McKinsey & Company

Rewired glossary

Executives leading and working on tech and AI transformations should be familiar with these terms.

by Alex Singla, Alex Sukharevsky, Kate Smaje, Rob Levin, and Eric Lamarre

April 2026

Agentic engineering

Also known as: agentic systems development

Definition: The discipline of designing, building, orchestrating, and governing a system of AI agents and workflows to do tasks autonomously with appropriate human oversight.

Why it matters: Ensures the development of a large-scale agentic AI workforce that can be orchestrated and managed effectively and safely.

Agentic systems

Also known as: multiagent systems; enterprise agent networks

Definition: Architectures composed of a system of different kinds of AI agents that coordinate, plan, reason, and execute a range of tasks and workflows. For example, an enterprise operations system may involve agents that jointly balance staffing, inventory, and logistics across regions; a corporate strategy system may involve agents that coordinate finance, sales, and supply-chain workflows to maximize profitability and resilience.

Why it matters: Agentic workflows need coordination mechanisms and standards so they can build on each other to become full agentic systems.

Agentic workflows

Also known as: multiagent workflows; agent-orchestrated workflows

Definition: A process where multiple AI agents work together to complete a series of steps in a specific workflow. For example, a procurement workflow may have one agent that gathers supplier data, another that evaluates risk, and a third that negotiates contract terms. A marketing workflow may involve agents that coordinate campaign planning, content generation, and performance reporting.

Why it matters: Workflows, not AI agents, should be the focus of change to generate value.

Agile development

Also known as: agile delivery; iterative development

Definition: A way of building products in small increments, testing frequently, and adjusting based on feedback.

Why it matters: Iterative development with cross-functional teams reduces waste and delivers products and solutions faster.

AI agents

Also known as: intelligent agents; autonomous agents

Definition: Software systems that perform multistep actions autonomously toward a goal, often orchestrating tools, models, and workflows. For example, a finance agent may pull data, reconcile variances, and draft journal entries; a customer care agent may triage, solve, and escalate tickets.

Why it matters: AI agents, when properly built, managed, and directed, can improve productivity and increase innovation.

AI evaluation (evals)

Also known as: AI model testing; AI quality checks

Definition: Structured assessments of an AI system's performance against well-specified criteria.

Why it matters: Creates a standard scorecard for accuracy, bias, safety, and cost so executives can approve launches, monitor performance of AI systems, compare vendors, and manage trade-offs with facts.

AI mesh

Definition: A way of organizing AI so that different parts of the business can share models, data, and tools instead of building their own from scratch.

Why it matters: Prevents duplication, reduces cost, and allows AI to scale across the company without creating a central bottleneck.

AI/ML platform

Also known as: AI infrastructure; ML platform

Definition: A shared set of tools and services that teams use to build, deploy, and consistently manage AI and ML models.

Why it matters: Reduces the time and cost of bringing AI to production by providing teams with a consistent, standardized (and secure environment rather than rebuilding plumbing for each new use case.

Application programming interface (API)

Also known as: integration end point; software interface

Definition: Rules and interfaces that allow different software systems to communicate with each other. For example, a payments API can be used by an app to accept credit cards, or an HR API can be used to synchronize employee profiles for use in tools.

Why it matters: A thoughtful API strategy makes it easier to add and replace applications and systems, and build capabilities (similar in concept to stacking LEGO blocks).

Artificial general intelligence (AGI)

Also known as: strong AI; human-level AI

Definition: While there is no single agreed-upon definition, the term generally refers to an AI system with broad, human-level intelligence across many tasks.

Why it matters: Offers the promise of AI systems undertaking more human tasks.

Artificial intelligence (AI)

Also known as: machine intelligence

Definition: Machines or software that perform tasks that previously required human intelligence.

Why it matters: Using AI frees humans to do the work that only humans can do while automating time-consuming tasks.

Capability-as-a-service (CaaS)

Also known as: internal capability commercialization

Definition: Turns an internal strength of your company (for example, payments, underwriting, identity) into a service that other companies can use and pay for.

Why it matters: Creates new revenue from capabilities you have already invested in and spreads costs across more users.

Chapter

Also known as: guild; community of practice

Definition: A group of individuals sharing a common skill or role to develop expertise and standards.

Why it matters: Raises craft quality and mobility across teams.

Cloud

Also known as: cloud infrastructure; on-demand IT

Definition: Computing services that can be accessed on an as-needed basis through the internet.

Why it matters: Shifts expenditures from fixed to operational, speeds provisioning, and enables easy access to modern data/AI architectures.

Context engineering

Definition: The discipline of designing prompts, rules, and data retrieval mechanisms so AI systems have the right context to produce reliable outputs.

Why it matters: Directly influences performance quality, reliability, and safety of AI systems.

Continuous integration/continuous deployment (CI/CD)

Also known as: build and release automation; continuous delivery

Definition: DevOps practice that automates software build, testing, and deployment.

Why it matters: Increases release frequency and quality while reducing operational risk and rollback pain.

Data architecture

Also known as: enterprise data design; data blueprint

Definition: High-level design of how data is collected, stored, managed, and accessed.

Why it matters: Good architecture lowers the total cost of data and makes trustworthy insights available when needed.

Data center

Also known as: server farm/room/bank/center

Definition: A facility housing compute, storage, and networking equipment—with redundant power, cooling, connectivity, and physical security—to run applications and store data.

Why it matters: Sets the guardrails for latency, resilience, cost, and compliance. These are the choices between on-premises, co-location, and cloud drive strategy for hybrid/edge, data residency, and sustainability.

Data engineers

Also known as: data developers

Definition: Engineers who build and maintain data pipelines, platforms, and integrations to enable data access and analysis.

Why it matters: They enable data to be reliably accessed, used, and reused, which is critical for analytics and AI throughput at scale.

Data fabric

Also known as: unified data layer; data integration layer

Definition: A shared layer that makes it easier to find and use approved data without building custom connections every time. In practice, teams find approved data through standard catalogs and access controls and can plug it into their analytics or AI solutions without stitching together ad hoc feeds.

Why it matters: Gives teams governed self-serve access to data—including who can see what, from where—reducing duplicate feeds and cutting time to onboard new data sources.

Data governance

Also known as: data oversight; information governance

Definition: The policies and processes ensuring data quality, security, and compliance.

Why it matters: Reduces confusion resulting from a lack of clarity regarding ownership and decision rights.

Data lake

Also known as: raw data repository

Definition: Central repository storing raw data in its native format.

Why it matters: Provides flexible storage for advanced analytics and ML, if paired with metadata and governance.

Data mesh

Also known as: decentralized data architecture

Definition: Operating model where business domains, such as claims or sales, own and manage their data as a product—each with clear owners, quality standards, and interfaces—so other teams can easily find and use that data.

Why it matters: Shrinks the “ask the data team” queue and speeds delivery by pushing ownership to the business units that generate the data, while enforcing shared standards so products interoperate.

Data monetization

Also known as: data commercialization

Definition: Extracting financial value from data assets, either directly or indirectly.

Why it matters: Unlocks the value of proprietary data, both by creating new data-driven products and by improving pricing, cross-selling, and retention through insights embedded in decisions.

Data platform

Also known as: analytics platform; data hub

Definition: A unified technology environment that ingests, stores, processes, and serves data securely and at scale for analytics and AI use cases. Examples include a cloud-based data platform that integrates transactional, third-party, and streaming data for enterprise reporting, or a governed data platform that enables self-service analytics while enforcing data quality and access controls.

Why it matters: The data platform is the operating system for data/AI, standardizing the ways teams ingest, transform, discover, and serve data.

Data product

Also known as: data asset; information asset

Definition: Curated and packaged data sets designed to be easy to access through APIs.

Why it matters: Data products make data easier to find, trust, and use for anyone who needs it for analytics, AI, or decision-making.

Data product owner

Also known as: data product lead; data product manager

Definition: The role accountable for the vision, value, and life cycle of a data product, ensuring it meets user needs and business outcomes.

Why it matters: A dedicated role focused on developing value from data.

Data scientist

Also known as: data analyst

Definition: Professionals who use data and AI to improve decisions and performance, frame problems, build models, run experiments, and communicate impact.

Why it matters: Translates business questions into models and experiments that move KPIs, not just reports.

Data steward

Also known as: data custodian; data governance steward

Definition: The role responsible for ensuring data quality, definitions, and governance standards within a specific data domain.

Why it matters: Data stewards focus on standards, quality, and risk to ensure the data foundation remains sound.

DataOps

Also known as: data operations; agile data engineering

Definition: Agile approach to managing data pipelines and analytics.

Why it matters: Improves data reliability and lead times through automation, testing, and observability.

Deep learning

Also known as: neural networks; multilayer learning

Definition: A subset of machine learning that uses very large, multilayered neural networks to learn complex patterns from data automatically. It can be used to detect defects from camera feeds and forecast demand using multivariate time series.

Why it matters: Deep learning fundamentally changes what computers can do with unstructured data, which traditional software couldn't handle well.

DevOps research and assessment (DORA)

Also known as: DevOps performance metrics

Definition: A set of four metrics (lead time, deployment frequency, mean time to recovery, change failure rate) used to measure software delivery performance.

Why it matters: Provides objective benchmarks for engineering quality and delivery speed.

DevSecOps

Also known as: DevOps with security

Definition: A methodology for building security checks directly into the software development and operations process instead of adding them at the end.

Why it matters: The practice of integrating security into every stage of the software development and deployment process. It enables speed without compromising compliance or safety.

Digital factory

Also known as: AI factory; innovation lab; tech hub

Definition: Dedicated environment for cross-functional teams to rapidly develop digital products.

Why it matters: Concentrates talent, tools, and governance to accelerate priority use cases from idea to scale. The digital/AI factory is a good model to use for companies early in their tech and AI transformation.

Digital trust

Definition: The confidence that AI systems protect data, operate securely, behave reliably, and are deployed responsibly.

Why it matters: Without trust, customers stop using your products, regulators intervene, and AI programs stall.

Digital twin

Also known as: virtual replica; simulated model

Definition: A virtual representation of a physical asset, person, or process. For example, a digital twin may simulate a production line to optimize throughput or model a building to cut energy costs.

Why it matters: A digital twin serves as a living model of a real-world system, enabling organizations to monitor, test, and improve it without touching the live version.

Domain

Also known as: business domain; value stream; end-to-end process; customer journey

Definition: A major area of the business (for example, sales, claims, fulfillment with clear ownership and outcomes. It's the anchor focus for digital, data, and AI transformation.

Why it matters: Clarifies accountability and defines the scope of the transformation so it is large enough to matter but small enough to manage. For example, for an insurer, claims is a business domain spanning from first notice of loss through adjudication and payment.

Domain owner

Also known as: business domain leader; value stream owner; capability owner

Definition: Senior business leader accountable for outcomes within a transformation domain.

Why it matters: Establishes ownership for value, budget, and backlog, preventing incentives from being split between business and technology.

Enterprise architecture

Also known as: EA

Definition: The structured blueprint of systems, platforms, integrations, and standards that define how technology supports business strategy.

Why it matters: Determines speed, scalability, reuse, and long-term flexibility.

Explainable AI

Also known as: AI explanation (AIX)

Definition: The ability of an AI system to provide clear, understandable, and transparent reasons for its outputs or actions (where feasible).

Why it matters: Builds regulatory confidence and reduces risk by allowing leaders to understand and defend AI-driven decisions.

FinOps

Also known as: cloud financial management

Definition: A financial management discipline for managing and optimizing cloud spending with clear accountability.

Why it matters: Prevents runaway cloud costs, maximizes spend efficiency, and aligns engineering velocity with financial discipline.

Foundation model

Also known as: pretrained model; base model

Definition: A large, general-purpose AI model trained on huge amounts of data that can then be adapted to many different tasks. For example, a language foundation model can power chatbots, document summarization, and code generation across the enterprise.

Why it matters: Choosing the right foundation model governs cost, accuracy, speed, and IP risk across dozens of use cases—this is a portfolio decision, not a one-off.

Gateway

Definition: A control point that manages and secures communication between technology systems.

Why it matters: Controls access, enforces security policies, and ensures reliable communication between systems—key elements for speed.

Generative AI (gen AI)

Also known as: content-generating AI

Definition: AI based on foundation models to create new content (for example, text, images, audio, video). For example, gen AI can be used to draft first-pass marketing copy and product images for campaigns that teams refine. It can also generate code and test cases to accelerate feature delivery.

Why it matters: Gen AI is the first technology that can produce novel, human-quality content on demand.

Generative pretrained transformer (GPT)

Definition: A large language model widely used in generative AI applications, capable of generating text, reasoning over inputs, and supporting conversational and content-generation use cases.

Why it matters: It powers generative and agentic AI applications but requires strong governance and integration capabilities to deliver enterprise value.

Global capability center (GCC)

Also known as: global engineering hub

Definition: A dedicated team located in another geography that provides technology, AI, or operations support to the company.

Why it matters: Expands access to talent and can accelerate AI development when integrated strategically.

Infrastructure as code (IaC)

Also known as: programmable infrastructure; infrastructure automation

Definition: The practice of provisioning and managing infrastructure through code so environments can be created automatically and consistently.

Why it matters: Creates repeatable, secure environments on demand, thereby reducing outages, degradation over time, and lead times for new initiatives.

Key performance indicators (KPIs)

Also known as: performance metrics; success metrics

Definition: Quantifiable measures used to track the progress or performance of a business objective.

Why it matters: Focuses teams on specific, easy-to-track outcomes. KPIs should be tied to objectives and key results.

Knowledge graph

Also known as: knowledge network; semantic network

Definition: A structured map of business information and how items relate to each other, making it meaningful for gen AI.

Why it matters: Improves AI accuracy and reasoning by making data easier to understand.

Large language model (LLM)

Also known as: advanced language model

Definition: A type of AI foundation model trained on text to perform language-related tasks (for example, to generate human-like responses). For instance, an LLM can be used to create a knowledge assistant that cites policy pages to answer employee questions, or it can create a customer service copilot that summarizes cases and proposes next actions.

Why it matters: Language is the interface for almost everything humans do, and LLMs are the first systems that can engage with language flexibly and competently across virtually any domain.

Learning and skill development journeys

Also known as: training pathways; capability-building programs

Definition: Structured pathways to build employee skills aligned with organizational goals.

Why it matters: Scales new ways of working, reduces dependency on external hires for critical

skills, and ensures the wider organization understands how to work in new tech- and AI-enabled ways—not just the tech teams.

LiveOps (live operations)

Also known as: continuous operations (postlaunch

Definition: A system for managing the ongoing updates and engagement of digital products after launch.

Why it matters: Lifts engagement and revenue after launch. The continuous tuning of features, offers, and pricing keeps customers active and reduces churn.

Machine learning (ML)

Also known as: predictive modeling; statistical learning

Definition: A form of AI that can learn without explicit programming by a human (some ML algorithms are specialized in training themselves to detect patterns; this is called deep learning.

Why it matters: ML algorithms can detect patterns and learn how to make predictions and recommendations by processing data and experiences, rather than by receiving explicit programming instructions.

MLOps

Also known as: machine learning operations; AI operations

Definition: Practices and tools for deploying and maintaining ML models in production.

Why it matters: Keeps models dependable, monitored, and cost-effective, so pilots don't die on the path to scale.

Model context protocol (MCP)

Definition: A standard method that allows AI systems to securely connect to company tools and data (similar to what APIs do for non-AI systems.

Why it matters: Reduces integration complexity in AI systems.

Modular architecture

Also known as: decoupled architecture

Definition: A system design approach in which components are treated like independent blocks that are connected through APIs, making it easy to change or upgrade parts of the system without affecting the rest of it.

Why it matters: Makes the technology systems faster and less fragile—one upgrade doesn't break everything else.

Multimodal AI

Also known as: cross-modal AI; multisensory AI

Definition: AI that can use more than one type of input or output together (for example, text plus images or audio) in a single system. For instance, a multimodal AI assistant may read a technical document, look at a photo of equipment, and listen to a short voice note to suggest a fix.

Why it matters: Unlocks higher-impact use cases (field service, safety, design because real work spans documents, images, speech, and other signals, not just text.

Objectives and key results (OKRs)

Also known as: goal-setting framework

Definition: A team-level goal system that links clear objectives to important and measurable results.

Why it matters: Creates transparency into progress on priority outcomes, particularly around how much value has been captured.

Observability

Definition: The ability to see in real time whether systems are working properly, and quickly identify what is wrong if they are not.

Why it matters: Reduces downtime, accelerates time to addressing issues, and protects customer experience.

Opaque AI models

Also known as: black-box AI

Definition: AI models whose internal workings are not meaningfully interpretable to people.

Why it matters: A degree of model opacity should drive governance choices. In situations where you need auditability or fairness reviews, you can pick models that are explainable or add controls to do so (for example, monitoring, human-in-the-loop.

Persistent funding

Also known as: product-based funding; continuous funding model

Definition: A funding approach in which budgets are allocated to teams or domains once a year based on business priorities, the value identified in the strategic road map, and the teams' track record.

Why it matters: Replaces stop–start projects with durable teams, accelerating value realization and reducing coordination waste.

Physical AI

Also known as: AI in robotics; embodied AI

Definition: AI applied in physical machines such as autonomous vehicles and robots to perceive, reason, and act in the real world.

Why it matters: Moves AI from screens to the floor. When effectively implemented, it could reduce injuries, fill labor gaps, and improve throughput and quality in plants, warehouses, and field operations.

Platform

Definition: A reusable technology capability used by multiple parts of the company.

Why it matters: Drives speed, reuse, scalability, and lower cost across the enterprise.

Platform engineering

Also known as: internal platform development; infrastructure engineering

Definition: The development of internal self-service platforms and tools that software teams share across the organization.

Why it matters: Gives teams secure and tailored processes for CI/CD, data, and AI, with fewer bespoke stacks, faster delivery, and lower risk.

Platform reuse

Also known as: technology reuse

Definition: Deliberately reusing shared platform components (for example, sign-on and logging) across products instead of rebuilding them each time.

Why it matters: Cuts build time and run cost, improves reliability, and simplifies audits by standardizing a few well-managed services.

Platform teams

Also known as: enablement teams; infrastructure teams

Definition: Teams providing foundational technology services used by other delivery teams.

Why it matters: Centralizes widely shared and essential capabilities (security, reliability that product teams need, enabling them to ship faster and more safely.

Product and platform operating model

Also known as: product-centric model; platform-centric model

Definition: Organizational model in which delivery is organized around products and platforms, minimizing handoffs and project-based funding to enable durable, outcome-owning teams.

Why it matters: Aligns teams to value, not projects. Enables the agile team model to scale.

Product management

Also known as: product strategy; product leadership

Definition: The discipline of steering a digital product's direction to meet user needs and business goals.

Why it matters: Ensures scarce capacity is invested where it moves the P&L and customer outcomes most.

Product owner

Also known as: product lead; feature owner

Definition: The role responsible for defining and prioritizing work to maximize product value.

Why it matters: Converts strategy into a clear, ordered backlog that prevents duplication or misalignment.

Product teams

Also known as: feature teams; experience teams

Definition: Cross-functional teams that are responsible for the end-to-end life cycle of a product.

Why it matters: Persistent product teams build better products more quickly.

Prompt

Also known as: input query

Definition: The input or instruction given to a generative AI model to guide its response or output for a specific task.

Why it matters: The quality and specificity of prompts directly affect the quality, safety, and cost of AI outputs.

Quarterly business reviews (QBRs)

Also known as: quarterly performance reviews

Definition: Structured leader meetings to review team progress and plan next actions.

Why it matters: QBRs bridge the gap between long-term strategic goals and short-term operational priorities. They provide a regular check-in where senior leadership and teams can review progress against objectives, reprioritize work, and resolve dependencies.

Replatforming

Also known as: tech stack migration; cloud modernization; architecture overhaul; application re-engineering

Definition: This involves moving legacy systems to modern platforms in a way that improves speed, security, and flexibility (and not simply copying them over.

Why it matters: Avoids migrating old problems onto new platforms, unlocking flexibility, security, and speed while reducing total cost of ownership and enabling AI integration.

Responsible AI

Also known as: trustworthy AI; AI ethics/safety

Definition: The practice of governing and deploying AI in a safe, compliant, and ethical way while still enabling innovation.

Why it matters: Helps organizations protect customers, employees, and the public; reduce regulatory risk; and build the trust needed to scale high-value AI programs without surprise headlines.

Retrieval-augmented generation (RAG)

Also known as: retrieval-enhanced generation

Definition: A method that directs AI to look up trusted company documents before generating an answer.

Why it matters: Reduces hallucinations and ensures AI answers are accurate, current, and auditable.

Semantic layer

Also known as: business logic layer

Definition: A layer that translates complex technical data structures used by AI models into consistent business logic that is easy for humans to understand.

Why it matters: Prevents fragmentation of meaning across domains and ensures AI systems operate on aligned business definitions.

Shadow AI/shadow IT

Also known as: unapproved AI usage; unsanctioned AI; rogue AI projects

Definition: Unofficial AI or IT initiatives developed outside formal governance structures, often driven by business units trying to progress on their own.

Why it matters: Signals unmet demand and breakdowns in governance. Shadow builds can lead to security and compliance exposure.

Skills inference

Also known as: skill profiling; competency mapping

Definition: Using data to estimate employees' skill levels and identify gaps.

Why it matters: Enables targeted learning, smarter staffing, fairer promotion by grounding decisions in evidence, and more accurate budgeting for capability-building investments.

Site reliability engineers (SREs)

Definition: Engineers responsible for ensuring software systems run reliably in production by maintaining uptime, resilience, and performance.

Why it matters: Critical for scaling digital and AI solutions safely and reliably.

Skills taxonomy/architecture

Also known as: competency framework; skills map

Definition: Classification of skills within an organization, organized by categories and proficiency levels.

Why it matters: Provides a shared language for hiring, learning, and workforce planning, which are foundational for AI-driven talent tools.

Small language model (SLM)

Also known as: lightweight LLM

Definition: Smaller, domain-specific AI model with fewer parameters than foundation models to accomplish fit-for-purpose tasks. Examples include an on-device SLM powering offline sales coaching.

Why it matters: Allows economically viable use of AI in narrower settings, improving unit economics by matching model size to specific tasks and devices.

Software-as-a-service (SaaS)

Definition: Cloud-based software solutions delivered as subscription services.

Why it matters: Reduces complexity for utility systems, but it must be weighed against strategic differentiation needs.

Software development life cycle (SDLC)

Also known as: software development cycle; application development cycle

Definition: The structured process for planning, building, testing, deploying, operating, and continuously improving software.

Why it matters: Companies increasingly are relying on their ability to develop, deploy, and maintain software. When streamlined and automated, SDLC increases speed, quality, and reliability.

Talent win room

Definition: A dedicated HR unit focused on accelerating recruitment, onboarding, management, and development of critical technology talent.

Why it matters: Enables rapid build-out of digital and AI capabilities without waiting for full HR transformation.

Tech and AI transformation

Also known as: digital transformation; gen-AI-driven transformation

Definition: Reinventing how an organization operates to create, deliver, and capture value by

embedding digital technology, data, and AI into how work is done across the business. It often leads to major changes in products, processes, and organizational design.

Why it matters: It is the fastest route to step-change EBIT growth, creation of a sustainable competitive advantage, and resilience—modern firms grow faster, operate leaner, and adapt more quickly than legacy peers.

Technical debt

Also known as: tech debt; code debt; legacy debt

Definition: The future cost of software development that results from coding shortcuts or deferred maintenance.

Why it matters: Unmanaged debt slows delivery and raises incident risk; leaders must fund intentional pay-down.

Transformation office

Definition: A centralized leadership and governance structure responsible for leading and monitoring large-scale transformation initiatives.

Why it matters: Prevents drift, resolves conflicts quickly, and ensures promised value is delivered.

Unstructured data

Definition: Data that is not organized in predefined schemas (for example, emails, maintenance logs, audio, design files.

Why it matters: Unstructured data has traditionally been difficult to access. LLMs can now access it, although, to be useful, unstructured data still needs to be organized.

Upskilling versus reskilling

Also known as: skill enhancement versus role transition

Definition: Upskilling builds up existing skills; reskilling prepares for a different role.

Why it matters: Helps an organization focus on skills rather than on roles in building capabilities. Upskilling and reskilling require ongoing commitments, especially as the pace of change shifts skills needs more frequently.

Use case

Also known as: business application; AI application

Definition: A specific, practical application of digital or AI capabilities to solve a narrowly defined problem or deliver a clear outcome.

Why it matters: Use cases are the core elements for delivering solutions. Use cases need to be integrated and orchestrated so they enable broader adoption and scaling goals.

Value road map

Also known as: transformation plan; strategic road map

Definition: A plan linking initiatives to business value over time, including use cases, resources, timelines, and expected impact.

Why it matters: Provides a view of resource and capability needs over the course of a transformation; identifies specific funding tied to measurable economic outcomes rather than activities, enabling quarterly reprioritization based on actual impact.

Vector stores

Also known as: embedding databases

Definition: A specialized database that helps AI find information based on meaning, not just keywords.

Why it matters: Makes AI search smarter and more context-aware, improving answer quality.

Vibe coding

Definition: An informal approach to programming where developers rapidly build software by iterating with AI tools in a conversational, intuitive flow, focusing more on intent and experimentation than on detailed up-front design.

Why it matters: Dramatically lowers the barrier between idea and execution, enabling faster experimentation, shorter product cycles, and broader participation in software creation across the organization. The result could be to accelerate innovation and competitive advantage.

Workforce skill gaps

Also known as: talent gaps; capability gaps

Definition: The difference between the skills an organization needs and the skills its workforce currently has.

Why it matters: Quantifying gaps focuses investments on the few skills that unlock disproportionate value, often using skills inference and other analytics to decide where to act.