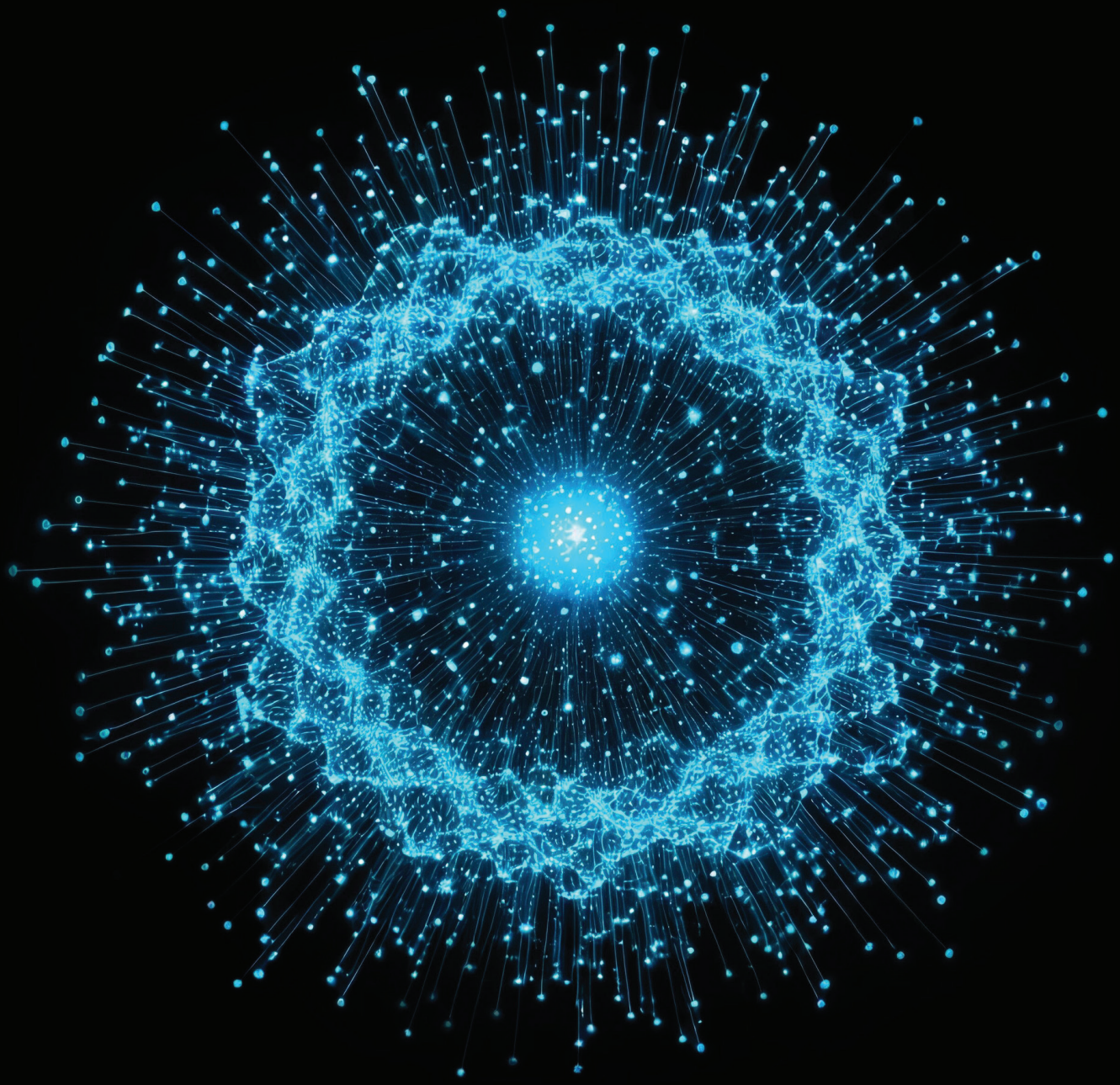
Deloitte.



AI-native workforce:
Future of work and skills in engineering and product value chain

February 2026

Table of contents

Executive summary

The technology landscape is undergoing a structural transformation driven by two converging forces: The maturity of Generative AI (GenAI) into "agentic" workflows and the stabilisation of hybrid work models. Software engineering is transitioning from a discipline of *creation* to one of *orchestration*, and product management evolves from *coordination* to *strategic acceleration.*

Current data indicate that GenAI tools can improve new product development times by 50 percent and significantly accelerate software development tasks, with productivity gains of 30–40 percent witnessed in engineering teams.[1] However, realising this value estimated at ~7 percent of global GDP[7] requires organisations to look beyond automating individual tasks and redesign entire workflows, develop skills and reimagine talent and leadership to accommodate human-agent collaboration.

The macro environment: Drivers of change

The era of agentic AI

We have moved past simple code completion. The future workforce will be a collaboration of people, agents and robots. It is suggested that while ~52 percent of worker tasks in the US could be completed faster with the same level of quality with the available technology; the goal is augmentation rather than displacement.[1,6] AI agents are becoming "virtual coworkers" capable of planning and executing multistep workflows, such as migrating legacy code or autonomously managing sales leads.

The remote work "productivity paradox"

A significant disconnect exists between engineering data and leadership sentiment.

Developer reality

About 64 percent of developers report higher productivity working remotely.[2] Data show a 4 percent increase in "focused work" (keystrokes per minute) and a 5 percent increase in coding during core business hours among remote workers.[2]

Leadership scepticism

Only 12 percent of leaders express complete confidence in remote productivity.[3]

Implication

Organisations must implement observability tools to bridge this trust gap rather than enforcing mandates that risk attrition.

We can see these trends playing out across the talent landscape and in how the roles of software engineers and product managers are evolving

New rules of technology talent

Global talent divergence[1]

From coder to conductor: The new rules of tech hiring

AI is causing a seismic shift in tech hiring, decoupling productivity from headcount and favouring specialised, senior talent.

The old hiring playbook: Focus on volume

Goal: Increase headcount for productivity

More engineers were hired to write more raw code and manually build features

The ideal candidate: "The coder"

Valued for proficiency in specific languages and ability to write code quickly

Roles now in decline

Data analysts ↓ Software testers ↓

Goal: AI-led supervision

Hiring fewer senior specialists to guide, validate and orchestrate AI outputs.

The ideal candidate: "The architect of intelligence"

Valued for designing systems where AI agents perform tasks securely and reliably

Roles now in demand

AI research ↑ **ML engineers** ↑

New focus on higher education

40–45 percent of roles in the Americas and Europe now demand a **Masters or PhD**

The new AI-driven playbook: Focus on value

Hiring trends reveal a geographical split in engineering value chains. The Americas and Europe are increasingly focusing on highly specialised talent (PhDs, masters) for AI research and model architecture. Conversely, South and Southeast Asia are witnessing high demand for application-based and operational engineering roles.

Volume vs. value

As AI automates routine tasks such as code generation, bug fixes and UI scaffolding, the demand for entry-level "coders" is softening. The focus has shifted to engineers capable of "AI-led Supervision" guiding, validating and integrating AI outputs rather than writing raw code.

The "Orchestrator" profile

Hiring is prioritising "Cross-disciplinary skills." Candidates must grasp adjacent domains, blending core engineering with data pipelines, model behaviour and governance risks.

Net new roles

Recruitment is opening for entirely new job titles, including AI enablement engineers, Internal Developer Platform (IDP) product managers, agent orchestration engineers and context (RAG) engineers, which are multi-skilled hybrids/variants from the "existing" jobs.

From roles to skills

Talent acquisition is shifting from "role-based hiring" to "capability/skill-based hiring." Recruiters are looking for evidence of "AI fluency" "ethical judgment" and "adaptability" rather than just years of experience in a specific language.

Internal build

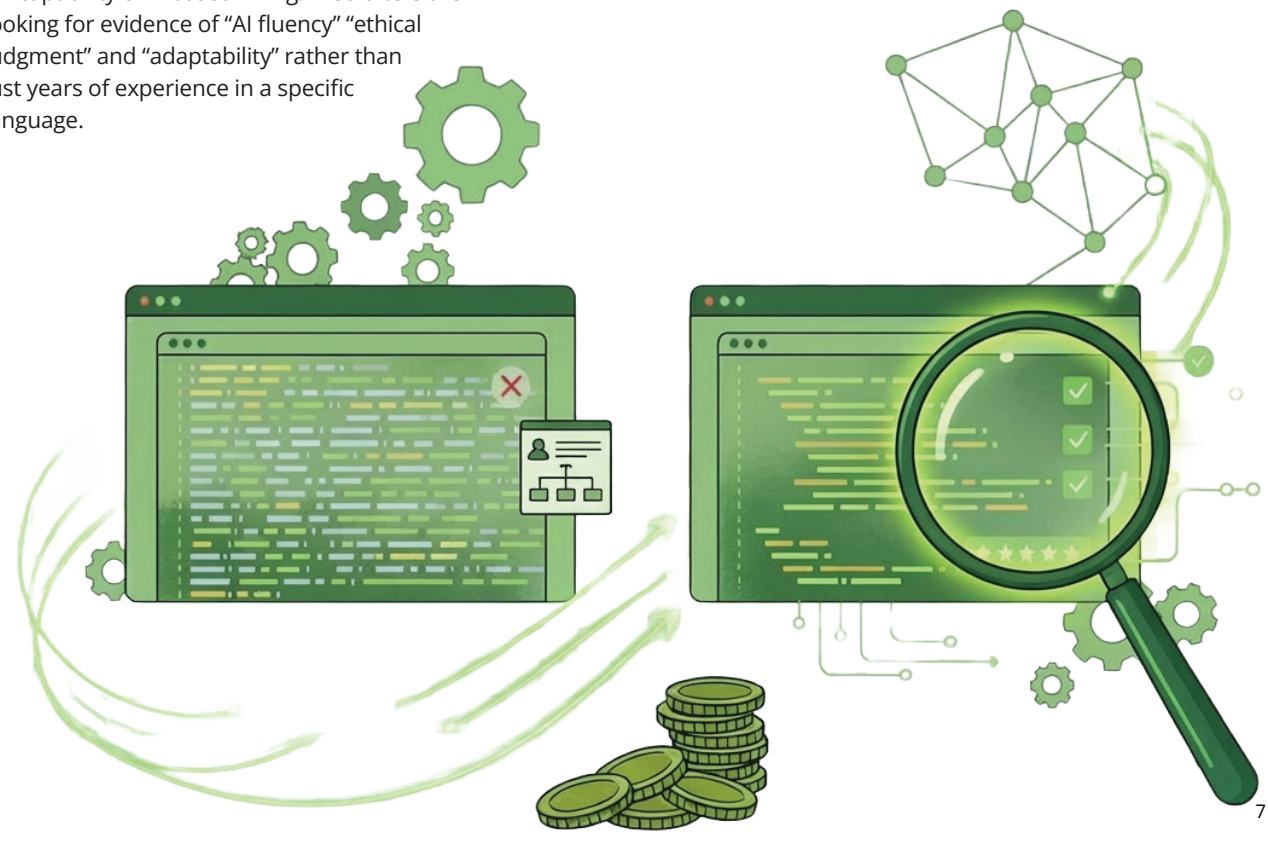
There is a strong preference for "reskilling over replacing." Instead of mass layoffs, companies are reskilling existing engineers for AI-driven roles.

Ecosystem sourcing

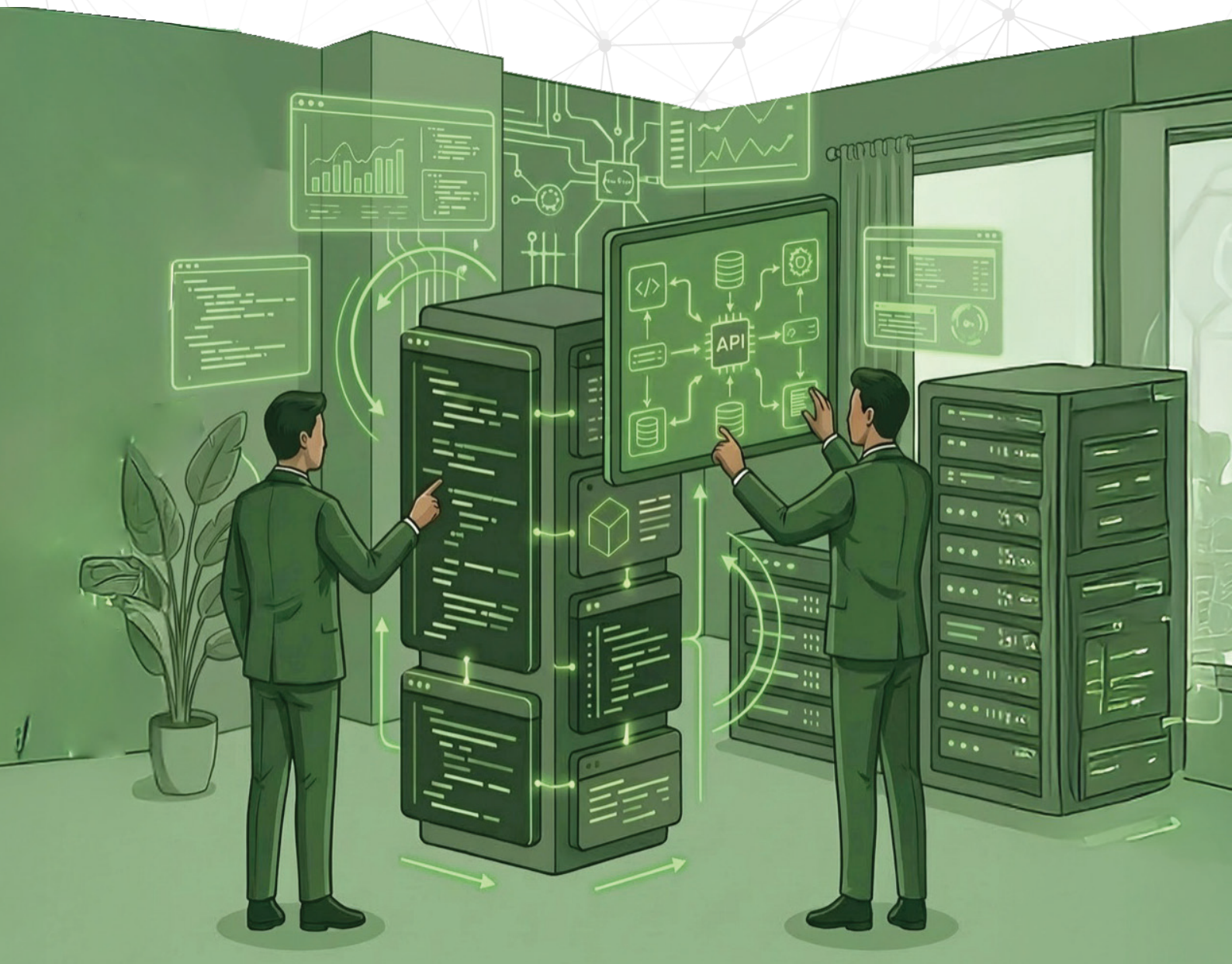
External hiring is increasingly focusing on non-traditional pools, such as open-source communities, hackathons and AI research collaborations, rather than relying solely on job portals such as LinkedIn.

Graduate expectations

For new graduates (next 24–36 months), companies are specifically looking for capstone projects with external sponsors, coursework that embeds AI into non-AI subjects (e.g., AI in OS or DB courses) and evidence of "disciplined AI usage" (e.g., maintaining AI logs and model critiques).



Evolution of software engineering roles



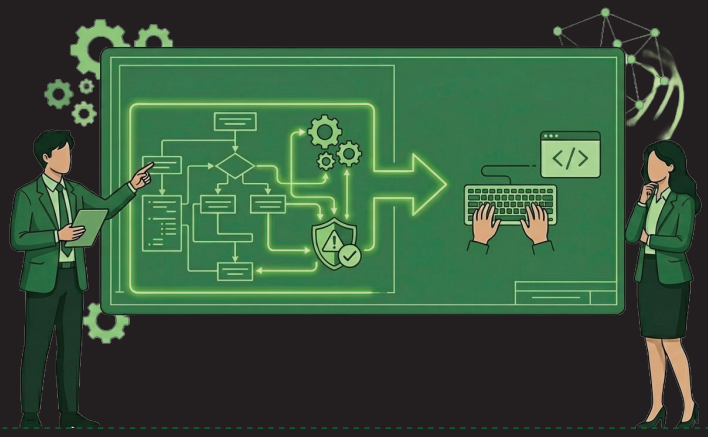
Snapshot of the emerging changes in the software engineering role

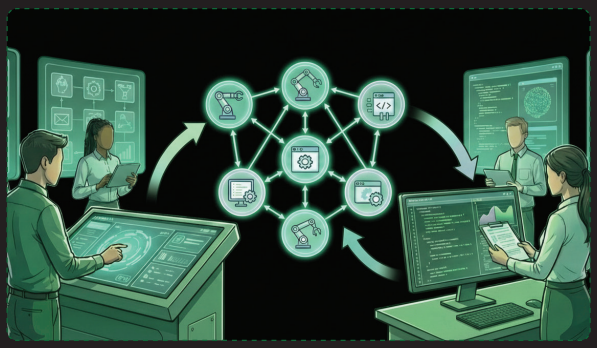
Software engineering is undergoing a major evolution, known as “SE 3.0.” This new paradigm, driven by AI, redefines the role of an engineer from a simple “coder” to a strategic “orchestrator” who manages and collaborates with AI agents to build complex systems.

The great reallocation: From creation to orchestration

Manual coding and reviews

The time spent on manual coding and is projected to drop significantly.





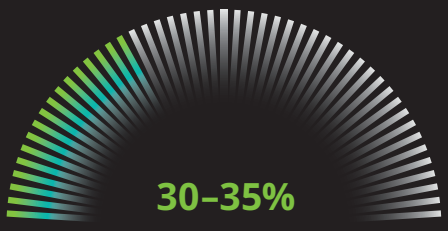
New focus: AI orchestration and oversight

Engineers will now define tasks for co&-genetation agents and manage their output.

System design over implementation

Architectural planning and defining risk are critical than hands-on coding

Productivity gains and emerging roles



30-35% Overall Productivity Lift

AI integration is expected to boost productivity across the entire lifecycle

Productivity impact across development stages

Coding and implementation

Testing

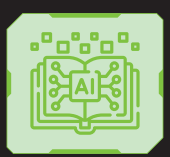
Architecture and design

Rise of the AI-native specialist

The traditional engineer role is fracturing into new specialisations

Agent orchestration engineer

AI enablement engineer

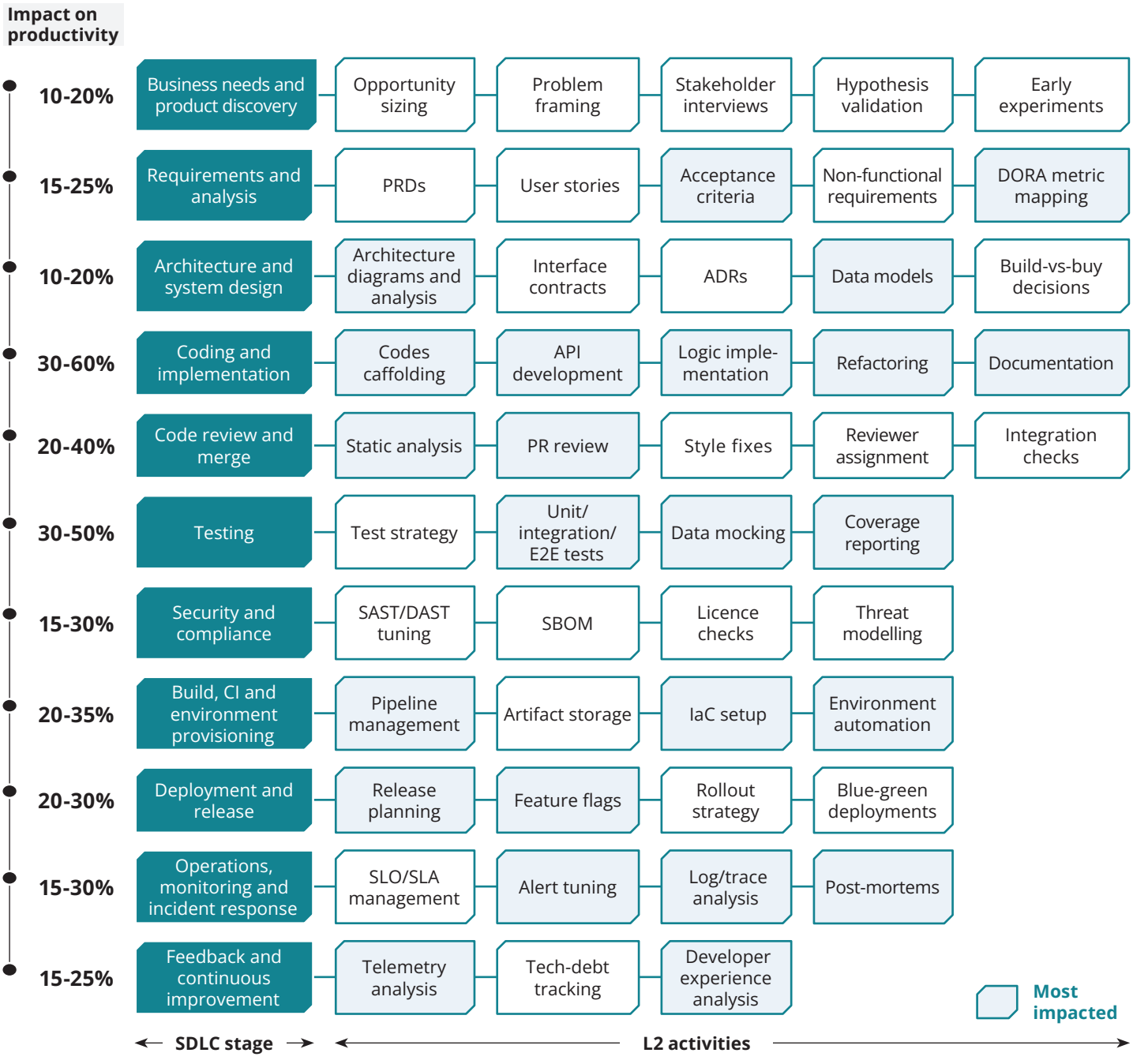


Deep dive: The future of Software Engineering (SE)

Deloitte research defines the future state of engineering as **"SE 3.0"**, characterised by a transition from manual "code factories" towards high-value oversight and human-AI collaboration.

The software development life cycle will witness productivity gains of **30-35 percent** due to the evolution of technology that includes internal developer platforms, re-usable code and GenAI-led tooling of coders.

Impact on software developement lifecycle in the age of technology and tooling



The majority of the impact will be seen in the areas of coding, implementation, reviews and testing, where the roles of a software development engineer and an engineering manager are expected to undergo fundamental shifts. We have specifically studied the impact of technology and tooling on four roles - Software development engineer, Machine learning engineer, Production service engineer and Engineering manager.

Software development engineer: From "Doer" to "Reviewer" and "Orchestrator"

The fundamental nature of coding is changing. Traditionally, Software Development Engineers (SDEs) spent approximately **70 percent of their time on code development and reviews.**[1] In the future state, this is projected to drop to **40 percent**, with the remaining capacity shifting towards AI/Agent Orchestration.[1]

The review burden

Mid-senior level engineers must develop advanced forensics skills to validate AI-generated code. Developers must shift from writing code to reviewing code for security flaws and subtle hallucinations.

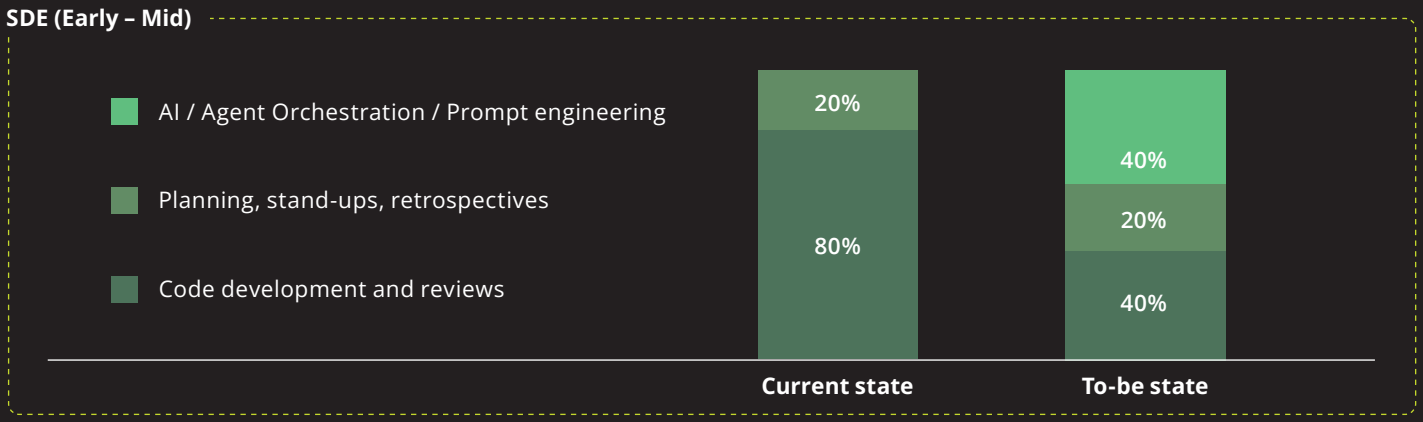
Productivity gains

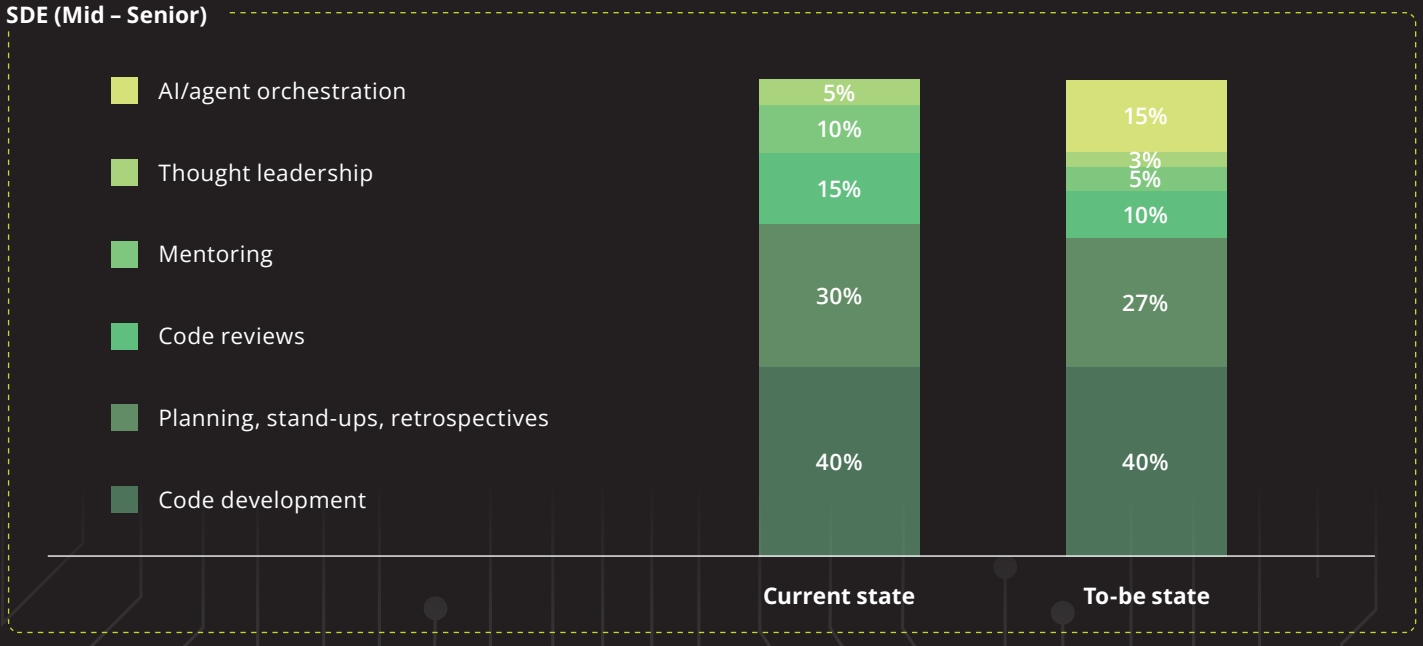
GenAI tools allow developers to insert 1.3x more characters per keystroke, indicating a massive reduction in typing and an increase in editing and orchestration.

Phase-specific impact

Productivity gains are uneven across the lifecycle: 30–50 percent in coding/ implementation versus only 10–20 percent in architecture and design, which remains highly human-centric.[1]

Software Development Engineer (SDE)





Machine learning engineers: Increased focus on "reliability"

Due to the wide availability of AI-led tooling, the nature of work for machine learning engineers in the future is expected to shift from data preparation to data model evaluation, monitoring and reliability.

Data preparation

Today, where the majority of time is spent on cleaning and labelling data as well as maintaining data quality, is expected to move towards curating a knowledge corpus and data contracts with synthetic data generation governed through agent-led data pipelines

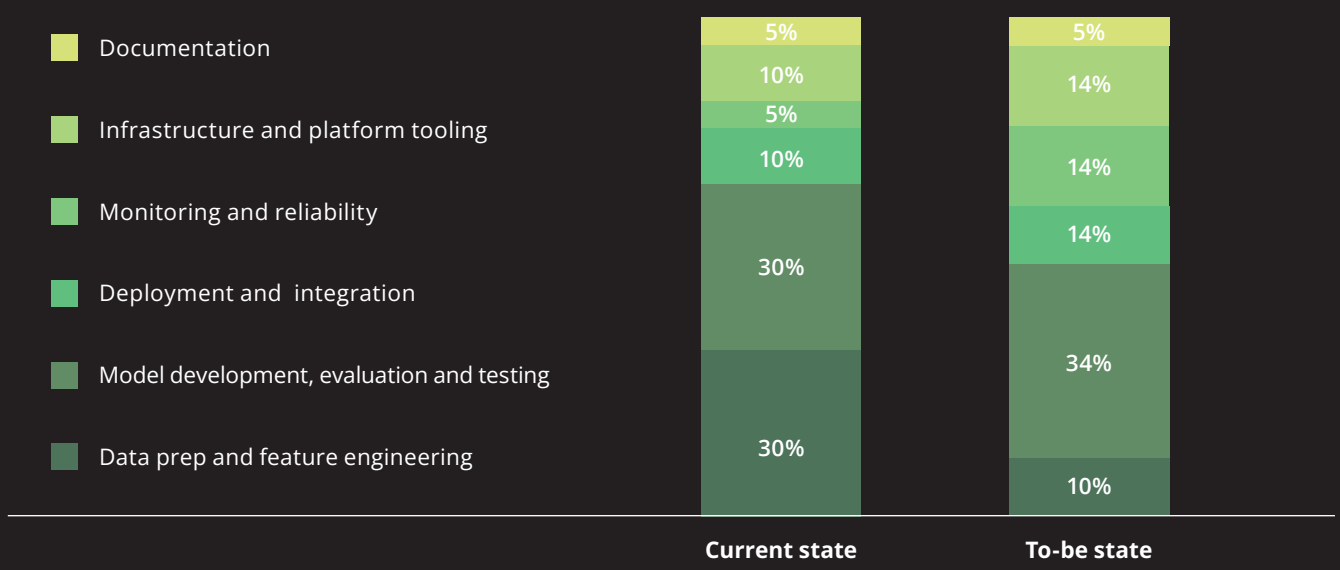
Model development and testing

In the future-state, while "systems" and "agents" will focus on selective fine-tuning of models, about 33 percent of the time available will be focused on model testing through rigorous evaluations of models focused on tasks, safety, robustness, hallucinations, etc., aligned to global safety requirements.

Monitoring and reliability

Time spent on monitoring and reliability will grow by 3X due to increasing visibility through telemetry, tracing and prompt lineage to ensure outputs are in line with expectations.

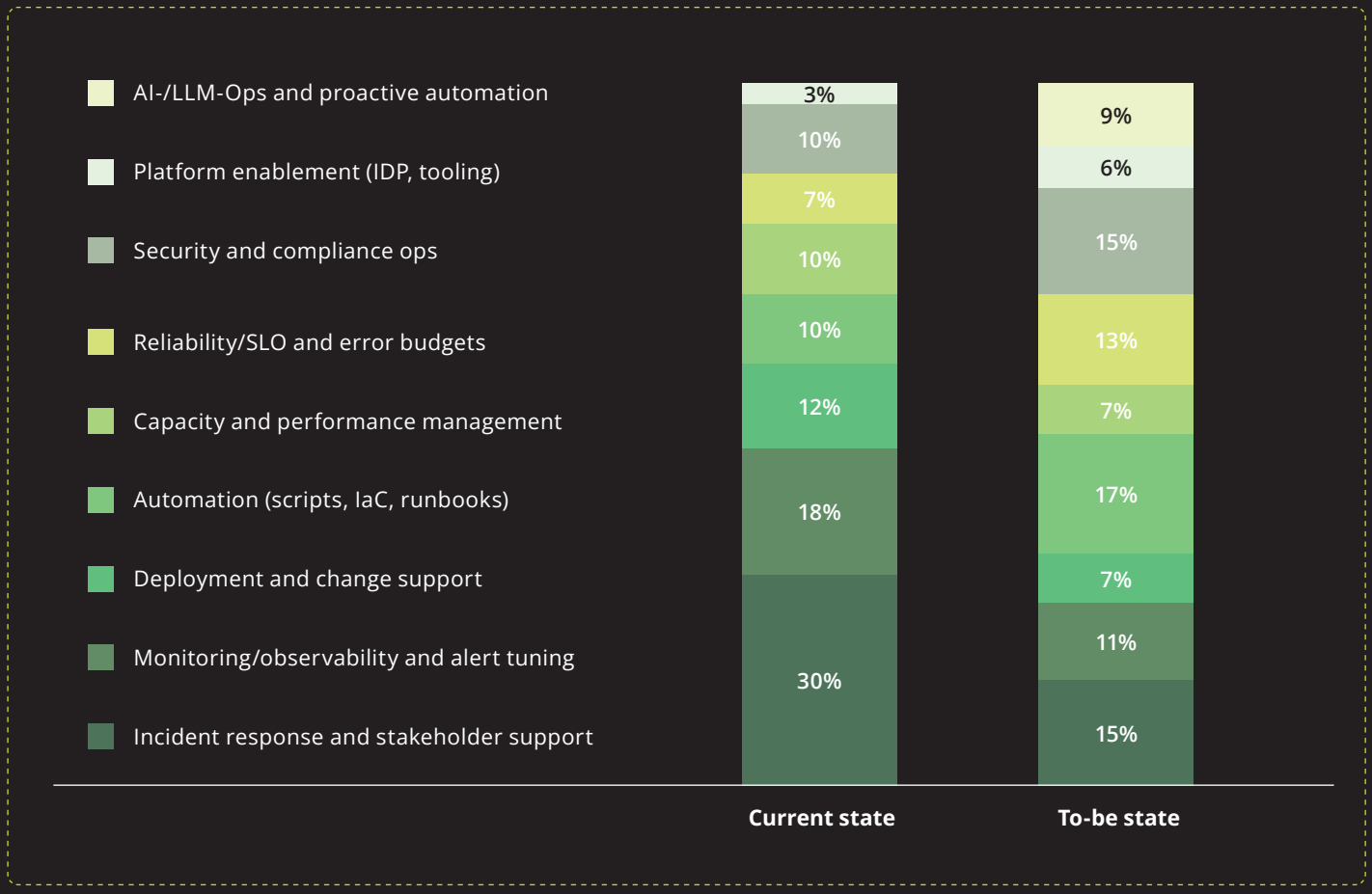
Machine learning engineers



Production service engineer: Declining manual ops

DevOps is solidifying into a cloud-first reliability discipline, where infrastructure is treated entirely as code and manual monitoring is replaced by AI-driven observability. The role of production engineers is shifting from incident response and stakeholder management (enabled by self-service portals and internal developer platforms) to developing automations for scripts and runbooks, while maintaining reliability, ensuring compliance and ensuring security.

Production service engineer



Engineering Manager: Re-balancing managerial responsibilities

Technical oversight and reliability will become the primary focus, while tasks such as delivery coordination and reporting will become less prominent.

AI-driven changes

Managerial roles are shifting from supervision to strategic focus, supported by tools for analytics, code reviews and dependency management.

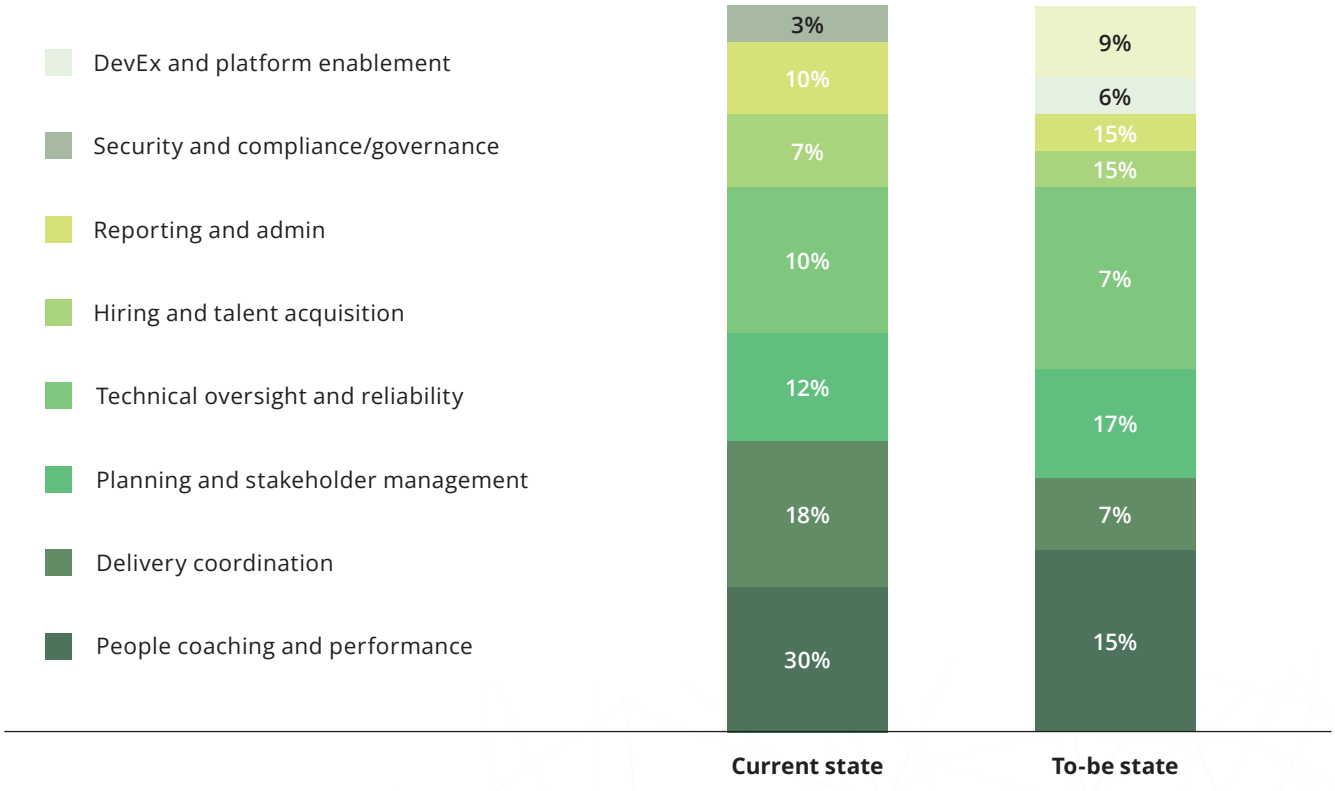
Reduction in tactical work

AI and internal developer platforms will reduce cognitive load, automate routine tasks and enable self-service environments.

Transformation enablers

Adoption of internal developer platforms, standardised metrics, error-budget policies, clarified team structures and AI agents with guardrails will streamline processes to improve reliability, reduce firefighting and enhance communication.

Engineering manager



Emerging roles and “seismic” shifts

Legacy roles are fragmenting into specialised, high-value functions:

Agent orchestration engineers

Professionals who compose AI agents, define guardrails and manage model selection. These engineers build “logic” to drive agent behaviour and ensure safety, reliability and observability.

Platform engineers

A shift from manual infrastructure management to building Internal Developer Platforms (IDPs). This role focuses on “paved roads” and automation-first orchestration.

AI enablement leads

New roles dedicated to measuring and improving developer velocity through AI toolchains by curating “golden paths” for AI-assisted coding and reviews.

Context/Retrieval engineers

Specialists who build RAG (Retrieval-Augmented Generation) layers, so development agents understand internal APIs and documentation.

Synthetic data engineers

Generate synthetic datasets to increase data diversity, address privacy and enable model training under compliance, especially in highly regulated industries such as healthcare, defence and finance.

AI governance and compliance engineer

Focuses on AI model inventory, risk classification, audit trails and policy adherence (per NIST AI RMF / EU AI Act, or any applicable).

Case study 1

Emerging trends in the engineering function – Global design platform

AI-augmented engineering and productivity

Use case	Emerging roles in the ecosystem
• Use GitHub/Copilot-like tools to scaffold functions, auto-generate tests and accelerate debugging • Internal tools auto-summarise documentations and recommend reusable components	• LLM engineers • AI product managers

1

Embedded AI / ML roles within Product Squads

Use case	Emerging roles in the ecosystem
• Embedded GenAI tools for text and image generation, integrated directly into product PODs • ML models powering content ranking and personalisation	• ML engineers • Context engineers • AI product managers

3

Platform engineering as a force multiplier

Use case	Emerging roles in the ecosystem
• Platform exposing APIs and runtime environments for internal/external developers to build extensions • Data-as-a-product enabled with service aligned data platform for scaling analytics	• Platform engineers • DevEX engineers

2

Responsible AI as core

Use case	Emerging roles in the ecosystem
• Campaigns on ethical use and bias reduction in creative AI • Content moderation with safeguards against unsafe/copyright content in AI outputs	• Trust and safety engineers • Compliance engineers • ML engineers

4

Evolution of product management roles

Snapshot of the future of Product Management (PM)

The traditional PM: The coordinator

Primary focus: Administrative toil

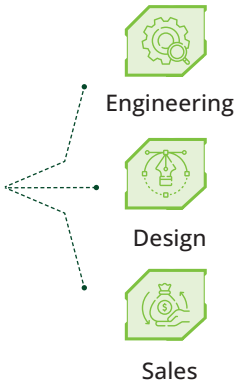
Responsibilities included backlog grooming, meeting summaries and routine documentation.

Role as a communication hub

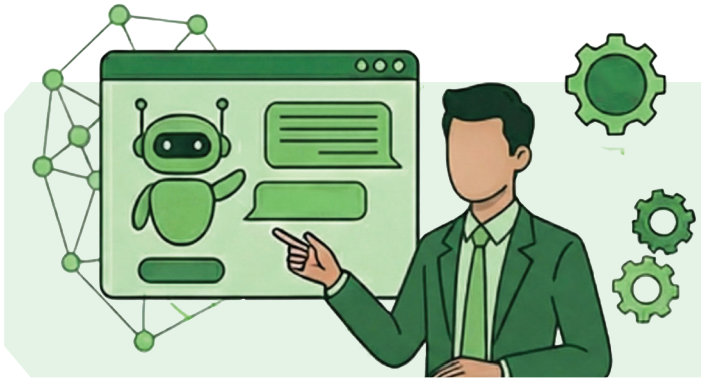
Acted as the central point of contact between engineering, design and sales.

Core skills: Process management

Expertise was centred on agile methodologies and stakeholder management.



The future PM: The AI-augmented "mini-CEO"



Focus shifts to strategic vision

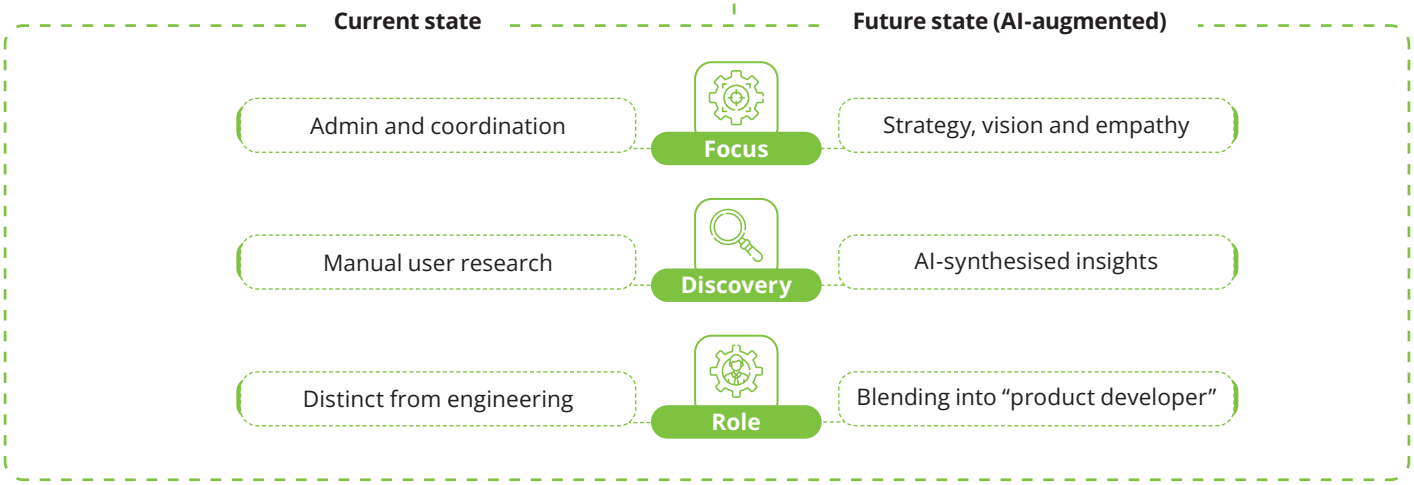
PMs use AI to synthesise data, enabling them to lead product strategy.

40% increase in productivity

GenAI tools reduce "high toil" tasks and accelerate decision-making.

New skills: "Agentic" fluency and risk management

PMs must master complex AI systems and safely integrate ethical safeguards.



Deep-dive into the emerging changes in the PM role

The AI-Augmented "Mini-CEO"

Product managers are using GenAI to compress the Product Development Life Cycle (PDLC), shifting from coordination to strategic acceleration.

Acceleration: Use of GenAI in the PDLC accelerates new product development by up to 50 percent.[5 4]

High-impact areas: The most significant gains are in "content-heavy" tasks. GenAI has nearly twice the positive impact on tasks such as synthesising user research, drafting press releases and creating Product Requirements Documents (PRDs) compared with "content-light" data visualisation tasks, leading to efficiency gains of ~10 percent in the role of product managers.[18]

Shifts in responsibilities and skills

As administrative toil decreases, the PM role is pivoting:

Strategic alignment

PMs can focus on product vision and stakeholder management, which remain largely human-centric tasks.

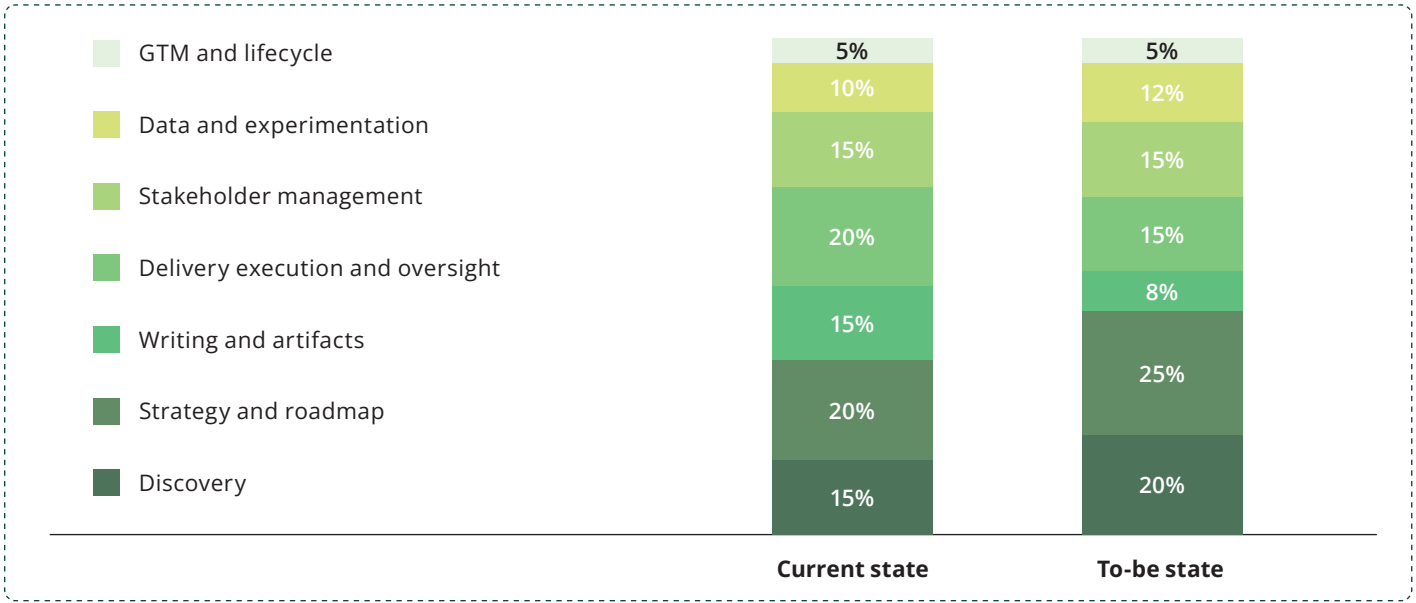
Agentic frameworks

PMs must now understand how to deploy LLMs that work together to complete tasks (e.g., agentic frameworks"), requiring proficiency in low-code tools and iterative prompting, leading to a decrease in MVP cycles and documentation.

Risk stewardship

PMs are increasingly responsible for working with risk experts to integrate safeguards into the product definition phase, addressing liability, data privacy and trust deficits.

Product management



Role convergence: The PM and developer roles could eventually merge into a singular "Product Developer" persona. This individual would use AI tools to define requirements and immediately generate the code, prepare an MVP, rather than building out a traditionally long PRD/BRD.

The experience gap: Senior PMs derive higher quality outputs from GenAI because they possess the "product sense" required to effectively critique AI output. Junior PMs gain speed but often at the expense of quality, necessitating new mentorship models.

Emerging skills of the future

The skills matrix: 2025–2030

The shelf-life of technical skills is shrinking. Organisations must pivot from role-based planning to **skill-based planning.**

Category	Emerging/High demand	Stable/Core	Declining/Legacy
Engineering	**AI fluency:** Prompt engineering, RAG, agent orchestration, assisted code development (7x growth in demand).	**Core languages:** Python, Java, SQL.	**Legacy scripting:** Node.js (in some contexts), Unix Shell.
	Cloud-native security: "Shift-left" security, identity governance.	**DevOps:** CI/CD, SRE, Kubernetes.	**Manual QA:** Replaced by AI-augmented automated testing.
	System thinking: Designing distributed, agentic architectures.	**System design:** Microservices architecture.	**Rudimentary analysis**: Business analysis, BI, statistics
	DevEx product skills: Internal developer platforms implementations, including "golden paths" and policy implementation.	**Privacy/Data governance**: Data quality and drift.	
Product	**AI Literacy:** Understanding model capabilities/costs.	**Communication:** Stakeholder management.	**Basic admin:** Routine documentation, meeting notes, basic backlog grooming.
	User empathy: Interpreting unarticulatd needs.	**Lifecycle management:** Planning and execution.	
	Data analytics: Real-time data synthesis.		
Human skills	**Problem-solving:** Top soft skill (21% focus).	**Collaboration:** Working in hybrid/ remote teams.	**Rote execution:** Following rigid, pre-defined process steps.
	Adaptability/Resilience: Psychological safety in rapid-change environments.		
	Decision quality: Supplement technical proficiency with judgment-led reasoning.		

Case study 2

Emerging trends in the engineering function | Large global e-commerce and cloud services player

AI and LLMs as product fabric

Use case	Contributing skills
GenAI to support search, product recommendations, seller tools and chat-based personalisation	• LLM engineers • Applied scientists • AI product managers

1

Observability, SRE and self healing

Use case	Contributing skills
Comprehensive monitoring, chaos engineering and automated recovery for critical events	• SRE practices • Monitoring • Anomaly detection • Automated remediation

4

Cloud native and compostable architectures

Use case	Contributing skills
Scalability during peak events, faster deployment cycles and robust multi-cloud resilience	• Kubernetes • Service mesh • API design • DevEx platform development

2

Real-time and streaming analytics

Use case	Contributing skills
Real-time personalisation, fraud detection, inventory and pricing decisions, and logistics optimisation	• Streaming data pipelines • Online ML • Reinforcement learning

5

Edge computing and low-latency

Use case	Contributing skills
Last-mile logistics IoT devices use edge computing	• Edge system engineering • Offline-first design

3

Advanced AdTech and measurement

Use case	Contributing skills
Integration of commerce and advertising with advanced attribution and RTB capabilities	• AdTech engineering • Growth analytics • Causal inference for ad incrementality

6

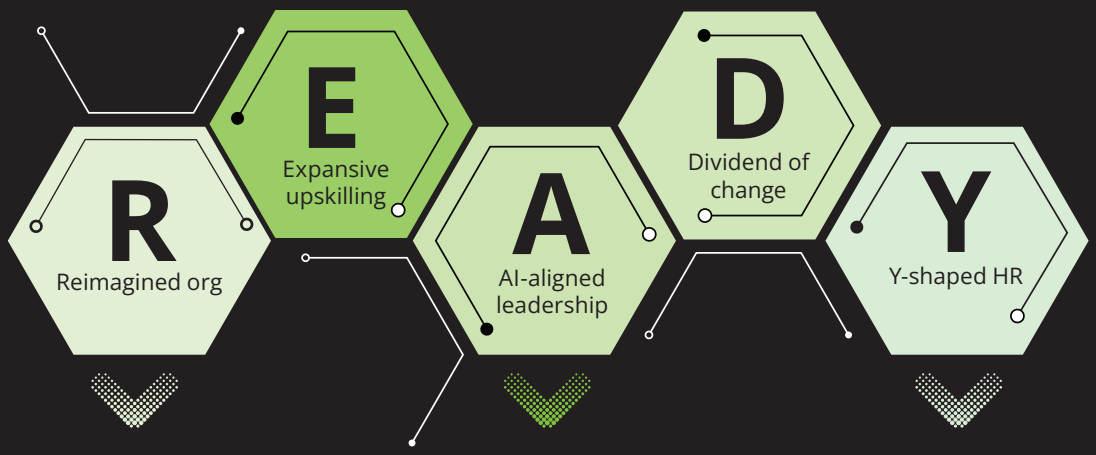
Organisations must be READY to embrace this change

The R.E.A.D.Y Framework

To capitalise on the ~US$7 trillion opportunity[7] supported by advances in GenAI, natural language processing and automation, organisations must move beyond piecemeal tool adoption. The following recommendations utilise the **R.E.A.D.Y.** framework to guide the transformation.

Shift training from "doing" to "reviewing."Build skills in architectural thinking and forensic review of AI-generated output.

Measure outcomes, not inputs, to prove AI value. Track metrics such as delivery velocity and cycle time reduction, not just output volume.



Redesign workflows around AI-human collaborations. Shift humans from task execution to high-value roles such as oversight and architectural validation.

Build trust and manage wider, AI-augmented teams. Implement automated governance (Policy as a Code) to ensure AI safety and build confidence.

Evaluate the "human+agent" as the new unit of productivity. Redefine performance based on the leverage an individual extracts from their AI tools.

Trust deficit
66% of developers cite distrust in AI output as a challenge.

Leadership span
Manager spans of control are rising from ~5.9 to ~7.7 reports.

Skills demand
The demand for AI fluency has increased sevenfold in two-year period.

R Reimagined organisation and AI-human workforce

Shift from "task automation" to "workflow redesign". The most significant economic value comes from redesigning entire workflows to accommodate a collaboration between people, agents and robots, rather than just speeding up individual tasks.

- **Implement "agentic" workflows:** Restructure teams away from linear hand-offs towards agent-orchestrated loops. For instance, in software modernisation, redesign the workflow so that an *Assessment Agent* scans dependencies, *a Coding Agent* migrates the code and the human engineer acts as the "Architect", validating integrity.
- **Define "human-in-the-loop" protocols:** As AI agents take over execution, such as drafting clinical reports or managing sales outreach, the human role must shift to high-value oversight and handling exceptions.
- **Operating model evolution:** Move towards a two-tier operating model comprising "Product Feature Verticals" (customer outcomes) and "AI/platform capability horizontals" (tooling and enablement). This ensures agent orchestration and guardrails are standardised.

E Expansive upskilling and capability building

Solve the "mentorship gap" and prioritise AI fluency. As AI automates entry-level tasks such as coding syntax and documentation, junior employees lose the traditional "on-the-job" training ground.

- **Formalise mentorship 2.0:** Senior engineers must actively "shadow" junior staff and engage in co-coding sessions not to teach syntax, but to transfer judgment, architectural thinking and risk assessment.
- **Target "upstream" and "review" skills:**
  - **Engineering:** Shift focus from writing code to reviewing code. Engineers require advanced forensic skills to identify subtle anomalies or security flaws.
  - **Product management:** Upskill PMs in "upstream" tasks such as user research synthesis and strategic visioning, as AI handles content-heavy drafting.
- **Mandate AI fluency:** Demand for AI fluency has grown sevenfold in two years. Training must go beyond basic prompting to include "agentic frameworks" — understanding how to chain multiple LLMs together to complete complex tasks.

A AI-aligned leadership perspective

Orchestrate trust and manage wider spans of control. Leadership must pivot from directive supervision to facilitative orchestration.

- **Bridge the "trust deficit":** A significant barrier is the lack of trust in AI output; 66 percent of developers cite distrust as a challenge.[17] Leaders must implement "Policy as Code" automated governance that evaluates AI code against security policies within the pipeline to build confidence.
- **Expand spans of control:** AI automation allows for flatter hierarchies. As routine oversight is automated, managers can oversee larger teams, with spans of control at major tech firms already rising from ~5.9 to ~7.7 reports.[1]
- **Standardise tooling:** Leadership must standardise the AI toolchain to prevent fragmentation. General-purpose tools (such as ChatGPT) have shown higher adoption rates among product managers than specialised tools, suggesting a need for careful selection.

D Dividend of change and adoption readiness

To fully realise the productivity dividend from AI, organisations must address the adoption gap, not just deploy new tools. Many leaders experience a "productivity paradox," where activity levels appear high, but business impact remains unclear. Closing this gap requires reframing how productivity is defined, observed and reinforced.

- **Measure outcomes, not outputs:** Move beyond legacy measures such as lines of code or hours logged. Redefine productivity in terms that leaders and teams can trust, i.e., cycle-time reduction, faster time-to-market (e.g., ~50 percent reduction in NPD times), quality improvements and customer impact. Clear outcome metrics help teams understand what good looks like in an AI-enabled environment.
- **Reinvest efficiency gains to reinforce adoption:** Time saved through AI (often 30-50 percent in coding, analysis or sales tasks) should be deliberately reinvested, not absorbed. Redirect this capacity towards higher-value work, such as innovation, stakeholder engagement and problem-solving, to visibly demonstrate the benefits of adoption and sustain momentum.
- **Build transparency through observability, not surveillance:** Implement development observability and workflow analytics to provide leaders with confidence and teams with clarity on how work flows end-to-end. This enables trust-based management, supports flexible work models and reduces the need for blunt controls such as rigid return-to-office mandates.

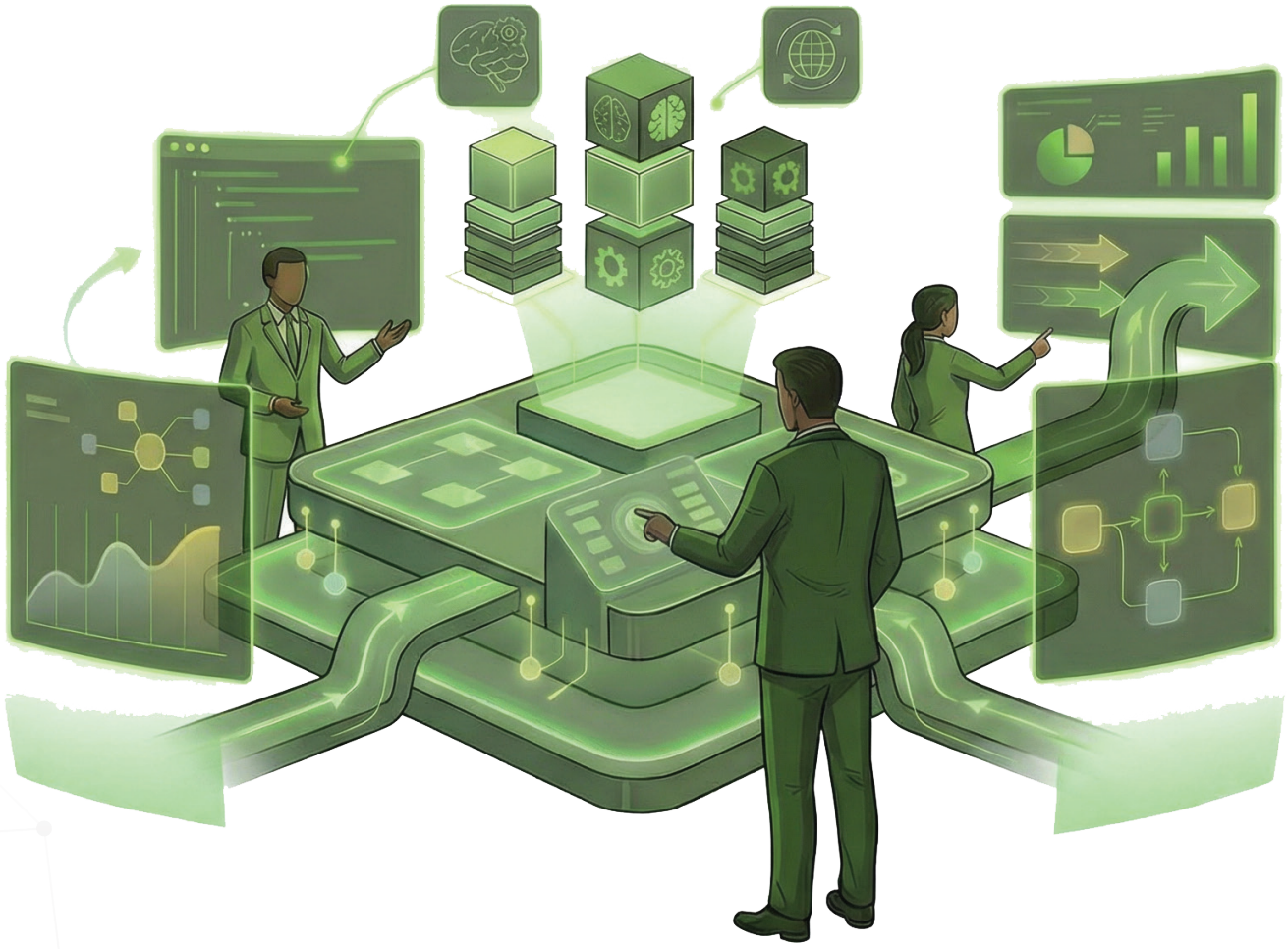
Change dimension	Leaders (CXO/ExCo)	Managers (Engineering, Product)	Teams (SDEs, PMs)
Mindset shift	Move from *activity assurance* to *outcome confidence;* accept that productivity is no longer visually observable	Shift from *supervising effort to* enabling flow and *removing friction*	Shift from "doing more work" to "delivering more value"
What success looks like	Business outcomes: faster time-to-market, quality, customer impact	Predictable delivery, reduced cycle time, fewer handoffs	Clear goals, fewer low-value tasks, meaningful impact
Metrics and measurement	Sponsor outcome-based metrics (velocity, quality, cycle time) and stop asking for legacy proxies	Translate outcome metrics into team-level signals and dashboards	Understand and align daily work to outcome metrics
Behavioural change required	Trust AI-enabled teams; resist reverting to old controls	*Coach teams* on prioritisation and reinvestment of freed-up time	Actively use AI to *offload routine work* and focus on judgment
Reinvestment of time saved	Mandate intentional reinvestment (innovation, growth, customer focus)	Actively reallocate 30–50 percent of time savings to higher-value work	*Take ownership* of higher-order problem-solving and innovation
Adoption role	Visible sponsorship; reinforce "new definition of productivity" consistently	Role-model AI usage and new ways of working	Normalise AI as part of everyday work, not a special tool
Observability and trust	Use observability as a confidence mechanism, not a control lever	Use workflow insights to unblock teams, not micromanage	View observability as support for better outcomes, not surveillance
Risk to manage	Reverting to RTO or activity tracking due to a lack of confidence	Over-interpreting metrics or managing by dashboards	Fear that AI usage will be misunderstood or penalised
Primary change ask	"Redefine productivity and reward outcomes"	"Enable flow, protect focus and reinvest capacity"	"Adopt AI, elevate work and own outcomes"

Sustained AI value comes not from tool deployment alone, but from aligning metrics, leadership behaviours and reinvestment choices to drive lasting change and adoption.

Y-shaped HR and people function evolution

Treat skills as data and redefine performance. HR must evolve into a discipline with deep expertise in people strategy, combined with technical data capabilities.

- **Skills-based strategic planning:** Move away from role-based hiring to skill-based planning. Use AI inference tools to scan internal code repositories and external profiles to create a real-time inventory of the organisation's skills (e.g., "we need RAG capabilities," not just "a developer").
- **Performance management of the "micro-unit":** Revamp performance reviews to evaluate the combined output of the human and their AI agents. Performance should be measured by the leverage an individual extracts from AI tools.
- **Sourcing from non-traditional pools:** As degree-based requirements become less relevant than skill proficiency, expand sourcing to ecosystem collaborations and hackathons to fill critical shortages in cloud-native and AI security roles.



The way forward

The future software organisations will look fundamentally different from the past. Historically, much of the engineering effort was spent on manual, repetitive activities, such as writing boilerplate code, conducting routine tests and managing low-value execution work. With the emergence of AI agents, a sizable portion of this effort is now automated and accelerated.

As a result, the role of engineers and product managers is shifting up the value curve. The focus moves from execution speed to design judgment, architectural decisions and end-to-end orchestration. Human talent increasingly owns the “what” and the “why”, setting direction, shaping product strategy, reviewing outputs for quality and risk and ensuring that multiple AI-enabled components work together seamlessly.

In this new model, competitive advantage is no longer defined by how quickly teams can produce code, but by their ability to envision the proper outcomes, make informed trade-offs and consistently deliver high-quality, business-relevant solutions at scale.

References

01. *Deloitte primary research and analysis conducted in collaboration with Adobe on Future of Software Development & Engineering: Project Aurora 2025*
02. *Future of Work Report, Data on the Impact of Remote Work on Software Development, Software.com*
03. *Hybrid Work Is Just Work. Are We Doing It Wrong?, Work Trend Index Special Report, Microsoft*
04. *Deloitte research and analysis on future of product management*
05. *Cooper, Robert & McCausland, Tammy. (2024). AI and New Product Development. Research-Technology Management. 67. 70-75. 10.1080/08956308.2024.2280485.*
06. *GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models, Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock, August 22, 2023*
07. *Generative AI could raise global GDP by 7%, Goldman Sachs, Apr 5, 2023*
08. *Agentic AI: Autonomous Generative AI agents – Deloitte India 2025*
09. *The Agentic Reality Check: Preparing for a silicon-based workforce – Deloitte Tech Trends 2026*
10. *AI Overviews Reduce Clicks by 34.5% - By Ryan Law, Xibeijia Guan*
11. *The great rebuild: Architecting an AI-native tech organization - By Lou DiLorenzo Jr., Anjali Shaikh, Michael Caplan, Erika Maguire -– Deloitte Tech Trends 2026*
12. *The Deloitte 2025 Horizon Architecture Survey*
13. *"The AI Index 2025 Annual Report," AI Index Steering Committee, Institute for Human-Centered AI, Stanford University, Stanford, CA, April 2025*
14. *The "2025 Tech Value Survey" by the Deloitte Center for Integrated Research*
15. *DevSkiller Future Skills Report 2025*
16. *GitHub Developer Survey, 2024*
17. *Stackoverflow Developer Survey, 2025*
18. *Rewriting product and engineering roles in the age of AI, Deloitte, 2025*

Connect with us

Nitin Razdan
Partner and Human Capital Leader,
Technology & Transformation
Deloitte South Asia
nitinrazdan@deloitte.com

Deepan Dasgupta
Partner, Consulting
Deloitte India
deedasgupta@deloitte.com

Contributor

Naishadh Chawla

Deloitte.