



The neural CFO

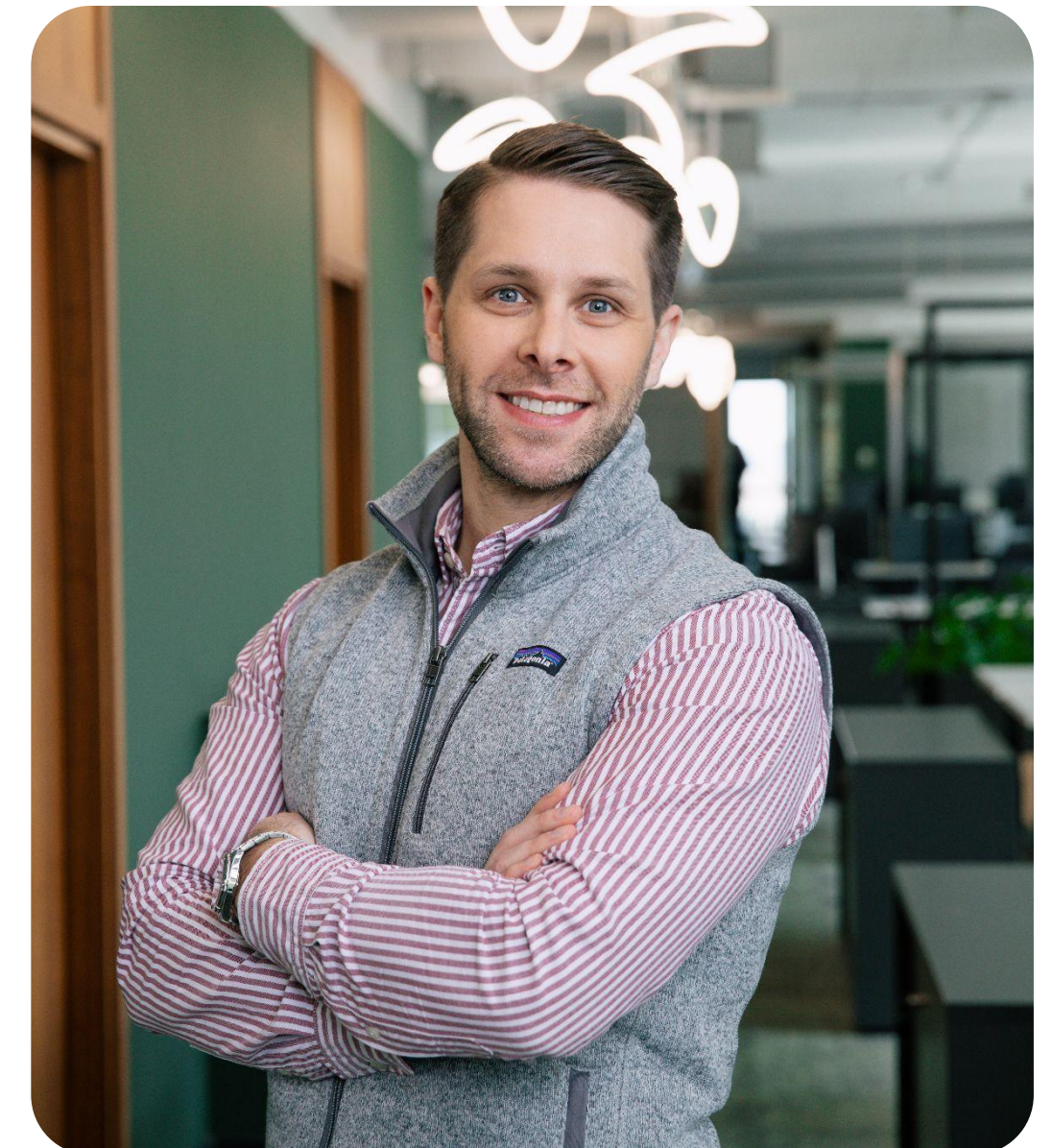
Building adaptive, learning finance systems for the AI era

Payhawk

Payhawk: Who we are



Georgi Ivanov
Storyteller-in-Chief, AI &
Finance



Felix Brückner
Country Director – DACH

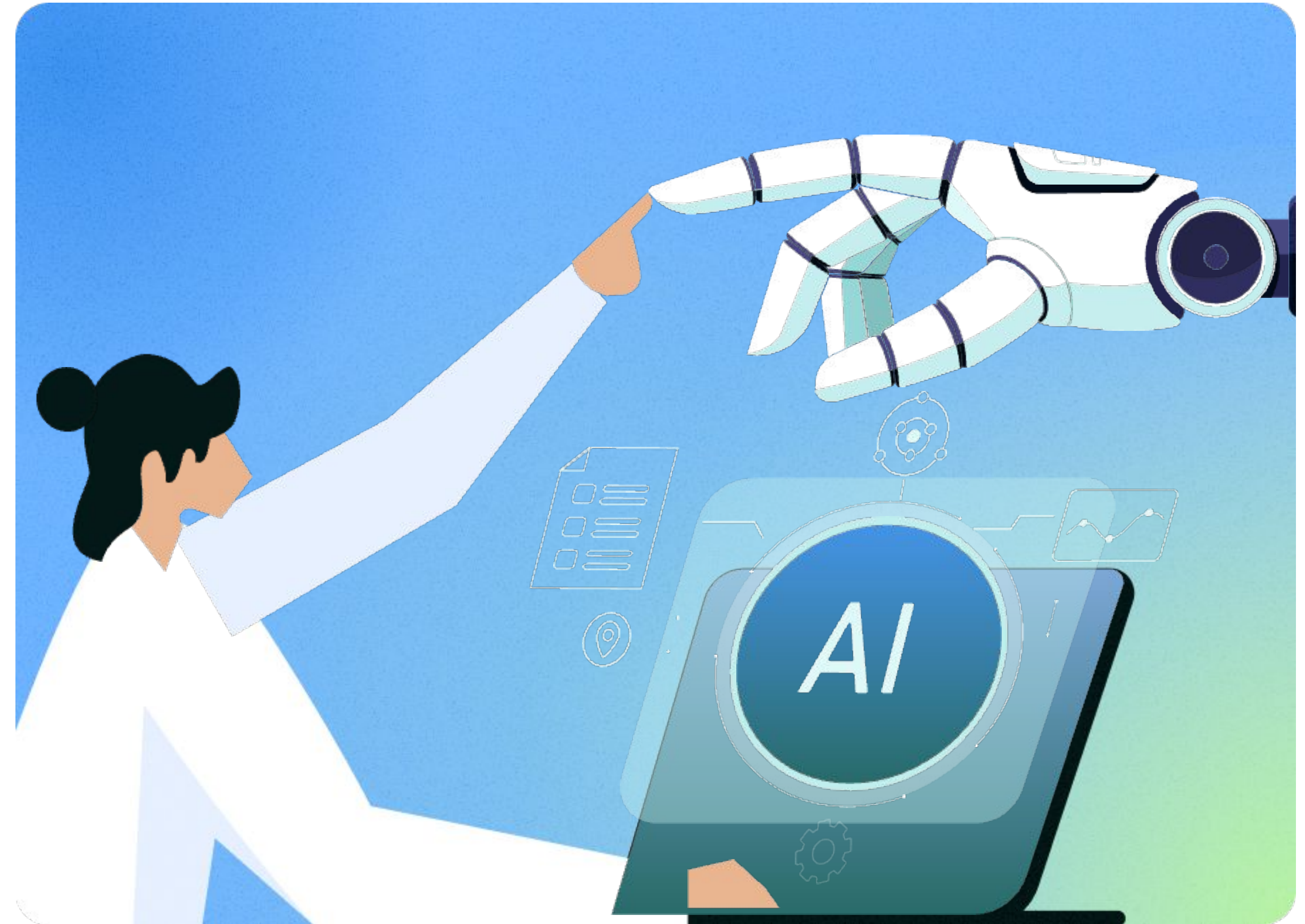
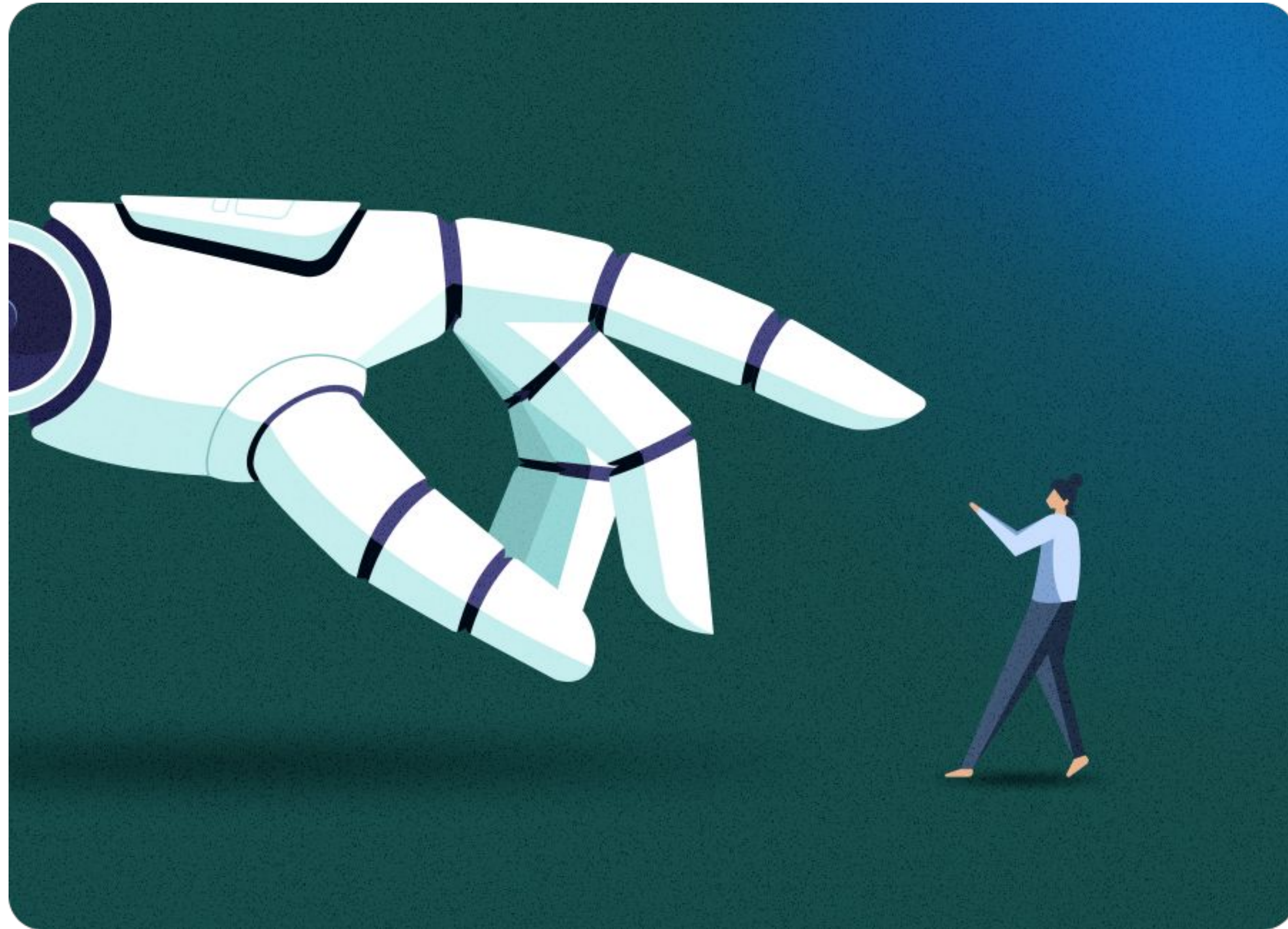
Autopilot didn't
remove pilots.
It redesigned
the cockpit.





Myth: AI will replace finance jobs

Reality: AI replaces tasks, not people



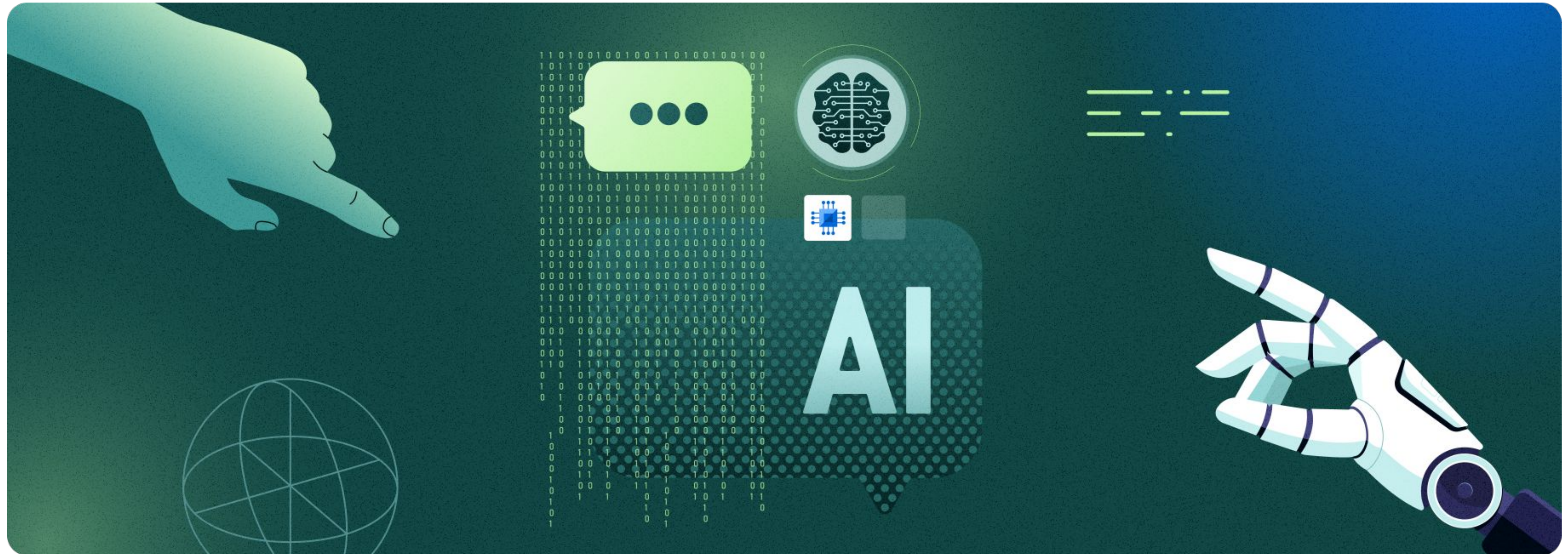
Myth: AI is too risky for finance

Reality: Risk comes from bad governance



Myth: AI is only for the big guys

Reality: AI has been democratised



Payhawk Finance AI Evolution Framework

Attribute \ Stage	0. Digitise	1. Automate	2. Augment	3. Advance	4. Autonomous Ecosystem
Tagline	Get paper into pixels	Cut the clicks	Humans + predictive power	Close in real time	Self-optimising finance
Dominant capability	OCR / e-invoicing	Auto-coding & matching	Anomalies + rolling forecast	Continuous close + drift guard	Multi-agent STP
Hard KPI gate	≥ 80 % inbound docs digital	< 20 % manual touches	MAE ≤ 4 % · alert precision > 90 %	Close ≤ 24 h · drift ≤ 2 %	STP ≥ 99 % · real-time liquidity
Agent archetype	—	Lookup bots	Action agents (HITL)	Autonomous agents (bounded)	Agent orchestration (multi-agent system)

Trust Layer: Unified truth · Explainable AI · Continuous assurance

Agent archetypes at a glance

Lookup

- Rights: Suggest-only
- Tasks: Parse · Enrich · Match
- Controls: Lineage · Confidence
- Gate → S2: <20% touches

Action / HITL (Human-in-the-loop)

- Rights: Propose → approve
- Tasks: Accruals · Anomalies · Forecast recos
- Controls: Model cards · Approvals
- Gate → S3: MAE ≤4% · precision >90%

Autonomous (bounded)

- Rights: Execute within limits
- Tasks: Routine JEs · Auto-PO · Auto-book
- Controls: SoD/RBAC · Invariants · Drift
- Gate → S4: Close ≤24h · drift ≤2%

Agent orchestration (multi-agent / MAS)

- Rights: Orchestrate cross-workflow
- Tasks: P2P + Travel + Treasury sequencing
- Controls: End-to-end provenance · Telemetry
- Target: STP ≥99% · Live liquidity

Archetypes persist across stages; earlier layers continue running underneath later ones.

Illustrative examples:

Payhawk agents across the archetypes



Financial Controller Agent

- Lookup (S1): Status lookups; receipt parsing/grouping
- Action (S2): Smart reminders; doc-chase; unusual spend flags
- Autonomous (S3): Auto-follow-ups within policy; escalate on breach
- MAS (S4) : Coordinate with Payments/Travel for near-STP expense flows



Procurement Agent

- Lookup (S1): Intake normalisation; category/vendor suggestions
- Action (S2): Drafts requests/POs; cites policy; routes approvals
- Autonomous (S3): Request→PO/card pathing within budgets & policy
- MAS (S4): Sequence releases with AP/Payments/Treasury



Travel Agent

- Lookup (S1): Traveller profile/policy checks; coding hints
- Action (S2): Suggests compliant options; triggers approvals
- Autonomous (S3): Auto-book within policy;trip report § attach invoices; basic changes
- MAS (S4): Coordinate cards, Controller & AP for straight-through travel



Payments Agent

- Lookup (S1): Payment status lookups across rails
- Action (S2): Investigates; drafts resolution steps; routes to workflows
- Autonomous (S3): Auto-resolve standard info-only cases; escalate cash-moving items
- MAS (S4): Sequence timing with Procurement & Treasury (cash ladder)

Illustrative examples:

Payhawk agents across the stages

Agent \ Stage	1. Automate	2. Augment (HITL)	3. Advance (bounded)	4. Autonomous Ecosystem
Financial Controller	Doc validation prompts; receipt matching; anomaly pre-flags	Propose fixes; chase docs; anomaly evidence packs	Auto-hold/auto-chase small, high-confidence cases; escalate exceptions	Swarm with Payments/Travel to enforce complete trip/report packs; near-STP reimbursements
Procurement	Request intake normalised; policy lookup; budget check preview	Draft POs/requests; policy citations; route approvals	Auto-PO/auto-approve catalog + threshold buys; 3-way match triggers	Coordinate with AP/Treasury/Payments to sequence P2P; optimise discounts vs. cash
Travel	Traveller profile/policy gating; expense coding hints	Policy-compliant itinerary proposals; approvals; Trip Report assembly	Auto-book within policy; auto-attach invoices; auto-reconcile per-diems	Coordinate cards funding, Controller, AP for near-STP travel spend
Payments	Status lookup; data stitching across rails	Draft responses; investigate payment issues; route to workflows	Auto-resolve standard cases (e.g., resend remittance); never move cash outside bounds	Sequence payments with Procurement/AP/Treasury vs. cash ladder; surface exceptions only

The trust layer: how autonomy stays **safe**

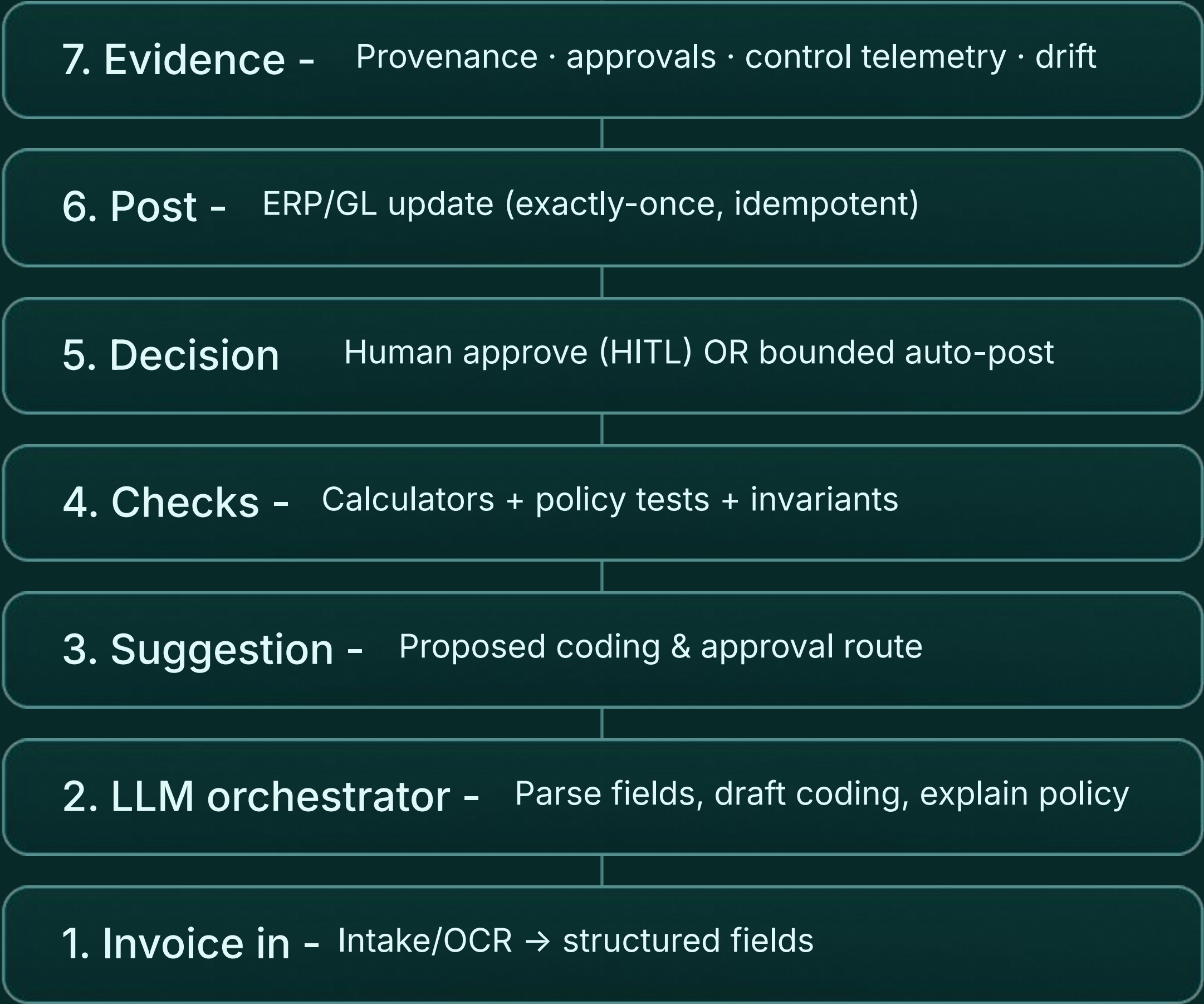
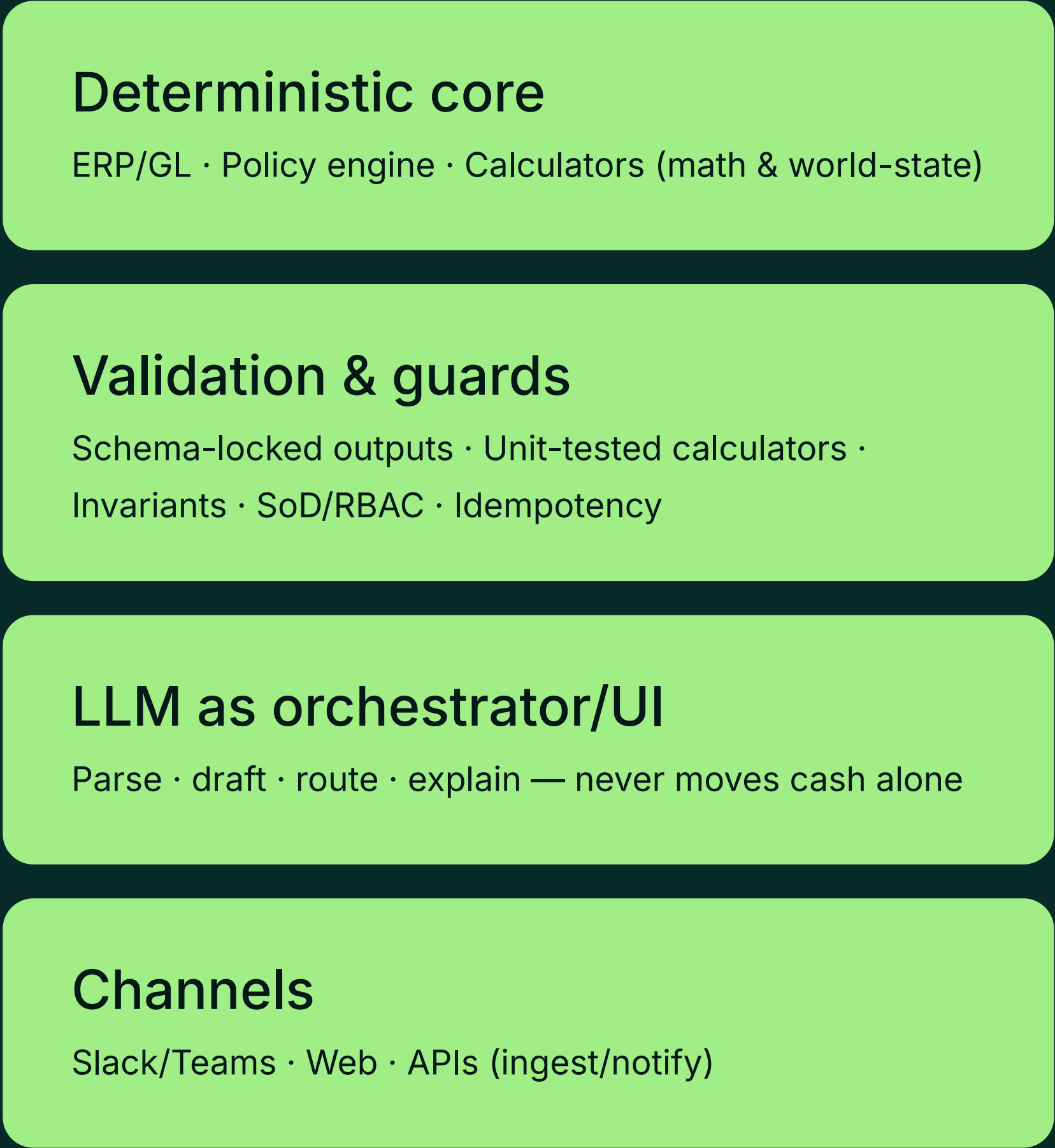


Unified truth
One governed finance data lake
("single spreadsheet in the sky")

Explainability
Model cards, bias scans, approvals
in trail

Continuous assurance
Telemetry, drift monitors,
kill-switches

Architecture: where it runs



Why LLMs \neq finance brains

LLMs aren't world models

- Predict words, not the world (no persistent state / hard constraints)
- Fluent but rule-breaking
- Struggle when math/state must stay consistent (hallucinations, recency bias)

Failure modes to avoid

- LLM posting JEs or moving cash
- LLM editing mappings/drivers without approvals & tests
- Narrative numbers not linked to calculators/data (no provenance)

Finance needs a world model

- Math + rules + state you can audit (ledgers, subledgers, policy limits, covenants, identities)
- Enforce invariants ($A=L+E$; tie-outs), limits, and exactly-once posting
- Time-indexed state & deterministic replay (period locks, adjustments) with full provenance.

Design rules

- LLM = interface/orchestrator (parse, draft, route, explain)
- Deterministic core = truth (calculators, policy engine, ERP/GL)
- Guards = schema-locked outputs, invariants, SoD/RBAC, idempotency, drift monitors; fallback to HITL on breach

Words aren't the world. Ledgers are.

The Vision: 2030

From clicks to intent
Say what you want; policy-bound agents do the rest.

From after-the-fact to by-design controls
Policies compile into guardrails; evidence travels with every action.



From month-end to moment-ready
The ledger behaves like a living digital twin - publish when you're ready.

From apps to an agentic fabric
Specialised agents coordinate across spend, travel, and cash without tickets.

From number-making to decision-making
Humans handle scenarios and exceptions; machines handle the rest.



Your job is not to close the books.
It's to open **the future.**

Payhawk