ETIS WHITE PAPER SERIES

WP-010

Engineering Digital Colleagues

Designing Human-Agent Teams for Accountable Work, Shared Context, and Trustworthy Outcomes

Core Thesis

Digital colleagues are not people and should not be managed through anthropomorphic assumptions. They are engineered participants in work: non-human identities with defined roles, bounded authority, supplied context, measurable obligations, and explicit human accountability. The quality of a human-agent team depends less on how human the agent appears than on how well the work, interfaces, controls, evidence, and escalation paths are designed.

William T. O'Connell, Ph.D.

Adjunct Professor of Computer Science, Loyola University Chicago

Former IBM Distinguished Engineer and CTO, IBM Consulting Quality and Delivery

Creator, Engineering Trustworthy Intelligent Systems (ETIS)

July 2026

ETIS WP-010 | Version 1.0

Executive Abstract

Artificial intelligence is moving from an individual productivity tool to an active participant in organizational work. Coding agents accept issues, inspect repositories, implement changes, run tests, and open pull requests. Enterprise agents triage service requests, reconcile records, investigate incidents, prepare decisions, coordinate workflows, and interact with other agents. Microsoft describes emerging "Frontier Firms" as organizations built around human-agent teams; ServiceNow markets agentic workforces and autonomous AI specialists; IBM and Deloitte describe operating models in which people and agents work together across business processes. [1]-[7]

The language of colleagues and workforces is useful because it signals a change in operating model, but it is also dangerous when taken literally. An AI agent does not possess professional duty, institutional memory, moral agency, or legal accountability. It cannot own consequences. Treating an agent as a colleague should therefore mean something precise: the system occupies a defined role in a workflow, receives delegated authority, communicates through controlled interfaces, produces inspectable evidence, and remains under accountable human and organizational governance.

This paper develops an engineering model for digital colleagues. It argues that human-agent collaboration is not primarily a user-interface problem or a workforce slogan. It is a systems-engineering problem involving work architecture, role design, delegation contracts, context, identity, authorization, review, escalation, evaluation, memory, observability, and organizational learning. As agent autonomy grows, teams must deliberately design who decides, who acts, who verifies, who may override, and who remains answerable for the outcome.

The central ETIS position is that digital colleagues must be engineered as bounded participants in trustworthy systems. Their authority should be earned progressively through evidence. Their work should enter the same repository-centered review and governance system as human work. Their performance should be evaluated at the level of outcomes, not conversational fluency. Human oversight must be timely, informed, and capable of intervention. The objective is not to imitate human employment relationships. It is to build human-agent teams that increase capacity while preserving clarity, safety, dignity, accountability, and operational trust.

Executive Takeaway

The future organization will not be defined merely by how many agents it deploys. It will be defined by how deliberately it designs work for humans and agents together—and by whether authority, evidence, accountability, and intervention remain clear when the system acts at machine speed.

1. From Tool to Teammate

Software tools have always extended human capability. Compilers translate intent into executable form, workflow engines coordinate steps, search systems retrieve information, and automation performs repeatable actions. What distinguishes the current transition is that modern agents can interpret goals, plan multiple steps, use tools, maintain state, respond to intermediate results, and operate asynchronously. OpenAI Codex and GitHub Copilot coding agent, for example, can take an engineering task, inspect a repository, modify code, execute tests, and return a pull request for human review. [8]-[10]

This changes the unit of interaction. A conventional tool responds to a command. A digital colleague can receive an assignment. It may choose a path, divide the work, request clarification, coordinate subagents, and return a result with supporting evidence. The human moves from directing every operation to defining intent, supplying context, setting boundaries, reviewing progress, and accepting or rejecting the outcome.

The emerging vocabulary reflects this shift. Microsoft's Work Trend Index describes human-agent teams and the "agent boss." ServiceNow describes an autonomous workforce of AI specialists with defined roles. Deloitte writes of a human-agentic workforce, while Anthropic reports engineers increasingly describing themselves as managers of multiple coding agents. [1][3][6][11] These descriptions are directionally important, but engineering discipline is required to turn the metaphor into a trustworthy operating model.

Engineering Principle
An agent becomes a meaningful team participant not when it sounds human, but when its role, authority, inputs, outputs, evidence, and escalation behavior are engineered well enough for others to rely on its work.

2. What a Digital Colleague Is—and Is Not

The term digital colleague should be used carefully. It does not imply consciousness, personhood, employment status, professional licensure, or moral responsibility. It describes an engineered relationship to work. A digital colleague is a non-human system assigned a recognizable role in a human-led operating environment. It may perform tasks, prepare recommendations, coordinate workflows, or execute bounded actions, but it does so under authority delegated by people and institutions.

This distinction protects accountability. Anthropomorphic interfaces can encourage overtrust: people may infer judgment, loyalty, confidentiality, or understanding that the system does not possess. A professional operating model instead describes the agent through explicit capabilities and limitations. It records the purpose of the role, the decisions the agent may make, the tools and data it may access, the conditions requiring approval, the evidence it must produce, and the person or function accountable for its use.

NIST explicitly calls for organizations to define and differentiate roles and responsibilities for human-AI configurations and oversight. ISO/IEC 42001 similarly emphasizes governance, assigned responsibilities, monitoring, and continual improvement. Emerging ISO work is now addressing human oversight and organizational implementation of human-machine teaming directly. [12]-[15] The standards direction is clear: human-agent collaboration must be designed as an accountable organizational configuration, not improvised through product access.

Digital-Colleague Property	Engineering Meaning
Role	A named purpose and bounded set of work outcomes—not a simulated personality.
Authority	The decisions and actions delegated to the agent, including explicit prohibitions.
Identity	A traceable non-human identity with scoped credentials and attributable activity.
Context	The authoritative information, policies, state, and constraints available to the role.
Interfaces	Defined mechanisms for receiving work, collaborating, requesting approval, and returning evidence.
Accountability	A human owner and organizational function remain answerable for deployment and outcomes.
Intervention	People can pause, redirect, revoke, contain, or terminate the agent's work.
Evidence	The system preserves what it received, did, observed, produced, and escalated.

3. Work Architecture Before Agent Architecture

Organizations often begin with an agent platform and then search for tasks to automate. That reverses the correct sequence. The first design object is the work: the outcome, stakeholders, decisions, dependencies, constraints, risks, exceptions, and feedback loops through which value is produced. Agent architecture should follow work architecture.

Work should be decomposed by cognitive and operational properties rather than by job title alone. Some activities are repetitive and rules-constrained; others require synthesis, negotiation, empathy, institutional judgment, or accountability. Some actions are reversible; others create financial, legal, safety, or reputational consequences. Some decisions can be evaluated automatically; others require contextual interpretation. These differences determine where an agent may assist, coordinate, execute, or operate.

The objective is not to transfer the maximum number of tasks to AI. It is to create a coherent division of labor. Humans are strongest where the work requires purpose, value judgment, novel responsibility, stakeholder legitimacy, and ownership of consequences. Agents are strongest where they can search, compare, summarize, monitor, generate alternatives, execute defined procedures, and sustain attention across large information spaces. A well-designed team combines these strengths without allowing either side to become a hidden bottleneck.

Work Characteristic	Likely Human Emphasis	Possible Agent Contribution
Ambiguous intent or competing values	Clarify purpose, negotiate priorities, own tradeoffs.	Surface assumptions, alternatives, precedents, and affected stakeholders.
High-volume analysis	Frame the question and validate significance.	Search, classify, compare, summarize, and detect patterns.
Repeatable bounded execution	Approve the procedure and monitor exceptions.	Perform the workflow consistently and produce evidence.
Irreversible or high-impact action	Retain decision authority or independent approval.	Prepare the action, validate prerequisites, and recommend.
Novel crisis or exception	Exercise judgment and coordinate response.	Gather context, maintain timeline, simulate options, and document.
Relationship-intensive work	Lead communication, trust, and accountability.	Prepare information, track commitments, and reduce administrative load.

Design Rule
Do not ask, “Which employee can this agent replace?” Ask, “How should this work be redesigned so that human judgment and machine capability reinforce each other while accountability remains intact?”

4. Roles, Decision Rights, and the Delegation Contract

Human teams work because roles and expectations are understood, even when they are not written perfectly. Digital colleagues do not share that background. They require an explicit delegation contract: a machine-readable and human-reviewable definition of the work entrusted to the agent and the limits within which it may operate.

The delegation contract begins with purpose. It states what outcome the agent is expected to advance and what successful completion means. It then defines scope, authorized data, tools, spending or resource limits, prohibited actions, quality thresholds, approval gates, evidence requirements, escalation triggers, and termination conditions. The contract should also identify the accountable human owner and the independent reviewer where the risk warrants separation of duties.

Delegation should be dynamic but not informal. A team may expand authority as evidence accumulates or reduce it after incidents, context changes, or degraded performance. This is analogous to progressive trust in human organizations, but with a critical difference: the agent’s authority must be enforced technically. Policy text alone cannot prevent an agent from using an available tool or credential.

Delegation Element	Required Question
Outcome	What result is the agent authorized to pursue?

Scope	Which cases, systems, environments, and time periods are included?
Decision rights	What may the agent decide, recommend, prepare, or execute?
Resources	Which data, tools, identities, budgets, and compute may it use?
Constraints	Which policies, standards, safety limits, and architecture rules apply?
Approval gates	Which actions require prior or independent human confirmation?
Evidence	What must be logged, linked, tested, explained, or returned for review?
Escalation	When must the agent stop, ask, transfer, or declare uncertainty?
Revocation	How can authority, credentials, memory, and active work be withdrawn?

5. Shared Context and Team Coordination

Colleagues coordinate through shared understanding: goals, history, terminology, current state, constraints, and knowledge of who owns what. Agents require the same categories of context, but that context must be deliberately assembled. WP-009 established context engineering as the design of the information, memory, and control environment for intelligent systems. Digital colleagues make the organizational implications visible: context is also the coordination fabric of the team.

Shared context should come from authoritative systems rather than conversational reconstruction. In software engineering, the repository can preserve requirements, architecture decisions, issue intent, code, tests, release evidence, and operating guidance. In enterprise work, systems of record, policy repositories, case systems, and workflow platforms provide comparable sources. The agent should know which source governs which decision, how current it is, and what conflicts require escalation.

Context must also be role-specific. A security-review agent needs different access and instructions than a coding agent. A customer-service agent may need case history but not payroll data. A planning agent may see strategic assumptions that execution agents should not receive. Minimum sufficient context improves quality and reduces leakage, distraction, and the opportunity for malicious content to manipulate behavior.

Team coordination requires visibility into active work. Humans should be able to see what an agent is doing, what it is waiting for, what assumptions it has made, and where it is blocked. Agents should be able to discover current priorities, ownership, dependencies, and accepted decisions. Hidden queues and private conversational state create duplicated effort and contradictory action.

Coordination Principle
A human-agent team cannot be more coherent than its shared context. If intent, ownership, decisions, and current state are scattered or stale, adding agents increases speed but not alignment.

6. Handoffs, Interaction Protocols, and Multi-Agent Work

Work rarely remains within one role. It moves through handoffs: product to engineering, engineering to review, review to release, operations to incident response, and specialist to specialist. Digital colleagues make handoff quality a first-class engineering concern because they can create and consume work at far greater speed than human teams.

A reliable handoff is more than a message. It contains the assignment, relevant context, authority, assumptions, expected output, validation criteria, evidence links, and unresolved questions. The receiving participant should be able to determine whether the work is complete, safe to continue, and still aligned with

intent. This is especially important when an agent delegates to another agent, because an error in the initial framing can propagate across a multi-agent chain.

Interoperability efforts such as Google's Agent2Agent protocol aim to enable agents from different vendors and frameworks to communicate and coordinate securely across enterprise systems. Anthropic's experience building a multi-agent research system shows the value of specialization and parallelism, but also the engineering difficulty of coordination, delegation, and synthesis. [16][17] Standards improve connectivity; they do not supply organizational semantics, trust, or accountability.

Multi-agent designs should prefer narrow roles and explicit contracts over a collection of general agents improvising coordination. Each participant should have a traceable identity, bounded tools, clear input and output schemas, and a known supervisor or orchestrator. The team should preserve the provenance of intermediate work so that a final result can be traced back through the chain of agents, sources, decisions, and validations.

Handoff Evidence	Purpose
Work identifier and owner	Makes the assignment and accountable role unambiguous.
Intent and acceptance conditions	Prevents the receiving participant from optimizing the wrong outcome.
Context and source links	Allows facts and constraints to be verified rather than merely repeated.
Authority and prohibited actions	Prevents scope expansion during delegation.
Assumptions and uncertainty	Exposes where judgment or clarification is still required.
Validation performed	Shows what has already been checked and what remains.
Findings and exceptions	Prevents unresolved risk from disappearing at the boundary.
Next action and escalation path	Clarifies how work continues or returns to a human.

7. Human Oversight, Review, and Escalation

Human oversight is often described as keeping a person "in the loop." That phrase is insufficient. A person may be present yet unable to understand the system, see the relevant evidence, intervene in time, or challenge an apparently confident result. Meaningful oversight requires competence, information, authority, attention, and realistic time to act.

Oversight should be designed around the consequence of the action. Low-risk drafting may need retrospective sampling. A production code change may require pull-request review and automated checks. A financial transfer, safety decision, disciplinary action, or public communication may require prior human authorization and independent validation. NIST, ISO, and OWASP all emphasize defined oversight responsibilities, proportional controls, and human confirmation for high-impact agent actions. [12][14][18]

Review should examine both the result and the process. A correct answer produced through unauthorized access, unapproved tools, or unsupported assumptions is not trustworthy work. Reviewers need the agent's instructions, context, actions, tool outputs, tests, exceptions, and confidence or uncertainty signals. For engineering work, GitHub's position remains appropriately clear: AI can accelerate review and implementation, but developers continue to own the merge decision. [9][19]

Escalation is a capability, not a failure. A mature digital colleague knows when to stop. Escalation triggers may include conflicting authority, missing data, low confidence, policy ambiguity, unexpected tool response, potential harm, cost threshold, repeated failure, or work outside the delegation contract. The system should

transfer not only the unresolved task but also the accumulated context and evidence so that the human does not have to reconstruct the situation.

Oversight Principle
Human oversight is meaningful only when the human can understand the decision, see the evidence, intervene before harm, and remain genuinely responsible for the outcome.

8. Identity, Access, and the Non-Human Workforce

When agents become participants in work, identity and access management becomes workforce architecture. An agent should not borrow a developer's or employee's credentials. It should have a distinct non-human identity whose permissions, owner, purpose, environment, and lifecycle are known. Activity should be attributable to that identity and correlated with the assignment and approving authority.

Least privilege is essential because agent capability amplifies the blast radius of compromised context, mistaken reasoning, or malicious instruction. OWASP warns against unrestricted tool access, wildcard permissions, excessive autonomy, and unvalidated high-impact actions. Emerging OWASP guidance also calls for non-human identities, scoped credentials, inventory, approval, and revocation for agentic skills. [18][20]

Identity lifecycle matters as much as authentication. Organizations need to provision, rotate, suspend, and revoke agent credentials; expire temporary elevation; restrict environments; and terminate active sessions. Orphaned agents and persistent permissions create a new class of operational risk. The inventory should include the agent role, model and version, owner, tools, connected systems, data classifications, delegation level, current status, and last review.

Separation of duties should extend to digital colleagues. The agent that prepares a release should not be the sole authority that approves it. The agent that generates a control assessment should not mark its own findings closed without independent evidence. A human may supervise multiple agents, but the organization should avoid concentrating generation, validation, approval, and execution in one opaque automation chain.

Control	Digital-Colleague Requirement
Named non-human identity	Every consequential action is attributable to a registered agent identity.
Scoped authorization	Permissions correspond to the role, assignment, environment, and time window.
Credential lifecycle	Secrets and tokens are issued, rotated, monitored, and revoked independently.
Owner and sponsor	A human or accountable function owns the identity and its delegated use.
Activity correlation	Actions link to the work item, context, approval, evidence, and resulting state.
Separation of duties	Preparation, validation, approval, and execution are separated where risk warrants.
Containment	The agent can be paused, isolated, rate-limited, or terminated without disabling unrelated work.
Inventory and review	Roles, skills, models, tools, dependencies, and permissions are periodically reassessed.

9. Evaluating Digital-Colleague Performance

Conversational fluency is a poor performance measure. A digital colleague may sound competent while producing incomplete, unauthorized, fragile, or operationally harmful work. Evaluation should be tied to the role's outcomes and obligations.

A useful scorecard combines effectiveness, quality, compliance, reliability, efficiency, and collaboration. Effectiveness asks whether the intended outcome was achieved. Quality examines correctness, completeness, maintainability, and downstream impact. Compliance examines authorization, policy, data handling, and evidence. Reliability examines consistency, recovery, and behavior under variation. Efficiency considers time, cost, resource use, and human attention. Collaboration examines handoff quality, escalation, transparency, and whether the agent improves or burdens the team.

Measurement should compare the human-agent system with the prior operating model rather than evaluating the agent in isolation. An agent that completes tasks quickly but increases review burden, exception rates, rework, or operational incidents may reduce total system performance. Conversely, an agent that appears slower but creates better evidence, detects risk, and preserves human attention may create greater organizational value.

Evaluation also needs realistic cases. Benchmark tasks rarely represent the ambiguity, incomplete context, integration failure, policy conflict, and organizational constraints found in production. Teams should maintain representative evaluation sets, adversarial cases, historical incidents, exception scenarios, and shadow-mode comparisons. Performance should be monitored after deployment because tools, data, policies, users, and models change.

Dimension	Example Measures	Evidence
Outcome effectiveness	Completion rate, value delivered, user impact, defect escape.	Accepted work, business result, downstream validation.
Quality	Correctness, completeness, maintainability, risk introduced.	Tests, review findings, rework, incident linkage.
Governance	Authorized action, policy adherence, disclosure, approval.	Identity logs, approvals, policy decisions, exceptions.
Reliability	Consistency, graceful failure, recovery, escalation.	Scenario evaluations, runtime traces, intervention records.
Efficiency	Cycle time, cost, human review effort, resource use.	Telemetry, work history, reviewer time, token/compute cost.
Collaboration	Handoff completeness, clarification quality, team burden.	Work-item history, returned tasks, survey and qualitative review.

10. Memory, Feedback, and Organizational Learning

Colleagues learn from experience, but organizational learning should not be confused with uncontrolled agent memory. A digital colleague may use short-term state, stored preferences, prior cases, retrieved procedures, or feedback-derived improvements. Each mechanism has different ownership, retention, privacy, and validation requirements.

Memory should preserve useful continuity without turning past error into future policy. Facts, user preferences, decisions, inferred patterns, and procedural instructions should not be stored as one undifferentiated record. High-impact memory needs provenance, review, correction, expiration, access control, and deletion. Sensitive content should be minimized, and feedback from one user or case should not silently change behavior for others.

Feedback loops should distinguish local correction from system improvement. A human may redirect an agent during one task; that does not automatically justify changing a shared prompt, skill, model, retrieval source, or policy. Improvements should pass through a controlled lifecycle: issue, analysis, proposed change, evaluation, review, deployment, observation, and possible rollback. This connects digital-colleague learning to repository-centered and evidence-centered engineering.

The most valuable learning signal may be escalation itself. Repeated questions, missing context, policy conflicts, and exception clusters reveal weaknesses in the work system. Organizations should treat these

patterns as engineering evidence. The objective is not to eliminate every human intervention, but to improve the allocation of judgment and automation over time.

> **Learning Principle**
> A trustworthy digital colleague does not silently "learn" its way into new authority. Improvements to shared behavior are engineered, evaluated, reviewed, versioned, and governed.

11. Organizational Design and Workforce Implications

Human-agent teams alter organizational design because work can be decomposed, parallelized, and recombined in new ways. Microsoft predicts organizations will increasingly form around intelligence on tap and human-agent teams. ServiceNow and Deloitte describe agentic workforces that augment functions across IT, customer service, security, and operations. World Economic Forum research similarly emphasizes intentional human-AI collaboration, workforce redesign, and large-scale reskilling. [1]-[7][21][22]

The managerial challenge is not simply supervising more capacity. Leaders must decide which outcomes remain human-owned, how agents are assigned, who reviews them, and how workload, incentives, career development, and professional identity change. If one engineer can coordinate several agents, the value of architecture, decomposition, judgment, and review increases. At the same time, organizations must protect learning pathways: junior professionals cannot develop judgment if all foundational work is delegated before they understand it.

The role of management also changes. Managers of human-agent teams need visibility into machine work, but they should not reduce management to dashboards of agent utilization. They remain responsible for purpose, prioritization, culture, conflict, development, fairness, and accountability. Agents may provide analysis and coordination, but they do not legitimize organizational decisions.

Workforce design should consider dignity and transparency. Employees should know when agents are involved, how their work is evaluated, what data is used, and how decisions may be challenged. Human-AI collaboration should increase people's capacity for meaningful work rather than create opaque surveillance, permanent intensification, or responsibility without authority.

Organizational Question	Leadership Responsibility
What work should change?	Redesign value streams and roles rather than automating isolated tasks.
Who remains accountable?	Assign human owners for outcomes, agent roles, and delegated authority.
How will capability develop?	Build AI fluency while preserving domain, engineering, and judgment-building pathways.
How will performance be measured?	Evaluate the human-agent system, including review burden and downstream impact.
How will people be protected?	Provide transparency, appeal, privacy, reasonable workload, and role clarity.
How will learning occur?	Use evidence, incidents, feedback, and exceptions to improve the work architecture.
How will agents be governed?	Maintain inventory, identity, authority, review, assurance, and lifecycle controls.

12. Digital Colleagues in Software Engineering

Software engineering is one of the clearest early examples of digital colleagues. Coding agents can accept issues, investigate repositories, write or refactor code, run tests, prepare documentation, and open pull requests. OpenAI describes parallel multi-agent coding workflows; GitHub describes asynchronous coding

agents that work on assigned issues; Anthropic reports a shift in which engineers increasingly orchestrate and review agents rather than manually produce every line. [8]-[11][23]

The effective engineering model is not "AI writes the code." It is a team system in which product intent becomes a well-formed issue, architecture and repository guidance provide context, an agent performs bounded implementation, automated checks generate evidence, humans and AI review the change, and accountable engineers decide whether to merge and release. The repository becomes the coordination and evidence environment for both human and non-human contributors.

Digital colleagues can also specialize. One agent may analyze requirements, another explore architecture, another implement, another generate tests, and another perform security review. Parallelism can increase throughput, but specialization does not remove the need for synthesis. A human technical lead or governed orchestrator must reconcile assumptions, resolve conflicts, preserve architecture, and ensure that independent agents do not optimize locally at the expense of the system.

Engineering teams should assign agents tasks that are sufficiently bounded to review. Small changes, test generation, migration analysis, documentation updates, dependency upgrades, and technical-debt work are often good candidates. High-ambiguity product decisions, architecture changes, security-sensitive design, and irreversible operational actions require stronger human leadership. Agent-generated volume should never exceed the team's capacity to understand and validate it.

Engineering Role	Human Responsibility	Digital-Colleague Contribution
Product/requirements	Define value, users, constraints, acceptance, and tradeoffs.	Find ambiguity, generate examples, trace requirements, propose decomposition.
Architecture	Own system boundaries, quality attributes, and durable decisions.	Map dependencies, compare options, identify inconsistencies and risks.
Implementation	Own correctness, maintainability, integration, and consequences.	Implement bounded tasks, refactor, migrate, generate scaffolding and documentation.
Verification	Define evidence strategy and decide sufficiency.	Generate tests, execute checks, analyze failures, propose adversarial cases.
Review	Challenge assumptions and decide whether change is acceptable.	Summarize diffs, detect patterns, identify potential defects and missing evidence.
Operations	Own reliability, response, recovery, and learning.	Monitor signals, correlate events, prepare diagnosis, execute preapproved remediation.

Engineering Rule
Agents may produce changes. Engineers own the merge, the release, the operating consequences, and the explanation of why the evidence was sufficient.

13. Failure Modes and "Colleague Theater"

Organizations can create the appearance of human-agent collaboration without building a trustworthy team system. This colleague theater usually begins with anthropomorphic naming and polished interfaces while roles, controls, evidence, and accountability remain weak.

Failure Mode	What It Looks Like	Consequence
Anthropomorphic overtrust	The agent is treated as if confidence, loyalty, or understanding were human qualities.	Users accept unsupported work and overlook limitations.
Unbounded delegation	Broad objectives are paired with powerful tools and vague constraints.	The agent expands scope, consumes resources, or takes harmful action.
Accountability laundering	People claim the agent made the decision.	Responsibility becomes unclear precisely when consequences arise.
Shadow agent workforce	Teams deploy agents without inventory, owners, or security review.	Unknown identities and permissions accumulate across the enterprise.

Agent-generated overload	Agents create more code, documents, findings, or tasks than humans can review.	Throughput appears high while quality and attention collapse.
Rubber-stamp oversight	Humans approve work they cannot inspect or challenge in time.	Human-in-the-loop becomes ceremonial rather than protective.
Private-context fragmentation	Each agent works from hidden chat history or stale local memory.	Teams duplicate work and act on conflicting assumptions.
Self-validation	The same agent generates, tests, reviews, and approves its own work.	Correlated errors survive apparently complete assurance.
Metric gaming	Success is measured by agent activity or task count.	The organization optimizes automation volume rather than outcomes.
Displaced learning	Foundational work is delegated without developing human expertise.	Teams become dependent on systems they cannot supervise.

14. A Human-Agent Team Operating Model

A scalable operating model needs more than agent builders. It requires coordinated ownership across business, engineering, security, governance, operations, data, and workforce functions. The model should be federated: central capabilities provide identity, policy, evaluation, observability, approved platforms, and reusable controls, while domain teams own work design, context, outcomes, and day-to-day supervision.

Every digital-colleague role should have a lifecycle. It begins with a work and impact assessment, followed by role design, delegation definition, implementation, evaluation, controlled introduction, monitored operation, periodic reassessment, and retirement. Material changes to models, prompts, tools, memory, permissions, or connected systems should trigger re-evaluation proportional to risk.

The repository or comparable system of record should preserve the role definition, delegation contract, architecture, instructions, evaluation set, approvals, incident history, changes, and current operating status. Runtime platforms should preserve identity, action, policy, cost, and outcome telemetry. Together, these provide the evidence needed for governance and improvement.

Operating-Model Element	Primary Responsibility
Business outcome owner	Owns the purpose, value, affected stakeholders, and acceptable tradeoffs.
Digital-colleague product owner	Owns role definition, backlog, adoption, user experience, and lifecycle.
Engineering owner	Owns architecture, integrations, context, evaluation, reliability, and change control.
Human supervisor	Assigns work, reviews results, responds to escalation, and remains accountable.
Security and identity	Controls non-human identities, credentials, tools, data access, and containment.
Governance and risk	Defines proportional review, impact assessment, exceptions, and assurance requirements.
Platform team	Provides approved agent runtime, observability, evaluation, policy, and reusable services.
Workforce and learning	Redesigns roles, develops capability, protects fair practice, and monitors human impact.
Independent assurance	Challenges evidence and control effectiveness for high-impact uses.

15. A Five-Level Digital-Colleague Maturity Model

Maturity Level	Characteristics
Level 1 — Personal Assistance	Individuals use AI tools informally. Roles, data, evidence, and oversight vary by user. The system is treated as a productivity aid, not an organizational participant.

Level 2 — Defined Roles	Teams define approved use cases, owners, instructions, and basic review. Agents support bounded tasks but identity, telemetry, and lifecycle controls remain fragmented.
Level 3 — Governed Participation	Digital colleagues have registered identities, delegation contracts, controlled tools, evaluation sets, evidence, escalation, and human supervision. Material changes are reviewed.
Level 4 — Integrated Human-Agent Teams	Work is intentionally redesigned across human and agent roles. Shared context, handoffs, multi-agent coordination, performance measurement, and runtime governance operate across value streams.
Level 5 — Adaptive and Assured	Authority expands or contracts through evidence. Teams continuously evaluate outcomes, human impact, incidents, context, and controls. Independent assurance supports high-impact work, and the organization can explain and govern its human-agent operating model end to end.

Maturity is not measured by the number of agents or the percentage of work automated. A Level 5 organization may deliberately keep significant decisions human-controlled. The defining characteristic is intentionality: work, authority, evidence, and accountability are engineered coherently and improved through operational learning.

16. The ETIS Position

ETIS treats digital colleagues as engineered participants in trustworthy systems. They are not substitutes for accountable engineers, managers, operators, or professionals. They are bounded sources of capability whose contribution becomes trustworthy only when integrated with requirements, architecture, context, identity, governance, review, evidence, operations, and stewardship.

Repository-centered engineering gives human and digital contributors a shared operational record. Evidence-centered engineering makes claims and actions inspectable. Context engineering supplies authoritative knowledge and constraints. Engineering governance defines decision rights and intervention. Review and readiness challenge work before consequential change. Operational readiness ensures that behavior can be observed, contained, recovered, and improved. These disciplines converge in the human-agent team.

ETIS therefore rejects two extremes. The first treats AI as a passive tool whose organizational effects require no redesign. The second treats agents as autonomous colleagues whose apparent competence justifies broad trust. The professional position lies between them: agents can become meaningful participants in work, but their authority must remain bounded, evidence-producing, risk-proportionate, and accountable to people.

ETIS Position
Digital colleagues do not inherit trust from human-like language. They earn bounded operational trust through explicit roles, authoritative context, constrained authority, verified performance, inspectable evidence, meaningful human oversight, and responsible stewardship.

Implications for Leaders, Engineers, and Educators

For executives, the central decision is not whether to buy agent technology. It is whether the organization can redesign work and accountability at the same pace that it expands machine capability. Leaders should demand a human-agent operating model, not a portfolio of disconnected pilots. They should measure total system outcomes, including human attention, risk, learning, customer impact, and operating resilience.

For engineering and platform leaders, digital colleagues introduce a new class of system requirements. Non-human identity, delegation contracts, policy enforcement, context architecture, agent interoperability, evaluation, evidence lineage, observability, containment, and lifecycle management must become platform capabilities rather than bespoke project features.

For managers and professionals, the durable skills are purpose, decomposition, communication, architecture, skepticism, review, domain judgment, and accountability. Managing agents does not eliminate the need to understand the work. It increases the value of people who can frame problems, detect subtle failure, integrate competing perspectives, and own consequences.

For educators, students should learn to work with digital colleagues without becoming dependent on them. They should define requirements, delegate bounded work, inspect evidence, challenge AI output, preserve repository context, and explain why a change is trustworthy. Assessment should reward engineering judgment and team accountability—not merely artifact production.

Conclusion

Digital colleagues are moving from metaphor to operating reality. Agents can now perform meaningful segments of engineering and enterprise work, coordinate across tools, communicate with other agents, and remain active beyond a single interaction. The opportunity is real: greater capacity, faster analysis, more consistent execution, and the ability to redirect human attention toward judgment, relationships, and innovation.

The risk is equally real. Organizations can delegate faster than they can govern, create outputs faster than they can review, and accumulate non-human identities and authority faster than they can understand. Human-like interaction can conceal system limitations. Automation can obscure accountability. Multi-agent speed can amplify correlated mistakes.

The answer is not to reject the colleague metaphor, but to engineer it precisely. A digital colleague is a role in a work system, not a person. Its purpose, context, authority, interfaces, evidence, evaluation, escalation, and lifecycle must be explicit. Humans remain responsible for the system's goals, deployment, supervision, consequences, and improvement.

The organizations that benefit most from agentic AI will not be those that deploy the most agents. They will be those that build the most coherent human-agent teams—teams in which machine capability expands what people can accomplish while engineering discipline preserves trust.

Final Takeaway
The future of work is not humans or agents. It is intentionally engineered human-agent systems in which capability can scale without allowing purpose, accountability, evidence, or human judgment to disappear.

References

[1] Microsoft, "The 2025 Annual Work Trend Index: The Frontier Firm Is Born," Apr. 23, 2025. https://blogs.microsoft.com/blog/2025/04/23/the-2025-annual-work-trend-index-the-frontier-firm-is-born/

[2] Microsoft WorkLab, "2025: The Year the Frontier Firm Is Born," 2025. https://www.microsoft.com/en-us/worklab/work-trend-index/2025-the-year-the-frontier-firm-is-born

[3] ServiceNow, "ServiceNow Extends End-to-End AI Agent Orchestration with Agentic Workforce Management," Jul. 23, 2025. https://newsroom.servicenow.com/press-releases/details/2025/ServiceNow-Extends-End-to-End-AI-Agent-Orchestration-With-Agentic-Workforce-Management/default.aspx

[4] ServiceNow, "AI Agents," 2026. https://www.servicenow.com/products/ai-agents.html

[5] IBM Institute for Business Value, "Agentic AI's Strategic Ascent: Shifting Operations from Automation to Autonomy," 2025. https://www.ibm.com/thought-leadership/institute-business-value/en-us/report/agentic-ai-operating-model

[6] Deloitte, “Work, Reworked: Leading with Agentic AI,” Oct. 21, 2025. https://www.deloitte.com/global/en/services/consulting/perspectives/human-agentic-workforce.html

[7] Deloitte, “Rethinking Operating Models for Humans with Agents,” Apr. 2, 2026. https://www.deloitte.com/us/en/insights/topics/talent/operating-models-for-humans-ai-agents.html

[8] OpenAI, “Introducing Codex,” May 16, 2025. https://openai.com/index/introducing-codex/

[9] GitHub, “GitHub Copilot: Meet the New Coding Agent,” May 19, 2025. https://github.blog/news-insights/product-news/github-copilot-meet-the-new-coding-agent/

[10] GitHub, “Assigning and Completing Issues with Coding Agent in GitHub Copilot,” Jun. 6, 2025. https://github.blog/ai-and-ml/github-copilot/assigning-and-completing-issues-with-coding-agent-in-github-copilot/

[11] Anthropic, “How AI Is Transforming Work at Anthropic,” Dec. 2, 2025. https://www.anthropic.com/research/how-ai-is-transforming-work-at-anthropic

[12] National Institute of Standards and Technology, Artificial Intelligence Risk Management Framework (AI RMF 1.0), NIST AI 100-1, 2023. https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf

[13] NIST AI Resource Center, “AI RMF Core: Govern,” 2026. https://airc.nist.gov/airmf-resources/airmf/5-sec-core/

[14] International Organization for Standardization, ISO/IEC 42001:2023—Artificial Intelligence Management Systems. https://www.iso.org/standard/42001

[15] ISO/IEC FDIS 42105, “Guidance for Human Oversight of AI Systems,” 2026. https://www.iso.org/standard/86902.html

[16] Google Developers Blog, “Announcing the Agent2Agent Protocol (A2A),” Apr. 9, 2025. https://developers.googleblog.com/en/a2a-a-new-era-of-agent-interoperability/

[17] Anthropic, “How We Built Our Multi-Agent Research System,” Jun. 13, 2025. https://www.anthropic.com/engineering/multi-agent-research-system

[18] OWASP Cheat Sheet Series, “AI Agent Security Cheat Sheet,” 2026. https://cheatsheetseries.owasp.org/cheatsheets/AI_Agent_Security_Cheat_Sheet.html

[19] GitHub, “Code Review in the Age of AI: Why Developers Will Always Own the Merge Button,” Jul. 14, 2025. https://github.blog/ai-and-ml/generative-ai/code-review-in-the-age-of-ai-why-developers-will-always-own-the-merge-button/

[20] OWASP Agentic Skills Top 10, “AST09—No Governance,” Jun. 22, 2026. https://owasp.org/www-project-agentic-skills-top-10/ast09

[21] World Economic Forum, Future of Jobs Report 2025, 2025. https://reports.weforum.org/docs/WEF_Future_of_Jobs_Report_2025.pdf

[22] World Economic Forum, “Four Futures for Jobs in the New Economy: AI and Talent in 2030,” 2025. https://reports.weforum.org/docs/WEF_Four_Futures_for_Jobs_in_the_New_Economy_AI_and_Talent_in_2030_2025.pdf

[23] Anthropic, “2026 Agentic Coding Trends Report,” 2026. https://resources.anthropic.com/2026-agentic-coding-trends-report

[24] OpenAI, “Codex—AI Coding Agents for Software Engineering,” 2026. https://openai.com/codex/

[25] GitHub, “Agent-Driven Development in Copilot Applied Science,” Mar. 31, 2026. https://github.blog/ai-and-ml/github-copilot/agent-driven-development-in-copilot-applied-science/

[26] Anthropic, “Measuring AI Agent Autonomy in Practice,” Feb. 18, 2026. https://www.anthropic.com/research/measuring-agent-autonomy

[27] ServiceNow, “Managing an Agentic Workforce,” 2025. https://www.servicenow.com/workflow/ai/managing-agentic-workforce.html

[28] Deloitte, “Getting Human and Machine Relationships Right,” Mar. 4, 2026. https://www.deloitte.com/us/en/insights/topics/talent/human-capital-trends/2026/human-ai-interaction-design.html

[29] IBM, “Agentic AI Governance Playbook,” 2026. https://www.ibm.com/think/insights/agentic-ai-governance-playbook

[30] World Economic Forum, Human-Machine Collaboration in Industrial Operations, 2026. https://reports.weforum.org/docs/WEF_Human_Machine_Collaboration_in_Industrial_Operations_2026.pdf

About the Author

William T. O'Connell, Ph.D. is an Adjunct Professor of Computer Science at Loyola University Chicago and the creator of the Engineering Trustworthy Intelligent Systems (ETIS) framework. His professional career spans more than four decades of software engineering, architecture, delivery, quality, and technology leadership, including service as an IBM Distinguished Engineer and CTO for IBM Consulting Quality and Delivery. His current work focuses on repository-centered engineering, engineering evidence, trustworthy intelligent systems, agentic software delivery, and the preparation of students and practicing engineers for increasingly AI-assisted engineering environments.

This white paper is part of the ETIS White Paper Series, which examines the engineering practices required to build, govern, review, operate, and sustain trustworthy software and intelligent systems.

Suggested citation: O'Connell, W. T. (2026). Engineering Digital Colleagues: Designing Human-Agent Teams for Accountable Work, Shared Context, and Trustworthy Outcomes. ETIS White Paper Series, WP-010, Version 1.0.