

Cents-Grace Exit: Bringing Back the Premarket Bot's \$0.10/\$0.05 Stop

Research findings, 2026-09-18. Prepared for review. Not wired into the live bot.

The idea

Bring back a mechanism that actually ran live in the sibling “premarket” bot (premarket_backup_2026-09-08_2010/): a flat \$0.10 trailing stop off the position's own peak, with one \$0.05 “grace” extension if the stock's own health signals don't confirm real deterioration at the moment the stop is touched. That project's own config notes record 14/14 trades touching the stop and needing the grace extension just to survive the first touch on one real day — i.e. even there this was a load-bearing mechanism on this same \$5-15 price cohort, not a rare edge case.

Port, not copy

The original's “health” gate was a separate intraday_health.py module (price/VWAP slope + volume participation + structure classification) that doesn't exist in this project. exit_cents_grace_experimental.py instead reuses THIS bot's own existing “healthy” definition — exit.py's own Layer 3 vote (price vs VWAP, VWAP slope, price trend, lower-highs/lower-lows), which already logs “healthy (N/4 deterioration signals)” when fewer than 2 of 4 agree. “Still healthy” means the same thing everywhere in this codebase now, not a second, differently-tuned health concept. Wired into simulate.py via --cents-grace-exit, with --cents-grace-base, --cents-grace-amount, and --cents-grace-max-extensions to sweep all three original knobs.

Backtest sweep (2026-09-18, max_positions=3)

Baseline (no extra exit layer): 10 trades, 5W/5L, \$559.56.

Base	Grace	Max ext.	Trades	W/L	Total P/L
\$0.10	\$0.05	1	20	9/11	\$497.28
\$0.10	\$0.05	2	12	6/6	\$547.31
\$0.10	\$0.10	1	12	6/6	\$547.31
\$0.10	\$0.10	2	13	6/7	\$432.13
\$0.15	\$0.05	1	12	6/6	\$540.07
\$0.15	\$0.05	2	13	6/7	\$427.11
\$0.15	\$0.10	1	12	5/7	\$315.33
\$0.15	\$0.10	2	11	6/5	\$288.26
\$0.20	\$0.05	1	13	6/7	\$429.09
\$0.20	\$0.05	2	12	6/6	\$437.48
\$0.20	\$0.10	1	13	6/7	\$426.87

Base	Grace	Max ext.	Trades	W/L	Total P/L
\$0.20	\$0.10	2	11	5/6	\$577.74
\$0.25	\$0.05	1	12	5/7	\$309.27
\$0.25	\$0.05	2	11	6/5	\$337.95
\$0.25	\$0.10	1	11	6/5	\$337.95
\$0.25	\$0.10	2	10	5/5	\$559.56 (never fires)

The one config that beat baseline — and why it doesn't count

\$0.20 base + \$0.10 grace × 2 extensions finished at **\$577.74** (11 trades, 5W/6L) — \$18.18 above baseline, the first configuration across this entire investigation (PRF, giveback-vote, resistance-timer, velocity, and now this) to beat the no-extra-exit day at all. Tracing the actual trade list shows why that's not real evidence the mechanism works:

- GLOO's exit reason under this config is **still** “deterioration confirmed: 3/4 signals” — the real exit.py layer, at the identical 18:17:27 timestamp as baseline. The cents-grace layer never touched this trade at all.
- The improvement comes entirely from LTRX and SECZ's first attempts getting cut earlier by the cents-grace floor, which happened to free a slot for GLOO to re-enter at a slightly better price (\$4.93 vs baseline's \$5.07) — a slot-recycling timing coincidence, not the exit design solving reversal detection.
- FWDI, SBET, and SECZ's second position all still ride to END_OF_DAY exactly as in baseline — this config never actually intervenes on any of the day's real winners or its real motivating loser either way.
- Sensitivity is a cliff, not a gradient: the adjacent config one grace extension tighter (\$0.20/\$0.10×1) scores \$426.87 — a \$150 swing from a single integer knob. A real effect should degrade smoothly as a parameter moves; a cliff this sharp on one day's data is itself evidence of noise, not signal.

Treat \$577.74 as a coincidence of this one day's specific re-entry cascade, not a validated result.

What the sweep actually shows

Every other configuration underperforms baseline, most by a wide margin (\$288–\$547). At the original project's own literal defaults (\$0.10/\$0.05×1), this cohort's normal noise band is wide enough on this day's names that the layer fires constantly (20 trades vs baseline's 10, turning ordinary chop into 11 separate exits), consistent with that project's own historical note about 14/14 trades needing the grace extension just to survive a first touch. Loosening the base distance (matching this project's \$5-15 range more generously) mostly still lands well below baseline. Nothing here demonstrates the ported mechanism solving the reversal-vs-continuation problem on this data.

Recommendation

- Do not wire exit_cents_grace_experimental.py into the live bot — same standing as every other experimental exit this session.

- This is now the fifth distinct rule shape tested against the same problem (structure/imbalance state machine, fixed giveback floor, fixed recovery-window clock, decline velocity, and now a flat-dollar trailing stop with health-gated grace) with the same outcome: no configuration reliably beats simply not adding a fifth exit layer, on the one day available.
- If revisited, do not tune further on 2026-09-18 alone — the sharp sensitivity found above is a specific warning sign that more knob-turning on this single day will keep surfacing coincidences, not a real edge.