

PLAN_T1_ROADMAP_2_WEEKS_v3.4

Status: Draft to-confirm **Erstellt:** 2026-05-17 **Vorgänger:** v3.3 (Draft 2026-05-16) **Korrekturen-Basis:** Operator-Urteil zur v3.3 (3 Punkte: A1-Stop-Gate-Konsistenz, environment-aware Binance-Gate, Pflicht-/Optional-Filter-Trennung) **Start frühestens:** Montag 2026-05-19 nach Operator-GO G0 PHASE A1

1. Executive Summary

Zweck: 2-Wochen-Plan für T1-Quality-Refactor. v3.4 schärft 3 Punkte aus v3.3:

- **A1-Stop-Gate** auf MS-Runner / T3-Forward-Bridge / Main-Scanner eingegrenzt — External-Channel-Pfade sind explizit ausgenommen und werden separat beobachtet bis Phase B sie übernimmt.
- **Phase B environment-aware:** bei `BINANCE_TESTNET=true` zählt Testnet exchangeInfo als Execution-Gate. Mainnet-public-listing nur als Metadata, niemals als Freigabe.
- **Pflicht- vs. optionale Binance-Filter:** `PRICE_FILTER`, `LOT_SIZE`, `MIN_NOTIONAL/NOTIONAL` = Pflicht. `PERCENT_PRICE`, `PERCENT_PRICE_BY_SIDE`, `ICEBERG_PARTS`, ... = optional, fehlende werden nur als Metadata geloggt, nicht als Reject.

Alles andere aus v3.3 bleibt unverändert: - T1 = Binance only, robust, Quality > Quantity - T2 = Solana Pump only, DEACTIVATED - T3 = Copy Trading only, DEACTIVATED - Quality erst Shadow, dann Enforce - Risk-Guard vor Quality-Enforce - Partial Profit separate Phase mit eigenem Datenvertrag - Stop-Gates marktunabhängig - DB-CHECK legacy-safe (`t2_pump_dump` bleibt allowed) - T3-Bridge Log-once Startup - Phase A2 entkoppelt (kein Blocker für B/C1/D) - External-Channel-Reklassifizierung erst in Phase B mit Pair-Gate

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

2. Was an v3.3 geändert wurde

| Bereich | v3.3 | v3.4 |
|-------------------------------------|--|---|
| A1 Stop-Gate Konsistenz | „0 neue <code>t2_pump_dump</code> Decision-/Trade-/Snapshot-Rows nach Cutover“ — kollidiert mit „External-Channels unverändert lassen“ | eingegrenzt auf MS-Runner / T3-Forward-Bridge / Main-Scanner-Mapping. External-Channel-Pfade bleiben bis Phase B unverändert und werden separat beobachtet |
| A1→B Reihenfolge | „direkt nach A1“ empfohlen | präziser: wenn External-Channel-Signale aktiv sind, B unmittelbar nach A1 ausführen , nicht lange warten |
| Binance-Gate Source-of-Truth | exchangeInfo allgemein | environment-aware: <code>BINANCE_TESTNET=true</code> → Testnet exchangeInfo zählt; Mainnet-Listing nur als Metadata <code>mainnet_listed: true/false</code> , kein Execution-Gate |
| Reject-Reason für Universum | nur <code>symbol_not_on_binance</code> | erweitert um <code>symbol_not_on_binance_testnet</code> (falls Coin nur Mainnet) |
| Filter-Behandlung | „ <code>PERCENT_PRICE_BY_SIDE</code> als Pflicht-Filter“ | Pflicht = <code>PRICE_FILTER</code> , <code>LOT_SIZE</code> , <code>MIN_NOTIONAL/NOTIONAL</code> . Optional = <code>PERCENT_PRICE</code> , <code>PERCENT_PRICE_BY_SIDE</code> , <code>ICEBERG_PARTS</code> , ... (Metadata-only, kein Reject bei Fehlen) |
| Risk Register | R-16/17/18 aus v3.3 | + R-19 (Mainnet-Listing als Execution-Freigabe missverstanden), R-20 (External-Channel-Pfad nach A1 noch nicht durch Gate) |

Alles andere bleibt identisch zu v3.3.

3. Neue Zielarchitektur

| | |
|------------------------|---|
| T1 = Binance only | ACTIVE (Spot, USDT, environment-aware execution gate) |
| T2 = Solana Pump only | DISABLED bis separate Shadow-Phase |
| T3 = Copy Trading only | DISABLED bis separater Plan |
| legacy_unknown | historical only – keine neuen Rows |
| t2_pump_dump | LEGACY – Read-only, keine neuen Writes |

3.1 T1 Binance-only — environment-aware

Execution-Gate Source-of-Truth:

Wenn `BINANCE_TESTNET=true`:
Pair muss in Binance Testnet exchangeInfo als TRADING + isSpotTradingAllowed stehen.
→ `reject_reason = "symbol_not_on_binance_testnet"` wenn nicht.

Wenn `BINANCE_TESTNET=false` (Mainnet explizit erlaubt – derzeit nicht):
Pair muss in Binance Spot Mainnet exchangeInfo als TRADING stehen.
→ `reject_reason = "symbol_not_on_binance"` wenn nicht.

Mainnet-public-listing darf zusätzlich als Metadata geloggt werden
(`mainnet_listed: true/false`), aber niemals als Testnet-Execution-Freigabe
verwendet werden.

Pflicht-Filter (Reject bei Fehlen):

- `PRICE_FILTER` (tickSize)
- `LOT_SIZE` (stepSize, minQty)
- `MIN_NOTIONAL` oder `NOTIONAL` (mind. eine der beiden Varianten muss vorhanden sein)

Optional-Filter (nur Metadata, kein Reject bei Fehlen):

- `PERCENT_PRICE`
- `PERCENT_PRICE_BY_SIDE`
- `ICEBERG_PARTS`
- `MARKET_LOT_SIZE`
- `MAX_NUM_ORDERS`
- `MAX_NUM_ALGO_ORDERS`
- `TRAILING_DELTA`

Wenn ein optionaler Filter vorhanden ist, wird er in `decision_logs.metadata_json.binance_pair_check.optional_filters` geloggt. Wenn er fehlt, ist das kein Reject.

Quote-Asset: USDT (Phase 1 only).

Nicht erlaubt in T1: Solana-Fallback, T2-Fallback, DEX-Trading, Non-USDT-Quotes.

3.2 T2 Solana-only — nur reserviert

Unverändert zu v3.3: Code-Konstante reserviert, DB-Constraint ab A2 erlaubt den Wert, kein Code-Pfad schreibt ihn. GUI höchstens Disabled-Indicator.

3.3 T3 Copy-Trading-only

Unverändert zu v3.3: deaktiviert, Bridge in A1 archiviert mit Log-once.

4. Korrigierte Phasenreihenfolge

| Phase | Name | Aufwand | Block-Status | Reihenfolge |
|-------|---|---------|---|--|
| A1 | TIER-ARCH-CONTRACT-1-CODE | 0.5 d | Mapping + T3-Bridge archive + GUI-Disable-Indicator (KEINE External-Reklass) | zuerst |
| B | T1-BINANCE-SYMBOL-GATE-1 | 1.0 d | environment-aware Hard-Gate + External-Channel-Reklassifizierung mit Pair-Check | wenn External-Channels aktiv sind: direkt nach A1 |
| C1 | T1-QUALITY-SCORE-SHADOW-1 | 1.5 d | log-only, 24-48h | nach B |
| D | T1-RISK-GUARD-1 | 2.0 d | BEAR-DCA-Block, SL-Invariant, Vol-Sizing | nach C1-Start (parallel ok) |
| C2 | T1-QUALITY-SCORE-ENFORCE-1 | 1.0 d | nach C1-Auswertung | nach Decision-Gate |
| E1 | T1-EXIT-OPTIMIZER-BE-TIMESTOP-1 | 1.0 d | BE + Trailing-Gate + Time-Stop | nach D |
| E2 | T1-PARTIAL-PROFIT-1 | 2.0 d | separater Plan + Datenvertrag | nach E1 |
| F | T1-POST-TRADE-LEARNING | 1.0 d | Analyse-Only | nach E1 oder E2 |
| G | EXEC-MODE-LABEL-3 Phase 3a | 0.5 d | Cleanup, profitneutral | wenn Woche stabil |
| H | REFACTOR-VS-REWRITE-PDF + T2-SOLANA-SHADOW-PLAN-PDF | 0.5 d | reine Doku | jederzeit |
| A2 | STRATEGY-GROUP-DB-CONSTRAINT-COMPAT | 0.5 d | DB-CHECK legacy-safe erweitern | entkoppelt — Operator-Empfehlung: kurz vor T2-SOLANA-SHADOW, nicht vorher nötig |

Empfohlene Reihenfolge v3.4

A1 → B → C1 → D → DECISION-GATE → C2 → E1 → E2 → F → G → H

A2: kurz vor T2-SOLANA-SHADOW. Nicht zwingend früher.

Wichtig: A1 und B sollten zeitlich nah aneinanderliegen, solange External-Channel-Signale aktiv sind, weil A1 diese Pfade nicht anfasst und B die kanonische Reklassifizierung mit Pair-Gate einführt.

Mandatory Decision-Gate

Unverändert zu v3.3: nach Phase D Operator-Decision-Gate (Option A weiter mit C2, Option B Shadow verlängern, Option C Reihenfolge anpassen). Kein automatischer Übergang.

5a. Phase A1 — TIER-ARCH-CONTRACT-1-CODE

Ziel

Code-/Mapping-Seite der neuen Zielarchitektur durchziehen, **ohne** DB-Touch. **Keine** External-Channel-Reklassifizierung — das passiert erst in Phase B mit Binance-Pair-Gate.

Beziehung zu STRATEGY-GROUP-CONTRACT-1

Unverändert zu v3.3:

STRATEGY-GROUP-CONTRACT-1 (commit `a6a629d` , 2026-05-16) hat das Mapping `breakout / volatility_sweep` → `t2_pump_dump` eingeführt. Diese Klassifikation wird durch TIER-ARCH-CONTRACT-1 semantisch ersetzt, weil T2 in der neuen Zielarchitektur **Solana-only** ist und der MS-Runner Binance handelt.

Scope A1

- 1. `trading/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, # v3.4 – geändert von T2_PUMP_DUMP
    "volatility_sweep":   STRATEGY_GROUP_T1, # v3.4 – geändert von T2_PUMP_DUMP
}
DEFAULT_STRATEGY_GROUP = STRATEGY_GROUP_T1

# v3.4 – Reservation, kein Write-Pfad
STRATEGY_GROUP_T2_SOLANA_PUMP = "t2_solana_pump"

```

1. T3-Forward-Bridge archivieren mit Log-once:

```

_T3_BRIDGE_ARCHIVE_LOGGED = False

def _process_t3_forwarded_signals(...):
    global _T3_BRIDGE_ARCHIVE_LOGGED
    if not _T3_BRIDGE_ARCHIVE_LOGGED:
        logger.info(
            "T3 forward bridge archived / disabled "
            "(TIER-ARCH-CONTRACT-1 v3.4) – no further forwards processed"
        )
        _T3_BRIDGE_ARCHIVE_LOGGED = True
    return

```

1. External-Channel-Inputs bleiben unverändert in A1:

Solange Phase B nicht aktiv ist, behalten External-Channel-Signale (Telegram-Signal-Channels, Discord später) ihre bisherige Klassifikation — auch wenn sie historisch als `t2_pump_dump` getaggt waren. **Erst Phase B definiert die kanonische Reklassifizierung mit Binance-Pair-Check als notwendiger Bedingung.**

1. **GUI-Disable-Indicator** für T2:
2. statisches Element: „T2 Solana Pump — disabled, siehe T2-SOLANA-SHADOW-Plan“
3. keine KPI-Cards, keine T2-Trade-Listen, keine T2-Charts
4. STRATEGY-GROUP-CONTRACT-1 Tests werden angepasst.

Files A1

- `trading/strategies/strategy_group_map.py`
- `trading/strategies/multi_strategy_runner.py` (Tests anpassen)
- `trading/main.py` (T3-Bridge log-once)
- `gui/app/Filament/...` (T2-Disable-Indicator)
- Tests: `test_strategy_group_contract_v3_4.py`, `test_t3_bridge_log_once.py`, `test_no_new_t2_pump_dump_writes.py`

Stop-Gates A1 (v3.4 präzisiert)

1. 3-Way MD5 Repo == Image == Container für betroffene Files.
2. Bot-Restart ohne Traceback.
3. T3-Bridge-Log genau 1× pro Bot-Startup.
4. 0 neue `t2_pump_dump` Rows aus
 - MS-Runner (`decision_logs`, `trade_logs`, `position_snapshots`)
 - T3-Forward-Bridge (archived → keine neuen Rows von hier)
 - Main-Scanner-Mapping (`strategy_group` default)
5. External-Channel-Pfade unverändert. Werden NICHT als Verletzung von Stop-Gate 4 gewertet, aber separat beobachtet
 - Telegram-Signal-Channel-Inputs
 - Discord-Inputs (später)
 - falls Channel-Code bisher `t2_pump_dump` schrieb, bleibt das vorerst möglich
 - finale Reklassifizierung passiert in Phase B
6. GUI rendert T2-Disable-Indicator.
7. Regressionstests grün.

Boundaries A1

- 0× DB-Migration
- 0× DB-Mass-Mutation
- 0× Mainnet
- 0× T2-Solana-Code
- 0× External-Channel-Reklassifizierung
- 0× Modification an Channel-Code-Pfaden vor Phase B

- Operator-GO G0 PHASE A1

Empfehlung Phase-Übergang A1 → B

Wenn External-Channel-Signale aktiv sind (Telegram-Bot empfängt Signals):
→ Phase B unmittelbar nach A1 ausführen.

Wenn External-Channels gerade nicht aktiv sind:
→ A1 darf vor B als Mini-Cutover laufen, B folgt geplant.

Beobachten zwischen A1 und B:

- tägliche Decision-/Trade-Log-Abfrage:
SELECT strategy_group, source, COUNT(*)
FROM decision_logs
WHERE created_at > '<A1_cutover>'
GROUP BY 1,2;
- wenn neue t2_pump_dump Rows aus Channel-Source auftauchen:
→ Phase B priorisieren, kein Bug

5b. Phase A2 — STRATEGY-GROUP-DB-CONSTRAINT-COMPAT (entkoppelt)

Status

Entkoppelt — kein Blocker für B/C1/D/E1/E2/F/G/H. **Operator-Empfehlung v3.4:** A2 kurz vor T2-SOLANA-SHADOW ausführen, nicht vorher nötig.

SQL (legacy-safe, unverändert)

```
ALTER TABLE decision_logs DROP CONSTRAINT IF EXISTS decision_logs_strategy_group_check;
ALTER TABLE decision_logs ADD CONSTRAINT decision_logs_strategy_group_check
CHECK (strategy_group IN (
  't1_core',
  't2_pump_dump',      -- LEGACY: Reads bleiben, keine neuen Writes
  't2_solana_pump',    -- RESERVED bis T2-SOLANA-SHADOW
  't3_copy_trading',
  'legacy_unknown'
));
-- analog trade_logs, position_snapshots
```

Was A2 nicht macht

- Keine UPDATE -Mass-Mutation
- Kein Drop von Legacy-Werten
- Keine historische Datenmutation

Migration-Pflicht

Unverändert: Dry-run, Backup, Rollback-SQL, Add-only.

Tests A2

```
tests/test_strategy_group_db_constraint_v3_4.py
- insert t1_core, t2_pump_dump, t2_solana_pump, t3_copy_trading, legacy_unknown → ok
- insert 'random_value' → fail
```

Stop-Gates A2

- Dry-run grün
- Add-only Migration fehlerfrei
- Bot/Worker können nach Migration weiter schreiben
- Legacy-Reads funktionieren weiter
- Rollback-SQL liegt bereit

Boundaries A2

- 0x historische DB-Mutation
- 0x Drop von Legacy-Werten
- Operator-GO G0 PHASE A2 separat

6. Phase B — T1-BINANCE-SYMBOL-GATE-1 (environment-aware)

Ziel

Hartes Gate: jedes Signal an T1 muss im **aktuellen Binance-Universum** (Testnet wenn `BINANCE_TESTNET=true`, sonst Mainnet) handelbar sein. **Plus:** External-Channel-Signale werden hier kanonisch reklassifiziert — nur wenn Gate positiv.

Scope B (v3.4 environment-aware)

1. Symbol-Service (`trading/exchanges/binance_symbol_service.py`):

```
def is_t1_tradable(symbol):
    is_testnet = os.getenv("BINANCE_TESTNET", "false").lower() == "true"
    universe = "testnet" if is_testnet else "mainnet"

    info = exchange_info_for(universe).get(symbol)
    if info is None:
        rr = "symbol_not_on_binance_testnet" if is_testnet else "symbol_not_on_binance"
        return False, rr, {"universe": universe}

    if info["status"] != "TRADING" or not info["isSpotTradingAllowed"]:
        return False, "binance_pair_inactive", {"universe": universe, "status": info["status"]}

    if info["quoteAsset"] != "USDT":
        return False, "binance_quote_not_usdt", {"universe": universe, "quote": info["quoteAsset"]}

    filters_by_type = {"filterType": f for f in info["filters"]}

    # MANDATORY
    if "PRICE_FILTER" not in filters_by_type:
        return False, "price_filter_missing", {}
    if "LOT_SIZE" not in filters_by_type:
        return False, "lot_size_missing", {}
    notional = filters_by_type.get("MIN_NOTIONAL") or filters_by_type.get("NOTIONAL")
    if notional is None:
        return False, "min_notional_missing", {}

    # OPTIONAL (metadata only)
    optional_present = {
        ft: filters_by_type[ft]
        for ft in ("PERCENT_PRICE", "PERCENT_PRICE_BY_SIDE", "ICEBERG_PARTS",
                  "MARKET_LOT_SIZE", "MAX_NUM_ORDERS", "MAX_NUM_ALGO_ORDERS",
                  "TRAILING_DELTA")
        if ft in filters_by_type
    }

    meta = {
        "universe": universe,
        "mainnet_listed": is_listed_on_mainnet(symbol), # nur zur Info
        "tick_size": filters_by_type["PRICE_FILTER"]["tickSize"],
        "step_size": filters_by_type["LOT_SIZE"]["stepSize"],
        "min_notional": notional.get("minNotional") or notional.get("notional"),
        "optional_filters": list(optional_present.keys()),
    }
    return True, None, meta
```

- exchangeInfo-Cache TTL 1h, getrennt pro Universum (`testnet` vs. `mainnet`)
- Mainnet-exchangeInfo wird **nur** für `mainnet_listed` -Metadata gelesen, nicht für Execution
- Force-Refresh-Hook: bei Reject `symbol_not_on_binance*` darf einmalig refresh versucht werden, dann Cache aktualisieren
- **Reject-Reason-Vocabulary** (v3.4 erweitert):

| | |
|--|-----------------------------|
| <code>symbol_not_on_binance</code> | (Mainnet-Pfad) |
| <code>symbol_not_on_binance_testnet</code> | (Testnet-Pfad, neu in v3.4) |
| <code>binance_pair_inactive</code> | |
| <code>binance_quote_not_usdt</code> | |
| <code>price_filter_missing</code> | (mandatory, neu in v3.4) |
| <code>lot_size_missing</code> | (mandatory, neu in v3.4) |
| <code>min_notional_missing</code> | (mandatory) |
| <code>permission_failed</code> | |

Hinweis: `PERCENT_PRICE_BY_SIDE` und Co. erzeugen **keinen** Reject mehr — werden nur als Metadata geloggt.

1. **Wiring:**
2. MS-Runner `_attempt_execution` ruft Gate vor allen anderen Checks
3. External-Channel-Inputs (Telegram-Signal-Channels, Discord später) gehen jetzt durch dieselbe Funktion:

```
def classify_external_signal(signal):
    symbol = signal["symbol"]
    ok, reject_reason, meta = is_t1_tradable(symbol)
    if ok:
        return {"strategy_group": "t1_core", "tradable": True, "binance_pair_check": meta}
    return {
        "strategy_group": "legacy_unknown",
        "tradable": False,
        "reject_reason": reject_reason,
        "binance_pair_check": meta,
    }
```

- **Nicht-Binance-Signale gehen NICHT auf t1_core.** Sie landen als `legacy_unknown` und werden nicht an MS-Runner-Execution weitergegeben.
- Mainnet-only Coins (im Testnet-Betrieb) sind auch `legacy_unknown` mit Reject `symbol_not_on_binance_testnet`, aber `mainnet_listed: true` in der Metadata.
- `decision_logs.metadata_json` :

```
{
  "binance_pair_check": {
    "universe": "testnet",
    "mainnet_listed": true,
    "tick_size": "0.00010000",
    "step_size": "0.10000000",
    "min_notional": "5.00000000",
    "optional_filters": ["PERCENT_PRICE_BY_SIDE", "MAX_NUM_ORDERS"]
  },
  "external_channel_source": "telegram_kw_signals",
  "external_reclass_applied": true,
  "reject_reason": null
}
```

- Bei Reject zusätzlich: `reject_reason` und optional Reject-spezifische Details.
- **Tests B:**

```
tests/test_binance_symbol_gate.py
- mock testnet exchangeInfo
- alle Pflicht-Reject-Reasons
- missing optional filter → kein Reject, nur metadata
- mainnet_listed=true wenn nur Mainnet → reject symbol_not_on_binance_testnet
- cache TTL respected
- mainnet_listed Metadata ist NIE ausreichend für tradable=True im Testnet-Modus

tests/test_external_channel_reclass.py
- Binance-Pair positiv → t1_core + external_reclass_applied=true
- nicht-Binance → legacy_unknown + reject_reason gesetzt
- mainnet-only Coin im Testnet → legacy_unknown + mainnet_listed=true Metadata
- kein Pfad emittiert t2_pump_dump
```

Keine echten Orders in Tests, Testnet/Mainnet strikt getrennt.

Stop-Gates B

- exchangeInfo-Cache funktioniert pro Universum
- alle Pflicht-Reject-Reasons in Unit-Tests grün
- Optional-Filter-Fehlen erzeugt KEINEN Reject
- External-Channel-Reclass-Logic in Unit-Tests grün (beide Pfade)
- Mainnet-Listing wird NICHT als Execution-Freigabe behandelt (regression test)
- Runtime stabil
- `metadata_json.binance_pair_check.universe` korrekt
- `metadata_json.external_reclass_applied` bei External-Signalen vorhanden
- Live-Event nicht zwingend

Boundaries B

- 0x echte Orders in Tests

- 0x Mainnet-API-Aufrufe für Execution (nur für `mainnet_listed`-Metadata)
- 0x Mutation an Binance-Markets-Cache während laufendem Bot
- 0x Umschalten von `BINANCE_TESTNET=true` → `false` durch B (Mainnet bleibt verboten)
- Operator-GO GO PHASE B

7. C1 — T1-QUALITY-SCORE-SHADOW-1

Unverändert zu v3.3 §7. Sub-Scores werden berechnet und in `decision_logs.metadata_json` geloggt, ohne zu blockieren. 24-48h Beobachtung.

8. Phase D — T1-RISK-GUARD-1

Sub-Module unverändert zu v3.3 §8 (BEAR-DCA-Block, No-DCA-without-Reclaim, Max-Loss-Simulation, SL-Invariant, Vol-Scaled Sizing, Telegram-Rate-Limit).

Boundaries D (v3.3-Formulierung beibehalten)

0x bestehendes Strategieparameter-Tuning
0x ATR-/Trailing-/Trigger-Parameter ändern, die nicht ausdrücklich Teil dieser Phase sind
0x Mainnet

ABER:
Phase D ändert bewusst Trading-Logik (BEAR-DCA-Block, SL-Invariant, Vol-Sizing) und braucht deshalb explizites Operator-GO PHASE D.

Diese Änderungen sind keine versteckte Parameter-Anpassung, sondern dokumentierte Risk-Guard-Logik mit Tests, Stop-Gates und Rollback-Möglichkeit.

9. C2 — T1-QUALITY-SCORE-ENFORCE-1

Unverändert zu v3.3 §9.

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

Unverändert zu v3.3 §10.

11. E2 — T1-PARTIAL-PROFIT-1

Unverändert zu v3.3 §11.

12. F — T1-POST-TRADE-LEARNING

Unverändert zu v3.3 §12.

13. G — EXEC-MODE-LABEL-3 Phase 3a

Unverändert zu v3.3 §13.

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

Unverändert zu v3.3 §14.

15. Stop-Gates (marktunabhängig)

Unverändert zu v3.3 §15:

1. Synthetische Tests grün.
2. Runtime stabil 1h nach Cutover.
3. 3-Way MD5 Repo == Image == Container.
4. Telemetry-Verfügbarkeit.
5. GUI rendert ohne Fehler.
6. Wenn Live-Event auftritt, korrekt klassifiziert.
7. Kein erzwungener Trade nur für Verify.
8. Kein künstlicher Live-Trade.
9. Operator-Review bestanden.

16. Tests

Neue/aktualisierte Test-Files in v3.4

| | |
|---|---|
| tests/test_strategy_group_contract_v3_4.py | (A1) |
| tests/test_t3_bridge_log_once.py | (A1) |
| tests/test_no_new_t2_pump_dump_writes.py | (A1 – scope eingegrenzt auf MS-Runner / T3-Bridge / Main-Scanner) |
| tests/test_external_channel_unchanged_in_a1.py | (A1 – verifiziert dass Channel-Code-Pfad nicht angefasst wird) |
| tests/test_strategy_group_legacy_reads.py | |
| tests/test_strategy_group_db_constraint_v3_4.py | (A2, entkoppelt) |
| tests/test_binance_symbol_gate.py | (B – testnet vs mainnet, mandatory vs optional filters) |
| tests/test_binance_mainnet_listing_metadata_only.py | (B – Regression: mainnet-listing NICHT als execution-gate) |
| tests/test_external_channel_reclass.py | (B) |
| tests/test_quality_score_shadow.py | (C1) |
| tests/test_risk_guard_bear_dca.py | (D) |
| tests/test_risk_guard_sl_invariant.py | (D) |
| tests/test_risk_guard_vol_sizing.py | (D) |
| tests/test_quality_score_enforce.py | (C2) |
| tests/test_exit_optimizer_e1.py | (E1) |
| tests/test_partial_profit_e2.py | (E2) |
| tests/test_post_trade_learning.py | (F) |

Test-Coverage pro Phase

| Phase | Unit | Regression | Smoke | Runtime |
|-----------------|---|--------------------------------------|---------------------|---------|
| A1 | ✓ Mapping + T3-Bridge log-once + no-new-t2_pump_dump (MS/T3/Scanner) + External-Channel-Pfad unverändert | ✓ STRATEGY-GROUP-CONTRACT-1 → legacy | ✓ Bot+Worker | ✓ |
| A2 (entkoppelt) | ✓ Constraint accepts 5 Werte + rejects random | ✓ Legacy-Reads | ✓ Migration dry-run | ✓ |
| B | ✓ Symbol-Service environment-aware + Pflicht-Filter + Optional-Filter-Metadata + External-Reclass + mainnet-listing-NOT-execution-gate | ✓ MS-Runner + Channels | ✓ Mock-Exchange | ✓ |
| C1 | ✓ Sub-Scores + Metadata | ✓ Decision-Log-Emit | — | ✓ |
| D | ✓ 4 Sub-Module | ✓ DCA + SL + Vol-Sizing | ✓ BEAR-Sim | ✓ |
| C2 | ✓ Enforce-Logic pro Regime | ✓ C1 ↔ C2 Symmetrie | — | ✓ |
| E1 | ✓ pro Exit-Kind | ✓ Trailing-Konflikt | ✓ | ✓ |
| E2 | ✓ Partial-Cases | ✓ PnL-Summen | ✓ GUI | ✓ |
| F | ✓ Felder + fail-soft | ✓ GUI-Filter | — | — |
| G | ✓ Label-Replace | ✓ keine Semantik-Drift | ✓ | ✓ |
| H | — | — | — | — |

Bestehende Tests die grün bleiben müssen

Unverändert zu v3.3: ~856 Safe-Runner-Tests + HISTORY-1, COMMAND-BUS-CLOSE-*, STRATEGY-GROUP-CONTRACT-1, DATA-LINK-1-FU2, LABEL-1.

17. Risk Register

| ID | Risiko | Wahrscheinlichkeit | Impact | Mitigation |
|------|---|--------------------|----------------|--|
| R-01 | DB-Migration bricht historische Rows | niedrig | niedrig | legacy-safe Constraint |
| R-02 | Quality-Enforce blockt 100% BUYS in BEAR | mittel | mittel | C1-Daten + 48h-Notbremse |
| R-03 | BEAR-DCA-Block trifft sinnvolle Rescues | niedrig | mittel | Reclaim + Max-Loss-Override |
| R-04 | SL-Invariant blockiert ATR-Adjust nach News | niedrig | mittel | News-Adjust ist Tighen |
| R-05 | Partial-Profit erzeugt PnL-Doppelzählung | mittel | hoch | E2-Datenvertrag-PDF |
| R-06 | exchangeInfo-Cache stale | niedrig | niedrig | TTL 1h + Force-Refresh |
| R-07 | Telegram-Rate-Limit verbirgt Anomalie | niedrig | mittel | Realtime-Eskalation für errors |
| R-08 | T3-Forward-Disable bricht GUI-Filter | niedrig | niedrig | Read-Layer mapping |
| R-09 | Post-Trade-Learning Felder fail | niedrig | niedrig | fail-soft |
| R-10 | Markt erzeugt 0 BUYS während D | hoch | niedrig | Stop-Gate marktunabhängig |
| R-11 | MS-Runner-Aktivierung ungewollt | niedrig | hoch | <code>MULTI_STRATEGY_ENABLED=false</code> |
| R-12 | Operator-Decision-Gate übersprungen | niedrig | hoch | Phasen nicht chain-fähig |
| R-13 | T3-Bridge-Log spammt pro Scan | niedrig | niedrig | Module-Var Guard |
| R-14 | GUI suggeriert T2 läuft | niedrig | mittel | nur Disabled-Indicator |
| R-15 | STRATEGY-GROUP-CONTRACT-1-Test bricht | mittel | niedrig | CONTRACT-1.legacy |
| R-16 | External-Channel-Signal als t1_core ohne Gate | niedrig | niedrig | External-Reklass erst in B |
| R-17 | A2 wird zwingend benötigt vor T1-Verbesserungen | niedrig | niedrig | A2 entkoppelt |
| R-18 | Phase D als reine Param-Anpassung missverstanden | niedrig | mittel | Boundary präzise formuliert |
| R-19 | Mainnet-Listing fälschlich als Testnet-Execution-Freigabe | niedrig | hoch | Symbol-Service environment-aware + Regression-Test <code>mainnet_listing_NOT_execution_gate</code> |
| R-20 | External-Channel-Pfad erzeugt zwischen A1 und B weiter t2_pump_dump Rows | mittel | niedrig | Stop-Gate A1 scope eingegrenzt + Empfehlung A1→B direkt, separate Beobachtungs-SQL dokumentiert |
| R-21 | Optional Binance-Filter (PERCENT_PRICE_BY_SIDE) führt zu False-Reject | niedrig | mittel | Pflicht-/Optional-Trennung explizit, Test <code>missing_optional_filter_is_not_reject</code> |

18. GO/NO-GO Empfehlung

Freigegeben in v3.4

G0 v3.4 Roadmap als Arbeitsgrundlage

G0 Phase A1 – TIER-ARCH-CONTRACT-1-CODE
 (Mapping + T3-Bridge log-once + GUI-Disable-Indicator)
 (KEINE External-Channel-Reklass)
 (Stop-Gate auf MS-Runner/T3-Bridge/Main-Scanner eingegrenzt)

G0 Phase B – T1-BINANCE-SYMBOL-GATE-1
 (environment-aware Testnet-First-Gate)
 (External-Reclass mit Pair-Check)
 (mandatory vs optional filters)
 (mainnet-listing nur als Metadata)
 EMPFEHLUNG: B direkt nach A1 wenn External-Channels aktiv

G0 Phase C1 – T1-QUALITY-SCORE-SHADOW-1

G0 Phase D – T1-RISK-GUARD-1

DECISION-GATE nach D → Operator entscheidet:

- G0 Phase C2 (Enforce)
- Verlängerung Shadow
- 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

Entkoppelt

G0 Phase A2 – STRATEGY-GROUP-DB-CONSTRAINT-COMPAT
 Empfehlung v3.4: kurz vor T2-SOLANA-SHADOW
 nicht vor B/C1/D nötig

Noch nicht freigegeben

C2 Quality-Enforce (wartet auf Decision-Gate nach D)
 E2 Partial-Profit (wartet auf E1 + Datenvertrag-PDF)
 T2 Solana
 T3 Copy-Trading
 CommandBus-v6 / MH-1
 Mainnet
 Solana-Wallet / Private-Keys
 Leverage / Futures
 T2 als Binance-Pump-Profil
 MS-Runner als finales T2
 Quality-Score sofort hart blockierend
 Partial-Profit im selben Schritt wie Break-even
 GUI-KPI-Cards für T2-Solana solange disabled
 Drop von t2_pump_dump aus DB-Constraint
 historische DB-Mass-Mutation
 External-Channel-Signale auf t1_core ohne Binance-Pair-Check
 Mainnet-Listing als Testnet-Execution-Freigabe
 PERCENT_PRICE_BY_SIDE als harte Pflicht

19. STOP

Kein Code vor v3.4-Bestätigung und Operator- G0 PHASE A1 .

Bot läuft weiter mit:

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

Stand 2026-05-17 früh:

- Bot Container `c1awbot` healthy, host-PID 94
- Worker Container `c1awbot-worker` healthy
- letzter Commit master: `a6a629d` (STRATEGY-GROUP-CONTRACT-1)
- 18 Commits Stand 2026-05-16 abends, keiner gepusht

Nächste Schritte

1. Operator liest v3.4 (Markdown oder PDF).
2. Bestätigt oder fordert v3.5 an.
3. Bei Bestätigung: `G0 PHASE A1` startet.
4. Empfohlen: Phase B direkt nach A1 wenn External-Channels aktiv (Sicherheitsanker).
5. A2 wartet bis kurz vor T2-SOLANA-SHADOW.
6. Bis zur Bestätigung: 0 Code, 0 Cutover, 0 Migration.

Boundaries v3.4-Doku

- 0x Code
- 0x DB-Migration
- 0x DB-Writes
- 0x historische Updates
- 0x Bot-/Worker-/GUI-Restart
- 0x Strategieparameter-Änderung
- 0x Mainnet
- 0x Mainnet-API-Aufrufe für Execution
- 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.