous contract (merge policy `MergeBackAdjusted` or `MergeNonBackAdjusted` in the Data Series dialog) shows a price jump at the roll unless back-adjusted. Swings, zones and the ATR behind the stops are built from prices, so a non-back-adjusted roll can produce false zones, a sweep and an oversized ATR for a while, and the same holds for each scanner row on its own timeframe. With back adjustment there is no jump. State the merge policy when comparing values across machines.

08

Compatibility

ITEM	STATUS
NinjaTrader version	8.1.x Desktop (tested 8.1.9.0). Not 8.0, not NinjaTrader Web or mobile.
Product type	Indicator, price panel overlay, with secondary data series of the chart's own instrument (24 at the defaults).
Verified so far	MNQ (CME) 1-minute, a session template with the overnight hours, `OnBarClose`, historical load. See the chapter on how this edition was verified. Everything below that goes beyond this is what the code is written for, not yet a recorded run.
Instruments	Any instrument with bars: futures, micro futures, forex, stocks. Prices print at the decimals of the instrument's tick.
Bar types	Time-based (second, minute, day, week, month, year). Tick, volume, range, Renko and other non-time bars run; on them the alert's age limit uses the smallest spacing of the first 300 bars, which can withhold an alert on a slow chart.
Calculate modes	`OnBarClose` (default), `OnEachTick`, `OnPriceChange`. The forming bar is not fed to the calculation in any mode: a bar is committed once, when it has closed. With `OnEachTick` or `OnPriceChange`, the last-bar block reruns at most once a second and Tick Replay takes the per-tick path.
Session close	A signal on a session's final bar appears after the reopen, at the next bar's first tick. See the chapter on sessions, time and volume.
Data requirements	Bars of the chart and of the other timeframes, daily bars included. A gap in the daily series shows as old previous-period levels and old 1D, 1W and HTF readings. See the chapter on sessions, time and volume. Volume is read only by the dashboard's Volume row and the optional Live Confluence filter.
Load and speed	At the defaults every tick updates 24 more data series. **Scanner Rows** Off drops to the series the levels need (one, at the defaults).

ITEM	STATUS
Dashboard height	About 615 pixels at Text Size Normal and 100 % display scaling. On a short panel the last rows are cut off: use Text Size Small.
Multi-instance	Yes. Every instance paints its own surface, and every alert id carries the instrument, the period and an instance hash.
Market Analyzer	Not tested. The code draws nothing without a chart and the eight series are plots, but each instrument would load every secondary series: use **Scanner Rows** Off. Not advertised as a scanner.
Strategies	Not tested. The eight series can be referenced, and every input except the colours is a `[NinjaScriptProperty]`. NinjaTrader expects a calling strategy to add the same secondary series.
Rendering	SharpDX in `OnRender` with DPI scaling and full resource cleanup on render-target change and termination. No draw objects. Bars are placed linearly across the view, which assumes the default equidistant bar spacing.
Dark and light charts	The colours are inputs. The dashboard's own colours are fixed. There is no automatic switch between chart backgrounds.
Templates and workspaces	All settings serialize (colours through the standard brush serializer).
Minimized chart	Keeps calculating and alerting while its chart window is minimized (`IsSuspendedWhileInactive` off). A bar that closed more than two bar periods plus five seconds ago is not alerted, so a reconnect or a catch-up does not produce a burst of stale alerts.
Webhook	Not part of this edition. The indicator makes no network call of its own; e-mail goes through NinjaTrader's Mail share service.
Order handling	None. The indicator never touches orders, positions or accounts.

09

Troubleshooting

SYMPTOM	CAUSE AND FIX
Nothing on the chart	Check that the licence is active for your NinjaTrader login and that the indicator is in the Configured list (Ctrl+I). It paints in the price panel: with the indicator in its own panel it is not visible.
The last dashboard rows are cut off	The dashboard is taller than the panel: about 615 pixels at Text Size Normal and 100 % display scaling. Set **10 Dashboard > Text Size** to **Small** (about 510 pixels), or give the price panel more height.
No BoS or CHoCH lines, no tinted candles	By design. The structure-break events do not fire in any edition, so those inputs have no visible effect.
Chart Trend always reads NEUTRAL, HTF reads Divergent	The same cause. Neither affects a signal.
Every signal reads `***` and `Score: 3/5`	The score is fixed in the calculation and does not vary.
Minimum Confluence Score or Signal Cooldown does nothing	They do not filter a signal in any edition. Use **13 Signal Filters**.
Previous-period levels, the HTF row or the 1D row look old	NinjaTrader's daily series lags or has a gap, while the chart's bars are current. These readings then come from the last daily bars before the gap, with no message. Compare the `1D High` and `1D Low` tags with the previous session. Use the chart's right-click menu > Reload All Historical Data on a connection that delivers daily history. See the chapter on sessions, time and volume.
A scanner row stays at `--`	The state is flat, or that timeframe has fewer than 50 closed bars, which is common for `1M` on a young instrument. **Scanner Warm-up Marker** on shows a dot on a row that has delivered nothing. **Scanner Rows** Off leaves every row at `--`.
The Output line reads fewer than 24 of 24 secondary series	NinjaTrader did not add some series. The rows and levels behind them stay empty and the rest keeps working. Send the Log tab's lines (see below).

SYMPTOM	CAUSE AND FIX
No BUY or SELL	The state needs all its conditions on one closed bar, and a position must be flat first. **Enable Buy/Sell Signals** off stops them. A filter in **13 Signal Filters** can suppress them.
The stop or target line was crossed and the dashboard still shows the trade	The drawn levels are not the levels that close the signal state. See the table in the quick start.
The SL and TP lines disappeared	The signal state closed, which removes them, or a new signal replaced them. Only the newest set exists.
Grey boxes on the chart	Zones a close has gone through. They stay until the newest 501 boxes push them out. The zone counts limit active zones only.
A zone or a label is missing when zoomed far out	**Max Chart Objects** caps zones painted in view at two objects each, and **Label Pool Size** caps labels in view, signal and sweep history going first.
A price tag is not at its line's natural end	**Clamp Right-Edge Labels** on pins a tag inside the chart and left of the dashboard. The price on the tag is exact. Off, tags past the edge need room: chart Properties > Right side margin.
A level lies outside the panel	The price axis follows the bars, not the drawings. Drag the price axis to see it.
No signal on the last bar	The bar is still forming, or the session has ended. A signal on a session's final bar appears after the reopen, at the next bar's first tick.
OnEachTick shows no signal earlier	By design. The forming bar is not calculated in any Calculate mode.
Alerts do not fire	Alerts fire only in realtime, only for bars that close after the chart loaded, with **Enable Alerts** on. **Alert On** defaults to signals only, so a sweep needs `SignalsAndSweeps` or `All`, and **Highlight Liquidity Sweeps** off stops sweep events. A bar that closed more than two bar periods plus five seconds ago is not alerted, and that includes a session's final bar after a long break.
No row in the Alerts Log	**Enable Log Alerts** must be on.
Sound not played	**Enable Sound Alerts** must be on. **Sound File** is a name inside `<NinjaTrader 8 install>\sounds\` or a full path to a .wav file. Blank plays nothing.
No e-mail	**Enable E-mail Alerts** must be on with **E-mail To** filled in, and the Mail share service configured under Tools > Options > Share.
Looking for a webhook	A webhook is not part of this edition. **Webhook JSON Format** changes the alert text only; the indicator makes no network call of its own.
The chart feels slow or takes long to load	At the defaults the indicator adds 24 data series. **Scanner Rows** Off drops to those the levels need. A larger **Scanner Warm-up Bars** loads more.
Values differ from another platform on the same symbol	Bars differ by feed and session template; compare on identical bars (see the chapter on how this edition was verified).
The picture differs between two charts	The zones on screen are built from the bars before them. Compare charts with the same **Days to load** and Trading Hours template.
Text is tiny or huge	The sizes are fixed per class and follow the Windows display scaling. The dashboard has **Text Size**.
An error line names Smart Liquidity Matrix in the Log tab	Set **16 NinjaTrader > Debug Mode** on, reproduce it, and send `Documents\NinjaTrader 8\slm_nt_debug.log` with the chart interval, instrument and NinjaTrader version.
Everything is drawn twice	Two instances are configured (Ctrl+I, Configured list). Remove one.

A daily row, a VWAP or a previous-day level looks wrong or stops updating

NinjaTrader can keep using an older copy of its daily or minute history on your PC, and reloading the chart does not refresh it. Open **Control Center > Tools > Historical Data**, expand **Download**, pick the instrument, select **Day**, set the dates to cover the missing days and press **Download**. Do the same with **Minute** if intraday bars are missing, then reload the chart.

10

How this edition was verified

The verification has two stages, which measure different things and are reported separately. Stage 1 ran on forex bars, stage 2 on NinjaTrader futures bars, so the two figures are never added together.

Stage 1: the engine

The platform-free engine that ships here is the same code as the cTrader edition's engine apart from its namespace and using lines. That engine was replayed bar by bar there on EURUSD 1 hour, 23,103 bars from January 2023 to 17 September 2026, at the default inputs, and compared column by column at 1e-9 with reference values for the same bars.

GATE	COMPARED	MISMATCHES
everything that produces a drawn object, a signal or an alert	42 columns, 970,326 cells	0
the four level values from other timeframes (higher-timeframe average, previous-period high and low, daily open)	4 columns	0
the scanner's `1M` row	1 column	0

The other scanner rows are not at zero in stage 1. Fourteen more rows were compared there, and each disagrees on some bars, from 45 to 1,029 chart bars per row, because one disagreeing higher-timeframe bar spans many chart bars. Collapsed to one value per daily bar, the 1D row differs on 36 of 963 bars (3.7 %), in 31 separate runs of one to three bars, in both directions. The same state machine on the chart's own bars is at zero, so the cause is which other-timeframe bars the replay had, not the machine. Nine rows (`4m`, `6m`, `7m`, `8m`, `9m`, `20m`, `6H`, `8H`, `12H`) have no stage 1 comparison at all. They run the same code. The scanner feeds no signal, no drawn object and no alert.

Stage 2: NinjaTrader host vs the engine on identical NinjaTrader bars (2 November 2026)

With **Dump Bars + Outputs CSV** on, the indicator writes every committed bar (time, open, high, low, close, volume) together with the engine outputs as the host saw them: the ATR, RSI, EMA and volume series, the swing state, the structure, equal-level and sweep flags, the signal state and its drawn levels, the level values from other timeframes, the zone counts and all 24 scanner rows. It also writes the bars of each other timeframe. The same bars are replayed through a build of the identical engine outside NinjaTrader, with no host in between, and every column is compared.

RUN	BARS	SERIES	CELLS	MISMATCHES
MNQ 1 Minute, historical load, NinjaTrader 8.1.9.0, default inputs	4,923	70	344,610	0
the same run, the nine further scanner columns, compared by position	4,923	9	44,307	0

Bar times matched on every row, and the replay reads its prices from the same file. The bars run from 27 October 22:00 UTC to 2 November 2026 12:02 UTC, overnight hours included. On this run the engine produced 60 BUY and 39 SELL signals and 12 upside and 30 downside sweeps, on the same bars on both sides, and no structure-break event.

What stage 2 proves: on a historical load the host feeds the engine one closed bar per chart bar, in order, maps the other timeframes the way the replay does, and the engine's outputs inside NinjaTrader equal a replay outside it.

Live run, 2 Nov 2026: MNQ 1 Minute with Calculate set to OnEachTick, across the switch from history to live bars, with about 20 bars closing live: 4,191 bars, 70 series, 293,370 cells, 0 mismatches, 0 bar-identity mismatches. The 9 positional scanner columns matched too: 37,719 cells, 0 mismatches; 4 of the 9 changed value during the run.

Complete daily data, 2 Nov 2026: after the daily and minute history were downloaded, a further live run had no holes (5,371 bars, 27 October to 2 November) and the daily series delivered new bars for 26 to 30 October: 70 series, 375,970 cells, 0 mismatches, 0 bar-identity mismatches.

What it does not cover:

- Live alerts have no parity figure.
- The historical run above saw one previous-period high and low, those of 23 October, because the test PC's daily series had no bars for 26 to 30 October. The complete-data run below does not have that limit.
- One instrument, one interval, default inputs.
- How the surface paints is checked on a chart, not by this diff.
- None of this says anything about future performance.

11

Licensing

License model

The product is distributed as a compiled NinjaScript Add-On and licensed through NinjaTrader's user-based Vendor Licensing (NinjaTrader 8.1.5.0 or later). A licence is tied to the customer's NinjaTrader account login, not to a machine: the same login works on every installation of NinjaTrader Desktop the customer uses, and no Machine ID has to be sent to the vendor. Licences are created in the NinjaTrader Ecosystem Vendor Dashboard against the customer's NinjaTrader account email address, with a licence type and an expiry date.

The indicator carries the `VendorLicense(productId)` call in its constructor, with the Product ID of its add-on in the Vendor Dashboard. NinjaTrader verifies the licence whenever the indicator is used.

No free trial

The product is sold as a paid licence only; no free trial is offered. A licence is issued to the buyer's NinjaTrader account email after payment, normally within a few hours.

Activation and transfer

Nothing to activate: log in to NinjaTrader with the licensed account. A new PC needs no transfer. NinjaTrader allows two simultaneous market-data sessions per account, which is what prevents a licence from being shared.

Behavior without a valid licence

NinjaTrader shows its own licence notice (invalid or expired) and stops calling the indicator's calculation and plot methods: the indicator draws nothing and raises no alerts, while its settings still load. It never places orders (it has no order code) and exposes no proprietary data. A licensing-server outage is handled by NinjaTrader's verification (it retries against a secondary server) and must never crash the chart.

Licensing is a practical access-control layer, not a guarantee against copying.

12

Support

Questions go through the contact page at https://kestutisbalciunas.com/contact/ with the NinjaTrader version, the instrument and the chart interval. For an error, switch **16 NinjaTrader > Debug Mode** on, reproduce it once and attach `slm_nt_debug.log` from the `Documents/NinjaTrader 8` folder; switch Debug Mode off afterwards.

The product page, with the documentation and the change log, is https://kestutisbalciunas.com/ninjatrader/smart-liquidity-matrix/. Updates and refunds follow the terms on that page.

13

Risk disclosure

Trading involves substantial risk. Past, simulated, and backtested performance is not necessarily indicative of future results.

Futures, foreign currency and options trading contains substantial risk and is not for every investor. An investor could potentially lose all or more than the initial investment. Risk capital is money that can be lost without jeopardizing one's financial security or lifestyle. Only risk capital should be used for trading and only those with sufficient risk capital should consider trading.

Smart Liquidity Matrix is a technical analysis tool. It draws order blocks, fair value gaps, equal levels, liquidity sweeps, previous-period levels, signals and trade levels computed from closed bars; it does not place, modify or close orders and it gives no trade advice. The entry, stop and target lines are a risk picture at the chosen settings, and they are not the levels that close the signal state: the indicator does not report whether they are reached. The stars and the score on a signal are fixed values, not a measurement. Nothing in the software or its documentation is a promise of returns, win rates or funded-account success.

NinjaTrader is a registered trademark of NinjaTrader Group, LLC. No NinjaTrader company has any affiliation with the owner, developer, or provider of the products or services described herein, or any interest, ownership or otherwise, in any such product or service, or endorses, recommends or approves any such product or service.