DOI: 10.30280/ISPJ.202608_1(1).0001 | ISSN: 3135-4440 (Provisional) | Available online 18/8/2026

Submitted on 20/4/26 | Accepted on 15/6/26 after two rounds of revision | Accepting Senior Editor: Robert M Davison

ispj Information Systems Practice Journal

香港城市大學出版社
City University of Hong Kong Press

2026 ARTICLE 3

Managing at the Jagged AI Frontier: Delegation Contracts for Workflow-Embedded Agentic AI

Jonny Holmström [1]

Abstract

Managing agentic AI at the jagged technological frontier is fundamentally a delegation problem, not a resource-allocation problem. In this article, agentic AI refers to AI systems embedded in organizational workflows that can interpret tasks, use tools or data sources, recommend or execute actions, and materially shape records, transactions, communications, or decisions. The problem is urgent because recommendation can become execution through interface defaults, permissions, and process redesign before governance catches up. When capability is uneven across adjacent workflow steps, organizations cannot safely infer authority from apparent competence. Success in one step may conceal failure in the next, while fluent outputs, shifting boundaries of reliability, and organizational overreliance make implicit delegation hazardous. The article uses the jagged-frontier insight as a starting point but extends it from individual knowledge-work assistance to operational workflow governance. It introduces the delegation contract as a workflow-specific authorization record for bounding machine authority. The contract specifies purpose boundaries, permissions, evidence requirements, escalation triggers, autonomy tiers (Tier 1 Advisory, Tier 2 Conditional Automation, and Tier 3 Delegated Execution), and recertification rules, making machine authority explicit, reviewable, and revocable. The article shows when such contracts are unnecessary, useful, or essential and explains how they can be integrated with governance, risk, compliance, agile, DevOps, low-code/no-code, and open-source practices. Effective management requires not simply using more AI but continuously monitoring and periodically renegotiating delegated authority as models, tools, and work practices evolve.

KEYWORDS

agentic AI; AI governance; delegation contracts; workflow automation; jagged technological frontier; organizational authority

[1] **Jonny Holmström**
Swedish Center for Digital Innovation
Department of Informatics
Umeå University
Address: Umeå, Sweden
Email: jonny.holmstrom@umu.se

Jonny Holmström. "Managing at the Jagged AI Frontier: Delegation Contracts for Workflow-Embedded Agentic AI" *Information Systems Practice Journal*. 2026: 1:1.

Introduction

Managing agentic AI has become urgent because these systems increasingly sit inside operational workflows rather than outside them. Once connected to enterprise systems, data repositories, workflow tools, or communication channels, the difference between advice and action can become a matter of permissions, interface defaults, or process redesign. A model-generated recommendation can therefore become an organizational message, case classification, record update, routing decision, approval step, or customer commitment. The managerial challenge is not only model risk in the abstract; it is the quiet conversion of machine output into organizational action before decision rights, evidence requirements, escalation paths, and accountability have been specified.

By agentic AI, we mean AI systems embedded in organizational workflows that can interpret a task, use tools or data sources, recommend or execute actions, and materially shape records, transactions, communications, or decisions (Baird & Maruping, 2021; Jarrahi & Ritala, 2025). This is narrower than every business use of generative AI (GenAI). A manager prompting ChatGPT, Claude, Gemini, or another large language model (LLM) to summarize a document, edit a memo, or find references is usually using AI as an individual productivity aid. By contrast, a workflow-embedded AI system that retrieves records, drafts operational communications, classifies cases, updates fields, triggers handoffs, or sends messages is receiving organizational authority. Delegation contracts matter primarily in the second situation: recurrent organizational workflows where AI output can become organizational action.

The focus is not the label attached to the technology. A deterministic algorithmic agent, a digital worker, or a GenAI agent may each require similar controls if it receives authority to act in a workflow. Agentic AI becomes distinctive here because its capability boundaries are uncertain and fluent outputs can make unsupported action look decision-ready. That urgency is sharpened by the jagged technological frontier. A system may summarize records well, classify routine cases well enough, and still mishandle the adjacent exception that carries legal, financial, or reputational consequences. At what Dell'Acqua et al. (2026) call the jagged technological frontier, success in one workflow step is weak evidence for the next: similar-looking tasks can sit on opposite sides of the frontier, while fluent output makes boundary failures difficult to detect in time.

Dell'Acqua et al. (2026) provide the capability diagnosis, not the organizational setting for this article. Their field experiment with management consultants shows that AI capability varies across tasks that look similar to knowledge workers. This article extends that insight into operational workflow management. The contribution is to show that uneven capability becomes a problem of delegated authority when AI systems are embedded in recurrent workflows: what the system may do, when it must stop, who remains accountable, and how authority can be withdrawn when capability or context changes.

Managers therefore need to make delegation explicit. The central question is not simply whether the model produces useful text or whether the organization has access to an AI resource. It is what authority is being transferred to the agent, under what constraints, on what evidence, with what escalation routes, and how that authority can be limited, suspended, or withdrawn as capability, context, and risk shift. Familiar routines for rollout, feedback, and refinement (Grashoff & Recker, 2024) remain useful, but they do not by themselves specify decision rights. Treating agentic AI primarily as a resource can obscure those rights and misframe failure as a problem of utilization or compliance rather than misdelegation.

This is the practical problem that remains under-addressed. Existing discussions of AI governance often point to policies, model evaluations, risk classifications, human oversight, or post-hoc audits (Holmström, 2026). These instruments matter, but they can leave a crucial operational gap: who or what is authorized to act in a particular workflow, how far that authority extends, what evidence must support action, when the agent must stop, and who is accountable for recertifying the arrangement as the frontier moves. Without an explicit delegation mechanism, organizations may continue to expand agentic authority through local optimization while believing that they are only improving efficiency.

To close this gap, managers need a workflow-specific authorization record: the delegation contract. Here, 'contract' does not mean a legal agreement. It means an internal authorization record that specifies the terms under which machine output may become organizational action. It makes machine authority explicit, bounded, reviewable, and revocable. This article asks how organizations should govern agentic AI when machine competence is uneven across adjacent tasks and authority can move from recommendation to action within the same workflow. It makes three contributions. First, it reconceptualizes managing agentic AI as a delegation problem rather than a resource-allocation problem. Second, it introduces the delegation contract as a workflow-level authorization record for specifying, bounding, and periodically recertifying machine authority. Third, it identifies six governance dimensions of bounded delegation (purpose boundaries, permissioning, evidence, escalation, autonomy tiers, and recertification) and presents an agentic AI delegation contract.

The remainder of the paper is as follows: The next section shows why the jagged frontier creates four distinct management problems for agentic AI. The following section explains why these problems are better understood through delegation than through resource deployment. The third section develops the delegation contract as the managerial resource that follows from that shift.

Management challenges at the jagged AI frontier

The jagged frontier matters because it replaces a comforting picture of steady improvement with a harder operational fact: capability varies across adjacent tasks, and the boundary between reliable performance and subtle failure is difficult to see in advance. In Dell'Acqua et al.'s (2026) field experiment, GPT-4 boosted speed and quality on tasks inside the frontier while reducing correctness on an outside-the-frontier managerial task. The central lesson is not merely that AI sometimes fails, but that perceived quality and actual correctness can diverge within the same workflow.

To make the problem concrete, consider frontline billing support: the first-line handling of routine customer billing inquiries such as invoice-status requests, standard fee explanations, payment-routing questions, basic account-information checks, and case routing. At Tier 1 Advisory, an agent may retrieve account history, draft a response, and recommend a routing code. These activities can appear benign because a human still decides what leaves the organization. The risk changes when workflow design gives the same system one-click sending rights, write access to case fields, or automatic authority over a narrow class of 'routine' tickets. Those moves shift the agent toward Tier 2 Conditional Automation or Tier 3 Delegated Execution. At that point the agent is no longer merely producing text for an employee; it is exercising bounded organizational authority inside a live process.

Table 1 uses this example to define the terms that recur throughout the article. The point is not that billing support is unusually risky. It is that adjacent tasks within the same ticket can involve very different forms of authority. Reading account data, drafting a reply, sending a message, updating a field, waiving

Term	Plain-language meaning	Example in frontline billing support
Agentic AI	Workflow-embedded AI that can use tools, data, or permissions to recommend or execute actions.	Retrieves account records, drafts a reply, recommends a routing code, updates a case field, or sends a response under defined conditions.
Authority	The organizational decision or action the AI is allowed to perform.	Draft only; classify a ticket; send a standard invoice-status response; never waive a fee without human approval.
Permissioning	The technical access that makes authority possible.	Read-only access to invoices versus write access to account fields, case status, or outbound-message systems.
Autonomy tier	The level of independent action allowed.	Tier 1 Advisory drafting, Tier 2 Conditional Automation for straightforward invoice-status cases, or Tier 3 Delegated Execution for tightly specified subcases.
Escalation trigger	A condition requiring the AI to stop and route the case to human review.	Missing ledger match, contradictory records, hardship claim, fraud signal, repeated complaint, policy conflict, or low confidence.
Recertification	Periodic reapproval of delegated authority after change or evidence of drift.	Review after a model update, prompt change, billing-policy change, new integration, incident, or override-pattern shift.

Table 1. Key terms for managing agentic AI, illustrated through frontline billing support

Autonomy tier	What the AI may do	What remains human-controlled	Frontline billing support example
Tier 1 Advisory	Retrieve information, draft text, provisionally classify, or recommend a routing code.	A human reviews, edits, sends, updates records, and accepts accountability before any organizational action occurs.	AI drafts an invoice-status reply and cites the invoice record; a human sends or revises it.
Tier 2 Conditional Automation	Execute narrowly defined actions only when pre-specified evidence, confidence, and exception checks are satisfied.	Humans handle exceptions, ambiguous cases, hardship claims, fee waivers, policy conflicts, and repeated complaints.	AI sends a standard invoice-status message only when the ledger match is complete, the policy text is approved, and no escalation trigger appears.
Tier 3 Delegated Execution	Complete a defined class of routine actions without pre-action human approval, subject to logging, monitoring, limits, and revocation.	Humans retain accountability, monitor drift, investigate incidents, and recertify or revoke authority.	AI closes a narrow class of routine invoice-status cases after sending an approved response, but cannot waive charges, alter terms, or make legal representations.

Table 2. AI autonomy tiers for workflow-embedded agentic AI

a fee, and escalating a hardship claim may occur in the same interface, but each requires different permissions, evidence, autonomy, escalation, and recertification conditions.

Movement from Tier 1 to Tier 2 or Tier 3 is not an adoption milestone; it is a new authorization decision (see Table 2 for details). Each increase in autonomy requires workflow-specific evidence, updated permissions, defined escalation triggers, monitoring thresholds, and recertification triggers.

For managers, this means AI governance cannot be a one-time approval exercise. The boundary shifts as models, prompts, data, interfaces, and work practices change. In workflow-embedded agentic AI, the distinction between development and use can blur as teams revise prompts, retrieval sources, handoff routines, and interface defaults while the system is already in operation (Waardenburg & Huysman, 2022). Sundberg and Holmström (2024) similarly show that value creation depends on fusing domain knowledge with machine knowledge inside routines. Governance must therefore track changing conditions of use rather than assume stable capability.

Taken seriously, the jagged AI frontier changes what managers must govern. Success on one task does not justify delegation on adjacent tasks because workflows mix tasks with different epistemic demands, reversibility, and detectability of error. With the frontline billing example in view, the four implications below unpack that point by showing how adjacency creates false confidence, how plausible outputs become organizational hazards, why the frontier is hard to locate ex ante, and why overreliance is a systems property.

1) Workflow adjacency creates false confidence

For agentic deployments, the inside/outside-frontier pattern is amplified because adjacent workflow steps share inputs, vocabulary, artifacts, and interfaces. When an agent performs well on one step, it becomes easy to assume it can also handle the next. The jagged frontier is precisely what makes that inference unsafe.

Workflow decomposition amplifies the problem. Teams automate the steps that look tractable, but 'small' is not the same as 'low-stakes.' A micro-step can carry disproportionate epistemic risk, and once it is delegated, downstream steps can reinforce the error through additional coherence and polish. AI is not merely wrong in these moments; it can be wrong in ways that look professionally right.

Ethnographic work on AI development shows that adjacency often hides different kinds of knowledge. In van den Broek, Sergeeva, and Huysman's (2021) study of machine learning for hiring, the boundary between automatable inference and expert judgment was not knowable upfront; it emerged through hybrid practice and mutual learning. For agentic AI, that means nearby successes are weak evidence about the hardest adjacent steps. Adjacency also has a governance dimension. Mayer et al. (2025) characterize GenAI as a boundary resource that expands output while forcing governance to adapt. Inside firms, success in one domain can therefore become an argument to stretch scope in another faster than controls recalibrate.

2) 'Plausible' failure modes become organizational hazards

A second problem is that AI output can be coherent, polished, and wrong. When systems are agentic, this becomes a risk-propagation problem: confident error travels farther because it is easier to accept and harder to contest.

Plausibility is hazardous because AI-produced artifacts can travel across systems as if their assumptions were already validated. Generative outputs can become portable 'data objects' that compress uncertainty into something actionable and then travel across systems and teams (Alaimo & Kallinikos, 2022; 2024). Once such an AI-produced object becomes a reference point, its origin conditions and assumptions are easily forgotten.

3) The frontier is hard to locate ex ante – and does not stay put

The jagged-frontier result also shows that workers can misjudge where AI will help versus harm. For management, the real question is not whether the model can do the task, but under what conditions it does the task safely enough for a specific workflow authority. That question cannot be answered once and for all.

In workflow-embedded AI, development and use blur as teams adjust prompts, retrieval sources, interface defaults, and integrations while the system is already part of operations (Waardenburg & Huysman, 2022). Van den Broek et al.'s (2021) ethnography offers the micro-level corollary: hybrid practices redraw the frontier in use rather than fixing it once in advance.

The frontier also shifts because complements accumulate around the model: prompt libraries, retrieval pipelines, plugins, and fine-tuned variants alter what the agent can do in practice. Once agentic AI becomes shared infrastructure, local innovations redraw both capability and dependency faster than static policy can keep up.

4) Overreliance is not a user flaw; it is a systems property

Outside the frontier, users may accept outputs that feel good enough, invest less in independent checking, and defer to a plausible answer even when closer supervision is needed. In agentic workflows, this reliance pattern becomes organizational rather than merely individual: the workflow itself can make the machine output the easiest path forward.

That systems view matters because reliance is engineered into workflow design. Interface defaults compress search, drafting, checking, and execution into a single path of least resistance; throughput targets reward fast acceptance; and the persuasive surface quality of AI output suppresses the cues that would otherwise trigger scrutiny. Under those conditions, verification is not merely neglected. It becomes the slower, socially awkward, and economically disfavored option.

Overreliance becomes more likely when AI turns incomplete or uncertain information into fluent, decision-ready outputs. Managers should therefore design verification into the workflow instead of relying on users to notice misplaced confidence. Checks should be attached to the moments where action occurs: before an outbound message is sent, a field is updated, a recommendation is routed, or an exception is closed. The same system that productively augments work in one moment can invite premature closure in the next, especially when fast, fluent output suppresses the cues that would otherwise trigger skepticism.

Overreliance also accumulates across handoffs. As teams redesign processes around agentic output, upstream workers do less primary analysis and downstream workers assume that prior checks have already occurred. Verification becomes everyone's responsibility and no one's explicit task. Over time,

escalation thresholds drift upward, tacit expertise atrophies, and intervention becomes harder precisely because humans remain formally accountable while becoming less substantively engaged.

Taken together, the four challenges above point to a common conclusion: the central management problem is not simply whether organizations possess AI capability, but how they delimit, supervise, and periodically renegotiate machine authority. Once failure is understood as misdelegation rather than mere misuse, the conceptual shift from a resource view to a delegation view becomes necessary. The next section develops that shift; the section after that turns it into a managerial resource through the delegation contract.

Delegation as a useful governance lens

If the jagged frontier makes capability uneven and difficult to predict, then 'AI as a resource' is an incomplete abstraction. Resource-oriented capability arguments are useful when a technology can be treated as a relatively stable capacity that is procured, allocated, and governed through standard controls (Mikalef & Gupta, 2021). But that assumption breaks down when the system interprets objectives, sequences actions, and behaves differently across adjacent tasks. Following Baird and Maruping (2021), the relevant issue is delegation: the transfer of decision rights, action permissions, and accountability relationships between human actors and AI-enabled systems under conditions of uncertainty. Ribes et al. (2013) similarly show that delegation is not just a matter of efficiency; it reorganizes work, redistributes responsibility for decision-making, and alters what becomes visible or invisible in organizational practice.

Seen this way, the managerial question is not whether the firm possesses AI capability, but what kind of authority it is passing to the agent, with what reversibility, and under what conditions that authority must be constrained or withdrawn. Agentic AI therefore requires something different: structured delegation in a form that is explicit, workflow-specific, reviewable, and revocable.

Where this framework applies – and where it does not

The delegation-contract lens is most useful where agentic AI participates in recurrent workflows, can materially shape consequential actions, and has access to operational systems. It is less central for low-stakes exploration, one-off brainstorming, ad hoc drafting, or settings where no organizational authority is transferred. The practical test is simple: can the AI's output become an organizational action without substantial human rework? If yes, ask what has been delegated. Table 3 translates this test into three levels of delegation-contract intensity.

This also clarifies what the delegation-contract view adds beyond adjacent governance tools. Model evaluations test performance; enterprise AI policies state general rules; and AI impact assessments surface broader legal, ethical, or risk considerations. A delegation contract works one level closer to execution. It specifies what this agent may do in this workflow, with what permissions, evidence standards, escalation triggers, and review cadence. Its distinctive contribution is workflow-specific authorization rather than organization-wide principle setting or ex ante benchmarking. That difference is why, for recurrent workflows in which AI output can become organizational action, a contract is necessary rather than optional: without a workflow-level artifact that authorizes and limits action at the point of execution, machine authority migrates informally through interface design, permissions, and process redesign.

Use type	Example	Delegation-contract implication
Casual or exploratory use	Individual brainstorming, editing, reference search, or one-off drafting.	No full contract is needed if no organizational authority is transferred and the user substantially reviews the result.
Business workflow use	Billing support, claims handling, procurement routing, HR case classification, or customer-case triage.	A delegation contract should be treated as required before recurrent operational use or autonomy expansion.
Acute or high-stakes use	Safety-critical control, regulated decisions, irreversible financial or legal action.	A contract is required but not sufficient; additional validation, assurance, testing, and formal controls are needed.

Table 3. Criticality and delegation-contract intensity

	Resource view of AI capability	Delegation-contract view of workflow-embedded agentic AI
Basic metaphor	AI is a tool, asset, or capacity to be allocated.	AI is a delegated actor receiving bounded authority.
Core managerial problem	How do we deploy, scale, and utilize it efficiently?	What exactly may it decide or do, under what conditions, and with what limits?
Assumption about capability	Capability is stable enough to generalize across similar tasks.	Capability is uneven, jagged, and task-specific. Similar-looking tasks may require very different levels of trust.
Unit of management	The model, platform, seat, license, or application.	The task, workflow step, decision right, and action permission.
Primary performance lens	Utilization, adoption, throughput, and cost savings.	Correctness, reversibility, calibration, escalation quality, and blast-radius control.
Main risk model	Misuse, downtime, noncompliance, or poor integration.	Overdelegation, confident-but-wrong action, persuasive error, and boundary crossing.
Human role	User or operator of a resource.	Supervisor, exception-handler, reviewer, and ultimately accountable principal.
What governance focuses on	Policies, access rights, procurement, and standard controls.	Scope of authority, evidence requirements, escalation triggers, approval gates, and kill switches.

Table 4. Resource view versus delegation-contract view of agentic AI

Table 4 summarizes the shift from a resource view to a delegation-contract view of workflow-embedded agentic AI and clarifies why the two frames generate different governance questions, performance criteria, and risk assumptions.

The point of the contrast is not semantic. Each view directs managerial attention toward different objects: the resource view toward rollout, scale, and utilization; the delegation-contract view toward bounded authority, reversibility, escalation, and accountability. Once the unit of management shifts from the tool to the delegated decision, the relevant metrics, risks, and human roles shift with it.

Concretely, delegation contracts reorient management from utilization metrics to authorization design, from generic controls to task-specific routing, from output quality alone to operational safety, and from episodic governance to continuous monitoring and recertification. Framed this way, the contract is not an administrative overlay added after deployment; it is the operating logic through which bounded autonomy is maintained as the frontier shifts.

Capability is not authority. An organization may have evidence that a model can summarize documents, classify requests, draft recommendations, or execute a narrow sequence of tool calls. None of that, by itself, justifies granting the system the right to alter records, send commitments, approve exceptions, or trigger irreversible downstream actions. Capability answers whether the system can generate a useful output. Authority answers whether the organization is prepared to let that output change the state of the world. At the jagged frontier, those are not the same question. Delegation therefore cannot be inferred from apparent competence; it must be separately designed and bounded (Dell'Acqua et al., 2026; Baird & Maruping, 2021).

Authority often migrates through interface design. Organizations delegate unintentionally through defaults, permission settings, throughput targets, and workflow redesigns that make acceptance of machine output the path of least resistance. Once an agent drafts in the same interface from which a user can send, or classifies in the same system that can trigger case routing, recommendation quietly shades into execution. The risk is not only excessive trust in the model; it is operational authority being granted without being named as such. Ribes et al. (2013) show that delegation is embedded in artifacts and routines, not only in formal job descriptions. A delegation contract makes those latent transfers visible and discussable.

Governance shifts when the unit of management is the delegated decision. Under a resource lens, managers ask how many seats are used, how much time is saved, and which functions have been automated. Those metrics matter, but bounded machine authority requires different questions: Which actions are reversible? Which errors are detectable before harm propagates? Which decision points create financial, legal, reputational, or safety exposure? How quickly can authority be withdrawn when operating conditions change? This reframing aligns with Jarrahi and Ritala's (2025) principal-agent perspective: the challenge is not only to extract productive effort from the system, but to manage divergence between delegated objectives and situated performance.

Contracts coordinate business, technical, and risk teams. Frontline managers tend to see AI through workflow efficiency and service levels. Technical teams think in terms of models, prompts, retrieval pipelines, integrations, and tool reliability. Risk, legal, compliance, security, and audit teams focus on exposure, auditability, and policy conformance. The contract creates a common unit of analysis: this agent, in this workflow, with these permissions, evidence rules, escalation triggers, monitoring obligations, and review intervals. In that sense, it functions as a boundary object linking domain knowledge, technical knowledge, and governance knowledge (Sundberg & Holmström, 2024; Mayer et al., 2025).

Contracts support recertification because the frontier moves. Governance cannot stop at ex ante approval. Organizations need a way to convert operational experience into revised authority boundaries. A delegation contract specifies what should be observed: override rates, escalation patterns, incident types, evidence failures, and changes in task mix. These are not just performance indicators; they are signals about whether the current terms of delegation still fit the work. In workflow-embedded agentic AI, development and use can blur as prompts, retrieval sources, integrations, and work practices evolve (Waardenburg & Huysman, 2022). The line between automatable inference and expert judgment can also emerge only through practice (van den Broek et al., 2021). Contracts should therefore be treated as living instruments that accumulate organizational learning and trigger renegotiation when models, tools, or workflow conditions change.

The goal is accountable innovation, not caution for its own sake. Delegation contracts do not exist to slow useful experimentation or imply that humans always outperform machines. Their purpose is to preserve correspondence between authority and answerability. Human managers remain accountable for outcomes even when they did not directly execute the steps that produced them. A well-designed contract specifies not only what the agent may do but who must inspect what, when exceptions become mandatory, and how revocation can be enacted without organizational paralysis. Because agentic systems produce artifacts that look decision-ready, requiring evidence traces, source visibility, and explicit escalation conditions resists the slide from fluent output to unearned authority (Alaimo & Kallinikos, 2022; 2024).

The delegation contract as a managerial resource

A delegation contract gives the delegation lens an operational form, but it should be understood first as a managerial resource rather than as a legal instrument. Its function is to help managers decide whether demonstrated AI capability should be converted into organizational authority. The contract asks: What may the agent do? What must it not do? What evidence must exist before action? When must it stop and escalate? Who remains accountable? When must the arrangement be reviewed, narrowed, or withdrawn?

This distinction matters because capability and authorization are different managerial problems. An agent may perform well on a narrowly tested task, yet still lack legitimate authority to act in a production workflow. Conversely, an organization may specify careful controls, while the agent has not yet demonstrated sufficient reliability for the delegated task. The delegation contract helps managers connect these two questions without collapsing one into the other.

Figure 1 summarizes this logic as a delegation-readiness matrix. The vertical axis captures AI-workflow capability fit: whether the agent has demonstrated reliable performance in the specific recurrent workflow under consideration. The horizontal axis captures delegation-contract maturity: whether authority, permissions, limits, evidence, escalation, monitoring, accountability, and review have been explicitly specified.

The target state is authorize and monitor: contracted delegation. Capability has been demonstrated in the defined workflow, and the organization has made the terms of authority explicit; managers can deploy within those limits, monitor outcomes, and recertify or revise the arrangement as conditions change. The most hazardous quadrant is unsafe scaling risk: capable but under-contracted. Here, apparent usefulness can encourage scaling even though authority, limits, escalation, or ownership

	Low delegation-contract maturity	High delegation-contract maturity
High AI-workflow capability fit	Unsafe scaling risk: capable but under-contracted	Authorize and monitor: contracted delegation
Low AI-workflow capability fit	Exploratory use: experiment only	Test before authorization: governed pilot

Figure 1. Delegation-readiness matrix: aligning AI capability with managerial authorization

remain informal; the contract should therefore be written or revised before authority expands. The test-before-authorization quadrant is safer but still incomplete: controls are specified, but capability has not yet justified broader authority, so managers should continue testing, narrow the scope, redesign the workflow, or retain human approval. The exploratory-use quadrant marks the boundary of the framework: neither workflow fit nor delegation terms have been established, so one-off prompting for brainstorming, editing, reference search, or exploration should remain low-stakes and outside operational authority.

Managers can use the matrix at three moments. Before deployment, it asks whether the proposed use case should be kept in exploratory use, enter a test-before-authorization pilot, or be authorized and monitored as contracted delegation. When expanding autonomy, it asks whether demonstrated performance justifies additional authority or whether the organization is simply normalizing informal delegation. After a material change, such as a model update, new integration, data change, policy change, or incident pattern, it asks whether the current contract should be recertified, narrowed, or withdrawn.

The delegation contract is therefore more than a checklist. It is a diagnostic record because it reveals when productivity has outrun governance. It is an authorization record because it records what authority has actually been granted. It is a coordination record because business owners, technical teams, risk functions, auditors, and frontline staff can inspect the same terms. It is also a recertification record because it defines the conditions under which delegated authority must be reviewed or revoked. The accountable workflow manager should own the contract; technical teams should validate capability and integrations; risk, legal, compliance, security, and audit owners should review the authorization terms; and a senior business owner should approve the autonomy tier before the agent receives write access, external communication rights, or authority to trigger consequential workflow steps.

Using the delegation contract in existing governance routines

Managers should create the contract before a workflow agent receives write access, external communication rights, or authority to trigger consequential workflow steps. The accountable workflow owner should own it because that person is closest to the operational consequences. The technical team supplies evidence about capability, integrations, data access, logging, and failure modes. Risk, legal, compliance, security, and audit functions review the authorization terms. A senior business owner approves the autonomy tier. Operations then monitors incidents, overrides, and escalation patterns, and the contract is recertified after material changes.

Governance moment	Managerial question	Contract output
Before deployment	Is this experimental, a governed pilot, or delegated execution?	Initial autonomy tier and owner.
Before write access or external communication	What authority is actually being granted?	Authorized and prohibited actions; evidence requirements.
After model, workflow, data, or policy change	Has the frontier shifted for this workflow?	Recertification, narrowed scope, or new pilot.
After incident, override drift, or repeated escalation	Should authority be suspended, narrowed, or revoked?	Revised contract, kill-switch action, or human-only fallback.

Table 5. Using the delegation contract in existing governance routines

A practical sequence is: (1) the workflow owner proposes the AI use case; (2) the technical team validates capability and integration risk; (3) risk, legal, compliance, security, or audit owners review authority, evidence, escalation, and revocation terms; (4) the business owner approves the autonomy tier; (5) operations monitors overrides, incidents, and escalation patterns; and (6) the contract is recertified after model, workflow, data, policy, or incident changes. Table 5 summarizes the corresponding governance moments, managerial questions, and contract outputs.

In practice, managers can populate the contract through six dimensions of bounded delegation. These dimensions translate the matrix into operating decisions.

Purpose and task boundaries. Define the agent's remit in task-level terms, not capability slogans. 'Help with customer support' is too broad; 'draft replies for frontline billing inquiries using approved knowledge-base articles' is closer. Because similar-looking tasks can sit on different sides of the jagged frontier, boundaries must be specific enough to route edge cases away from autonomy.

Authority and permissioning. Separate read access from write access, and define which systems the agent may affect. In agentic contexts, the riskiest failures are not wrong text but wrong commitments: sending, buying, deleting, approving, merging, closing, or recording something that the organization must later treat as consequential. Authority should therefore be scoped to contain the blast radius of failure.

Evidence and justification requirements. Specify what evidence must exist before the agent may act. Depending on the workflow, this may include citations to internal sources, structured calculations, tool logs, confidence thresholds, exception checks, or policy confirmations. Evidence requirements are a direct countermeasure to persuasive but unsupported output.

Escalation and human-in-the-loop triggers. Do not treat human oversight as a slogan. Define the conditions under which the agent must stop and escalate: missing data, policy conflict, low confidence, ambiguous requests, unusual customer vulnerability, repeated overrides, or any task class known to be near the edge of the frontier. The point is not to slow every case down; it is to route the right cases away from autonomy.

Autonomy tiers and earned trust. Treat autonomy as a ladder rather than a switch. Tier 1 Advisory means the agent may draft, summarize, classify provisionally, or recommend while a human acts. Tier 2 Conditional Automation means the agent may execute narrowly defined actions only when pre-specified evidence, confidence, and exception checks are satisfied. Tier 3 Delegated Execution means the agent may complete a defined class of routine actions without pre-action human approval within approved limits, logging, monitoring, and revocation rules. Autonomy tiers should therefore be treated as earned and reversible operating states, not as a simple adoption ladder.

Change management, monitoring, and recertification. Version and recertify delegation contracts because the frontier moves. Model updates, tool changes, policy changes, data changes, unexplained shifts in override rates, and incident patterns should trigger reevaluation. A contract that was safe under one set of workflow conditions may become unsafe after an integration, prompt change, or change in operating targets.

A brief example makes the matrix concrete. In frontline billing support, the first question is not whether the agent is 'good at customer service' in general. It is where the proposed workflow sits in the matrix. If the agent has only been used to draft responses and no one has specified write permissions, the use case should be kept in exploratory use or treated as casual use. If tests show strong performance on standard invoice-status questions but no one has specified when the agent may send a response, the case is an unsafe scaling risk: capable but under-contracted. If the terms are clear but performance remains uncertain, the case belongs in a test-before-authorization pilot. Only when both conditions hold should managers treat the use case as contracted delegation.

In contracted delegation, the billing agent might retrieve account history, cite approved policy text, draft a reply, and recommend a routing code. It might send a response only when the case fits a narrow class, such as a straightforward invoice-status question or a standard explanation of an already-applied fee. It should not waive charges, make legal representations, alter payment terms, close a complaint involving a vulnerable customer, or communicate externally when account data are incomplete. These limits matter because adjacent billing tasks can look similar on the screen while depending on very different kinds of judgment.

The same example shows how the six dimensions translate into operating controls. The mandate defines the objective as drafting or sending responses for standard frontline billing inquiries, not 'handling billing' in general. Authority and permissioning separate read access from write authority and reserve fee waivers, policy exceptions, and account corrections for humans. Evidence requirements oblige the agent to reference the relevant invoice record, account status, approved knowledge-base article, and current retrieval-source version before any outbound response is sent. Escalation triggers include missing ledger matches, contradictory system data, repeated customer contact about the same issue, references to hardship or fraud, or any confidence pattern associated with prior overrides. Autonomy begins at Tier 1 Advisory before moving, if performance holds, to Tier 2 Conditional Automation for tightly specified subcases. Monitoring tracks unsupported claims, override rates, reopened tickets, complaint escalation, and the frequency with which cases initially classified as routine are later found to involve exceptions.

Appendix A operationalizes these governance dimensions in the form of a compact delegation-contract template. The appendix is not a competing framework but a granular implementation of the matrix: it records the workflow, system version, data-source or retrieval-source version, owner, autonomy tier, authorized and prohibited actions, evidence requirements, escalation triggers, monitoring

Contract field	Example entry
Use case / tier	Frontline billing support; Tier 2 Conditional Automation.
Authorized actions	Send a standard invoice-status response; update a routing code.
Prohibited actions	Waive fees; alter payment terms; close hardship complaints; make legal representations.
Evidence requirements	Invoice record, account status, approved knowledge-base article, successful ledger match, and current retrieval-source version.
Escalation triggers	Missing ledger match, contradictory records, fraud signal, hardship language, repeated complaint, low confidence, policy conflict.
Monitoring and logging	Log source records consulted, policy article version, ledger-match result, confidence and exception checks, outbound message text, human overrides, reopened tickets, and complaint escalations.
Kill switch / revocation protocol	Workflow owner or risk/compliance reviewer may suspend Tier 2 authority after unsupported outbound claims, repeated ledger mismatches, override-rate drift, policy change, or incident investigation.
Recertification triggers	Model update, prompt change, billing-policy change, data-source or retrieval-source change, new integration, incident, override-rate shift.
Risk boundaries / safeguards	Maximum blast radius defined by dollar value, customer class, record type, jurisdiction, and transaction category.

Table 6. Example entries: frontline billing support (Tier 2 Conditional Automation)

obligations, risk thresholds, revocation protocol, review cadence, recertification triggers, and approval responsibilities. Used this way, the delegation contract converts an abstract governance debate into a reviewable operating arrangement that frontline staff, technical teams, auditors, and risk owners can inspect against the same terms. Table 6 provides example contract entries for this Tier 2 use case.

Delegation contracts in agile, DevOps, low-code/no-code, and open-source contexts

Delegation contracts do not replace agile, DevOps, low-code/no-code governance, or open-source review. They specify the authorization terms that must be satisfied before an AI-enabled workflow receives operational authority. In agile settings, the contract can operate as acceptance criteria for moving from prototype to production or for expanding a user story from advisory use to action. In DevOps, it can become a release gate and recertification artifact tied to deployments, model updates, prompt changes, and new integrations.

In low-code/no-code settings, the contract is especially important because nontechnical users may assemble automations that grant write access, trigger actions, or send communications without recognizing that they have delegated authority. In open-source contexts, the contract does not assess

the general value of open-source modules; it records whether the assembled system is allowed to act in a specific workflow, under what evidence, escalation, logging, and revocation conditions. This preserves development speed by keeping experimentation lightweight while requiring explicit authorization when a prototype becomes production authority.

Contributions to practice

The delegation-contract perspective yields immediate implications for practice. Managers should treat workflow-embedded agentic AI as a decision-rights problem before they treat it as a productivity tool. A useful output is not enough to justify operational authority. A deployment is ready to move beyond experimentation only when the organization can state what task is being delegated, what action the agent may take, what systems it may read or write, what evidence must support action, which conditions force escalation, who owns monitoring, and what events trigger recertification or revocation.

The first practical step is to map the workflow at the level of consequential actions. Managers should identify the points where recommendation can become execution through permissions, defaults, one-click sending, automated routing, field updates, approvals, or downstream integrations. That mapping should focus on reversibility, detectability of error, and the operational consequences of a wrong action, not only on whether the AI appears technically capable.

The second step is to separate forms of authority that often become bundled in practice. Read access, draft generation, case classification, outbound communication, record changes, exception approvals, and financial commitments should be treated as separate delegation decisions. Moving from one to another should require explicit authorization rather than informal expansion through interface design or process redesign.

The third step is to make evidence, escalation, and monitoring operational. Before an agent acts, the contract should specify the approved source references, tool traces, validation checks, confidence thresholds, policy confirmations, or other auditable artifacts required for that workflow. It should also define the conditions that send work back to a human: ambiguity, missing data, policy conflict, repeated overrides, unusual confidence patterns, customer vulnerability, or signals that a supposedly routine case is actually an exception.

The fourth step is to manage autonomy as an earned and reversible state. Tier 1 Advisory, Tier 2 Conditional Automation, and Tier 3 Delegated Execution should be treated as distinct operating states. Each increase in autonomy is a new authorization decision, not an adoption milestone. It should be justified by workflow-specific evidence and accompanied by updated permissions, escalation triggers, monitoring thresholds, and recertification triggers. Model updates, prompt changes, new integrations, data-source or retrieval-source changes, policy revisions, incident patterns, or drift in overrides and escalation rates should trigger recertification rather than passive continuation.

The practical contribution of the delegation contract is therefore not only the template in Appendix A. It is a repeatable governance discipline for approval meetings, release gates, incident reviews, audits, and autonomy-expansion decisions. Used well, it helps managers keep innovation moving while preventing machine authority from expanding faster than evidence, accountability, and revocation mechanisms can support.

Concluding remarks

The jagged frontier reframes the central challenge of managing agentic AI. The problem is not that AI is unreliable in the abstract; it is that reliability is uneven, context-dependent, and difficult to anticipate even within the same workflow. Adjacent tasks can sit on different sides of the frontier, plausible outputs can conceal substantive error, and overreliance is produced not only by inattentive users but by socio-technical systems designed to be fast, persuasive, and frictionless. Under those conditions, managerial intuition about which tasks are similar enough to delegate becomes an unreliable guide.

From that diagnosis, the article makes three contributions. First, it reconceptualizes managing agentic AI as a delegation problem rather than a resource deployment problem. Second, it introduces the delegation contract as a workflow-level authorization record that is distinct from model evaluation, enterprise AI policy, or impact assessment because it operates at the level of workflow-specific authority. Third, it identifies six governance dimensions of bounded delegation – purpose boundaries, permissioning, evidence, escalation, autonomy tiers, and recertification – and operationalizes them through the eleven-field template in Appendix A.

The practical implication is that managing agentic AI is a continuing organizational design task. The relevant question is not whether the organization has AI capability, but what authority has been granted, with what reversibility, and with what triggers for human intervention. This argument is most salient where agentic AI participates in recurrent production workflows and can materially affect records, transactions, communications, or decisions; it is much less central where an individual uses an LLM for casual exploration, editing, or brainstorming and retains full review before any organizational action occurs.

Under those conditions, good management is not maximizing use but making delegation explicit, monitoring when its terms no longer fit, and revising or withdrawing authority as the frontier shifts. The purpose of the delegation contract is to specify and continuously govern workflow-specific machine authority – what an agent may do, on what evidence, with which escalation paths, and with what means of revocation – so that accountability remains human even when execution is partly delegated.

The criticality distinction matters. Casual uses can remain lightweight. Business workflows need explicit authorization and recertification. Acute or high-stakes workflows require the delegation contract plus stronger validation, assurance, testing, and formal controls. The delegation contract is therefore not a universal paperwork burden; it is a practical discipline for situations where machine output can become organizational action.

References

Alaimo, C., & Kallinikos, J. (2022). Organizations decentered: Data objects, technology and knowledge. *Organization Science*, 33(1), 19–37.

Alaimo, C., & Kallinikos, J. (2024). *Data Rules: Reinventing the Market Economy*. Cambridge, MA: MIT Press.

Baird, A., & Maruping, L. M. (2021). The next generation of research on IS use: A theoretical framework of delegation to and from agentic IS artifacts. *MIS Quarterly*, 45(1), 315–341.

Dell'Acqua, F., McFowland III, E., Mollick, E. R., Lifshitz, H., Kellogg, K. C., Rajendran, S., Krayer, L., Candelon, F., & Lakhani, K. R. (2026). Navigating the jagged technological frontier: Field experimental evidence of the effects of artificial intelligence on knowledge worker productivity and quality. *Organization Science*, 37(2), 403–423.

Grashoff, I., & Recker, J. (2024). How GuideCom used the Cognigy.AI low-code platform to develop an AI-based smart assistant. *MIS Quarterly Executive*, 23(3), 325–344.

Holmström, J. (2026). *AI Innovation in Business: How Artificial Intelligence Can Create Value for Organizations*. Abingdon: Routledge.

Jarrahi, M. H., & Ritala, P. (2025). Rethinking AI agents: A principal-agent perspective. *California Management Review Insights*.

Mayer, A.-S., Kostis, A., Strich, F., & Holmström, J. (2025). Shifting dynamics: How generative AI as a boundary resource reshapes digital platform governance. *Journal of Management Information Systems*, 42(2), 400–430.

Mikalef, P., & Gupta, M. (2021). Artificial intelligence capability: Conceptualization, measurement calibration, and empirical study on its impact on organizational creativity and firm performance. *Information & Management*, 58(3), 103434.

Ribes, D., Jackson, S. J., Geiger, R. S., Burton, M., & Finholt, T. (2013). Artifacts that organize: Delegation in the distributed organization. *Information and Organization*, 23(1), 1–14.

Sundberg, L., & Holmström, J. (2024). Fusing domain knowledge with machine learning: A public sector perspective. *The Journal of Strategic Information Systems*, 33(3), 101848.

van den Broek, E., Sergeeva, A., & Huysman, M. (2021). When the machine meets the expert: An ethnography of developing AI for hiring. *MIS Quarterly*, 45(3), 1557–1580.

Waardenburg, L., & Huysman, M. (2022). From coexistence to co-creation: Blurring boundaries in the age of AI. *Information and Organization*, 32(4), 100432.

Bio

Jonny Holmström is an information systems professor at Umeå University and the director and co-founder of Swedish Center for Digital Innovation and the AI Business Lab. His research is grounded in academic-practice collaborations and is focused on topics such as digital innovation, digital transformation, and human-AI collaboration. His work has appeared in journals such as *European Journal of Information Systems, Information & Management, Information and Organization, Information Systems Journal, Journal of Information Technology, Journal of the AIS, Journal of Strategic Information Systems, MIS Quarterly,* and *Research Policy*. He serves as a senior editor for *European Journal of Information Systems, Information and Organization*, and *Journal of Information Technology.*

Appendix A: Agentic AI Delegation Contract

Contract reference / version	[ID and version.]
Effective date / next review	[Approval and review dates.]
Delegated workflow / use case	[Workflow in which the AI agent participates.]
AI system / model / tool-stack version	[Model, agent, tools, retrieval source, and data-source versions.]
Responsible business unit / workflow owner	[Department and workflow owner responsible for oversight.]
Human owner / accountable manager	[Individual accountable for the delegation arrangement.]
Selected autonomy tier	☐ **Tier 1** Advisory ☐ **Tier 2** Conditional Automation ☐ **Tier 3** Delegated Execution Tier definitions: Tier 1 Advisory = AI recommends or drafts; human acts. Tier 2 Conditional Automation = AI executes narrowly defined actions only when evidence and exception checks pass. Tier 3 Delegated Execution = AI completes a defined class of routine actions without pre-action human approval, subject to logging, monitoring, limits, and revocation.
ARTICLE I – MANDATE	
Delegated objective / expected outcome / performance boundary	[Task purpose, expected outcome, and limits of acceptable performance.]
ARTICLE II – AUTHORITY	
Authorized actions	[Actions the agent may execute autonomously.]
Prohibited actions	[Actions forbidden or reserved for human approval.]
Permissions	☐ Read records ☐ Populate fields ☐ Trigger workflow steps ☐ External communication ☐ Financial commitment ☐ Modify master data
ARTICLE III – CONTROL CONDITIONS	
Evidence requirements	[Sources, retrieval/data-source versions, validation checks, confidence thresholds, or rule conditions required before action.]
Escalation triggers	[When control reverts to a human: ambiguity, low confidence, conflicts, exceptions, or out-of-scope requests.]
Monitoring and logging	[Required logs, evidence traces, prompt/tool records, audit owner, retention, and monitoring thresholds.]

ARTICLE IV – ASSURANCE AND RISK	
Performance metrics / thresholds	[Metrics used to evaluate delegation and threshold for continued authorization.]
Risk boundaries / safeguards	[Maximum blast radius by dollar value, customer class, record type, jurisdiction, or transaction category; financial limits, data restrictions, approval gates, and other safeguards.]
ARTICLE V – REVOCATION AND RENEWAL	
Kill switch / revocation protocol	[Who may suspend delegated authority, by what mechanism, and under what incident conditions.]
Review cadence / recertification triggers	[Review frequency; model, prompt, retrieval/data-source, tool, or policy changes; incident patterns; override drift.]
ARTICLE VI – APPROVAL	
Human owner / accountable manager (all tiers)	Name / signature / date:
Responsible business owner (all tiers)	Name / signature / date:
Technical / integration reviewer Required for Tier 2/3 or write/tool use.	Name / signature / date:
Risk / legal / compliance / security reviewer Required for Tier 2/3, regulated, or high-risk workflows.	Name / signature / date:
Scope	This contract bounds AI-agent authority in a named workflow and supports monitoring, escalation, revocation, and recertification.