

PLAN_T1_ROADMAP_2_WEEKS_v3.1

Status: Draft to-confirm **Erstellt:** 2026-05-16 **Vorgänger:** v3 (locked 2026-05-16 16:45 UTC) **Korrekturen-Basis:** plan-v3-1-adjust-prompt.md (Operator-Input 2026-05-16) **Start frühestens:** Montag 2026-05-19 nach Operator-GO GO PHASE A

1. Executive Summary

Zweck dieses Dokuments: den 2-Wochen-Plan für den T1-Quality-Refactor des Steve-TradingBot so überarbeiten, dass er zur neuen Zielarchitektur passt (T1 = Binance only, T2 = Solana Pump only, T3 = Copy Trading only) und gleichzeitig die Trading-Risiken durch eine Shadow-vor-Enforce-Trennung minimiert.

Kernsätze:

- T1 bleibt: robust, langweilig, Binance-only, Quality > Quantity.
- T2 ist Solana-reserviert und vollständig deaktiviert.
- T3 bleibt Copy-Trading-reserviert und deaktiviert.
- Quality-Filter laufen erst Shadow, dann Enforce.
- Risk-Guard kommt VOR der Quality-Enforcement.
- Partial Profit wird separate Phase mit eigenem Datenvertrag.
- Stop-Gates sind marktunabhängig formuliert.
- Boundaries Woche 1+2: 0x Mainnet, 0x T2-Code, 0x T3, 0x Leverage, 0x DB-Mass-Mutation, 0x Push.

Realistisches Wochenziel bleibt: **2-4 % / Woche** auf Testnet. 10 % / Woche bleibt Aspiration für nach echter T2-Aktivierung.

2. Was an v3 geändert wurde

Bereich	v3	v3.1
T2-Status	PLAN_ONLY (Binance-Pump-Label)	DEACTIVATED, Solana-reserviert
MS-Runner-Rolle	implicit T2-Vorläufer (breakout/volatility_sweep → t2_pump_dump)	T1-Binance-Research oder disabled, nicht finales T2
STRATEGY-GROUP-CONTRACT-1 Mapping	breakout / volatility_sweep → t2_pump_dump	alle 5 MS-Strategien → t1_core (in Phase A korrigiert)
Phase C Quality-Score	ein Schritt, sofort blockierend	gesplittet in C1 Shadow + C2 Enforce
Phasenreihenfolge	A → B → C → D → E → F → G → H	A → B → C1 → D → C2 → E1 → E2 → F → G → H
Phase E Exit-Optimizer	BE + Partial + Trailing + Time-Stop in einem Step	E1 = BE + Trailing + Time-Stop / E2 = Partial Profit separat
T1-Binance-Hardrule	implizit	explizit: Pair-Existenz + Filter-Check (tick / step / minNotional) als Pflicht-Gate
T2-Pump-Dump als Label	erlaubt für Binance-Pumps	verboten — T2 ist Solana-only
T3-forwarded-Logik	reduziert, nicht hard-disabled	als legacy markiert, in Phase A endgültig disabled/archived
Stop-Gates	teils marktabhängig (z.B. „min. 1 Partial-Event in 24h“)	marktunabhängig — synthetische Tests + Runtime-Stabilität reichen
DCA in BEAR	tighten	komplett blockiert
SL-Invariant	best practice	harte Regel: new_sl < old_sl verboten außer dokumentiertem DCA-Rebuild mit allen Guards
Telegram-Notifs	per Event	rate-limited Summary + Log statt Spam pro Block
Post-Trade Learning	Grade A-F als Live-Feed	reine Analyse-Layer, keine automatische Param-Änderung

3. Neue Zielarchitektur

T1 = Binance only	ACTIVE (Spot, Testnet zuerst)
T2 = Solana Pump only	DISABLED bis separate Shadow-Phase
T3 = Copy Trading only	DISABLED bis separater Plan
legacy_unknown	historical only – keine neuen Rows

3.1 T1 Binance-only

- Spot, kein Leverage, kein Futures.
- Quote-Asset: USDT (Phase 1 only).
- Jedes Signal muss Binance-handelbar sein:

```
Signal → exchangeInfo.symbols enthält Pair?  
yes → status == "TRADING", isSpotTradingAllowed == true → T1 may evaluate  
no → reject_reason = symbol_not_on_binance
```

- Zusätzliche Pflicht-Filter:
 - LOT_SIZE → minQty / stepSize
 - PRICE_FILTER → tickSize
 - MIN_NOTIONAL bzw. NOTIONAL → minNotional
 - PERCENT_PRICE_BY_SIDE (sanity-check)
- Spread / Top-of-Book Liquidity Gate (eigene Stufe)
- **Nicht erlaubt** in T1:
 - Solana-Fallback
 - T2-Fallback / Pump-Branch
 - Weiterleitung von Solana-only Tokens an T1
 - DEX-Trading
 - Non-USDT-Quotes

3.2 T2 Solana-only

- Solana SPL-Tokens / Pump.fun / Meme / Momentum.
- T2-Code, T2-Wallet, T2-Execution: alle deaktiviert bis explizit T2-SOLANA-SHADOW-Phase startet.
- Bisheriger Binance-Pump-/Breakout-Mix wird **nicht** als T2 weiter ausgebaut.
- MS-Runner gehört NICHT zu T2 — MS-Runner ist T1-Binance-Research solange er Binance handelt.

3.3 T3 Copy-Trading-only

- bleibt deaktiviert, eigener Plan kommt später.
- `_process_t3_forwarded_signals` wird in Phase A endgültig disabled / archived.
- Historie bleibt unverändert; keine neuen Forwards.

4. Korrigierte Phasenreihenfolge

Phase	Name	Aufwand	Block-Status
A	TIER-ARCH-CONTRACT-1	1.0 d	Pre-Code (Mapping-Refresh, DB-CHECK-Constraint, T3-Forward-Disable)
B	T1-BINANCE-SYMBOL-GATE-1	1.0 d	Hard-Gate
C1	T1-QUALITY-SCORE-SHADOW-1	1.5 d	log-only, 24-48h Beobachtung
D	T1-RISK-GUARD-1	2.0 d	BEAR-DCA-Block, SL-Invariant, Vol-Sizing
C2	T1-QUALITY-SCORE-ENFORCE-1	1.0 d	nach C1-Auswertung, gezielte Filter
E1	T1-EXIT-OPTIMIZER-BE-TIMESTOP-1	1.0 d	BE + Trailing-Gate + Time-Stop
E2	T1-PARTIAL-PROFIT-1	2.0 d	separater Plan + Datenvertrag
F	T1-POST-TRADE-LEARNING	1.0 d	Analyse-Only, keine Live-Params
G	EXEC-MODE-LABEL-3 Phase 3a	0.5 d	Cleanup, profitneutral
H	REFACTOR-VS-REWRITE-PDF + T2-SOLANA-SHADOW-PLAN-PDF	0.5 d	reine Doku, Ausblick

Netto: ~11.5 Arbeitstage über 2 Wochen — leicht mehr als v3 wegen Phase-Splits, aber besser steuerbar.

Mandatory Decision-Gate

Nach Phase D (Risk-Guard) **Operator-Decision-Gate**:

Option A: weiter mit C2 (Quality-Enforce) wie geplant
Option B: erst eine Beobachtungswoche, dann C2
Option C: Reihenfolge anpassen wenn Live-Erkenntnisse das nahelegen

Kein automatischer Übergang.

5. Phase A Detail — TIER-ARCH-CONTRACT-1

Ziel

Die aktuelle Strategie-Group-Semantik so umstellen, dass T2 = Solana-only und MS-Runner-Strategien alle als T1 laufen. Alte Mischlogik (T3-forwarded → T1, T2 = Binance-Pump-Label) endgültig zumachen.

Scope

1. `strategies/strategy_group_map.py` — Mapping refresh:

```
STRATEGY_ID_TO_GROUP = {
    "trend_follow":      STRATEGY_GROUP_T1,
    "mean_reversion":    STRATEGY_GROUP_T1,
    "vwap_mean_reversion": STRATEGY_GROUP_T1,
    "breakout":          STRATEGY_GROUP_T1,    # geändert von T2
    "volatility_sweep":  STRATEGY_GROUP_T1,    # geändert von T2
}
DEFAULT_STRATEGY_GROUP = STRATEGY_GROUP_T1
```

1. Neue Konstante `STRATEGY_GROUP_T2_SOLANA_PUMP = "t2_solana_pump"` (Code-Konstante, **nicht aktiv** — Reserve).
2. DB CHECK-Constraint Migration:

```
-- decision_logs.strategy_group, trade_logs.strategy_group, position_snapshots.strategy_group
ALTER TABLE ... DROP CONSTRAINT ..._strategy_group_check;
ALTER TABLE ... ADD CONSTRAINT ..._strategy_group_check
CHECK (strategy_group IN ('t1_core','t2_solana_pump','t3_copy_trading','legacy_unknown'));
```

`t2_pump_dump` wird aus der erlaubten Menge entfernt. Historische Rows mit `t2_pump_dump` werden **nicht** mutiert (D5 vom Operator) — stattdessen wird `legacy_unknown` als Fallback-Mapping in einem Read-Layer eingeführt, falls die GUI sie noch lesen will.

1. `_process_t3_forwarded_signals` in `trading/main.py` als **disabled / reject / archived** markieren:

```
def _process_t3_forwarded_signals(...):
    # ARCHIVED 2026-05-19 – T3 reinterpreted as Copy-Trading-only
    return
```

1. Externe Telegram-/Discord-Signal-Channel-Eingänge die bisher als `t2_pump_dump` getaggt waren werden auf `t1_core` umgeschrieben (Code-Pfad), nur für neue Inputs.

Files (geschätzt)

- `trading/strategies/strategy_group_map.py`
- `trading/strategies/multi_strategy_runner.py` (Tests anpassen)
- `trading/main.py` (Forward-Disable)
- `db/migrations/<n>_strategy_group_check_v3_1.sql` (neue Migration)
- `gui/app/Models/...` (falls `strategy_group`-Constants gespiegelt — read-only Anpassung)
- Tests: `tests/test_strategy_group_contract_*.py`, `tests/test_t3_forward_disabled.py`

Stop-Gates Phase A

- 3-Way MD5 Repo == Image == Container
- Migration läuft auf Testnet-DB ohne Datenverlust
- Bot-Restart ohne Traceback
- Worker liest neue Constraint korrekt
- GUI rendert `t1_core` für alle MS-Strategien
- `_process_t3_forwarded_signals` returnt sofort, 0 neue T3-Forward-Rows
- Regressionstests grün

Boundaries

- 0× Mainnet
- 0× T2-Code (Solana)
- 0× historische DB-Mutation
- 0× CommandBus-V6
- Operator-GO G0 PHASE A zwingend vor Cutover

6. Phase B Detail — T1-BINANCE-SYMBOL-GATE-1

Ziel

Hartes Gate: jedes Signal an T1 muss auf Binance handelbar sein. Frühe Ablehnung mit klarem Reject-Reason.

Scope

1. Symbol-Service erweitern (`trading/exchanges/binance_symbol_service.py` oder existierendes Äquivalent):
2. Lädt `exchangeInfo` einmal pro Session + caches mit TTL (z.B. 1h).
3. Bietet API: `is_t1_tradable(symbol) -> (bool, reject_reason)`.

4. Berücksichtigt:

- `status == "TRADING"`
- `isSpotTradingAllowed`
- `Quote-Asset == USDT` (Phase 1)
- Filter-Set komplett vorhanden

5. Reject-Reason-Vocabulary:

```
symbol_not_on_binance
binance_pair_inactive
binance_quote_not_usdt
min_notional_failed
lot_size_failed
price_filter_failed
permission_failed
```

1. Wiring:
2. MS-Runner `_attempt_execution` ruft Gate vor allen anderen Checks.
3. External-Channel-Inputs (Telegram-Bot, Discord später) gehen durch dasselbe Gate.
4. T3-Forwards sind disabled (Phase A) → kein Gate-Call nötig.
5. `decision_logs.metadata_json` enthält:
6. `binance_pair_check = {exists, active, quote, filters_ok}`
7. falls reject: `reject_reason + binance_pair_details` (kompakt)
8. Tests:
9. `tests/test_binance_symbol_gate.py` — mocked `exchangeInfo`, alle Reject-Reasons abgedeckt.
10. keine echten Orders.
11. Testnet/Mainnet-Endpoints strikt getrennt.

Stop-Gates Phase B

- `exchangeInfo`-Cache funktioniert (kein Live-API-Spam in Tests)
- alle Reject-Reasons in Unit-Tests grün
- Runtime: keine zusätzlichen Tracebacks
- `decision_logs.metadata_json.binance_pair_check` strukturiert
- Live-Event nicht zwingend — wenn auftritt, korrekt klassifiziert

Boundaries

- 0× echte Order-Calls in Tests
 - 0× Mainnet-API
 - 0× Mutation an `binance_markets`-Cache während laufendem Bot
 - Operator-GO G0 PHASE B
-

7. C1 — T1-QUALITY-SCORE-SHADOW-1 Detail

Ziel

Quality-Score berechnen und loggen, **ohne** zu blockieren. 24–48h Beobachtung um zu verstehen, welche Sub-Filter wie hart greifen würden.

Sub-Scores (Shadow only)

```
liquidity_score
spread_score
volume_burst_score
regime_fit_score
trend_quality_score
distance_to_resistance_score
btc_health_score
mtf_alignment_score
no_trade_zone_score
```

Jeder Sub-Score in $[0.0, 1.0]$. Combined-Score = gewichteter Mittelwert (Gewichte konfigurierbar, Defaults aus v3 erbst).

Pflichtfelder in `decision_logs.metadata.json`

```
{
  "quality_score": 0.72,
  "quality_threshold_shadow": 0.65,
  "quality_pass_shadow": true,
  "quality_components": {
    "liquidity_score": 0.81,
    "spread_score": 0.66,
    "...": 0.xx
  },
  "would_reject_reason": null,
  "regime_at_check": "BEAR_STABLE"
}
```

Bei Reject im Shadow:

```
{
  "quality_pass_shadow": false,
  "would_reject_reason": "trend_quality_below_threshold",
  "would_reject_subscore": "trend_quality_score=0.42"
}
```

Ziel C1-Auswertung

Nach 24–48h:

- Verteilung Combined-Score pro Regime
- Top-3 limiting Sub-Scores
- Wäre-Reject-Rate gesamt + pro Regime
- BUYs die durch Shadow-Block verloren gingen (hätten ginge)
- Realisierte PnL der zugelassenen Trades vs. Schatten-blockierte

→ Daraus Threshold-Set + welche Sub-Scores tatsächlich enforced werden.

Stop-Gates C1

- alle Sub-Scores schreiben metadata
- keine BUYs werden durch C1 blockiert
- regime-tagging korrekt
- 24h Mindest-Laufzeit Testnet
- mindestens $N=X$ (Operator-Zahl) Decision-Rows mit Sub-Score-Daten
- Runtime stabil

Boundaries

- 0x Block durch C1
- 0x Strategieparameter-Change
- 0x neue Reject-Reasons live wirksam

8. Phase D — T1-RISK-GUARD-1 Detail

Ziel

Den größten realen Schaden adressieren, **bevor** Quality enforced wird. Reihenfolge bewusst: Risk-Guard vor Quality-Enforce.

Sub-Module

8.1 BEAR-DCA-Block

- DCA in `BEAR_STABLE` oder `HIGH_VOL+BEAR` Kombination **komplett geblockt**.
- Block-Reason: `dca_blocked_bear_regime`.
- gilt für alle DCA-Trigger (Support-Fallback, ATR-Rescue, `vwap_revisit`).

8.2 No-DCA-without-Reclaim

- DCA nur erlaubt, wenn:
- aktueller Preis > letzter Reclaim-Level
- oder Price-Action-Reclaim-Bedingung positiv
- Sonst: Block-Reason `dca_no_reclaim`.

8.3 Max-Loss-Simulation

- Vor JEDEM DCA: simuliere SL-Hit nach DCA mit neuem Average-Entry.
- Wenn simulierter Loss > `MAX_LOSS_PER_TRADE_USDT` → Block-Reason `dca_max_loss_exceeded`.

8.4 SL-Invariant

Für Long-Trades: neuer SL darf nicht schlechter sein als alter SL, es sei denn ein dokumentierter DCA-Rebuild ist explizit erlaubt und alle Guards erfüllt sind.

Ausnahme-Trigger:

- DCA hat stattgefunden + Reclaim erfüllt
- Neuer simulierter Max-Loss < Cap
- Regime erlaubt
- Trade-Risk-Doku in `position_snapshots.metadata_json`

8.5 Volatility-Scaled Position Sizing

- Position-Size skaliert mit aktueller ATR / Median-ATR.
- High-Vol → kleinere Position (z.B. 0.5–0.7×).
- Low-Vol → leicht größere Position (max 1.2×).
- Cap bleibt absolut.

Telegram-Spam-Reduktion

- per-Block-Telegram raus
- statt: rate-limited summary alle 60min (max 1 Nachricht)
- echte Runtime-Anomalien (z.B. Exchange-Error) bleiben Realtime

Files (geschätzt)

- `trading/risk/risk_guard.py` (neu)
- `trading/execution/dca_rescue.py` oder Äquivalent (Block-Hooks)
- `trading/position_manager.py` (SL-Invariant + Vol-Sizing)
- `trading/notifier/reporter.py` (Rate-Limit Summary)
- Tests: `tests/test_risk_guard_*.py`

Stop-Gates Phase D

- 4 Sub-Module unit-test grün
- Synthetische BEAR-Simulation: DCA wird geblockt
- SL-Invariant-Regression: kein neuer schlechterer SL ohne Doku
- Vol-Sizing skaliert in Tests innerhalb Cap

- Telegram-Spam-Test: max 1 Summary/h in Stress-Sim
- Live-Trade-Veränderung: BUYs OK, DCA-Verhalten konservativer

Boundaries

- 0x Strategieparameter
- 0x Mainnet
- 0x DB-Migration
- Operator-GO GO PHASE D

9. C2 — T1-QUALITY-SCORE-ENFORCE-1 Detail

Voraussetzung

C1 mindestens 24h gelaufen + Operator-Auswertung + Decision-Gate aus §4 bestanden.

Scope

- Aus C1-Auswertung ausgewählte Sub-Scores werden enforced.
- Regime-individuelle Thresholds:

Regime	Combined-Threshold	Quality-Floor
BULL_STABLE	7.0	0.55
BULL_TREND	7.0	0.55
RANGE	7.5	0.60
BEAR_STABLE	8.0	0.65
HIGH_VOL + BEAR	block	—
HIGH_VOL + BULL	7.5	0.65

Werte sind v3-Defaults — final werden sie aus C1-Daten gewählt.

Reject-Reasons (live aktiv)

```
quality_below_combined_threshold
quality_below_floor
quality_high_vol_bear_blocked
quality_subscore_<name>_below_threshold (nur für enforced sub-scores)
```

Pflicht

- jeder Live-Reject hat `would_reject_reason` ↔ `reject_reason` Symmetrie mit C1
- Shadow-Mode-Code bleibt parallel als Fallback (Flag)
- keine Überfilterung — Operator entscheidet welche Sub-Scores enforced werden

Stop-Gates Phase C2

- Unit-Tests pro Regime + Subscore-Set grün
- Live-Reject-Rate plausibel zu C1
- Runtime stabil
- keine 0-BUY-Tage erzwungen (Notbremse: wenn 48h 0 BUYs → automatisch zurück in Shadow)

Boundaries

- Operator-GO GO PHASE C2 zwingend nach C1-Decision-Gate
- 0x Mainnet
- 0x Strategieparameter-Hardcode-Change

10. E1 — T1-EXIT-OPTIMIZER-BE-TIMESTOP-1 Detail

Scope

- **Früherer Break-even:** bei +0.5R oder +0.7R (statt v3-Default +1R) — Operator-Auswahl.

- **Momentum-Trailing-Gate:** Trailing aktiviert nur wenn Momentum-Score > Schwelle, sonst Standard-SL.
- **Time-Stop bei Seitwärtsbewegung:** nach `TIME_STOP_HARD_EXIT_HOURS` ohne Bewegung in Range R → exit.
- **bessere Exit-Metadaten:** `exit_reason_kind` v3-Subtypes erweitert um `time_stop_sideways`, `momentum_trail_loss`.

Stop-Gates E1

- Unit-Tests pro Exit-Kind grün
- Live: BE wird tatsächlich gesetzt wenn Trade > Schwelle
- Time-Stop greift in Sim
- keine Konflikte mit Trailing-Logik

Boundaries

- 0x Partial-Profit-Code in E1 (separater Schritt E2)
- 0x Strategie-Parameter-Change die nicht in Phase enthalten
- Operator-GO GO PHASE E1

11. E2 — T1-PARTIAL-PROFIT-1 Detail

Vorbedingung

E1 abgeschlossen, Runtime stabil. **Eigener Plan v3.1-E2** wird vor Implementierung erstellt.

Pflicht-Bestandteile

1. **Datenvertrag** dokumentieren:
2. Position behält ein Record, aber `closed_qty` / `remaining_qty` Trennung
3. Average-Entry bleibt erhalten
4. Realized-PnL inkrementell
5. Unrealized-PnL nur auf `remaining_qty`
6. **trade_logs:**
7. Partial-Exit erzeugt eigene Row mit `exit_kind = partial_profit_pX`
8. PnL nur auf verkaufter Menge
9. Original-Entry-Row bleibt offen mit reduzierter Qty
10. **position_snapshots:**
11. Snapshot nach Partial: `status=partial_closed`, `remaining_qty=X`, `last_partial_at=ts`
12. vollständiger Close: `status=closed`
13. **GUI / Chart-Marker:**
14. Partial-Markers eigene Symbolik
15. PnL nicht doppelt zählen
16. History zeigt alle Partial
17. **Idempotency / Retry:**
18. CommandBus oder direkter Pfad: Partial darf nicht doppelt ausgeführt werden
19. mtime-cookie / version-check wie in HISTORY-1
20. **Regressionstests:**
21. Partial → Full Exit
22. Partial → DCA → Partial
23. Partial → SL hit
24. Partial → Bot-Restart → Konsistenz

Stop-Gates E2

- Datenvertrag-Doku als PDF
- Alle Regressions grün
- GUI rendert Partial korrekt

- PnL-Summen stimmen
- Operator-GO G0 PHASE E2

Boundaries

- 0x Live-Aktivierung vor synthetischen Tests
- 0x Order-Calls in CI

12. F — T1-POST-TRADE-LEARNING Detail

Charakter

Analyse-/Metadata-Phase, nicht Trading-Entscheidung.

Pflicht

- keine automatische Strategieänderung
- keine Live-Param-Änderung
- nur metadata_json / GUI / Reports
- fail-soft bei fehlenden OHLCV

Inhalt

1. Post-Trade-Felder pro Closed Trade:
2. Entry zu früh? (Distance-to-recent-support / -breakout-trigger)
3. Exit sinnvoll? (war Preis nach Exit weiter / zurück)
4. MAE (Max Adverse Excursion)
5. MFE (Max Favorable Excursion)
6. Regime-passend? (regime-shift während Trade)
7. DCA hilfreich oder schädlich? (Vergleich gegen no-DCA-Sim)
8. Quality-Score rückblickend sinnvoll? (high-quality-trades vs PnL)
9. Grade A-F (operator wünscht numerische Felder behalten):
10. numeric_grade 0.0-1.0
11. letter_grade A/B/C/D/E/F (rein Display)
12. GUI:
13. neue Spalte in TradeLog: Grade-Pill (Color-Map)
14. Filter nach Grade
15. Aggregations-Widget: Grade-Verteilung letzten 7d / 30d

Stop-Gates F

- alle Felder werden geschrieben
- fail-soft funktioniert (kein Crash bei OHLCV-Gap)
- GUI rendert
- keine Live-Param-Änderung
- Unit-Tests + 1 Smoke-Run

Boundaries

- 0x automatische Strategieanpassung
- 0x DB-Mutation historischer Trades
- Operator-GO G0 PHASE F

13. G — EXEC-MODE-LABEL-3 Phase 3a

Ziel

Aufräum-Phase, nicht profitkritisch.

Scope

- Paper-Begriffe weiter entfernen wo seit EXEC-MODE-LABEL-2 noch Restbestände sind.
- Runtime-Labels bereinigen.
- 0x DB-Historie mutieren.
- 0x neue Semantik verstecken.

Stop-Gates G

- Code-Diff klein
- Tests grün
- Runtime stabil
- nur ausführen wenn die Woche stabil ist (kein Trade-Drama parallel)

14. H — T2-SOLANA-SHADOW Plan + REFACTOR-VS-REWRITE-PDF

Charakter

Reine Doku-Phase, kein Code, kein Trading.

Output 1: T2-SOLANA-SHADOW-PLAN.pdf

- T2-Architektur: Solana-Wallet, Helius-RPC, Pump.fun-Monitor (bereits vorhanden, aber als Signal-Only)
- T2-Execution-Plan: separate Wallet, separate Risk-Box, kein Mainnet vor Test
- Stop-Gates für T2-Aktivierung
- Boundaries: Wallet-Setup, Private-Key-Handling, Slippage-Caps

Output 2: REFACTOR-VS-REWRITE.pdf

- Bewertung: T1-Codebase weiter refactoren vs. T1 von scratch
- Empfehlung mit Pro/Contra
- Migrationsstrategie

Stop-Gates H

- PDFs publiziert
- Operator-Review
- kein Code

15. Stop-Gates (marktunabhängig formuliert)

Allgemeingültig für alle Phasen:

1. Synthetische Tests grün (Unit + Regression + relevante Smoke).
2. Runtime stabil (0 neue Tracebacks für mindestens 1h nach Cutover).
3. 3-Way MD5 Repo == Image == Container für betroffene Files.
4. Telemetry-Verfügbarkeit: relevante Felder werden geschrieben.
5. GUI rendert ohne Fehler (wo betroffen).
6. Wenn Live-Event im Beobachtungs-Fenster auftritt, wird es korrekt klassifiziert.
7. Kein erzwungener Trade nur für Verify.
8. Kein künstlicher Live-Trade.
9. Operator-Review bestanden (mündlich oder schriftlich dokumentiert).

Verboten als Stop-Gate:

- „min. 1 Reject in 24h“ — marktunabhängig
- „min. 1 Partial in 24h“ — marktunabhängig
- „mindestens 1 BEAR-DCA-Block live ausgelöst“ — marktunabhängig

Live-Events sind Bonus, nicht Pflicht.

16. Tests

Pro Phase Mindest-Coverage

Phase	Unit	Regression	Smoke	Runtime
A	✓ Mapping + Migration + T3-disable	✓ STRATEGY-GROUP-CONTRACT-1	✓ Bot+Worker	✓
B	✓ Symbol-Service + Reject-Reasons	✓ MS-Runner + Channels	✓ Mock-Exchange	✓
C1	✓ Sub-Scores + Combined + Metadata-Schema	✓ Decision-Log-Emit	—	✓ Shadow-Loop
D	✓ 4 Sub-Module einzeln + integration	✓ DCA + SL + Vol-Sizing	✓ BEAR-Sim	✓
C2	✓ Enforce-Logic pro Regime	✓ C1 ↔ C2 Symmetrie	—	✓
E1	✓ pro Exit-Kind	✓ Trailing-Konflikt	✓	✓
E2	✓ Partial-Cases + DCA-Mix + Restart	✓ PnL-Summen	✓ GUI-Render	✓
F	✓ Felder + fail-soft	✓ GUI-Filter	—	—
G	✓ Label-Replace	✓ keine Semantik-Drift	✓	✓
H	—	—	—	—

Pflicht-Test-Files (neue)

```
tests/test_strategy_group_contract_v3_1.py
tests/test_t3_forward_disabled.py
tests/test_binance_symbol_gate.py
tests/test_quality_score_shadow.py
tests/test_risk_guard_bear_dca.py
tests/test_risk_guard_sl_invariant.py
tests/test_risk_guard_vol_sizing.py
tests/test_quality_score_enforce.py
tests/test_exit_optimizer_e1.py
tests/test_partial_profit_e2.py
tests/test_post_trade_learning.py
```

Tests die unverändert grün bleiben müssen

- bestehende ~856 Safe-Runner-Tests
- HISTORY-1, COMMAND-BUS-CLOSE-* Tests
- STRATEGY-GROUP-CONTRACT-1 Tests (werden in A angepasst)
- DATA-LINK-1-FU2 Tests
- LABEL-1 Tests

17. Risk Register

ID	Risiko	Wahrscheinlichkeit	Impact	Mitigation
R-01	DB-Migration in Phase A fehlerhaft → blockierte Writes	mittel	hoch	dry-run + Rollback-SQL + Backup vor Migration
R-02	Quality-Enforce blockt 100% BUYs in BEAR	mittel	mittel	C1-Shadow-Daten als Threshold-Basis + 48h-Notbremse
R-03	BEAR-DCA-Block trifft auch sinnvolle Rescues	niedrig	mittel	Reclaim-Bedingung + Max-Loss-Sim als Override-Pfad
R-04	SL-Invariant blockiert ATR-Adjust nach News	niedrig	mittel	News-Adjust ist Tighten, kein Loosen → kein Konflikt
R-05	Partial-Profit-Logik erzeugt PnL-Doppelzählung	mittel	hoch	Datenvertrag E2 als PDF + Regressions vor Cutover
R-06	exchangeInfo-Cache stale → falsche Rejects	niedrig	niedrig	TTL 1h + Force-Refresh bei „symbol_unknown“-Reject
R-07	Telegram-Rate-Limit-Summary verbirgt echte Anomalie	niedrig	mittel	Realtime-Eskalation für <code>exchange_error</code> , <code>traceback</code>
R-08	T3-Forward-Disable bricht historische GUI-Filter	niedrig	niedrig	read-layer mappt <code>t3_*</code> -historische Werte auf <code>legacy_unknown</code>
R-09	Post-Trade-Learning Felder schreiben fehlschlagen	niedrig	niedrig	fail-soft, Wert bleibt NULL
R-10	Markt-Bedingungen erzeugen 0 BUYs während Phase D	hoch	niedrig	nicht als Bug werten — Stop-Gate ist marktunabhängig
R-11	MS-Runner-Aktivierung ungewollt durch Mapping-Change	niedrig	hoch	<code>MULTI_STRATEGY_ENABLED=false</code> bleibt, keine Aktivierung in v3.1
R-12	Operator-Decision-Gate übersprungen	niedrig	hoch	Phasen sind nicht chain-fähig ohne explizites <code>G0 PHASE X</code>

18. GO/NO-GO Empfehlung

Nach Operator-Review erwartete Freigabe:

```
G0 Phase A – TIER-ARCH-CONTRACT-1 (Mapping + Migration + T3-Disable)
G0 Phase B – T1-BINANCE-SYMBOL-GATE-1
G0 Phase C1 – T1-QUALITY-SCORE-SHADOW-1 (24–48h Beobachtung)
G0 Phase D – T1-RISK-GUARD-1
```

DECISION-GATE → Operator entscheidet:

- G0 Phase C2 (Enforce)
- oder Verlängerung Shadow
- oder Reihenfolge anpassen

```
G0 Phase E1 – Exit-Optimizer BE + Time-Stop
G0 Phase F – Post-Trade Learning
G0 Phase G – Exec-Mode-Label-3
G0 Phase H – Doku (T2-Shadow-Plan + Refactor-vs-Rewrite)
```

NO-GO (nicht freigegeben in v3.1):

```
T2 als Binance-Pump-Profil
MS-Runner als finales T2
Quality-Score sofort hart blockierend
Partial-Profit im selben Schritt wie Break-even
Solana-Wallet
Solana-Orders
CommandBus-v6 / MH-1
Mainnet
Leverage / Futures
T3-Copy-Trading Reaktivierung
```

19. STOP

Kein Code wird geschrieben, bevor v3.1 bestätigt ist und Operator `G0 PHASE A` gegeben hat.

Bot läuft weiter mit:

```
MULTI_STRATEGY_ENABLED=false
BINANCE_TESTNET=true
T2_SOLANA_EXECUTION=false
T3_COPY_TRADING=false
```

Stand 2026-05-16 Abend:

- Bot Container `c1awbot` healthy, host-PID 94
- Worker Container `c1awbot-worker` healthy
- letzter Commit master: `a6a629d` (STRATEGY-GROUP-CONTRACT-1)
- 18 Commits heute, keiner gepusht

Nächste Schritte

1. Operator liest v3.1 (Markdown oder PDF).
2. Bestätigt oder fordert Korrekturen für v3.2 an.
3. Bei Bestätigung: GO PHASE A startet Montag 2026-05-19 (oder nach Operator-Wunsch).
4. Bis dahin: 0 Code, 0 Cutover, 0 Migration.

Boundaries v3.1-Doku

- 0x Code
- 0x DB-Migration
- 0x DB-Writes
- 0x historische Updates
- 0x Bot-/Worker-/GUI-Restart
- 0x Strategieparameter-Änderung
- 0x Mainnet
- 0x Solana-Wallet
- 0x Private Keys
- 0x Solana-Orders
- 0x CommandBus-v6/MH-1-Aktivierung
- 0x Push ohne separates GO
- 0x Secrets
- 0x env dump
- 0x `docker compose config` mit Secrets

STOP.