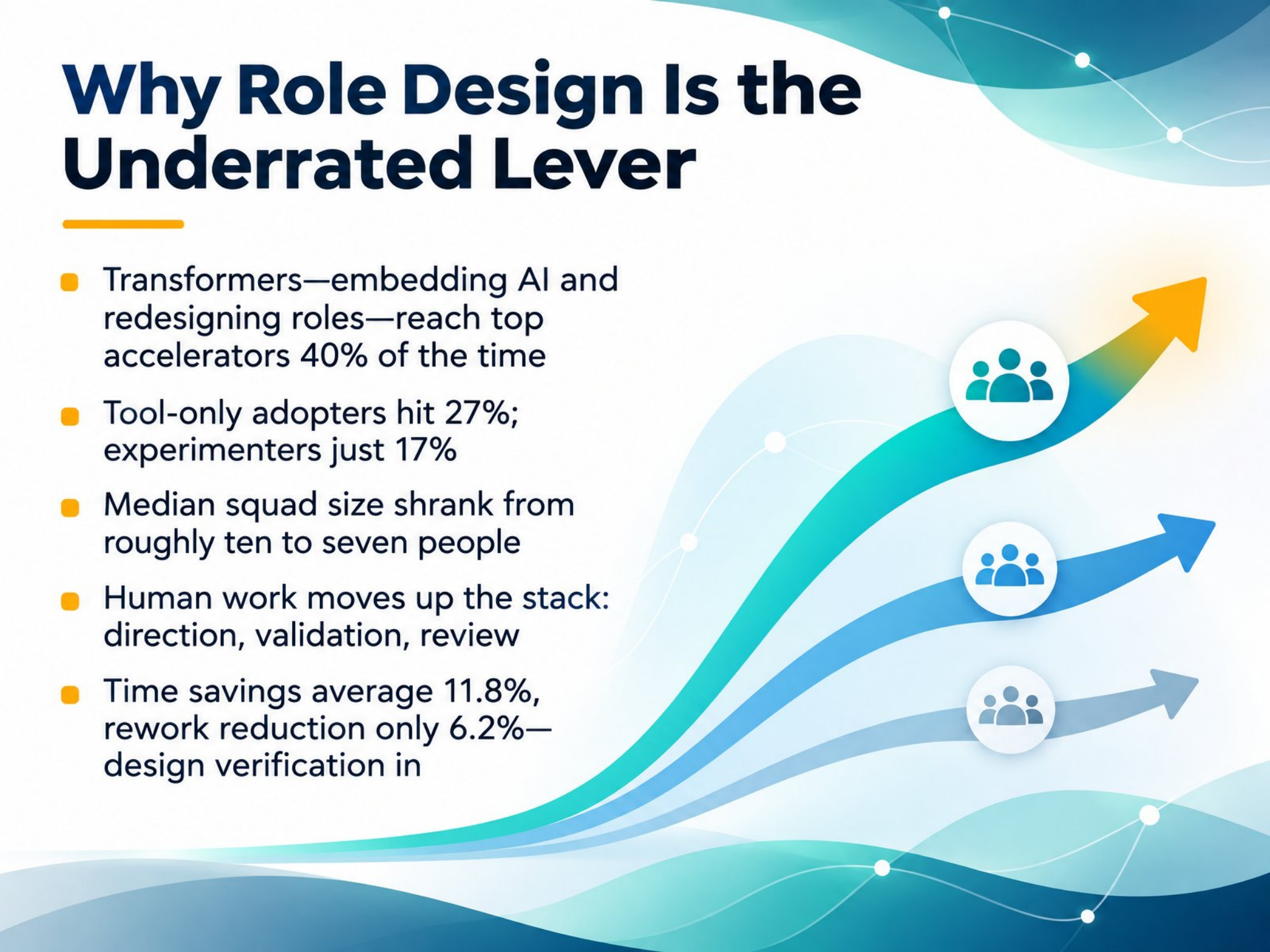
Generative AI Roles, Responsibilities, and Handoffs: Operating Model Training for Cross-Functional Teams

- One shared operating model for who does what across the generative AI lifecycle
- Delivery is a relay: application teams, shared harness, independent governance oversight
- Good looks like: one accountable owner, written decision rights, evidence-carrying handoffs
- Failure modes addressed: accountability diffusion, undocumented prompt changes, rubber-stamp reviews
- **Roadmap**: role map → decision rights → handoffs and gates → human review → incidents → workshop



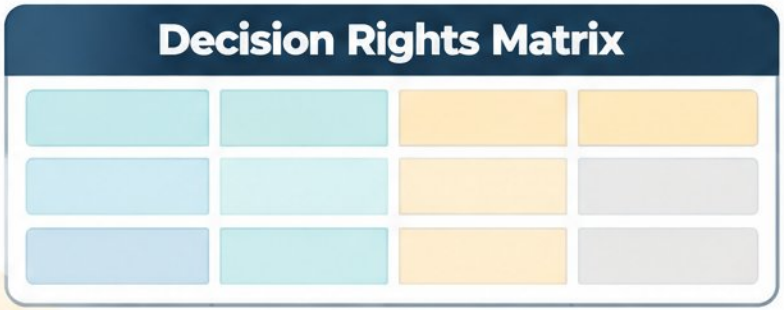
Why Role Design Is the Underrated Lever

- Transformers—embedding AI and redesigning roles—reach top accelerators 40% of the time
- Tool-only adopters hit 27%; experimenters just 17%
- Median squad size shrank from roughly ten to seven people
- Human work moves up the stack: direction, validation, review
- Time savings average 11.8%, rework reduction only 6.2%—design verification in



The Federated Model: Application Teams, Shared Harness, Independent Oversight





- Application teams own the workflow, success measures, and business outcome
- Shared harness team provides tooling, agent identity, permissions, evaluation infrastructure
- Risk, security, and governance hold independent oversight — outside the delivery chain
- Application teams and domain experts define evaluation cases and the acceptance bar
- Federation fails on ambiguity: unclear ownership surfaces later as stalled releases

Role Map Across the Generative AI Lifecycle

- **Lifecycle:** discovery, data prep, model selection, design, evaluation, deployment, monitoring, retirement
- **Delivery roles:** AI product owner, AI/ML and data engineers, LLMOps, evaluation lead, domain expert, enablement lead
- **Second line:** governance and risk lead, AI security engineer, legal and privacy — outside the delivery chain
- **Agents add owners:** agent owner, agent operations, named approvers, permission steward, red-team coverage

Executive sponsor holds budget and priority disputes; named AI leader runs the programme

Delivery roles

- AI product owner
- AI/ML and data engineers
- LLMOps
- Evaluation lead
- Domain expert
- Enablement lead



Discovery → Data prep → Model selection → Design → Evaluation → Deployment → Monitoring → Retirement

Outside the delivery chain

- Governance and risk lead
- AI security engineer
- Legal and privacy

Decision Rights Written Down: A RACI by Decision, Not by Team

- Exactly one accountable role per activity — two A cells means the right is unsettled
- Assign explicitly: new use case, model or vendor, data source, release, residual risk, incident, pause
- Separate accountability from execution: product owner builds and evaluates, business owner releases
- Build lightweight charts by decision type, not one team-wide matrix nobody reads
- Anti-patterns: accountability on "the team," shared A cells, every senior leader accountable

Decision	R	A	C	I
New use case	R	A	C	
Model or vendor	R	A		I
Data source		A	C	
Release	R	A		I
Residual risk		A	C	
Incident	R	A		I
Pause		A	C	

Risk Tiers Drive Proportional Review

- Classify beyond likelihood and cost: data sensitivity, reversibility, autonomy, vulnerability, regulatory exposure
- Tier 1: streamlined sign-off and reusable test suites
- Tier 2: enhanced review of model governance, logging, and recovery
- Tier 3: multidisciplinary review, documented mitigations, executive risk acceptance
- Tier the deployment, not the vendor logo
- Tiering sets testing depth, oversight, approval authority, and cycle times
- Intake must record owner, intended use, population, data category, environment



Same Vendor Model

Intended Use

Tier 1 — streamlined sign-off and reusable test suites

Tier 2 — enhanced review of model governance, logging, and recovery

Tier 3 — multidisciplinary review, documented mitigations, executive risk acceptance

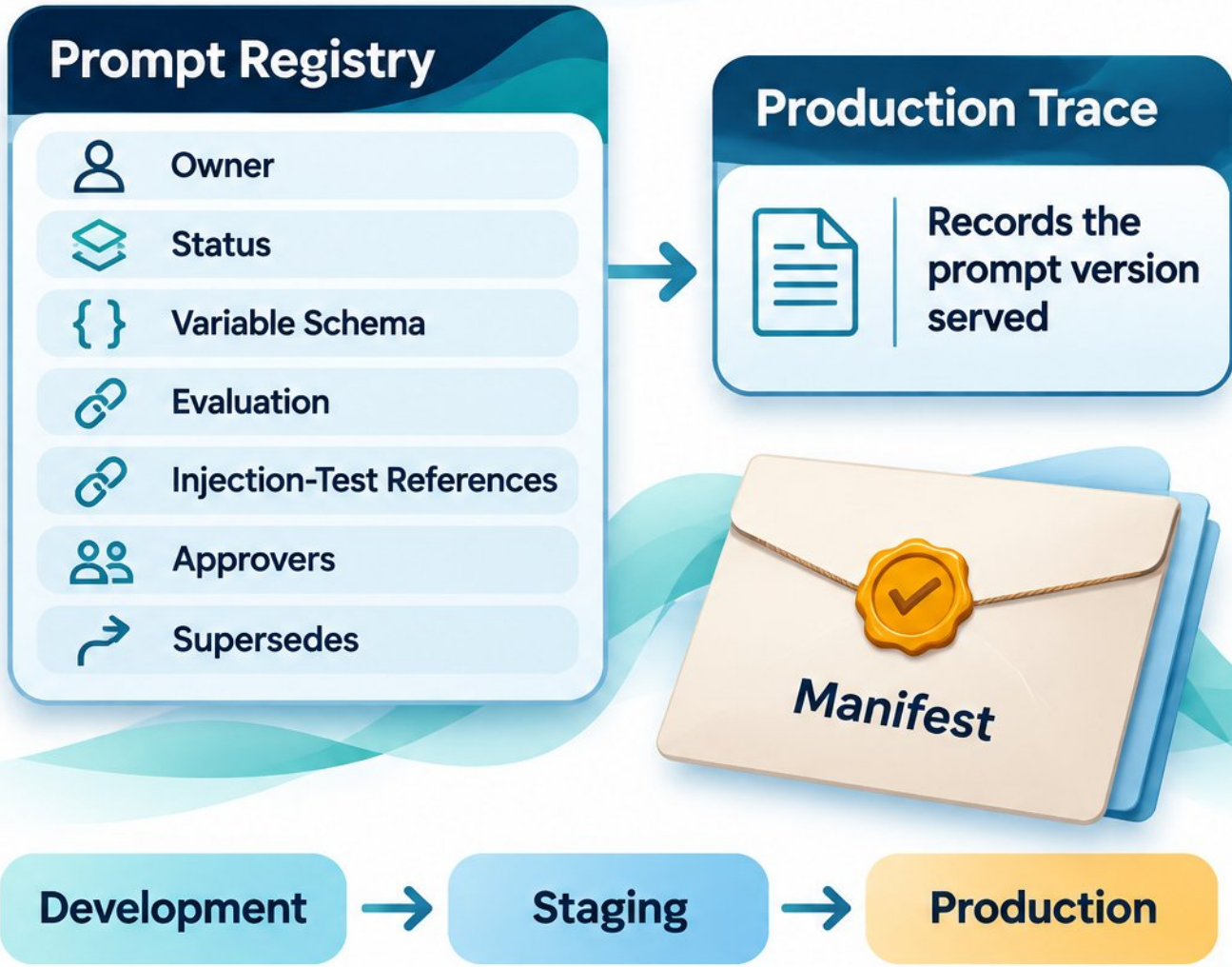
Handoffs That Hold: Artifacts, Gates, and Acceptance Criteria



- Define five handoff points: use case→design, design→evaluation, evaluation→release, release→operations, incident→remediation
- Required artifacts: use-case brief, risk tier record, evaluation plan, versioned results, release manifest, runbook
- Release manifest pins code, model config, prompt versions, retrieval settings; rollback rolls back the manifest
- Acceptance test: receiving team can use the artifact without the original builder
- Anti-patterns: throw-over-the-wall, undocumented prompt changes, missing evaluation evidence

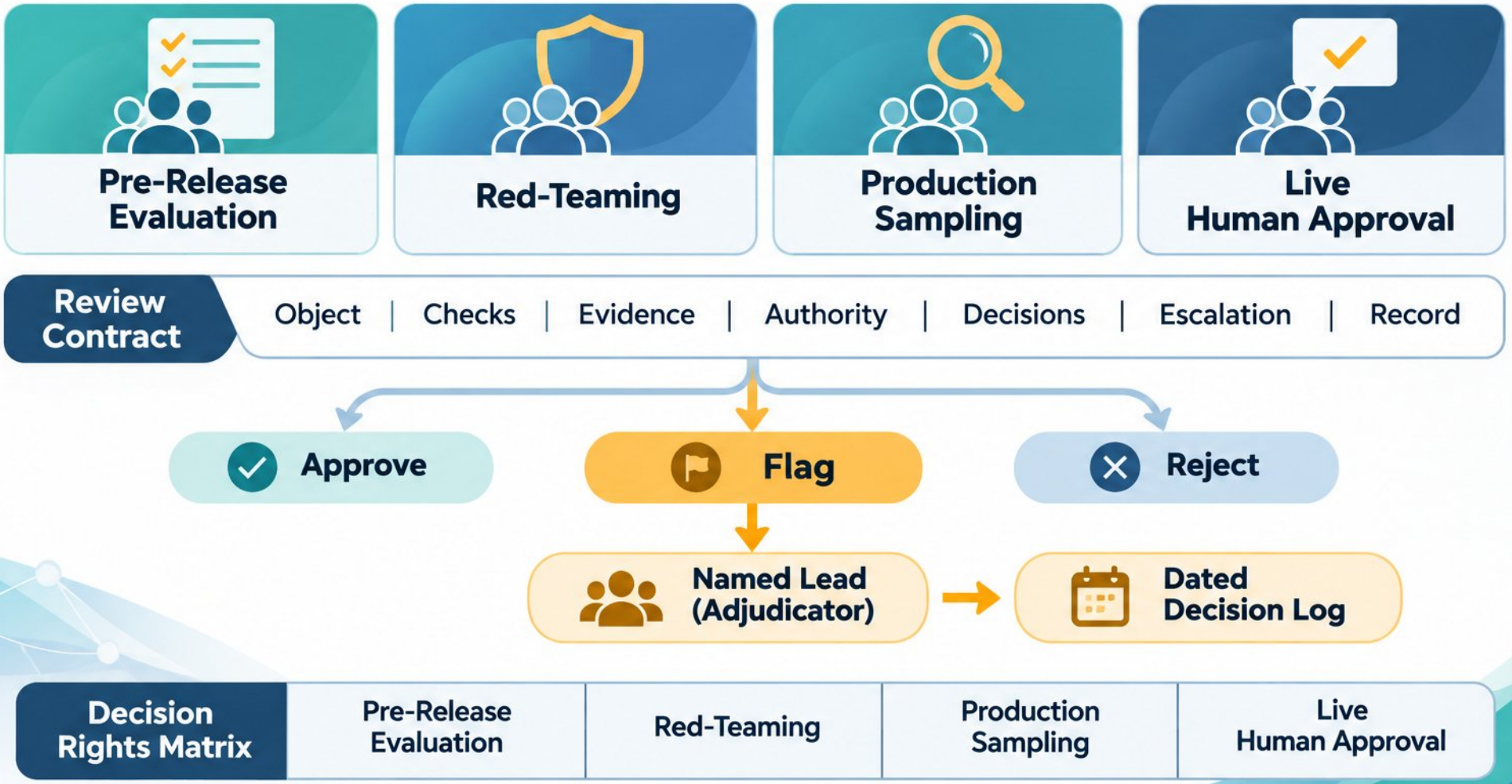
Versioning Prompts, Models, and Retrieval as Production Code

- Prompts are production artifacts: version, register, and link each version to qualifying evaluation results
- A version number names a performance profile, not just a template
- Registry fields: owner, status, variable schema, evaluation and injection-test references, approvers, supersedes
- Promote prompts through development, staging, and production like code
- Every trace records the prompt version served, enabling post-incident forensics



Domain Reviewers and Human-in-the-Loop: Design, Not Rubber Stamp

- Separate four review types with distinct owners: pre-release, red-team, production sampling, live approval
- Write a review contract per point: object, checks, evidence, authority, decisions, escalation, record
- Give reviewers a third option — flag — with a named lead and dated decision log
- Route strategically: disagreement, risk-based, novelty, and incident-driven sampling
- Calibrate and measure: gold set, inter-rater agreement, seeded-problem catch rate



Review Gates with Real Blocking Power

- Gates: pre-build, pre-release, pre-deployment, post-incident review
- Each gate has defined entry and exit evidence
- A gate works only if a finding can block launch
- Name the authority: "anyone can raise a concern" blocks nothing
- Pre-release evidence: ground-truth evaluation, failure taxonomy, rollback tested
- Keep gates proportionate to risk tier; low-risk work skips queues



Incident Ownership and Escalation Paths

- Name an incident commander with authority to pause the system and escalate
- Contain first, investigate in parallel: disable features, apply filters, throttle access
- Pre-agree who can disable, isolate, roll back, and authorize return to service
- Escalate immediately for harm, sensitive data, irreversible decisions, or incomplete logs
- Rehearse: system down, bad proposals, escalation owner absent



Governance Interfaces and Third-Party Responsibilities



- Map interfaces with security, privacy, legal, compliance, procurement, and vendors
- Name who attends a Red-tier event and who owns each vendor relationship
- Diligence covers data use, de-identification, provenance, red-team evidence, subcontractors
- Contracts need advance notice, rights to test new versions, and deprecation terms
- Treat material vendor model updates as production changes requiring gate requalification
- Document vendor versus customer responsibility for configurations and safeguards
- Customer-side settings can shift liability, so record who owns each configuration

Operating Rhythm, Metrics, and Continuous Improvement



- Standing forums: model review board, evaluation review, release readiness, blameless post-incident review — each with named output and audience
- Handoff metrics: rework rate, late-gate gaps, time to release, escalations resolved, reviewer correction rate, incident recurrence
- Feedback loop: route production failures and edge cases into evaluation sets as permanent regression tests
- Watch for reviewer burnout: rising bulk approvals and falling seeded-problem detection signal nominal review
- Refresh role map and decision rights quarterly, or when models, tools, regulations, or teams change materially

Workshop: Draft Your Team's Role and Handoff Charter

- **Exercise 1:** map one use case's lifecycle roles; one accountable owner per activity
- **Exercise 2:** define three critical handoffs with artifacts, evidence, testable acceptance criteria
- **Exercise 3:** name one accountability gap and one high-risk escalation trigger
- **Exercise 4:** assign a risk tier and name who can block release
- **Peer review rubric:** clear ownership, testable criteria, manageable reviewer load
- **Adopt** a one-page charter; set a 30-60-90 day plan

One-Page Charter

Exercise 1: Role Map



Exercise 2: Three Handoffs



Exercise 3: Accountability Gap



Exercise 4: Risk Tier and Blocker



30-60-90 Day Plan





PersonWise.

PersonWise • AI digital-human course platform

Scan the code or click anywhere to watch this course live, with voice Q&A

personwise.ai/t/6be3ea39/public