WHITE PAPER

Designing AI Agents That Deliver Outcomes

Six best practices for building an enterprise AI and automation capability in financial services

Roger Estrada · AI Strategy and Transformation · October 2026

ABSTRACT

AI agents can coordinate work across departments and fragmented systems, but most early efforts capture only a fraction of that value because they are run as tools and pilots rather than as a managed capability. This paper sets out six practices for designing agents that deliver measurable outcomes: build the operating model, prioritize by outcome, fix the process before automating it, match technology and sourcing to the problem, govern with graduated autonomy, and lead adoption against clear KPIs. It draws on my work in AI strategy and transformation at a financial institution, my graduate study in AI in business, my MBA in finance, and my Lean Six Sigma Black Belt.

1. Why design matters more than technology

The institutions that benefit most from a new technology are rarely those with the best tools. They are the ones that redesign how work is done around it (Mantia, Chatterjee and Lee, 2025). Agentic AI raises the stakes of that choice. Unlike a tool a person invokes, an agent system interprets a goal, decides on steps, calls other systems, and adapts. Because it acts across functional boundaries, it exposes whatever is weak in them: unclear ownership, undocumented decision rules, inconsistent data, and slow handoffs. In a regulated institution, where members expect fair, secure service and regulators expect demonstrable control, scattered experiments are not enough. The goal is a managed capability that produces outcomes, not activity.

2. What agents are, and how they differ from automation

Think of an agent system as a team of digital colleagues. Specialist agents do one job well, such as retrieving the right policy passage, drafting a letter, or opening a case. Coordinator agents interpret a request, route it to the right specialist, and assemble a finished outcome. Rules-based automation and RPA remain valuable for stable, repetitive work, but every exception must be scripted in advance and the automation breaks when systems change. Agents are built around an objective rather than a script, so they handle variation and can work across applications never designed to work together. The design question therefore shifts from "which steps can we mechanize?" to "what outcome are we accountable for, and what mix of people and agents delivers it most reliably?" Figure 1 (below) shows the reference architecture used in this paper.

3. Six best practices

Practice 1: Establish the function and its operating model

Without a clear owner, AI adoption fragments: overlapping tools, inconsistent controls, and results reported in incompatible ways. An enterprise AI and automation function, positioned close to technology leadership and partnered with business, risk, compliance, and operations, provides one point of accountability. Its operating model should define a charter and decision rights, a single intake path with a standard template, a transparent prioritization method, delivery standards for testing, security, oversight, and handover, and a rolling roadmap tied to strategy and architecture. A hub-and-spoke design works well: the center sets standards and maintains shared assets, while business owners own outcomes and adoption. A function that builds for the business without sharing accountability for the result creates dependency, not capability.

Practice 2: Prioritize by outcome and appoint value stream owners

Members experience journeys that cross functions, and what they want is fewer problems, not merely faster fixes. Agents aimed at isolated tasks speed up a fragmented process; agents aimed at an outcome redesign it. Every significant member journey is a value stream, and it needs a value stream owner who defines the outcome, directs people and agents, and answers for the result, with authority across departments. This is the role Mantia, Chatterjee and Lee (2025) call a mission owner; Lean practice has long given the same accountability to the owner of an end-to-end value stream. Choose use cases with a consistent rubric (Table 1). From a finance perspective, I weight expected value explicitly, including the cost of errors and the cost to serve, which for language-model systems includes usage-based pricing.

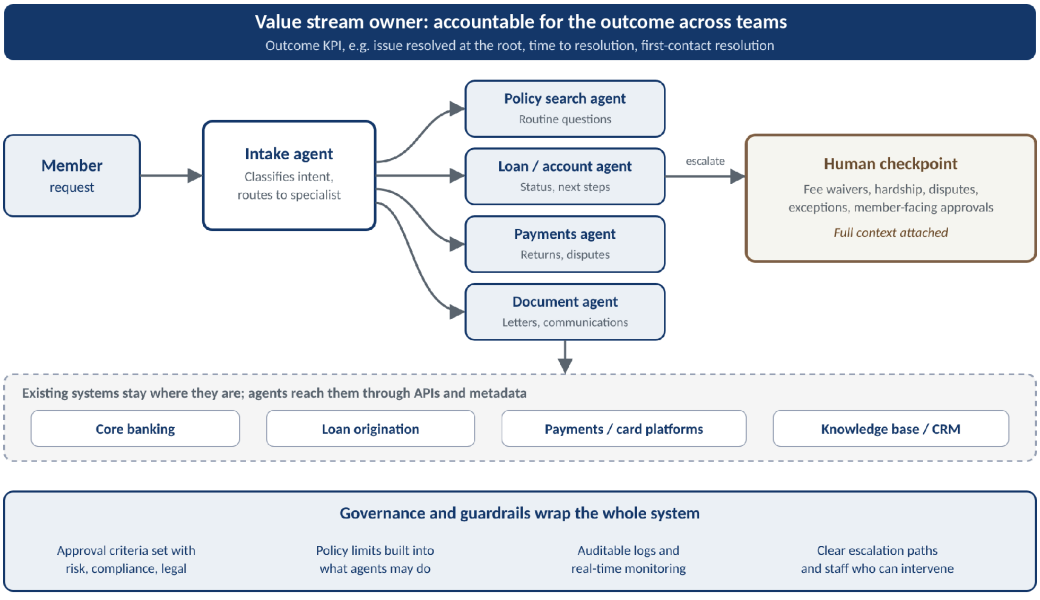


Figure 1. Reference architecture for an outcome-oriented agent system (illustrative).

Table 1. Illustrative use-case scoring rubric

Criterion	Question	Example indicators
Outcome value	Does it measurably improve the member or employee experience?	Resolution time; satisfaction; recurrence
Efficiency and scale	Does it cut effort and scale without proportional cost?	Cost per case; hours returned; volume
Risk reduction	Does it reduce errors, exposure, or control failures?	Error rate; exceptions; losses avoided
Feasibility	Are data, systems, and APIs accessible, and is the process stable?	Data quality; system access
Risk and autonomy tier	How consequential are the actions? (scored inversely)	Tier 1 to 4; regulatory sensitivity
Strategic fit	Does it advance strategy and reuse shared components?	Roadmap alignment; reuse

Practice 3: Fix the process before automating it

Automation applied to a flawed process produces faster flaws. Using Lean Six Sigma discipline (George, 2002), map the current state, baseline cycle time, defects, and rework, remove non-value-adding steps, and standardize the work before an agent touches it. The DMAIC cycle provides the backbone, and a control plan keeps gains from eroding. Then codify the business logic in four layers: the trigger and intended outcome, the decision rules and their sources, the systems and data consulted, and the exception and escalation conditions. This is the hardest and most valuable work, because the knowledge is usually tacit and varies between teams.

Do not wait for perfect data (Mantia, Chatterjee and Lee, 2025). Agents can operate across existing, siloed systems provided they have access through APIs, usable metadata, and information that is timely and accurate enough to act on; a single source of truth is helpful but not a precondition. Start narrow: one well-understood journey, validated end to end, yields reusable patterns and a short feedback loop. The mapping itself often exposes stale data and dependence on individuals, and fixing those improves the whole organization.

Practice 4: Match technology and sourcing to the problem

Most institutions will use several families of technology, and each fits a different problem (Table 2). Beyond agents, a limited set of approved enterprise AI assistants should be rolled out with acceptable-use guidance, role-based training, and adoption tracking, so productivity gains are measured rather than anecdotal.

Table 2. Technology-selection guide

Technology	Best suited to	Limits
Rules-based automation / RPA	Stable, high-volume, rule-driven tasks	Brittle to change; poor with ambiguity

Technology	Best suited to	Limits
Generative AI assistant	Individual productivity: drafting, summarizing, search	Needs human review; no direct system action
AI agent	Multi-step, goal-driven work across systems	Needs clear logic, guardrails, and monitoring
Analytics / machine learning	Prediction and pattern detection on structured data	Needs quality data and model validation

Few institutions will build everything, so decide build, buy, or partner by explicit criteria: whether the capability differentiates you, how much proprietary process knowledge it requires, speed, available talent, portability, and multi-year cost. In a buy-and-partner model, retain ownership of the outcome, the business logic, and the data. Treat vendors as governed partners under third-party risk expectations (Board of Governors of the Federal Reserve System et al., 2023): structured due diligence on security, data handling, and model provenance; contract terms on data use (including whether your data may train vendor models), audit rights, incident notice, and service levels; transparency on model changes; an exit plan; and performance reviews against agreed measures.

Practice 5: Govern with graduated autonomy

Proportionate governance neither stifles useful automation nor under-protects consequential decisions. Table 3 proposes four autonomy tiers, offered as a practitioner heuristic rather than a standard. Set the tier per action, not per agent, and revisit it as performance evidence accumulates.

Table 3. Four-tier autonomy model

Tier	Agent behavior	Typical use	Principal controls
1. Inform	Presents information; takes no action	Answering questions from approved policy	Approved sources; logging; accuracy sampling
2. Recommend	Proposes an action; a person decides	Drafting a letter; suggesting a resolution	Mandatory review; reasons shown
3. Act with approval	Executes once a person approves	Filing a request; updating a record	Approval records; reversibility; role-based permissions
4. Act autonomously	Completes defined, low-risk actions alone	Routine status updates; scheduling	Tight scope; real-time monitoring; kill switch

Anchor governance in recognized frameworks rather than inventing your own. The NIST AI Risk Management Framework organizes the work into Govern, Map, Measure, and Manage (NIST, 2023), its generative AI profile addresses risks specific to those systems (NIST, 2024), and model risk management principles remain relevant where AI informs decisions (Federal Reserve and OCC, 2011). Build in approval criteria agreed with risk, compliance, legal, data governance, and security before launch; policy constraints so agents cannot exceed their remit; least-privilege data access; recorded reasoning for explainability; auditable logs with real-time monitoring; and clear escalation paths. Involve control partners from the first design conversation, not the last.

Practice 6: Lead adoption, measure value, and build the team

Adoption is a change-management problem before it is a technical one. Employees need enough literacy to understand what their digital teammates do and when supervision is required, and leaders need to address fears about jobs and identity directly. Demonstrate rather than describe, invite staff to propose use cases, and involve frontline teams in tool and vendor evaluation. A structured method such as ADKAR helps plan communication, training, and reinforcement. Measure four categories together: outcomes (the mission KPI, recurrence of root-cause issues), efficiency (cycle time, cost to serve, share resolved without intervention), quality and risk (accuracy on sampled cases, escalation appropriateness, audit findings), and adoption (active use, employee confidence). Never report efficiency without the paired quality and risk measures. Staff the function with a deliberate blend of process improvement, solution integration, data, risk and control, vendor management, and change leadership, and run a regular horizon scan so adoption is neither reactive nor complacent.

4. Applied experience

I have applied these practices to AI and automation initiatives at a financial institution, each built on a vendor platform. In each case the work centered on the same disciplines: defining the outcome and its KPIs before go-live, codifying the business logic the platform would execute, setting the autonomy tier and controls with risk and compliance, and planning how the data would reach the institution's analytics environment.

- Agentic, conversational, and voice AI for member service. The design questions mirror Figure 1: which inquiries the AI resolves, which route to staff, what actions it may take, and how quality is tracked across channels. The same platform supports outbound collections outreach.
- AI-assisted origination for lending and deposit accounts. AI supports the workflow while credit decisions and exceptions stay with qualified people. The key work was codifying policy and decision rules, preserving explainability, and tracking funnel conversion and speed.
- Behavioral-biometric decisioning for fraud risk. Behavioral signals inform fraud decisions. The emphasis was on thresholds and response actions, balancing loss reduction against member friction, and a financial KPI: avoided fraud losses of at least 150% of annual cost.

Table 4. KPIs defined for each initiative

Initiative	KPIs
Conversational and voice AI (member service)	Service levels; call volume handled by staff; NPS; cost per contact (department expense divided by agent-supported calls, chats, and emails)
Conversational AI (collections)	Contact rate; payment acceptance rate
AI-assisted lending	Look-to-book (funded to total applications); funded-to-approved rate; time to approve (minutes); time to fund (days); share of loans with automated decisions
AI-assisted deposit origination	New-account pull-through (application start to funded account); speed of account opening (minutes); products per new member

Initiative	KPIs
Behavioral-biometric decisioning	Fraud-loss avoidance of at least 150% of annual cost

Two lessons recur. First, plan the data path early: vendor-supplied metrics typically live in the vendor's portal, and integrating them into the institution's analytics environment is a workstream in its own right. Second, some outcomes lag deployment. Fraud avoidance, for example, cannot be quantified on day one, so pair the lagging financial KPI with leading indicators such as alert volumes, analyst dispositions, and false-positive rates.

5. Roadmap

Sequence the work in phases governed by exit criteria rather than dates: (1) Foundation: charter, decision rights, intake, rubric, and baseline governance with risk and compliance. (2) Select: rank use cases, choose one high-friction journey, and name the value stream owner and KPI. (3) Improve and codify: map and improve the process, document the business logic, confirm data access, and complete vendor due diligence. (4) Pilot: build a narrow slice, assign autonomy tiers, and run in shadow or recommend mode. (5) Operate and adopt: release with monitoring, train staff, and track value against baseline. (6) Scale: extend to adjacent journeys, reuse patterns, and promote actions to higher tiers as evidence supports. Running agents in shadow mode before granting autonomy produces the evidence needed to promote actions responsibly.

6. Risks and conclusion

The framework does not remove risk. Language-model agents can produce plausible but wrong outputs, so tiering, sampling, and human checkpoints remain essential. Over-delegation is a risk of its own, because reviewers relax as agents perform well. Vendor dependence can limit flexibility and obscure model behavior, codified logic can ossify without ownership, and regulatory expectations continue to evolve. The evidence base for agent deployments in regulated sectors is still young, so treat this framework as a structured starting point to be tested, not a validated model.

The promise of AI agents is not faster execution of existing tasks. It is organizing work around the outcomes members care about, across boundaries that have traditionally frustrated them. Institutions that treat AI as a managed capability, with clear ownership, disciplined process improvement, sound sourcing, graduated governance, and sustained investment in people and measurement, will scale fastest and most safely.

References

Board of Governors of the Federal Reserve System, FDIC, and OCC. (2023). Interagency guidance on third-party relationships: Risk management.

Board of Governors of the Federal Reserve System and OCC. (2011). Supervisory guidance on model risk management (SR 11-7 / OCC 2011-12).

George, M. L. (2002). Lean Six Sigma: Combining Six Sigma quality with Lean speed. McGraw-Hill.

Mantia, L., Chatterjee, S., & Lee, V. S. (2025). Designing a successful agentic AI system. Harvard Business Review, October 2025.

National Institute of Standards and Technology. (2023). AI Risk Management Framework (AI RMF 1.0). NIST AI 100-1.

National Institute of Standards and Technology. (2024). AI RMF: Generative AI Profile. NIST AI 600-1.