

Cival Pattern Lab — Guide

Release 1.0.0-atas · archive SHA-256 b931972572d5ec73f4f5f1aa794837d7cc309f719fb8a4110dc7ee179d50c24e · 211,915 bytes

Contents: install guide · reading the indicator · trading playbook · settings reference · platform compatibility

Cival Pattern Lab — Install Guide

Version 1.0.0 · file Cival.PatternLab.dll · name in ATAS: **Cival Pattern Lab**

Cival Pattern Lab is a compiled ATAS plugin. This guide takes you from the downloaded zip to a verified, working indicator. The full list of settings is in 02-Settings-Reference.md.

1. Requirements

Item	What you need
Platform	ATAS Platform 8.0.15 or newer . The plugin runs on the .NET 10 runtime that ATAS ships; you install nothing else.
Data	Footprint data : bid and ask volume at each price, for the period you want to test. A bar without complete footprint rows can never trigger a pattern (see section 4).
Instrument	A symbol whose tick size ATAS knows. Stop distances are entered in ticks.
History	Enough chart history to produce a useful number of simulated trades. A few days of an active market is a starting point; more is better. The indicator processes at most the last Maximum bars back bars (default 10,000).
Windows	You can write to %APPDATA%\ATAS\Indicators (a normal user profile is enough).

2. Install with the included script

1. **Close ATAS completely** (check the Windows task manager that no ATAS process is left; a running ATAS locks the DLL).
2. Extract Cival-Pattern-Lab-v1.0.0.zip to a normal folder and open PowerShell in it.
3. Run:

```
powershell .\ATAS\Install.ps1
```

The script copies Cival.PatternLab.dll to %APPDATA%\ATAS\Indicators, keeps a backup of any DLL with the same name that was already there, and verifies the SHA-256 of the copied file against the value recorded in the package. If it reports a mismatch, stop and download the product again. 4. If Windows blocks the script because of the execution policy, run it for this session only:

```
powershell powershell -ExecutionPolicy Bypass -File .\ATAS\Install.ps1 5. Start ATAS.
```

3. Manual install

1. Close ATAS.
2. Check the file: `Get-FileHash .\ATAS\Cival.PatternLab.dll -Algorithm SHA256` and compare the value with the `Cival.PatternLab.dll` line in `SHA256SUMS.txt`.
3. If `%APPDATA%\ATAS\Indicators\Cival.PatternLab.dll` already exists, copy it somewhere safe first.
4. Copy `ATAS\Cival.PatternLab.dll` into `%APPDATA%\ATAS\Indicators`.
5. Start ATAS. ATAS only loads the new DLL when it starts; an ATAS session that was already open keeps the old one.

4. First load

1. Open (or create) a chart of the instrument you want to study. Use a timeframe you actually trade.
2. Open the chart's indicator catalog, search for **Cival Pattern Lab** and add it.
3. Wait for the historical calculation to finish. For a long history this can take a while: Pattern Lab reads every bar's footprint rows to detect later revisions by the data provider. If it feels slow, lower **Maximum bars back**.
4. The indicator panel appears at the top-left of the chart, about 50 pixels below the top edge.

What the first panel lines mean

What you see	Meaning
Cival Pattern Lab / Waiting for closed candles / historical calculation	ATAS has not delivered a closed bar to the indicator yet. Normal for a moment after loading.
Cival Pattern Lab – simulation	Calculation is running. This is always line 1 afterwards.
Instrument tick size unavailable	ATAS did not report a tick size. Nothing is simulated until it does.
Set holdout dates before judging results / Validation not configured / Test not configured	You have not entered the Validation / Test dates yet. Results are shown as one pool (All) only.
INVALID: test must follow validation	The Test date is not later than the Validation date (or the Test date is set while Validation is not). Pattern Lab stops marking new signals until you fix the dates.

You will probably see only three lines at first. With the default **Compact panel (title + two lines)** switched on, ATAS shows only the title and the next two lines: the rule and the simulation settings. The result lines (`All`, `Train`, `Validation`, `Test`, `Ambiguous`) are in the panel but hidden. Untick **Compact panel (title + two lines)** in the *Information panel* group to see all nine lines.

Markers. On historical bars where a rule matched you will see ▲ (below the bar) or ▼ (above the bar) with a label such as `LAB DeltaReversal`. If the chart is zoomed out so far that bars are narrower than **Minimum bar spacing for signal labels (pixels)** (default 6), the labels are hidden but the arrows stay.

Leave **Full analysis detail** switched on (group *00 Chart clarity*, default on). If you switch it off while **Compact display** is on, Pattern Lab hides its own markers and panel because it is a research tool, not a chart overlay.

5. Settings to touch first

1. **Pattern, Stop distance (ticks), Target / stop multiple, Maximum holding bars.**
 2. **Round-trip costs in price units:** the default is 0, which is not realistic. Enter your fees plus slippage per round trip in *price units* (not ticks, not currency).
 3. **Validation begins (chart time)** and **Untouched test begins (chart time):** see `04-Trading-Playbook.md` for how to use them without fooling yourself.
 4. **CSV export folder (blank disables)** if you want the trade list and bar features in files.
-

6. Verify it works — checklist

Tick each item.

1. `Get-FileHash` of `%APPDATA%\ATAS\Indicators\Cival.PatternLab.dll` matches the `SHA256SUMS.txt` line.
 2. After restarting ATAS, **Cival Pattern Lab** appears in the indicator catalog.
 3. Added to a chart, the panel title reads `Cival Pattern Lab – simulation` once the calculation finishes.
 4. After unticking **Compact panel (title + two lines)** the panel shows nine lines, including `All n=...` `PF=...` `DD=...`
 5. On an active instrument with several days of footprint history and the default rule, some ▲/▼ markers labelled `LAB DeltaReversal` appear on past bars. (If none appear, see the troubleshooting table.)
 6. Changing **Pattern** to another family changes the markers and the `All n=` count. Changing **Round-trip costs in price units** changes `PF` and the `E=` values but not `n`.
 7. Entering a **Validation begins** date turns the placeholder lines into `Train:` and `Validation:` lines. Entering a later **Untouched test begins** date adds `Test:`.
 8. With a CSV export folder set, `PatternLab-trades.csv`, `PatternLab-bars.csv`, `PatternLab-signals.csv` and `PatternLab-zones.csv` exist in that folder after the calculation finishes (the zones file only contains a header, because Pattern Lab draws no zones).
-

7. Updating

1. Close ATAS completely.
2. Extract the new zip and run its `.\ATAS\Install.ps1`. The previous same-named DLL is backed up automatically.
3. Start ATAS and re-add or reload the indicator on your charts.
4. Before updating, save a chart template and write down your key settings (rule, stop, target, costs, holdout dates). If a new version renames or removes a setting, ATAS may fall back to the default for

that one setting.

5. Keep the old zip. If you need to return to an older version, close ATAS and copy the older DLL back.

Changing any calculation setting recalculates the whole history, and so does a new version of the plugin. Read the documents in the new zip before comparing old and new results.

8. Uninstalling

1. Remove Cival Pattern Lab from your charts and templates.
2. Close ATAS.
3. Run `.\ATAS\Uninstall.ps1` from the extracted folder, **or** delete `%APPDATA%\ATAS\Indicators\Cival.PatternLab.dll` yourself.
4. Exported CSV files stay where you wrote them; delete them if you no longer need them.

9. Troubleshooting

Symptom	Likely cause	What to do
Cival Pattern Lab is not in the indicator list	ATAS was open during install or was not restarted; the DLL is in the wrong folder; the install script was blocked	Close ATAS fully, check the DLL is in <code>%APPDATA%\ATAS\Indicators</code> , start ATAS again, search for <code>Cival</code> . If the script was blocked use the <code>- ExecutionPolicy Bypass</code> form in section 2.
Panel shows only three lines and no results	Compact panel (title + two lines) is on (default)	Untick it in the <i>Information panel</i> group.
No panel at all	Show information panel is off; or other Cival panels fill the space (a gold message at the bottom-left says how many panels need space); or Full analysis detail is off together with Compact display	Tick Show information panel , hide unused panels, switch Full analysis detail back on.
Nothing drawn on the chart, <code>All n=0</code>	No footprint rows for the bars (data source or history), warm-up not finished, the rule did not match, or the history is too short	Check the data first: footprint rows must exist for the period (the ATAS footprint chart should show bid/ask numbers) and every bar in the lookback must be complete. Check warm-up (the first Prior lookback bars + 1 bars cannot signal), the loaded history, the symbol and timeframe, and any older saved template settings. A silent panel is not a measured zero; it means the rule did not match or the data was unavailable. Do not loosen the gates just to force signals: every change of Pattern , Volume / prior mean or Minimum absolute delta % is a new research trial.

Symptom	Likely cause	What to do
The newest signals are missing from the results, pending=1	A simulated trade needs later bars to reach its stop, target or time limit. Signals near the end of the loaded history are counted as <i>pending</i> , not as completed trades.	Wait for more bars. Do not count a pending trade as a result.
A period shows n=0	The Validation or Test date is outside the loaded history, or no completed trade fell inside that period (trades that cross a boundary are excluded)	Compare the dates with the first and last bar on the chart; adjust them or load more history.
The settings do not match this guide	A saved chart or template carries its own values and overrides the defaults	Open the indicator settings and compare each value with the defaults in <code>02-Settings-Reference.md</code> ; reset any you do not recognise.
CandleSequence never marks anything	The custom sequence contains a character other than U, D, N, or is shorter than 2 or longer than 20 characters. Nothing warns you.	Re-enter the sequence, for example <code>DDDUU</code> .
INVALID: test must follow validation	Test date is not after the Validation date	Set Validation first, then a later Test date; or clear the Test date. ATAS may display an unset date as <code>01/01/0001 00:00:00</code> ; that means <i>not set</i> .
Instrument tick size unavailable	ATAS gave no tick size for this instrument	Check the instrument settings in ATAS; reload the chart.
DLL is locked during install or update	An ATAS process is still running	Close ATAS, end leftover ATAS processes in the task manager, run the installer again.
Settings reset after an update	A setting was renamed or changed in the new version	Re-enter your key settings from your notes; load your saved template.
Results changed after reloading the chart	A different history length, a provider revising already-closed bars, or a changed setting. The whole history is recalculated whenever a setting changes or a closed bar is revised.	Note the history range and settings with every result you keep. Export the CSV files when you want a fixed snapshot.
CSV files do not appear	Folder blank or not writable; calculation not	Set a writable folder. If the panel shows a <code>Research export:</code> or <code>CSV export failed:</code> line (it appears as line 2), read the message. Files

Symptom	Likely cause	What to do
	finished	are refreshed when history finishes calculating and when new bars close.
Waiting for closed candles / historical calculation stays on screen	The calculation is still running, or the chart has no closed bars	Wait; reduce Maximum bars back ; confirm the chart has history.

Still stuck? Email support@civalsystems.com with the product version (1.0.0), your ATAS version, a screenshot with the panel expanded, and what you expected to see.

Cival Pattern Lab — Reading the Indicator

Pattern Lab is a **research instrument**. On every *closed* bar it asks one question: did the rule you chose match? If it did, it marks the bar, simulates the trade that *would* have been taken at the next bar's open, follows that simulated trade bar by bar to its stop, target or time limit, and adds the result to a scoreboard that is split into train, validation and test periods. The point is to find out whether a rule survives costs and unseen data before you risk money on it.

Everything below is taken from the shipped plugin (v1.0.0). Defaults are the ones the indicator starts with.

1. What is drawn on the chart

Element	Where	Meaning
▲ in the Bull color (default green #3CB371)	just below the signal bar's low	The rule matched on a closed bar with a long hypothesis.
▼ in the Bear color (default red #FF6347)	just above the signal bar's high	The rule matched with a short hypothesis.
Label LAB <Pattern>, for example LAB DeltaReversal	to the right of the arrow	Which rule matched. Hidden when bars are narrower than Minimum bar spacing for signal labels (pixels) (default 6) or when it would overlap another label (Avoid overlapping labels , default on). Labels are hidden, never moved. The arrow always stays.
Information panel	top-left of the chart, 50 px below the top edge	Rule, simulation settings and the results scoreboard (section 8).

There are no zones, lines or volume bubbles. Settings for those (**Maximum visible zones**, **Show touched levels**, **Show bubble volume numbers**, **Bubble number color**, **Compact volume numbers**) have no effect in this product.

An arrow is drawn for **every** match, including a match that happens while a simulated trade is already open. Only one simulated position is allowed at a time, so the number of arrows is usually larger than the number of trades in the scoreboard (in the demo: 111 arrows, 78 completed trades and 1 pending, section 11).

Nothing is drawn on the forming bar. A bar is judged when it has closed; the arrow appears when the next bar starts.

2. Gates every pattern must pass

Let **b** be the bar that just closed, **p** the bar before it, and *prior* the **Prior lookback bars** (default 20) bars before b.

Gate	Rule
Enough history	At least $\max(3, \text{Prior lookback bars} + 1)$ closed bars have been processed.
Footprint present and complete	b and every bar in <i>prior</i> have footprint rows, and the bid+ask total of those rows is within 1% of the candle's volume. One incomplete bar silences the rule until it leaves the lookback window.
Non-degenerate bar	High is above Low and the volume is above zero.
No gaps	The last $\max(\text{lookback} + 1, 5)$ bars (or the sequence length for <code>CandleSequence</code>) have consecutive bar numbers. A missing bar from the data provider silences the rule for a while.
Volume	Volume of b is at least the mean volume of <i>prior</i> times Volume / prior mean (default 1; 0 switches this gate off).
Delta size	$ \text{delta} / \text{volume} \times 100$ of b is at least Minimum absolute delta % (default 20; 0 switches this gate off).

Volume here is bid plus ask volume from the footprint. *Delta* is ask volume minus bid volume (aggressive buying minus aggressive selling). *Close location* is $(\text{Close} - \text{Low}) / (\text{High} - \text{Low})$: 0 means the bar closed on its low, 1 on its high. The 0.65 close-location threshold below is fixed in the plugin; it is not a setting.

3. The six patterns

After the gates, exactly one of these conditions decides the side: +1 is a long hypothesis, -1 a short hypothesis.

Pattern	Long (+1)	Short (-1)
Engulfing	b closed up, p closed down, b opened at or below p's close and closed at or above p's open (b's body covers p's body).	Mirror image: b down, p up, b opened at or above p's close and closed at or below p's open.
DeltaReversal (default)	p's delta was negative, b's delta is positive, b closed above p's close, and b's close location is at least 0.65 (top 35% of its range).	p's delta was positive, b's delta is negative, b closed below p's close, and b's close location is at most 0.35.

Pattern	Long (+1)	Short (-1)
Absorption	b's delta is negative (net aggressive selling) but b closed in the top 35% of its range (close location ≥ 0.65).	b's delta is positive but b closed in the bottom 35% (close location ≤ 0.35).
BreakoutRetest	Let H be the highest high of the <i>prior</i> bars excluding p . p closed above H, b traded back down to H or lower ($Low \leq H$), b closed above H, and b's delta is positive.	Let L be the lowest low of the same bars. p closed below L, b traded back up to L or higher, b closed below L, and b's delta is negative.
ReversalFiveBar	Closes of the last five bars $c_0 \dots c_4$ (c_4 is b): $c_0 > c_1 > c_2 < c_3 < c_4$ and b's delta is positive.	$c_0 < c_1 < c_2 > c_3 > c_4$ and b's delta is negative.
CandleSequence	The last N bars match the Custom sequence (U up / D down / N doji) string, U = close above open, D = close below open, N = close equal to open. N is the length of the string (2 to 20 characters). Default DDDUU. The side is the Custom sequence side (+1 / -1) setting (default +1; 0 or positive means +1).	Same, with side -1.

Notes that matter when you interpret a match:

- **The delta gate checks size, not direction**, for Engulfing and CandleSequence. A bullish engulfing bar with a negative delta of at least 20% of its volume still passes.
- **Absorption is a name for a pattern, not a measurement of absorption**. It means "net aggressive flow one way, close location the other way" on one bar. The Lab does not look at what happened at individual prices. Like every family here it is an evidence-based hypothesis about one bar; it does not reveal anyone's hidden positions or orders.
- A **CandleSequence** with a character other than U, D, N, or a length outside 2–20, never matches and shows no warning. The panel does not print the sequence either; it only prints the pattern name.
- **Require directional POC migration** (default off) adds one more gate to any pattern: the POC of b (the price row with the most volume; ties go to the lower price) must be higher than p's POC for a long, lower for a short.
- The default settings (DeltaReversal, 20 bars, volume $\geq 1 \times$, $|\text{delta}| \geq 20\%$) and the default DDDUU sequence are *examples*, not tuned recommendations.

4. From signal to simulated trade

1. **Signal bar**. The bar that matches is marked. It is closed; nothing about it will change.
2. **Entry**. The simulated trade enters at the **open of the next bar**. It is never filled inside the signal bar: a signal on bar 10 cannot enter at bar 10's own open or at any earlier price, so the first possible entry is the open of bar 11.
3. **Levels**. Stop distance = **Stop distance (ticks)** $\times$ tick size (default 20 ticks). Target distance = stop distance $\times$ **Target / stop multiple** (default 2). For a long, the stop is below the entry and the target above; for a short the reverse.
4. **Checked on every bar from the entry bar on** (the entry bar counts as bar 1):

Condition on the bar	Exit price	Reason shown in the CSV
The bar opened at or beyond the stop (a gap through the stop; cannot happen on the entry bar)	the bar's open (can be worse than the stop)	Stop gap
The bar opened at or beyond the target	the target price (the extra gap profit is not credited)	Target gap
The bar's range reached the stop	the stop price	Stop
The bar's range reached the target	the target price	Target
The count of bars since entry reached Maximum holding bars (default 20) and neither level was reached	the bar's close	Time

5. **Both levels inside one bar.** Intrabar order is unknown, so the Lab always assumes the **stop was hit first** and counts the trade as `Ambiguous`.
6. **Net result** = side × (exit – entry) – **Round-trip costs in price units** (default 0). Results are in **price units** (for a BTC perpetual quoted in dollars: dollars per 1 BTC), not in ticks and not in your account currency.
7. **One position at a time.** While a simulated trade is open, further matches are marked with an arrow but do not become trades.
8. **Censored.** If the indicator skips a bar while a trade is open (a hole in the data), that trade is dropped and counted as `censored`; it is not guessed.
9. **Pending.** A trade that has not finished at the end of the loaded history is counted as `pending` and is **not** added to the statistics.

MAE / MFE. Maximum adverse and favorable excursion are measured from the entry using the **full high-low range of each bar, including the bar where the trade exits**. Because the real order of highs and lows inside a bar is unknown, they are **bounds**, not exact excursions. They appear only in the CSV (`mae_bound`, `mfe_bound`).

5. Holdout periods: train, validation, test

Set **Validation begins (chart time)** and **Untouched test begins (chart time)** to split the history by date. A trade belongs to a period according to when it was known to have finished:

Period	A trade is counted when
Train	its exit time is before the Validation date
Validation	its entry time is on or after the Validation date and its exit time is before the Test date (if a Test date is set)
Test	its entry time is on or after the Test date
boundary-excluded (CSV only)	it entered before a boundary and exited at or after it. It is in no period, so no result leaks across a boundary.

The *exit time* the Lab uses is the open time of the bar **after** the exit bar, the earliest moment the result could be known. A trade that exits on the last bar before a boundary is therefore treated as crossing it.

ATAS may display an unset date as `01/01/0001 00:00:00`; that means *not set*. The dates must be in chronological order. If the Test date is set but not later than the Validation date (or Validation is not set), the panel prints `INVALID: test must follow validation` and **no new signals are marked or simulated** until the dates are fixed.

Until the dates are set, the panel prints `Set holdout dates before judging results, Validation not configured and Test not configured`. If you set only Validation, everything after it is *Validation*. Tip: leave the Test date empty while you explore, so that the `Test:` line cannot tempt you (see `04-Trading-Playbook.md`).

6. The panel, line by line

The panel has nine lines. By default only the first three are visible; untick **Compact panel (title + two lines)** (group *Information panel*) to see all of them. When an export error occurs, a line with the error is inserted as line 2.

#	Text (example from the demo)	What it means
1	<code>Cival Pattern Lab – simulation</code>	Title (gold).
2	<code>DeltaReversal; V≥1x Δ≥20%</code>	Active pattern, the Volume / prior mean gate (V) and the Minimum absolute delta % gate (Δ).
3	<code>Next bar open; stop 100t / 2R</code>	Entry rule, stop in ticks (t) and target multiple (R).
4	<code>All n=78 PF=0.61 DD=2931</code>	All completed trades: count, profit factor, maximum drawdown in price units. PF=n/a when there are no losing trades (including when there are no trades at all).
5	<code>Train: n=24 WR=41.7% E=-12.6667</code>	Count, win rate, expectancy (average net per trade, price units) for the train period. Replaced by <code>Set holdout dates before judging results</code> if no Validation date.
6	<code>Validation: n=27 WR=25.9% E=-46.2222</code>	The same for validation. Replaced by <code>Validation not configured</code> .
7	<code>Test: n=27 WR=29.6% E=-35.1111</code>	The same for the test period. Replaced by <code>Test not configured</code> .
8	<code>Ambiguous=0; censored=0; pending=1</code>	Trades where stop and target were both inside one bar; trades dropped for missing bars; simulated trades still open.
9	<code>MAE/MFE: full-bar bounds; stop-first</code>	Reminder of the two modelling assumptions. Reads <code>INVALID: test must follow validation</code> when the dates are wrong.

How the numbers are computed

Number	Definition
n	Completed trades in the group.
WR	Trades with net result above zero ÷ n × 100.
E	Mean net result per trade after costs, in price units.
PF	Sum of positive net results ÷ sum of absolute negative net results. Below 1 means the group lost money after costs.
DD	Largest fall of cumulative net result from a previous peak, trades ordered by exit bar, price units. Shown for All only.

A group with only a handful of trades produces numbers that are anecdotes, not evidence. The demo's 24 to 27 trades per period are already on the small side.

7. CSV exports

Set **CSV export folder (blank disables)** to write four files. Use a different folder for each chart or timeframe, because a second instance writing to the same folder overwrites the files.

File	Written	Columns
PatternLab-trades.csv	when history finishes calculating, then whenever trades or bars are added live	signal_bar, entry_bar, exit_bar, entry_time, exit_time, partition, side, entry, exit, net_price_units, mae_bound, mfe_bound, ambiguous, reason
PatternLab-bars.csv	same	bar, time, open, high, low, close, volume, delta, poc, complete
PatternLab-signals.csv	when history finishes calculating and whenever a new signal appears live	time, available_time, bar, price, side, kind, label, volume
PatternLab-zones.csv	together with the signals file	available_bar, low, high, poc, side, label, state, touched_bar, invalidated_bar (header only: Pattern Lab creates no zones)

Column notes:

- partition is train, validation, test or boundary-excluded, using the rules in section 5. With no dates set every trade is train.
- side is 1 (long) or -1 (short). ambiguous is True Or False. reason is Stop, Target, Stop gap, Target gap Or Time.
- net_price_units is after costs. mae_bound and mfe_bound are full-bar bounds (section 4).
- entry_time is the open time of the entry bar; exit_time is the time the result became known (open of the bar after the exit bar).
- bars.csv: volume and delta are footprint totals (bid + ask, and ask – bid); poc is the bar's highest-volume price; complete is True only when the bar's footprint rows exist and match the candle volume

within 1%. Bars without a footprint show volume 0, delta 0, `complete` False. The file keeps at most **Maximum bars back** bars.

- `signals.csv`: `time` is the signal bar's open time; `available_time` is the open time of the next bar, when the signal bar's close is known; `volume` is the marker size and is always 1 for Pattern Lab. Signals older than the last 1,000 retained closed bars are not written to this file.
- All times are in the chart's time base, written in ISO 8601. Numbers use a dot as decimal separator.
- Files are snapshots. Changing a setting recalculates the history and rewrites them, so copy files you want to keep. After you set a folder, files may appear only when the next bar closes or when history is next recalculated; if you do not want to wait, change any calculation setting and change it back.

8. Alerts

Alerts on closed signal / level touch (default off) plays the **Alert sound** (default `alert1`) when a bar closes with a match, with the text `Cival LAB <Pattern>: <price>`. In Pattern Lab an alert means "*the rule matched on the bar that just closed*". It is an observation for research or replay, not an instruction. Recalculating the history does not fire alerts for old signals.

9. Display and session settings

Setting (group)	Effect in Pattern Lab
Show information panel (<i>Appearance</i>), default on	Shows or hides the panel. Hiding the panel or the labels does not remove the underlying events or the CSV rows.
Panel top offset (pixels), Panel width (pixels)	Position/width of the panel. The offset only applies when Automatic panel placement is off; width is limited to 220–700 px and to the chart width.
Automatic panel placement, Panel font size (10–22, default 13), Compact panel (<i>Information panel</i>)	Stacks the panels of several Cival indicators in a column; font size; the three-line limit described above.
Show labels, Minimum bar spacing for signal labels, Avoid overlapping labels, Bull color, Bear color (<i>Appearance</i>)	Marker labels and colors.
Maximum bars back (<i>Performance</i>), default 10,000	Only the latest N bars are processed. The indicator keeps at most the 500 most recent signals in memory and 10,000 trades.
Compact display (full detail must be off), Full analysis detail (<i>00 Chart clarity</i>), default on/on	In the default state Pattern Lab draws normally. If Full analysis detail is off while Compact display is on, Pattern Lab hides its own markers and panel.
No effect here: Session start, Session timezone, Initial balance minutes, Show readiness/status panel, Maximum visible zones, Show	Pattern Lab does not use sessions, zones or bubbles.

Setting (group)	Effect in Pattern Lab
touched levels, Show bubble volume numbers, Bubble number color, Compact volume numbers	

Setting changes that alter the calculation (pattern, filters, stop, target, holding bars, costs, dates, **Maximum bars back**) recalculate the whole history. Each change is another trial of the rule, which matters for honest testing (see the playbook).

10. History revisions and what "closed bar" means here

Signals are decided only from bars that have closed; nothing is looked up from the future. However, Pattern Lab recalculates its entire history whenever you change a calculation setting and whenever the data provider revises an already-processed closed bar. Marks and statistics can therefore change after a recalculation. That is a property of the data and of your settings, not a live "repaint". Source corrections, a different history prefix (loading more or fewer bars), changes to any calculation setting and a change of timeframe can all legitimately recalculate the results, so a screenshot taken today may not match a chart opened next week. Panning, zooming or moving the cursor changes only what is drawn, never the underlying closed-bar results.

11. Worked example — real numbers from the demo

Source. The store demo: Hyperliquid BTC perpetual, 5-minute bars (tick size 1.0) built from trades recorded by the Cival order-flow recorder, 17–21 October 2026, 1,380 closed bars (about 4.8 days). It was produced by the shipped plugin. Settings: DeltaReversal with the default gates (volume $\geq 1 \times$ the prior-20-bar mean, $|\text{delta}| \geq 20\%$), **Stop distance (ticks)** 100, **Target / stop multiple** 2, **Maximum holding bars** 24, **Round-trip costs in price units** 24 (about 0.03% of price), Validation begins 19 Oct 00:00, Test begins 20 Oct 12:00. Raw output and settings file: `demo\out\pattern-lab.json`, `demo\settings\pattern-lab.json`.

A trade that worked. The 5-minute bar opening 15:45 UTC on 17 October (open 81,434, high 81,434, low 81,372, close 81,374) traded 137.0149 BTC against a prior-20-bar mean of about 22.0 BTC (6.2 $\times$). Its delta was -106.0531 , or -77.4% of its volume. The previous bar had delta $+7.3557$ and closed at 81,435. So: previous delta positive, this delta negative, close below the previous close, and the close at the very bottom of the range (close location 0.03, far under the 0.35 limit). The Lab marked ▼ at 81,434 (the bar's high). The simulated short entered at the next open, **81,374**; stop 100 units higher at 81,474; target 200 units lower at 81,174. It exited at the target on the 16:55 bar: gross $+200$, minus the 24 cost = **+176**.

A trade that did not. The 18:40 bar (high 81,529, low 81,501, close 81,507) traded 103.5589 BTC (about 2.2 $\times$ its prior mean) with delta -77.3548 (-74.7%) after a previous bar with delta $+5.7342$. The Lab marked ▼, entered at the next open **81,507**, and the 19:40 bar's high reached the stop at 81,607: gross -100 , minus 24 = **-124**.

The scoreboard (panel values; the profit-factor column for the partitions is computed from the trade list in the same raw output using the panel's own formula):

Group	Trades	Wins	Win rate	Expectancy per trade	Profit factor
All	78	25	32.1%	-32.05	0.61
Train	24	10	41.7%	-12.67	0.82
Validation	27	7	25.9%	-46.22	0.50
Test	27	8	29.6%	-35.11	0.60

Other numbers from the same run: maximum drawdown 2,931 price units; net result over all 78 trades -2,500 price units; 52 stops, 22 targets and 4 time exits; 0 ambiguous bars; 0 censored; 1 pending; 111 arrows in total.

The arithmetic of why. With a 100-unit stop and a 24-unit cost, a stop-out loses 124 and a target wins 176. Ignoring time exits, that needs about **41.3% winners** just to break even ($124 \div 300$). The rule won 32.1% overall. Even the train period, which came closest at 41.7%, still had negative expectancy.

Verdict. This rule lost money after costs in **all three periods**, including the test period it never saw. It is a rule to **reject**, and finding that out cost nothing. This is one rule on one market over a short sample, so the honest reading is "no evidence of an edge here", not "this pattern can never work". It also shows the correct response: do not go back and tune the test period away (see 04-Trading-Playbook.md).

12. What Pattern Lab can and cannot know

It can	It cannot
Apply a precisely defined rule to every closed bar of your loaded history	Know that a rule will work on future data
Charge a fixed cost per round trip, assume entry at the next open and stop-first inside a bar	Model real slippage, queue position, partial fills, latency, funding or margin
Keep results out of a period when a trade crosses a boundary	Stop you from peeking at the Test line and tuning on it
Report win rate, expectancy, profit factor, drawdown and ambiguity counts	Give a confidence interval, or tell you whether a sample is large enough
Export trades and bar features for your own analysis	Know where other participants' orders, stops or positions are
Show results in price units	Convert to your contract size or currency; do that yourself

Unavailable data is not a measured zero: with no footprint rows, too little history or a rule that never matched, the panel simply shows no signals. That says nothing about whether the rule works.

Every change to a setting after you have seen a result is another trial. Many parameter trials create **selection bias**: with enough trials, some rule will look good by chance. A zero-cost default is not realistic performance, and open simulations are never silently completed. Pattern Lab has no way to count your trials for you; keep your own log.

Cival Pattern Lab — Trading Playbook (a research workflow)

Most indicators ask you to trust what they draw. Pattern Lab asks the opposite: **do not trust any rule, including the ones in this indicator, until it has survived costs and data it has not seen**. This playbook is therefore a workflow for deciding whether a rule deserves your attention: **hypothesis** → **holdout** → **decision**. It does not tell you to take any trade, and none of the arrows on the chart is a recommendation.

Context first

- Pattern Lab can tell you what a precisely defined rule *would have done* on the bars you loaded, with the entry, stop, target and cost assumptions in `03-Reading-the-Indicator.md`. It cannot tell you what the rule will do next week.
 - A rule's results depend on the market, the timeframe, the time of day and the volatility regime. Test the market and timeframe you actually intend to trade.
 - Costs decide many rules. Fees plus slippage can be larger than the average gain a pattern offers. The default cost is 0, which is not realistic.
 - Every time you change a setting and look at a result again you have run another trial. The more trials, the more likely one of them looks good by luck (selection bias).
-

Playbook A — Test one hypothesis from start to finish

1. Write the hypothesis before you look. One sentence: market, timeframe, rule, direction, what you expect, and why. Example (this is the rule the demo tested): *"On 5-minute BTC bars, when delta flips against the previous bar's delta and the bar closes at the extreme of its range, the next bars reach twice the stop distance before the stop, after costs."*

2. Fix the costs. Set **Round-trip costs in price units** to your realistic round-trip fees plus slippage, converted to price units: money cost per unit ÷ money value of one price unit. If you are unsure, use a pessimistic number.

3. Fix the rule. Choose **Pattern**, the two gates (**Volume / prior mean**, **Minimum absolute delta %**), **Stop distance (ticks)**, **Target / stop multiple** and **Maximum holding bars** from your hypothesis. Do not tune them yet.

4. Choose the dates by the calendar, not by results. Look at the first and last bar on the chart. Keep the earliest part as *train*, the middle part as *validation*, the latest part as the *untouched test*. The demo used roughly two days, one and a half days and one and a quarter days. Enter **Validation begins (chart time)** but **leave Untouched test begins empty for now**: with no Test date the panel prints Test not configured, so you cannot be tempted by it.

5. Explore on train only. Untick **Compact panel (title + two lines)** so you can see the lines. Read the **Train:** line and the **A11** line. Check that the number of trades is not tiny and that **Ambiguous** is small. If you

change the rule, write down that you changed it (section "Journaling" below).

6. Check validation. Read the `Validation:` line. A rule that is positive after costs in train and negative in validation was probably fitted to train.

7. Freeze, then open the test once. When you will make no further changes, enter **Untouched test begins (chart time)** (it must be later than the Validation date, otherwise the panel prints `INVALID: test must follow validation` and stops marking signals). Read the `Test:` line once. If you then change the rule and look again, the test is no longer untouched; the next clean test is *later data*, which you have to wait for or load.

8. Decide with criteria you wrote down in step 1.

Outcome on the panel	Decision
Expectancy E is zero or negative after costs, or profit factor below 1, in validation or test	Reject. Do not keep tuning.
Positive in validation and test, but the number of trades is only a handful	Not enough evidence. Extend the history or wait for newer data; do not trade it.
Positive after realistic costs in both validation and test, similar to train, adequate trade count, few ambiguous trades, and not carried by a couple of lucky trades (check the CSV)	Candidate worth forward testing on replay or a simulated account. It is still only a hypothesis.

Cancel this playbook, or stand aside, when: the footprint data is incomplete for much of the history (check the `complete` column in `PatternLab-bars.csv`); you chose the dates after seeing the results; you changed the rule after seeing the Test line; the sample is too small to read; or your costs were set to zero.

A real rejection: the demo rule

The store demo ran exactly this workflow on 4.8 days of Hyperliquid BTC perpetual 5-minute bars (17–21 October 2026), with the delta-reversal rule, a 100-unit stop, a 2× target, a 24-bar limit and a 24-unit round-trip cost. Result: 78 simulated trades, profit factor 0.61, and a negative expectancy in train (–12.7), validation (–46.2) **and** the untouched test (–35.1). It needed about 41% winners to break even and won 32%. Decision according to the table above: **reject**. Nothing was lost finding this out. The correct next step is a different hypothesis, not a different stop for the same sample. The details are in section 11 of `03-Reading-the-Indicator.md`.

Playbook B — Does the result survive costs?

Setup. Keep a rule and dates fixed; only change **Round-trip costs in price units**.

Procedure. 1. Run with your realistic cost, then with a pessimistic one (a higher number), and note `A11` PF and the `Validation:` and `Test:` expectancy each time. 2. Work out the break-even win rate. With stop S, target multiple R and cost C (all in price units), a loss is $S + C$ and a win is $R \cdot S - C$, so the break-even win

rate (ignoring time exits) is $(S + C) \div (S \times (1 + R))$. In the demo $S = 100$, $R = 2$, $C = 24$ gives $124 \div 300 = 41.3\%$. 3. Compare to the win rate on the panel.

What the numbers mean. If the rule only has positive expectancy at zero or optimistic costs, it has nothing to trade. Costs that are a large part of the stop distance (24% in the demo) make rules much harder to pass.

Cancel when: your cost estimate is a guess you have not checked against real fills; the rule changes between runs; the win rate and the break-even rate are within the noise of a small sample.

Playbook C — A custom U/D/N candle-sequence hypothesis

Setup. Set **Pattern** to `CandleSequence`, type a sequence (U up, D down, N doji, 2 to 20 characters) and a side (**Custom sequence side (+1 / -1)**). The default `DDUUU` is an example, not a recommendation. The volume and delta gates still apply to the last bar of the sequence.

Discipline. Write down *why* the sequence should matter before trying it. Each additional sequence you try is another trial; with many sequences and little data, one will look good by luck. Test a handful at most, with holdouts, and expect most to fail.

Reference prices. The same as in Playbook A: entry at the next bar's open, the stop and target distances from your settings.

Cancel when: the sequence is a result of scanning many options on the same data; the number of trades is small; the sequence only works in one period.

Playbook D — Export, look closer, then replay

Setup. Set **CSV export folder (blank disables)** to a folder only this chart uses. After the calculation finishes, open `PatternLab-trades.csv` in a spreadsheet.

Questions to ask the file. - Filter on `partition`: do `validation` and `test` look like `train`? - How much of the net result comes from the best three trades? Remove them and re-add up `net_price_units`. - Split by `side`, by hour of the day and by `reason` (`Stop`, `Target`, `Stop gap`, `Target gap`, `Time`). A rule that only works for one side or one hour is a narrower rule than you thought. - Count `ambiguous` trades. They were scored stop-first, which is pessimistic for winners. - In `PatternLab-bars.csv`, how many bars have `complete` equal to `False`? Pattern Lab silences itself on those bars.

Move to replay. If a rule has survived Playbook A and this second look, run it in ATAS market replay or on a simulated account, following exactly the assumptions the Lab used: entry at the open of the bar after the arrow, the tested stop, the tested target, the tested time limit. If you enter differently (for example with a limit order), you are trading a different rule that the Lab has not tested.

Cancel when: the replay shows costs or fills very different from your assumption; the rule behaves differently on days outside the original sample; you notice yourself bending the rule to make a trade.

If a rule survives: what a plan could look like

This is a description of how a discretionary trader might *define* a plan, not an instruction to trade.

- **Entry.** Decided at the close of the arrow bar, executed at the next bar's open, as tested.
- **Stop.** The tested distance from the entry; it exists before you enter.
- **Management.** The tested target multiple, or the time limit; nothing improvised.
- **Cancel.** Footprint data incomplete on the arrow bar; an event or session change that makes costs larger than assumed; you have changed any parameter since the test.

Combining with other context

Use Pattern Lab to test a rule, then ask whether the rule needs a context filter to work (for example only in certain sessions, or only near levels from another tool). Any such filter is a new hypothesis and needs its own holdout. Do not add a filter after you have seen the Test line.

Journaling and review

For every rule you test, keep a log with: date; market and timeframe; the hypothesis; every setting, including costs and holdout dates; the All, Train, Validation and Test lines and the Ambiguous / censored / pending line (take a screenshot with the panel expanded); how many trials you ran before this one; the decision and the reason. Review the log monthly. If you cannot say how many variations you tried, assume the number was larger than you think.

Mistakes to avoid

- Reading the All line and ignoring Validation and Test.
- Leaving **Compact panel (title + two lines)** on and wondering where the results are.
- Leaving costs at 0.
- Looking at the Test line more than once and then adjusting.
- Treating the arrows as trade signals. An arrow means the rule matched, nothing more.
- Forgetting that the Lab takes one simulated trade at a time, while your own trading may differ.
- Loosening the gates (**Volume / prior mean, Minimum absolute delta %**) merely to get more signals or trades. That is a new trial, and it adds to the selection bias.
- Trusting a sample of a few dozen trades, or a single regime.
- Comparing results across settings changes without recording them.
- Taking results in price units as money: multiply by your contract size and compare against your actual costs.

Risk

Trading futures and crypto involves substantial risk of loss and is not suitable for everyone. Indicators describe what happened, not what will happen. Pattern Lab's results are a description of a past sample under simplified assumptions (entry at the next open, a fixed cost, stop-first inside a bar); real fills, slippage, funding and market conditions will differ. Test any rule on market replay or a simulated account first. Size positions so that a stop-out, or a run of them, is affordable. Past observations do not predict future results. Nothing in this product is financial advice.

Cival Pattern Lab — Settings Reference

Version 1.0.0. Generated from the compiled ATAS plugin: every value below is the default a fresh instance loads with.

Cival Pattern Lab

00 Chart clarity

Setting	Default	Allowed
Compact display (full detail must be off)	on	on / off
Full analysis detail	on	on / off

Alerts

Setting	Default	Allowed
Alerts on closed signal / level touch	off	on / off
Alert sound	alert1	String

Appearance

Setting	Default	Allowed
Show information panel	on	on / off
Panel top offset (pixels)	50	0 – 1500
Panel width (pixels)	380	180 – 600
Compact volume numbers	on	on / off
Avoid overlapping labels	on	on / off
Minimum bar spacing for signal labels (pixels)	6	0 – 30
Maximum visible zones	12	1 – 100
Show touched levels	off	on / off
Show labels	on	on / off
Show bubble volume numbers	on	on / off

Setting	Default	Allowed
Bubble number color	#FFFFFF	Color
Bull color	#3CB371	Color
Bear color	#FF6347	Color

Export

Setting	Default	Allowed
CSV export folder (blank disables)	(blank)	String

Holdout

Setting	Default	Allowed
Validation begins (chart time)	(not set)	DateTime
Untouched test begins (chart time)	(not set)	DateTime

Information panel

Setting	Default	Allowed
Automatic panel placement	on	on / off
Panel font size	13	10 – 22
Compact panel (title + two lines)	on	on / off
Show readiness/status panel	on	on / off

Performance

Setting	Default	Allowed
Maximum bars back	10000	100 – 100000

Rules

Setting	Default	Allowed
Custom sequence (U up / D down / N doji)	DDDUU	String
Custom sequence side (+1 / -1)	1	Int32
Pattern	DeltaReversal	Engulfing, DeltaReversal, Absorption, BreakoutRetest, ReversalFiveBar, CandleSequence
Prior lookback bars	20	3 – 500
Volume / prior mean	1	0 – 20
Minimum absolute delta %	20	0 – 100
Require directional POC migration	off	on / off

Session

Setting	Default	Allowed
Session start (minutes after midnight)	0	0 – 1439
Session timezone (blank = chart; otherwise input UTC)	(blank)	String
Initial balance minutes	60	1 – 240

Simulation

Setting	Default	Allowed
Stop distance (ticks)	20	1 – 10000
Target / stop multiple	2	0.1 – 100
Maximum holding bars	20	1 – 1000
Round-trip costs in price units	0	0 – 100000

Cival Pattern Lab — Platform Compatibility

The ATAS plugin is the reference implementation. Each port computes what its platform's data allows and labels anything approximate or unavailable inside the indicator itself. The sections below are the READMEs that ship next to each port's source file (ports/<platform>/README.md).

TradingView (Pine Script v6)

Cival Pattern Lab for TradingView

Port of the ATAS original **Cival Pattern Lab v1.0.0.0** to Pine Script v6 (CivalPatternLab.pine).

Pattern Lab is a **research tool**, not a signal service. It evaluates one of six transparent, rule-based candle/order-flow patterns on **closed bars**, then reports a deliberately conservative simulation of those signals (next-bar-open entry, stop and target in ticks, maximum holding time, round-trip cost, stop-first on ambiguous bars) split into train / validation / untouched-test partitions by date. Results are observed history under the stated rules, not a calibrated probability, and no setting here has been shown to have an edge. The script is an `indicator()`, not a `strategy()`: it places no orders, makes no network calls and sends nothing off the chart.

Requirements

Item	Requirement
Platform	TradingView, Pine Script v6

Item	Requirement
Footprint data (real buy/sell volume per bar, POC)	Premium or Ultimate plan (<code>request.footprint()</code>).
Any other plan	Runs, but delta is ESTIMATED from lower-timeframe bars, labelled APPROXIMATE in the panel. POC migration is unavailable.
Chart type	Standard candles / bars. Heikin Ashi, Renko and other synthetic types are not supported.
Symbol	Needs volume. Symbols without volume produce no complete bars and therefore no signals.

Install

1. Open the **Pine Editor** at the bottom of a TradingView chart.
2. Paste the full contents of `CivalPatternLab.pine` (or *Open > New blank indicator* and replace the text).
3. Click **Save**, then **Add to chart**.
4. Set the pattern and simulation inputs, then set the **Holdout** dates (Validation begins / Untouched test begins). Leave a date at `1970-01-01` for (*not set*).
5. Optional alerts: *Create alert > Condition: Cival Pattern Lab > Any alert() function call*. Alerts are **off** by default (input *Alerts on closed signal / level touch*), as in the original; they fire only when a bar closes.

The status line (second row of the panel) always says which data mode is active: `FOOTPRINT (real buy/sell rows)`, `ESTIMATED ... APPROXIMATE` or `MIXED` (both modes occurred in the loaded history).

Verification note: the script was compiled on TradingView's servers (0 errors, 0 warnings). It has not been observed running on a live chart or against a Premium footprint feed in the build environment. Inspect results on your own data before relying on them.

Feature matrix

Values: Full / Approximate / Not available / Not ported.

Feature	ATAS original	This port	Notes
Pattern: Engulfing	Yes	Full	Same formula. Volume/delta filters use footprint, or estimate (Approximate).
Pattern: DeltaReversal	Yes	Full / Approximate	Uses delta of this and previous bar. Approximate when either bar uses estimated delta (label shows <code>est. delta</code>).
Pattern: Absorption	Yes	Full / Approximate	Close-location 0.65 fixed, as in the original.
Pattern: BreakoutRetest	Yes	Full / Approximate	Range = highest high / lowest low of the prior lookback bars excluding the previous bar.
Pattern: ReversalFiveBar	Yes	Full / Approximate	Close sequence <code>a0>a1>a2<a3<a4</code> (and mirror) + delta sign of the signal bar.

Feature	ATAS original	This port	Notes
Pattern: CandleSequence (U/D/N, 2-20 chars, side +1/-1)	Yes	Full	Invalid sequence text = no signals. Still subject to the volume and delta filters, as in the original.
Volume / prior-mean filter	Yes	Full / Approximate	Volume = classified buy+sell volume of the bar.
Minimum absolute delta % filter	Yes	Full / Approximate	
Require directional POC migration	Yes	Full with footprint; Not available with estimated delta	With estimated delta the rule cannot be evaluated, so signals are suppressed while it is on (fail closed) and the panel says so.
Footprint-complete gating (classified volume within 1 % of candle volume; prior lookback bars too)	Yes	Full	Tolerance exposed as a port-only input (default 1 %). Applies to the estimate as well.
Closed-bar signals, no repaint	Yes	Full	All state changes happen on <code>barstate.isconfirmed</code> ; no <code>lookahead_on</code> ; no future data.
Signal marker + LAB <pattern> label	Yes	Full	Up/down label at bar low/high, bull/bear colours. Label overlap avoidance is Not ported.
Next-bar-open entry simulation	Yes	Full	Entry at the open of the bar after the signal.
Stop (ticks), target = stop x multiple, max hold bars	Yes	Full	Tick = <code>syminfo.mintick</code> .
Round-trip costs in price units	Yes	Full	Subtracted from every trade.
Stop-first on ambiguous bars, stop/target gap fills, ambiguity count	Yes	Full	Gap against the stop fills at the open; favourable gap is capped at the target.
One position at a time; open trade at end of history not counted	Yes	Full	Panel shows pending.
MAE / MFE bounds	Yes	Full	Full-bar extrema, therefore bounds, not executable excursions.
Chronological partitions: Train / Validation / Test, boundary-crossing trades excluded	Yes	Full	Same predicates as the original panel.
Partition guard (test must follow validation)	Yes	Full	Invalid order suppresses new signals and says INVALID .
Statistics: n, win %, expectancy, profit factor, max drawdown, ambiguous, avg MAE/MFE	Yes (n, win %, expectancy; PF / DD for All)	Full	Port adds a results table with all columns for All / Train / Validation / Test.

Feature	ATAS original	This port	Notes
Trade outcome availability time	Next candle time	Full	<code>exitTime</code> is set to the open time of the bar after the exit bar.
Information panel (9 lines)	Yes	Full content	Rendered as a Pine table. Default <i>Compact panel</i> shows title + two lines, as in the original; the results table carries the numbers.
Panel top offset / width in pixels, automatic panel placement	Yes	Not ported	Tables are placed by corner: use <i>Panel position</i> / <i>Results table position</i> .
Panel font size	Yes	Approximate	Mapped to TradingView's text size steps.
Chart-clarity compact mode (Compact display / Full analysis detail)	Yes	Approximate	Compact mode hides the signal markers; the cross-indicator "current chart read" board is Not available (scripts cannot see each other).
Alerts on closed signal	Yes (off)	Full	<code>alert()</code> once per bar close; off by default. Alert sound setting is Not ported (set in the TradingView alert dialog).
CSV export (PatternLab-trades.csv, PatternLab-bars.csv, signals/zones)	Yes	Not available	Pine cannot write files.
<code>pattern_research.py</code> walk-forward grid	Companion script	Not available	It reads the CSV export, which Pine cannot produce.
Share with Cival Decision Panel	Yes	Not ported	No cross-indicator bridge on TradingView.
Censored-trade counter	Yes	Not ported	Pine bar series has no index gaps; the original's counter would always be 0.
Presets / review mode	None in this indicator	n/a	

Settings mapping

Names and groups mirror the ATAS display names. Defaults come from `Cival1.PatternLab.settings.json`.

Group	Input	Default	Notes
Rules	Custom sequence (U up / D down / N doji)	DDDUU	
Rules	Custom sequence side (+1 / -1)	1	Any non-negative value = +1.
Rules	Pattern	DeltaReversal	Engulfing, DeltaReversal, Absorption, BreakoutRetest, ReversalFiveBar, CandleSequence
Rules	Prior lookback bars	20 (3-500)	
Rules	Volume / prior mean	1 (0-20)	
Rules	Minimum absolute delta %	20 (0-100)	

Group	Input	Default	Notes
Rules	Require directional POC migration	off	Footprint only.
Simulation	Stop distance (ticks)	20	
Simulation	Target / stop multiple	2	
Simulation	Maximum holding bars	20	
Simulation	Round-trip costs in price units	0	
Holdout	Validation begins (chart time)	not set	1970-01-01 = not set. The date picker produces a UNIX time which is compared with bar open times.
Holdout	Untouched test begins (chart time)	not set	
Appearance	Show information panel	on	Also controls the results table.
Appearance	Show labels	on	Off = arrow only.
Appearance	Bull color / Bear color	#3CB371 / #FF6347	
Information panel	Panel font size	13 (10-22)	Approximate mapping.
Information panel	Compact panel (title + two lines)	on	
Performance	Maximum bars back	10000	Bars older than this from the last bar are not processed.
Alerts	Alerts on closed signal / level touch	off	Pattern Lab has no levels; applies to signals.
00 Chart clarity	Compact display (full detail must be off) / Full analysis detail	on / on	

Port-only inputs (group *Port (TradingView only)*): Footprint ticks per row (POC resolution, default 1), Estimation timeframe (blank = automatic), Footprint completeness tolerance % (default 1), Panel position, Show results table, Results table position.

Not ported (no equivalent or no effect in the original):

ATAS setting	Reason
Session start / Session timezone / Initial balance minutes	Inherited from the base class; Pattern Lab does not use them.
Panel top offset, Panel width, Automatic panel placement	Pixel layout; replaced by corner positions.
Show readiness/status panel	No effect for this indicator class in the original.
Compact volume numbers, Show bubble volume numbers, Bubble number color	Pattern Lab draws no volume bubbles.

ATAS setting	Reason
Avoid overlapping labels, Minimum bar spacing for signal labels	Need pixel geometry that Pine cannot read.
Maximum visible zones, Show touched levels	Pattern Lab creates no zones.
CSV export folder	Pine cannot write files.
Alert sound	Chosen in the TradingView alert dialog.
Share with Cival Decision Panel	No cross-indicator bridge.

Known differences from ATAS

1. **Data source.** ATAS reads bid/ask per price from your data provider. TradingView footprint rows come from TradingView's own intrabar data and buy/sell classification, so volumes, delta and POC can differ from ATAS on the same instrument. Expect different signals bar by bar; the rules themselves are identical.
2. **Estimated delta (no footprint).** Each lower-timeframe bar inside the chart bar is classed as up or down by close vs open (ties keep the previous direction) and its volume is added to buy or sell. This is a proxy for aggression, not measured aggression. It is flagged APPROXIMATE everywhere it is used; bars whose classified volume does not match candle volume within the tolerance are treated as incomplete and produce no signal. Default lower timeframe by chart timeframe: 1D or higher uses 15m, 4h to under 1D uses 5m, 5m to under 4h uses 1m. Charts under 5 minutes fall back to chart-bar direction (no intrabar detail) unless you enter a lower timeframe such as 1s, which needs a plan with seconds data.
3. **POC.** Taken as the midpoint of TradingView's POC row (rows are *ticks per row* wide, default 1). ATAS takes the lowest price among equal-volume rows; TradingView's tie rule is not documented. Only the *direction* of POC change is used.
4. **Estimated-bar POC.** Estimated bars have no POC, so *Require directional POC migration* blocks signals that depend on them.
5. **Exit time.** Set to the open time of the next bar (ATAS: next candle time). For the newest trade it is provisional (bar close) until the next bar exists.
6. **Holdout dates.** Compared as UNIX timestamps with bar open/exit times. How the date picker maps a calendar time to UNIX time follows TradingView's input widget, not an ATAS "chart time" setting.
7. **Arithmetic.** ATAS uses `decimal`; Pine uses floating point. Equality at exact tick boundaries (touching a stop or target exactly) can differ by rounding.
8. **Markers.** TradingView keeps at most 500 labels; on very long histories only the most recent markers remain visible (statistics still use every processed signal).
9. **History depth.** TradingView limits how many bars and how much footprint/intrabar history your plan loads. Statistics cover only the loaded, processed bars; the trade list is capped at 10,000 like the original.
10. **Historical vs realtime estimates.** Lower-timeframe data can be revised between historical load and realtime; estimated bars may therefore differ slightly after a reload (TradingView documents this repainting risk for `request.security_lower_tf`).
11. **Live bar.** The original's live-bar handling does not apply: nothing is evaluated until the bar closes.

NinjaTrader 8 (NinjaScript)

Cival Pattern Lab — NinjaTrader 8

Compile on import: NinjaScript Editor → F5. This file has **not been compiled against NinjaTrader 8** (NT8 was not available where the port was written). It was type-checked at C# 7.3 against hand-written stand-ins for the NinjaScript / SharpDX types, and every NT8 API name was checked against the NinjaTrader 8 help guide, but only your F5 compile proves it against the real platform.

Summary and requirements

Cival Pattern Lab is a research indicator. It applies one of six transparent rules (Engulfing, DeltaReversal, Absorption, BreakoutRetest, ReversalFiveBar, CandleSequence) to closed bars, gated on volume versus the prior mean and absolute delta %, with an optional directional POC-migration filter. Every match is then run through a conservative simulation: entry at the next bar's open, a stop in ticks, a target equal to the stop times a multiple, a maximum holding time, a round-trip cost, stop-first on ambiguous bars. Results are split into train / validation / untouched test partitions by date and summarised in a panel (n, win %, expectancy, profit factor, drawdown). Trades and bar features can be exported to CSV. It is an **Indicator** (not a Strategy), places no orders and makes no network calls. It reports observed simulation results on your own data; it makes no profit, win-rate or accuracy claim, and changing parameters is another research trial, not independent evidence.

Requirements:

- NinjaTrader 8 with an Order Flow+ entitlement, and a chart using **Order Flow Volumetric Bars** (Ticks per level 1 recommended; delta type BidAsk recommended). Bar delta, classified volume and POC come from the Volumetric series.
- On a chart that is not Volumetric the patterns are **disabled** and the panel shows "Chart is not Volumetric (Order Flow+): delta/POC patterns disabled". Nothing is estimated or invented.
- `Calculate = OnBarClose`; every signal forms on a confirmed bar. Nothing repaints.

Install

1. Copy `CivalPatternLab.cs` to `Documents\NinjaTrader 8\bin\Custom\Indicators\`.
2. In NinjaTrader open **New → NinjaScript Editor** and press **F5** (Compile). The editor adds its own generated-code section at the bottom of the file on a successful compile.
3. Open a Volumetric chart, right-click → **Indicators → Cival Pattern Lab**.
4. The panel starts in Compact mode (title + two lines, as in the ATAS original). **Untick "Compact panel"** to see the train / validation / test statistics.
5. Set **Validation begins** and **Untouched test begins** (chart time, in order) before judging results. "(not set)" is `DateTime.MinValue` and shows as 1/1/0001 in the grid.

Feature matrix

Feature	ATAS original	This port	Notes
Six pattern families + custom U/D/N sequence	Yes	Full	Same rules, same 0.65 close-location constant. Needs Volumetric data (see below).
Volume / prior mean and delta % gates	Yes	Full	Prior mean over the previous <i>Lookback</i> bars.
Directional POC migration filter	Yes	Approximate	POC from <code>GetMaximumVolume</code> ; tie-break and Ticks-per-level behaviour can differ from ATAS (see differences).
Bar delta / classified volume source	Bid/ask footprint rows	Approximate	Volumetric <code>TotalBuyingVolume</code> – <code>TotalSellingVolume</code> ; follows the series' delta type.
Footprint completeness (1 % reconciliation)	Yes	Full	Classified volume must match bar volume within 1 %; otherwise the bar cannot trigger or serve as a prior bar.
Disabled-with-status on missing footprint	Yes	Full	Visible panel status on non-Volumetric charts.
Closed-bar processing, no repaint	Yes	Full	<code>Calculate.OnBarClose</code> .
Next-bar-open simulation, one position at a time	Yes	Full	
Stop ticks, target = stop × multiple, max hold, round-trip cost	Yes	Full	Prices compared with a 1e-6-tick tolerance (doubles instead of decimals).
Stop-first on ambiguous bars, gap fills, ambiguity count	Yes	Full	
MAE / MFE full-bar bounds	Yes	Full	Bounds, not executable excursions.
Censored / pending positions	Yes	Full	Missing bar index censors the position.
Train / validation / test partitions, boundary exclusion	Yes	Full	Chart time, see differences for the time basis.
Panel with n, win %, expectancy, PF, drawdown, ambiguous, censored, pending	Yes	Approximate	Drawn by the indicator (SharpDX); no cross-indicator auto-arrange.
Signal markers + "LAB " labels	Yes	Full	Triangle glyph + label with overlap avoidance and bar-spacing rule.
Compact display / Full analysis detail	Yes	Approximate	Full detail expands abbreviations as the original does; compact draws nothing for Pattern Lab (same as ATAS); the shared "current chart read" board is not ported.
CSV export: <code>PatternLab-trades.csv</code> , <code>PatternLab-bars.csv</code> , <code>PatternLab-signals.csv</code>	Yes	Full	Same columns. <code>PatternLab-zones.csv</code> is header-only (Pattern Lab has no zones). Blank folder = disabled.

Feature	ATAS original	This port	Notes
pattern_research.py walk-forward companion	Yes	Full	Reads the same PatternLab-bars.csv columns. Time column is NT chart time.
Alerts (off by default)	Yes	Approximate	NT Alert() in real-time only; sound name maps to <NT_install>\sounds\<name>.wav.
Rebuild when provider revises already-processed history	Yes	Not available	NT has no equivalent callback; reload the data / press F5 on the chart.
Share with Cival Decision Panel	Yes (ATAS-only bridge)	Not ported	Not part of the shipped settings reference.
Review mode / presets	None in the original	—	—

Settings mapping

ATAS group / name	NT8 property	Default	Notes
Rules / Custom sequence (U up / D down / N doji)	CandleSequence	DDDUU	
Rules / Custom sequence side (+1 / -1)	SequenceSide	1	
Rules / Pattern	Pattern	DeltaReversal	Enum; no [NinjaScriptProperty] on purpose (keeps NT's generated code free of the nested enum type).
Rules / Prior lookback bars	Lookback	20 (3–500)	
Rules / Volume / prior mean	VolumeMultiplier	1 (0–20)	double
Rules / Minimum absolute delta %	DeltaPercent	20 (0–100)	double
Rules / Require directional POC migration	PocConfirmation	false	
Simulation / Stop distance (ticks)	StopTicks	20	
Simulation / Target / stop multiple	RewardMultiple	2	double
Simulation / Maximum holding bars	HoldBars	20	

ATAS group / name	NT8 property	Default	Notes
Simulation / Round-trip costs in price units	Costs	0	double
Holdout / Validation begins, Untouched test begins	ValidationStart, TestStart	not set	DateTime
Appearance / Show information panel, Panel top offset, Panel width	ShowPanel, PanelTop, PanelWidth	true, 50, 380	
Appearance / Avoid overlapping labels, Minimum bar spacing, Show labels	AvoidLabelOverlap, MinimumLabelBarSpacing, ShowLabels	true, 6, true	
Appearance / Bull color, Bear color	BullColor, BearColor	MediumSeaGreen, Tomato	WPF Brush (NT convention)
Information panel / Panel font size, Compact panel	PanelFontSize, CompactPanel	13, true	
Export / CSV export folder (blank disables)	ExportFolder	blank	
Performance / Maximum bars back	MaxBarsBack	10000	Only the latest N bars are processed.
Alerts / Alerts on closed signal / level touch, Alert sound	EnableAlerts, AlertSound	false, alert1	
00 Chart clarity / Compact display, Full analysis detail	CleanChart, FullAnalysisDetail	true, true	

Not ported (and why)

ATAS setting	Reason
Session start / Session timezone / Initial balance minutes	Have no effect in the ATAS Pattern Lab; omitted rather than shown as dead controls.
Automatic panel placement	Needs ATAS's shared panel registry. In NT8 give each Cival panel a different <i>Panel top offset</i> .
Show readiness/status panel	Has no effect in the ATAS analytics indicators.
Compact volume numbers, Show bubble volume numbers, Bubble number color	Pattern Lab draws no volume bubbles.
Maximum visible zones, Show touched levels	Pattern Lab has no zones.

Known differences from the ATAS original

1. **Order-flow source.** ATAS builds delta and POC from bid/ask footprint rows; this port reads the Volumetric series (`BarDelta`, `TotalBuyingVolume`, `TotalSellingVolume`, `GetMaximumVolume`). With the default BidAsk delta type the numbers represent the same quantity; with `UpDownTick` they do not.
2. **POC tie-break.** ATAS picks the lowest price among equal-volume rows. NinjaTrader returns the highest-volume level's price and, when Ticks per level is above 1, the lowest price of that level; ties are not specified. This only matters when *Require directional POC migration* is on.
3. **Time basis.** NinjaTrader stamps time-based bars with their close time. Bar open time (used for the entry time and the *Validation / Test* comparisons) is derived exactly for Minute and Second bars (including the base period of Volumetric bars) and otherwise taken as the previous bar's stamp. The exit time is the exit bar's close stamp, the same "outcome known at the candle boundary" meaning as the original. Dates are compared in the chart's time (NinjaTrader's own clock); no time-zone conversion is applied.
4. **Numeric type.** Prices are doubles with a 1e-6 tick tolerance on equality-sensitive comparisons; ATAS uses decimals. A signal sitting exactly on a threshold can differ in rare cases.
5. **Compact panel default.** As in the original, the default panel shows only three lines. Untick *Compact panel* for statistics.
6. **Panels.** Panels are drawn with SharpDX at the *Panel top offset*; there is no cross-indicator auto-arrangement.
7. **Export cadence.** Files are rewritten on the last historical bars and then on each real-time bar close. Archive them before loading another chart; use a separate folder per symbol/timeframe.

No profitability or vendor-output parity is claimed. Synthetic and historical simulations validate software behaviour, not market edge.