

# REGIME-AWARE-PARAMS-1 — Plan-Only

**Datum:** 2026-06-05 **Modus:** PLAN-ONLY · 0× Code · 0× Push · 0× Orders · 0× Runtime-Apply **Trigger:** 180-d-Replay (MS-STRATEGY-REPLAY-20COIN-BACKTEST-1) zeigt **RANGE +26 %**, **WEAK\_TREND −30 %**, **BULL\_STRONG −9 %** **Heutiger XLM-SL-Loss (−11.17 USDT in BEAR-Regime)** ist ein Live-Beispiel des nicht-regulierten Risikos

## 1. Ziel

Per-Regime-Verhalten der MS-Pipeline einführen, um die paradoxe Eigenschaft („trend\_follow funktioniert nur in RANGE, verliert in Trends“) strukturell zu reparieren — **ohne** Coin-Allowlist/Denylist, **ohne** Exit-Parameter-Touch.

Drei Phasen: - P1.A — Shadow-Logger (log-only) - P1.B — Hard-Gate (erst nach 2-4 Wochen Shadow-Daten) - P1.C — Per-Regime-Params (optional, nach P1.B-Pilot stabil)

**Diese Plan-Phase deckt P1.A ab und bereitet die Regeln für P1.B vor.**

## 2. Existing Infrastructure (recon)

Im Bot existieren bereits: - MarketRegime Detector (5 Labels: STRONG\_TREND, WEAK\_TREND, RANGE, HIGH\_VOLATILITY, CHOP) - StrategyRouter Hard-Blocks (CHOP + HIGH\_VOL ohne base bereits blockiert) - Per-Regime-Strategy-Activation (STRONG\_TREND→trend\_follow, WEAK\_TREND→breakout+trend+oversold, RANGE→mean+vwap+vol\_sweep+oversold) - Decision-Log Regime im decision\_logs

→ **Regime-Detection ist bereits vollständig.** Was fehlt ist die regime-spezifische Edge-Filter-Schicht zwischen Router und Final-Decision.

## 3. Empirische Datenbasis (180d-Replay)

Regime	N	WR	Exp %	Total Net %	Verdict
STRONG_TREND	23	35 %	−0.37	−9	TOXISCH
WEAK_TREND	207	43 %	−0.15	−30	HOCH-TOXISCH
<b>RANGE</b>	<b>261</b>	<b>48 %</b>	<b>+0.10</b>	<b>+26</b>	<b>EDGE positiv</b>
HIGH_VOLATILITY	klein	−	−	−	gedämpft via Router
CHOP	0	−	−	−	bereits blockiert

**Kern-Befund:** Die MS- trend\_follow -Strategie ist eigentlich ein RANGE-Breakout-Detektor.

## 4. Cross-Impact-Matrix

Changed Area	Direct Effect	Downstream	Follow-up
multi_strategy_runner.py	neue _compute_regime_gate_shadow()	nur MS-Runner	Tests
multi_strategy_runner.py	metadata regime_gate	decision_logs, JSONL	KEIN DB-Migration
settings.py	6 neue Settings	nur MS-Runner	safe Defaults
position_manager.py, live_trade.py, trader_core.py	<b>0</b>	−	−
Strategy-Subklassen, Router, MarketRegime	<b>0</b>	−	−

**Mainnet/MS-Live:** unverändert.

## 5. Regime-Gate-Logik

### Klassifikation (Lookup)

Bot-Regime	base_regime	Class	Default Action
RANGE	−	range safe	ALLOW

WEAK_TREND	-	weak_trend_toxic	HARD_REJECT (P1.B)
STRONG_TREND	-	strong_trend_caution	SOFT_REJECT (P1.B)
HIGH_VOL	RANGE	high_vol_range_caution	ALLOW
HIGH_VOL	trend	high_vol_trend_toxic	HARD_REJECT
CHOP	-	chop	bereits durch Router

## Gate-Mode Wirkung

Mode	Toxic	Caution	Safe
shadow (Default)	log, allow	log, allow	allow
soft	reject mit score-Penalty	reject mit score-Penalty	allow
hard	reject	reject (low conf)	allow

## 6. Settings (NEU)

```

MS_REGIME_GATE_ENABLED:      bool = os.getenv('MS_REGIME_GATE_ENABLED', 'true').lower()=='true'
MS_REGIME_GATE_MODE:         str  = os.getenv('MS_REGIME_GATE_MODE', 'shadow')
MS_REGIME_GATE_BLOCK_WEAK_TREND: bool = os.getenv('MS_REGIME_GATE_BLOCK_WEAK_TREND', 'true').lower()=='true'
MS_REGIME_GATE_BLOCK_STRONG_TREND: bool = os.getenv('MS_REGIME_GATE_BLOCK_STRONG_TREND', 'false').lower()=='true'
MS_REGIME_GATE_BLOCK_HIGH_VOL_TREND: bool = os.getenv('MS_REGIME_GATE_BLOCK_HIGH_VOL_TREND', 'true').lower()=='true'
MS_REGIME_GATE_SOFT_SCORE_PENALTY: float = float(os.getenv('MS_REGIME_GATE_SOFT_SCORE_PENALTY', '1.5'))

```

**Cutover-Default:** enabled=true, mode=shadow (log-only).

## 7. Code-Spec

### 7.1 multi\_strategy\_runner.\_compute\_regime\_gate\_shadow(regime, decision)

```

def compute_regime_gate_shadow(self, regime, decision):
    s = self.settings
    if not s.MS_REGIME_GATE_ENABLED: return {'enabled': False}
    rg = regime.get('regime'); bg = regime.get('base_regime')
    if rg == 'RANGE': cls = 'range_safe'
    elif rg == 'WEAK TREND':
        cls = 'weak trend toxic' if s.MS_REGIME_GATE_BLOCK_WEAK_TREND else 'weak_trend_caution'
    elif rg == 'STRONG TREND':
        cls = 'strong trend toxic' if s.MS_REGIME_GATE_BLOCK_STRONG_TREND else 'strong_trend_caution'
    elif rg == 'HIGH VOLATILITY' and bg == 'RANGE':
        cls = 'high vol range caution'
    elif rg == 'HIGH VOLATILITY':
        cls = 'high vol trend toxic' if s.MS_REGIME_GATE_BLOCK_HIGH_VOL_TREND else 'high_vol_trend_caution'
    else: cls = 'unknown pass through'
    mode = s.MS_REGIME_GATE_MODE
    if cls.endswith(' toxic'):
        action = 'hard reject' if mode=='hard' else ('soft_reject' if mode=='soft' else 'allow')
    elif cls.endswith(' caution'):
        action = 'soft reject' if mode in ('soft','hard') else 'allow'
    else: action = 'allow'
    return {'enabled': True, 'regime': rg, 'base regime': bg,
            'regime confidence': regime.get('confidence'),
            'regime class': cls, 'gate mode': mode, 'gate_action': action,
            'gate reason': f"{cls} in {mode} mode",
            'would_have_blocked': action != 'allow'}

```

### 7.2 Integration in \_evaluate\_symbol

```

regime_gate = self.compute_regime_gate_shadow(regime, decision)
if regime_gate.get('enabled') and regime_gate.get('gate_mode') != 'shadow':
    if regime_gate['gate_action'] == 'hard reject':
        decision = build_REJECT(..., reject_reason='regime gate hard', metadata={..., 'regime gate': regime_gate})
    elif regime_gate['gate_action'] == 'soft reject' and decision['decision'] == DECISION_TRADE_CANDIDATE:
        new_score = (decision.get('final score') or 0) - self.settings.MS_REGIME_GATE_SOFT_SCORE_PENALTY
        if new_score < (decision.get('threshold') or 7.0):
            decision = build_REJECT(..., reject_reason='regime gate soft', ...)

```

### 7.3 Logging

`_log_decision` JSONL und `emit_decision` metadata bekommen `regime_gate`-Key.

## 7.4 Tests — mind. 20 neu

1-20: shadow/soft/hard × RANGE/WEAK/STRONG/HIGH\_VOL × allow/reject/penalty + Kill-Switch + JSONL/DB-meta

Regression-Sweep: phase\_n7/n8, ms\_mtf, ms\_dedup, ms\_stablecoin, recon-1, sync\_sanity, exit\_reason\_fix → ~250 grün.

## 8. Was NICHT geändert wird

Item	Bleibt
MULTI_STRATEGY_DRY_RUN	true
Protected-Toolkit (BE/Partial/Trailing)	unverändert
4h Hold	unverändert
Router Hard-Blocks (CHOP/HIGH_VOL ohne base)	unverändert
MarketRegime Detector	unverändert
<code>_is_4h_confirming</code>	unverändert
Strategy-Subklassen	unverändert
Live-Trade-Entscheidungen	unverändert

## 9. Cutover-Plan (SOT-1d)

1. Crontab freeze
2. Backup `/root/regime-gate-backup-{TS}/`
3. `docker compose build clawbot`
4. `docker compose up -d --force-recreate clawbot`
5. 3-Way MD5 (2 Files)
6. Bot healthcheck
7. Crontab thaw
8. Live-Verify `regime_gate` in `multi_strategy.log`
9. DB-Verify `regime_gate` in `decision_logs`
10. 24h-Monitor

## 10. Monitoring-Checkliste

**Phase A (24h Smoke):** 0 Tracebacks, `regime_gate`-Key in jeder Decision, `gate_mode='shadow'`, `gate_action='allow'`, `would_have_blocked` aggregierbar.

**Phase B (4 Wochen Shadow):** Total Decisions vs `would_have_blocked`, Verteilung über `regime_class`, Filter-Rate bei hypothetischem hard-Mode, BUY-Rate je Regime.

**Phase C (Backtesting):** 180d-Replay mit hard-mode auf Shadow-Daten, Vergleich Shadow vs hard, Akzeptanz-Check.

## 11. Akzeptanz für P1.B-Aktivierung

Schwelle	Wert
Shadow-Daten $\geq$ 200 <code>TRADE_CANDIDATES</code>	200
<code>WEAK_TREND</code> -Toxizität bestätigt	$WR < 35 \%$
<code>RANGE</code> -Edge bestätigt	$WR \geq 50 \%$
Total Net bei <code>WEAK_TREND</code> -Block positiv	$\geq +5 \%$ über Shadow-Period
Keine Coin-Allowlist/Denylist	(Boundary)
Operator-GO	ja

## 12. Risiken

#	Risiko	Severity	Mitigation
---	--------	----------	------------

1	Regime-Detector falsch klassifiziert	Mittel	seit Phase-N1 produktiv
2	RANGE-Edge im 180d-Sample war Markt-spezifisch	<b>Hoch</b>	Shadow > 4 Wochen
3	Toxic-Klassifikation blockiert echte Gewinner	<b>Hoch</b>	Shadow misst genau das
4	Doppel-Defense mit Router	Niedrig	Router-Blocks bleiben, Gate ist additive Schicht
5	Shadow-Period zu kurz	Mittel	4-6 Wochen
6	env-flip shadow→hard vergessen	Niedrig	Roadmap-Pin
7	metadata_json wächst	Niedrig	~300 Bytes/Row
8	Score-Penalty 1.5 heuristisch	Mittel	Sweep nach Shadow

### 13. Rollback-Plan

**Soft:** `git revert` + SOT-1d rebuild. **Hard:** `env MS_REGIME_GATE_ENABLED=false` oder `MS_REGIME_GATE_MODE=shadow`, recreate.  
**Phase:** `env MS_REGIME_GATE_MODE=shadow`, recreate, weiter Daten sammeln.

Backup: `/root/regime-gate-backup-{TS}/`.

### 14. Operator-Entscheidungen

Q	Frage	Default
Q1	Cutover im shadow-Mode?	<b>A</b> Ja, 4 Wochen log-only
Q2	Block WEAK_TREND in P1.B?	<b>A</b> Ja (−30 %)
Q3	Block STRONG_TREND in P1.B?	<b>B</b> Nein (Caution statt Toxic)
Q4	Block HIGH_VOL mit base in Trends?	<b>A</b> Ja
Q5	Soft-Score-Penalty?	<b>A</b> 1.5
Q6	Kill-Switch enabled?	<b>A</b> Ja, default true
Q7	Shadow-Period?	<b>B</b> 4 Wochen
Q8	Cutover-Zeitpunkt?	Wochenende
Q9	Bundle mit MS-MTF-D-SHADOW-LOGGER-1?	<b>A</b> Empfohlen
Q10	P1.B-Flip nach 2 Wochen erlaubt?	<b>B</b> Nein, mind. 4 Wochen

### 15. Aufwand

~145 LoC + 39 Tests + Cutover-Prep = **~5.5 h**.

### 16. Verhältnis zu anderen Phasen

- **MS-MTF-D-SHADOW-LOGGER-1:** Bundle empfohlen, gleicher Cutover
- **BTC-MACRO-CONFIRMATION-1:** unabhängig, später kombinierbar
- **LOSS-STREAK-CIRCUIT-BREAKER-1:** ergänzend
- **Strategy-Loss-Backtest-Audit:** paralleler P1, blockiert NICHT
- **MS-Live-Pilot:** DEFERRED

### 17. Boundaries

0× Code-Touch (außer Plan-Phase) · 0× Trading-State · 0× Orders · 0× MS-Live · 0× Mainnet · 0× Env-Änderung (außer optional 6 safe-Default Settings) · 0× DB-Schema/Migration · 0× ConfigProfile-Apply · 0× Bot/Worker-Recreate · 0× Push · 0× Roadmap-Commit · 0× Coin-Allowlist · 0× Coin-Denylist · 0× Protected-Toolkit-Param-Change.

### 18. STOP-Bedingungen

NICHT umsetzen wenn: Operator-GO fehlt, Bot/Worker unhealthy, State-Coherence-Conflicts aktiv, MS-Dry-Run-Anomalien letzte 24h, Bundle-Partner hat frische Tracebacks.

## 19. Plan-Closure (per CLAUDE.md §13)

---

- Root Cause: 180d-Replay RANGE +26/WEAK\_TREND -30; XLM-BEAR-Live-Loss heute
  - Files: 2 (multi\_strategy\_runner.py, settings.py)
  - Tests: 20 neu + Regression
  - Cross-Impact: §4
  - 3-Way MD5: post-Cutover Pflicht
  - Mainnet: weiter verboten
  - bewusst nicht: Protected-Toolkit, Subklassen, Router, Regime-Detector, Live-Pipeline
  - Follow-up: P1.B Mode-Flip, P1.C Per-Regime-Params, Score-Penalty-Sweep, BTC-Macro, Loss-Streak
- 

## 20. STOP

---

Plan abgeschlossen. **Keine Umsetzung.** Operator-Entscheidung zu Q1-Q10 + G0 EXECUTE REGIME-AWARE-PARAMS-1-P1A-SHADOW erforderlich für Code-Phase.