Case Study:

Real-Time Ad Launch Review:
Fast Approval, Deeper Checks Only When Needed

Prepared by **Rex Chang**

GenAI & Cloud Solution Architect / Program Lead





Results and Business Relevance

Problem

Deeper checks can mistakenly overturn a correct numeric decision.

Measured result

Freeze protection reduced wrong blocks from 1 to 0 in an 11-case replay.

Budget finding

Extra verification ran on 3 of 11 cases (27.3%), above the 15% cap.

Reviewability

All 11 replay decisions produced receipts.

Why This System Exists

The speed-vs-safety tradeoff in Ad launch moderation.

Business Context:

Minutes before launch, the platform must decide whether to **approve** an ad immediately or **hold** it for deeper review.

Measurable Pain:

1. Wrong blocks at launch
   → lost revenue and advertiser friction
2. Numeric misinterpretation (age, price, end date, budget)
   → high-impact moderation errors
3. Unpredictable compute usage
   → cost and latency spikes
4. Unexplainable decisions
   → audit escalations and slower rollouts

Why naive solutions fail:

Always fast

- Let risky edge cases slip through

Always slow/expensive

- delays launches and burns compute budget

Human override-heavy workflows

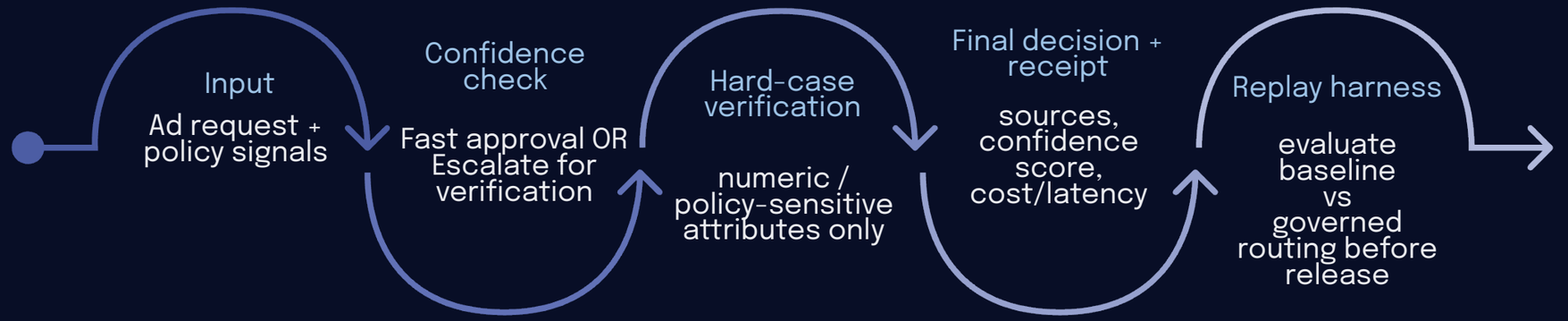
- do not scale and create inconsistent decisions

Economic Exposure:

Revenue loss, compliance risk, trust erosion.



End-to-End Decision Flow



Risk Control

- Risk flags route selected cases to deeper verification.
- Borderline or high-impact cases escalate for deeper verification

Measurability

- Every decision produces a structured receipt explaining tier choice
- Replay harness allows baseline vs governed routing comparison

Cost Control

- Most requests remain in the cheaper fast-lane tier
- Expensive verification is triggered only for low-confidence or flagged cases

Result integrity

- Freeze protection locks verified numeric outcomes
- Deeper verification cannot change them without proof

Deliberate Choices at Decision Points

Each row maps to the flow: *Confidence Check → Hard-Case Verification → Final Decision + Receipt → Replay Harness*

Flow Node	Deliberate Choice	Alternative	Business Implication	Trade-off
1/ Confidence Check	Confidence vs threshold routing	Static rules	Transparent fast-lane criteria	Requires calibration governance
2/ Hard-Case Verification	Seal clean numeric decisions	Always allow reinterpretation	Replay: 0 wrong blocks	Reduced flexibility
3/ Final Decision + Receipt	Structured decision receipts	Free-text explanation	Auditable decisions	Slight logging overhead
4/ Replay Harness	Hard release gate	Human approval	Sets release criteria	Slower iteration

Numeric Replay: Protection Works, Budget Needs Tuning

11-case replay: wrong blocks, verification usage and receipts.

Replay Experiment (Controlled Inputs)

Tested using a **deterministic replay** on an **evaluation set of 11 ad-review scenarios** designed to stress numeric edge cases such as age limits, prices, and dates.

Scenario	Wrong Blocks (out of 11)
Freeze protection ON	0
Freeze protection OFF	1

Wrong block = a correct ad that becomes incorrectly rejected after deeper verification changes a correct numeric interpretation.

Interpretation

- Freeze protection prevented the one wrong block observed in this replay.
- The result covers **the tested numeric edge cases**.

Governance Signal (Replay Outcome)

Signal	Observed Value	Interpretation
Advanced verification usage	27.3% (3 / 11 scenarios)	Exceeds configured 15% compute cap
Decision receipts	11 / 11 generated	Each decision includes traceable reasoning data

Advanced verification usage = % of requests routed to the expensive verification tier

Replay Findings and Release Criteria

- **Routing: 3 of 11 cases** used deeper verification, exceeding the 15% cap.
- **Decision protection:** freeze protection prevented 1 wrong block in this replay.
- **Release criteria** are designed to reject candidates that exceed configured limits.



Runtime Constraints and Decision Receipts

Runtime safeguards

- Latency, cost, and compute limits are explicit evaluation constraints.
- High-risk cases are escalated for deeper verification.

Deterministic Replay

- The same input and config reproduce the same routing outcomes for fair baseline-vs-governed comparison.

Auditable Decision Records

- Each request generates a structured trace with routing choice, verification path, and trace ID.

Decision integrity protection

- Verified numeric outcomes are locked unless new evidence proves the result incorrect.

Appendix

Technical Deep Dive



Architecture: Data Flow, Policy Flow, Evidence Flow

Data Flow

Request → Context → Decision → Receipt (tier, reasons, vitals).

Policy Flow

Thresholds + Guardrails → Routing Rules → Freeze / Override Logic.

Evidence Flow

Replay → Metrics → Release Gate → PASS/FAIL Artifacts.

Inputs & Context

- Request, flags, judge score, numeric/tool inputs
- Builds structured decision context

Ops & Governance

- Release gate
- Hard bars / caps
- Ship / hold decision

Measurement & Evidence

- Baseline vs cascade replay
- Ablation (freeze ON/OFF)
- Summary CSVs and per-task traces



Control Plane

- Confidence vs threshold routing
- Guardrail escalation
- Hard-case scoping

Generation & Response

- Reasoning gate
- Freeze-winner
- Proof-based override

State & Replay

- Golden set
- Decision traces
- Numeric mismatch logs

Deep Mechanism: Invariants That Prevent Regret

Fast Lane Earned

Rule: Fast lane is allowed only when confidence ≥ threshold and no guardrails trigger.

Why: Sends flagged requests for deeper checks.

Verification Scoped

Rule: Extra verification runs only when cascade mode is active, the request escalates, and hard-case signals appear.

Why: Keeps additional compute limited to borderline cases.

Numeric Results Locked

Rule: Freeze the result when: vote_share ≥ 0.66, no tie exists.

Why: Protects correct numeric outcomes from being degraded by later reasoning.

Proof Required

Rule: Override allowed only if |tool_value - frozen_value| > tolerance.

Why: Supports evidence-based corrections.

Fair Comparisons

Rule: Baseline and cascade must run on identical inputs.

Why: Supports a controlled baseline comparison.



Thank You!

Rex Chang

GenAI & Cloud Solution Architect / Program Lead

Portfolio: rexchang.dev | LinkedIn: linkedin.com/in/rex-chang-tw | Email: rex@rexchang.dev