WHITE PAPER

Building the Expertise Factory

How organizations turn human expertise into scalable, governed operating capability

DAVE WATKINS

2026

The central proposition

As software becomes easier to create, advantage shifts away from owning generic applications and toward the ability to identify valuable problems, capture expertise, and convert that expertise into working capabilities.

Abstract

For most of the digital era, software was the scarce asset. It was expensive to design, slow to procure, difficult to integrate, and costly to change. Those economics shaped the modern enterprise. Organizations purchased large standardized platforms, modified their operating processes to fit those platforms, and hired consultants to bridge the distance between business intent and technical implementation.

Artificial intelligence is beginning to alter that arrangement. AI-assisted development does not make reliable software free, nor does it remove the need for engineering discipline. It does, however, reduce the effort required to create many forms of analysis, workflow, interface, and application logic. Randomized field experiments have found meaningful productivity gains in some software-development settings, while other controlled research has found that experienced developers working in mature codebases can be slowed by current tools. The apparent contradiction is revealing: the economic effect depends less on access to a model than on the problem, context, architecture, and method of use.[1][2]

As application creation becomes faster, the bottleneck moves upstream. The scarce capability becomes deciding what deserves to be built, understanding how work is actually performed, capturing the judgment of capable people, connecting trustworthy information, designing appropriate human oversight, and measuring whether the resulting system improves an outcome.

This paper introduces the Expertise Factory: a repeatable organizational system for turning human expertise into scalable, governed operating capability. In this model, organizations create purpose-specific applications around consequential decisions, workflows, customer situations, temporary initiatives, and sources of specialized knowledge. Some applications become durable infrastructure; others are modified, combined, or retired as the need changes. The unit of enterprise technology moves from the software category toward the business problem.

The shift also changes consulting. Advice that once ended in a presentation can increasingly be encoded into a living capability: a decision-support application, a managed research feed, a guided workflow, or a monitored agentic process. The consulting firm becomes part strategist, part knowledge engineer, part application studio, and part operating partner. Its role is to help the client build and operate an Expertise Factory. The core product is not a report or a license. It is the client's improved ability to act.

The future belongs to organizations that can turn what they know into what they can repeatedly do.

Executive Summary

Enterprise technology is approaching an inversion. In the traditional model, organizations selected a durable software product and adapted their work to its structure. In the emerging model, they will increasingly define a valuable outcome, capture the expertise and context surrounding that outcome, and create a focused application around it. This inversion does not replace systems of record. It creates a more adaptive layer above them.

Three forces drive the change. First, natural-language interfaces and AI-assisted development reduce the distance between an operational idea and a working prototype. Second, modern models can interpret unstructured information that historically remained outside formal software systems. Third, the same systems

can increasingly coordinate tools and actions, allowing an application to participate in work rather than merely display information.

The result will not be an effortless world of automatically generated software. Production systems will still require security, evaluation, integration, maintenance, and accountable ownership. Indeed, the easier it becomes to generate code and interfaces, the more important those disciplines become. The significant change is that a larger number of smaller, more specific business needs can justify application development.

This expands the addressable territory of consulting. Every organization contains valuable but underused expertise: the questions an experienced manager asks, the signals a senior salesperson notices, the exceptions a compliance specialist recognizes, and the context an executive uses to distinguish signal from noise. Historically, that expertise was difficult to formalize and expensive to distribute. Expertise applications provide a mechanism for making it more accessible without pretending that judgment can be fully automated.

The paper develops five implications:

- Software becomes less scarce, but dependable organizational understanding becomes more scarce.
- Purpose-specific applications become practical for narrower, shorter-lived, and more contextual problems.
- Expertise becomes infrastructure when human judgment, evidence, workflow, and feedback are combined in a governed operating system.
- Consulting moves from episodic advice toward embedded and continuously improving capabilities.
- The enterprise must govern an evolving application portfolio without allowing speed to become sprawl.

The strategic question is therefore not simply how much AI an enterprise adopts. It is whether the enterprise can systematically turn human expertise into secure, measurable, and continuously improving operating capability. That is the purpose of the Expertise Factory.

The Expertise Factory

The Expertise Factory is the operating model that moves an important opportunity through five connected stages: discover the valuable problem, capture human expertise and organizational context, design the application and the human role, deploy it with appropriate controls, and improve or retire it based on measured outcomes.

This distinction matters commercially. “AI adoption” is broad and easily commoditized. The Expertise Factory is a concrete executive agenda, a transformation program, and a continuing management capability. It connects strategy, consulting, application development, governance, and operational improvement under one accountable model.

The keynote identifies the opportunity. The engagement builds the Expertise Factory.

1. When Software Stops Being Scarce

The modern enterprise was built around the assumption that software is expensive. A new application required specialized labor, a defined budget, an implementation calendar, and a long enough useful life to justify the effort. This favored broad platforms over narrow tools and durable categories over temporary needs. A payroll system, customer relationship platform, or enterprise resource planning suite could justify large investments because the underlying functions were stable and shared across organizations.

That logic produced enormous benefits. Standard platforms gave companies reliable transaction processing, common controls, and economies of scale. But the same economics also forced countless organizations to compress distinctive work into standardized fields and workflows. The software could not economically adapt to every business. The business therefore adapted to the software.

Consulting grew around this gap. Strategy consultants defined what the organization should do. Systems integrators translated that intent into platform configuration. Change-management teams trained employees to operate within the resulting process. The chain was logical because each software decision was expensive, slow, and consequential.

AI changes the cost structure unevenly rather than universally. In field experiments covering 4,867 developers at Microsoft, Accenture, and a Fortune 100 company, access to an AI coding assistant was associated with a 26.08 percent increase in completed tasks.[1] Yet a separate randomized study of experienced open-source developers working in mature repositories found that early-2025 AI tools increased completion time by 19 percent.[2] These findings should not be forced into a single headline. They demonstrate that code generation is not the same as software creation and that productivity depends on context, task type, codebase familiarity, verification burden, and the user's ability to direct and evaluate the system.

> **The important change is not that software becomes free. It is that many more problems become economically eligible for software.**

A working prototype can now be created before a requirements document is complete. A functional user interface can be assembled while business users are still refining the workflow. A narrow research tool, intake system, advisory application, or monitoring agent can be tested without first winning approval for a multiyear platform program. This changes the option value of experimentation. An organization can learn by creating, observing, and revising rather than attempting to specify the complete future state in advance.

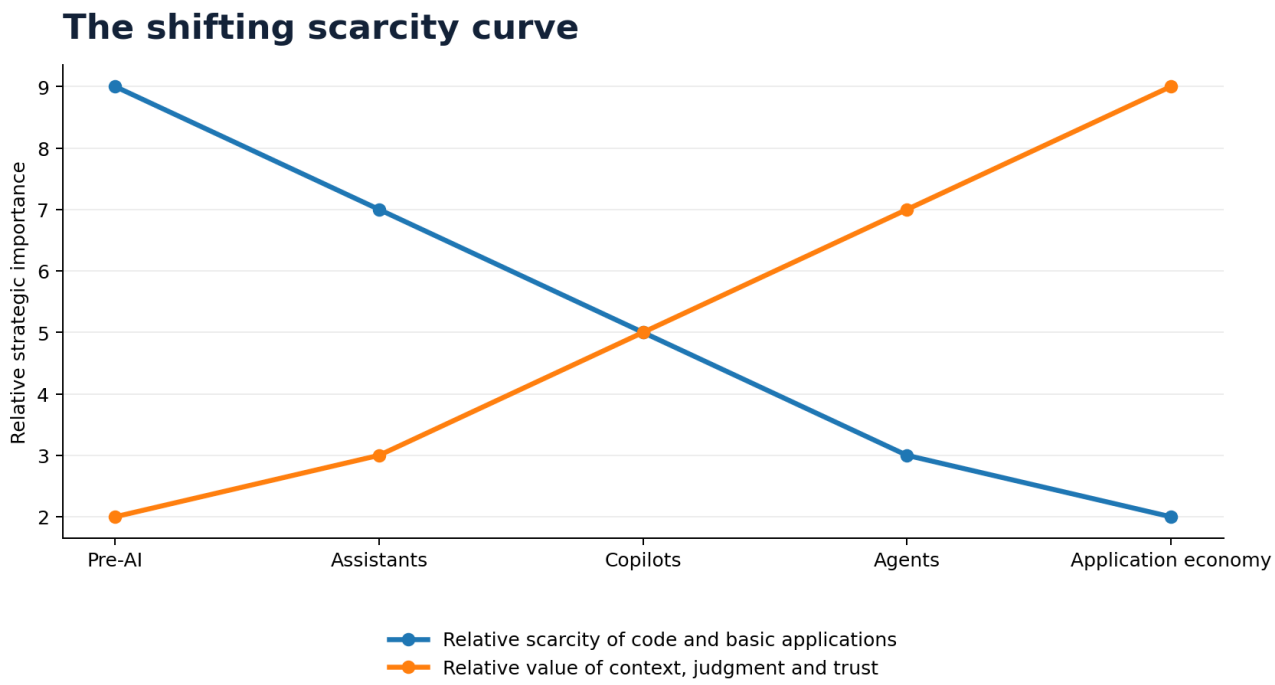

Figure 1. As application creation becomes easier, strategic value shifts toward organizational context, judgment, trust, and integration.

The shift is already visible in adoption patterns. Federal Reserve analysis of multiple U.S. surveys estimated that approximately 18 percent of firms had adopted AI by the end of 2025, while work-related generative AI use among individuals had reached roughly 41 percent by November 2025.[3] The difference between firm-level and worker-level measures suggests that adoption often begins inside the work before it appears as a formal enterprise program. Employees discover useful applications first; governance and operating models follow later.

This pattern matters because it reverses the traditional diffusion path. Enterprise software once entered through procurement and descended into the organization. AI often enters through individual experimentation and rises into formal operating systems. That creates both opportunity and disorder. The organization gains a distributed source of invention, but it also risks fragmented data practices, inconsistent quality, duplicated applications, and hidden dependencies.

The new bottleneck

When prototypes are plentiful, selection becomes strategic. The enterprise must distinguish between an interesting demonstration and a capability worth operating.

A running scenario: Northstar Components

Consider Northstar Components, a fictional 3,000-person industrial manufacturer. Northstar has competent technology teams, established enterprise systems, and a growing backlog of requests from sales, operations, compliance, and human resources. Its most experienced employees understand the company's customers, equipment, exceptions, and history. Much of that understanding is undocumented. Several senior specialists are nearing retirement, managers spend hours assembling information before routine decisions, and business units complain that enterprise projects take too long to approve.

Under the traditional model, Northstar would prioritize a small number of large systems. Smaller problems would remain in spreadsheets, email, and individual memory. Under the emerging model, Northstar can create focused applications around selected outcomes: preparing complex quotations, interpreting policy exceptions, monitoring regulatory changes, supporting frontline managers, and preserving the reasoning of retiring experts. The central challenge is no longer whether such applications can be prototyped. It is whether Northstar can determine which ones deserve to exist and operate them responsibly.

2. The Enterprise Technology Inversion

The most consequential effect of cheaper application creation is a change in sequence. In the old order, the category of software came first. The enterprise selected a customer platform, analytics platform, workflow platform, or knowledge platform and then searched for the business processes that could be fitted into it. In the new order, the enterprise begins with a valuable outcome and assembles the minimum capability required to improve it.

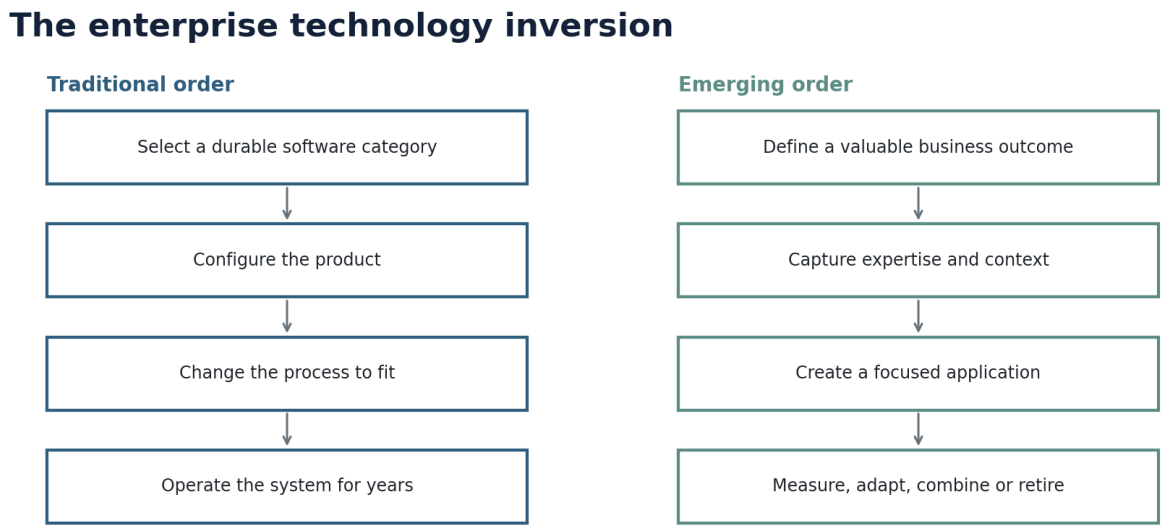

Figure 2. Enterprise technology shifts from selecting durable categories to creating, measuring, and adapting applications around specific outcomes.

This inversion changes what counts as an application. A system no longer needs to serve an entire department or remain in place for a decade. It may support one high-value decision, one customer segment, one regulatory transition, or one temporary initiative. A useful application can be short-lived without being a failure. Its lifespan can match the lifespan of the problem.

Imagine that Northstar enters a market requiring new environmental disclosures. Under the old model, the company might wait for an enterprise vendor to add the relevant capability or launch a large compliance-system project. Under the new model, it can build a focused application that assembles internal operating data, monitors external requirements, drafts evidence packages, and routes exceptions to specialists. When the regulatory transition stabilizes, the application can be simplified, absorbed into a broader system, or retired.

This is application disposability, but the term should not imply carelessness. A disposable application still requires appropriate controls while it operates. The idea is economic rather than aesthetic: not every useful capability should be forced into a permanent software category. Temporary scaffolding can be valuable precisely because it is built for a changing structure.

The inversion also changes procurement. A conventional request for proposal begins with a defined product category. An outcome-centered process begins with a business condition: reduce quotation cycle time, improve policy consistency, detect project risk earlier, or preserve expertise before retirement. Vendors and

internal teams are then evaluated on their ability to create and operate the capability, not merely on the breadth of a product checklist.

Budgeting must change as well. Large software purchases are capital-like commitments that encourage organizations to defend the system after the original rationale has weakened. Smaller applications can be funded as a portfolio of options. The enterprise can expand those that demonstrate value and close those that do not. This resembles disciplined experimentation more than conventional application ownership.

The danger is obvious: when creation becomes easier, application sprawl can grow faster than value. Gartner has already highlighted the need for adaptive governance that preserves agility without allowing SaaS and application proliferation to increase risk.[4] The response cannot be to restore the old bottleneck by requiring every experiment to pass through a multiyear approval process. The organization needs portfolio governance proportionate to the application's authority, data sensitivity, user population, and reversibility.

A portfolio, not a pile

The application-defined enterprise requires explicit ownership, lifecycle status, evaluation criteria, data permissions, and retirement rules. Speed without portfolio discipline becomes technical and organizational debt.

3. Expertise Becomes Infrastructure

Organizations often describe themselves as knowledge businesses, yet much of their most valuable knowledge is not stored in a usable form. Documents contain policies and facts. Databases contain records. Neither necessarily captures how an experienced person interprets an ambiguous situation, weighs conflicting evidence, recognizes an exception, or decides when to escalate.

This distinction separates information from expertise. Information describes the world. Expertise shapes attention and action within it. It identifies which facts matter, which sources deserve trust, which pattern is familiar, and which apparently routine case carries hidden risk.

Expertise is therefore not a document to be uploaded. It is a relationship among questions, evidence, context, judgment, action, and feedback. An application becomes an expertise application only when it reflects this complete relationship.

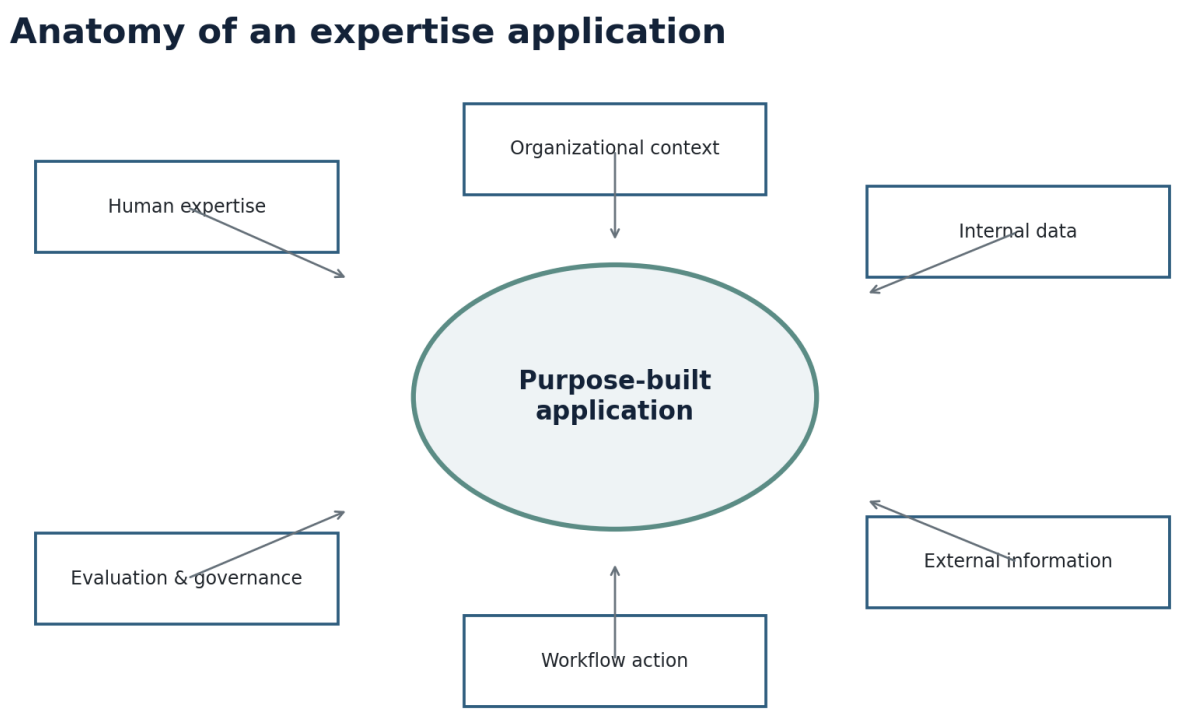

Figure 3. Expertise applications combine human judgment, organizational context, trustworthy information, workflow action, and continuous evaluation.

At Northstar, quotation specialists do more than calculate price. They know which specifications are likely to change, which customer requests imply unusual warranty exposure, which plant can actually meet the tolerance, and when a low-margin quote may still be strategically important. A generic language model may summarize the request. An expertise application must incorporate customer history, plant capabilities, pricing rules, prior exceptions, and the specialists' own diagnostic questions.

The application's first role may be modest. It can assemble evidence, identify missing information, and prepare a structured briefing. As confidence grows, it may draft a quote, recommend an exception path, or coordinate approvals. The human remains accountable, but the expert's scarce attention is redirected toward unusual cases rather than repeated information gathering.

This model also changes succession planning. Traditional knowledge-transfer programs ask experienced employees to document procedures or train successors. Those activities remain useful, but they often capture what experts say they do rather than the judgments revealed in real cases. Expertise applications can be developed through interviews, observation, historical-case analysis, and iterative review. The system becomes a structured record of how the organization reasons, not merely what it knows.

The objective is not to freeze one person's judgment into software. Expertise can be wrong, outdated, or overly dependent on local habit. The application should expose expert assumptions to comparison, measurement, and revision. In this sense, operationalizing expertise can improve the expertise itself. Practices that once survived through authority or repetition can be examined against outcomes.

Trust depends on attribution. A high-value application should distinguish original evidence, internal records, expert interpretation, and generated analysis. Users need to know whether a recommendation is grounded in a

policy, a historical precedent, a current data point, or the model's synthesis. The application should make it easier to challenge a conclusion, not merely easier to accept one.

Expertise becomes infrastructure when judgment is made accessible, evidence is traceable, and learning returns to the system.

4. Consulting Moves From Advice Into Operations

Artificial intelligence will automate parts of consulting. Research, document review, first-draft analysis, presentation production, and routine benchmarking can already be accelerated. The superficial conclusion is that consulting becomes less valuable. The deeper conclusion is that the deliverable must change.

A recommendation has always been an intermediate product. The client ultimately wants a different result: faster decisions, better performance, lower risk, greater capacity, or a new strategic position. Historically, consultants stopped at the recommendation because building and operating the corresponding system required a separate organization. When applications become easier to create, that boundary weakens.

The consulting firm of the next era can diagnose the problem, redesign the work, capture expert judgment, build the supporting application, and remain involved as the capability evolves. It combines several disciplines that clients previously purchased separately: strategy, domain advisory, knowledge engineering, application development, data integration, evaluation, and managed operations.

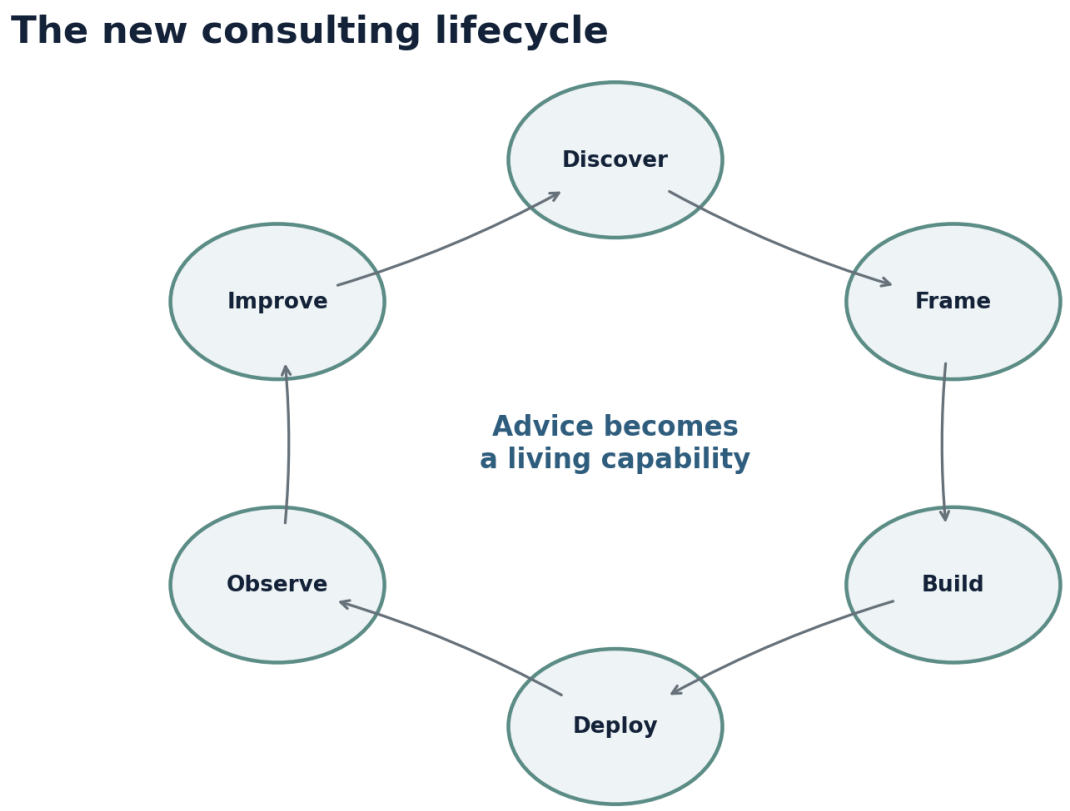

Figure 4. The consulting engagement becomes a continuous loop in which advice is embedded, observed, and improved through use.

This changes the economics of the relationship. Traditional consulting primarily prices labor and reputation. Software primarily prices access to a product. The emerging model can price a combination of advisory access, application creation, managed information, operational support, and measurable outcomes. The client does not simply receive hours or licenses. It receives an improving capability.

At Northstar, a conventional workforce engagement might deliver a manager framework, training materials, and a set of recommended interventions. An embedded-capability engagement would also create a manager preparation application that assembles relevant context, prompts the manager with appropriate questions, cites company policy, records follow-up commitments, and improves based on reviewed outcomes. The consultant's methodology enters the workflow while human responsibility remains explicit.

The same model applies to research. A consulting team may conduct a market study and present its conclusions once. A managed intelligence service can preserve the analytical framework, monitor selected sources, connect new events to the client's strategic concerns, and involve human experts when conditions exceed the application's confidence or authority. The research relationship becomes continuous without requiring a full consulting team to repeat the same baseline work each month.

This does not mean every consultant should become a software company. The highest-value role remains framing the real problem, understanding context, challenging assumptions, aligning stakeholders, and designing accountability. But the firm must be able to carry insight farther into operation, either directly or through an integrated delivery network.

The new provider resembles a studio more than a conventional pyramid. Small multidisciplinary teams work directly with client experts. They create narrow applications, test them in real settings, and expand only after evidence appears. Senior judgment cannot be spread thinly across many projects because the quality of problem selection and operating design determines the value of everything that follows.

The commercial opportunity

The strongest advisory position will belong to firms that can combine independent judgment with the ability to leave behind a functioning, governed, and continuously improving capability.

From keynote to operating model

The public idea and the client work should reinforce one another. A keynote can explain why the economics of software are shifting and why human expertise is becoming the scarce asset. A workshop can identify where expertise is trapped. An assessment can establish readiness and priorities. A transformation engagement can then build the first applications, governance practices, and operating cadence. Managed services can keep the portfolio current as information, models, and business conditions change.

This creates a natural commercial ladder: Expertise Factory keynote, executive workshop, readiness assessment, pilot application, portfolio transformation, and continuous expertise operations. Each step produces a useful outcome while creating evidence for the next.

5. The Expertise Factory

The application-defined enterprise requires more than a collection of AI projects. It requires an operating model for repeatedly converting human expertise into capability. This paper calls that operating model the Expertise Factory. Like a factory, it combines a dependable foundation, a defined production method, quality controls, skilled judgment, and feedback from actual use. Its output is not standardized physical goods, but a governed portfolio of purpose-specific capabilities that embody how the organization understands problems and takes action.

The Expertise Factory does not eliminate enterprise architecture. It makes architecture more important. The

organization needs stable foundations below and adaptable capabilities above. Systems of record continue to manage authoritative transactions, identities, and permissions. An intelligence and governance layer retrieves information, coordinates models and tools, evaluates outputs, and enforces policy. Purpose-specific applications sit at the top, changing as business needs change.

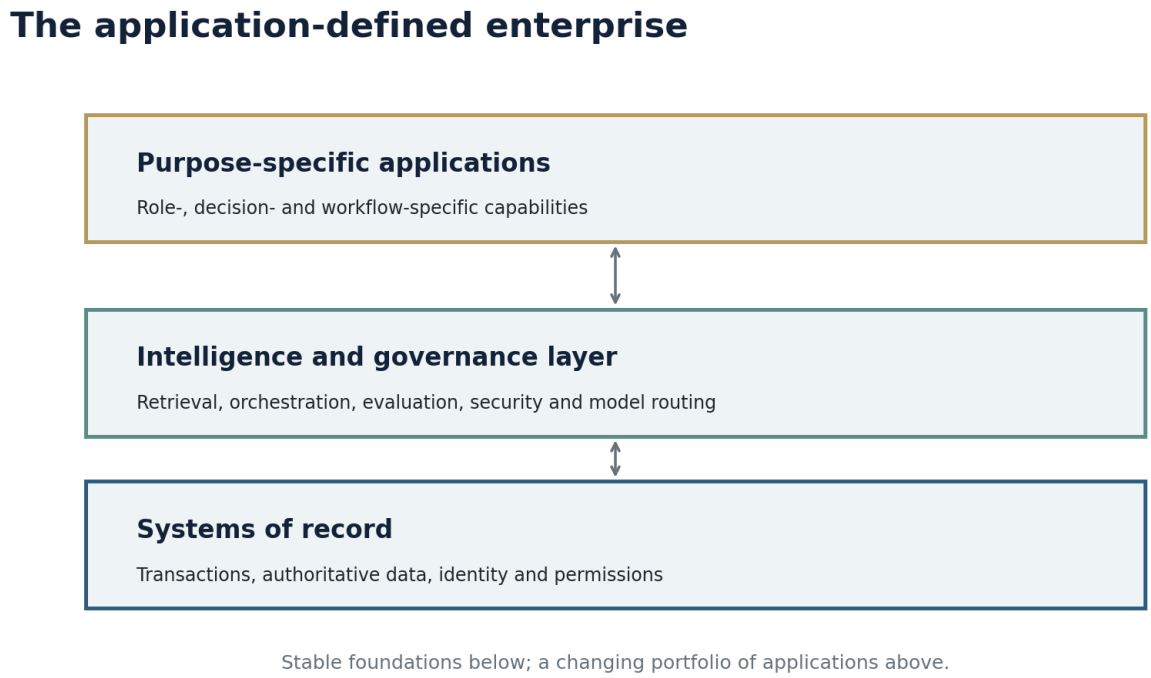

Figure 5. Stable systems of record support a governed intelligence layer and an evolving portfolio of purpose-specific applications.

This structure clarifies the roles of business and technology teams. Business leaders own outcomes, expert definitions, and adoption. Technology teams own secure foundations, shared integration patterns, observability, and reliability. Risk, legal, security, and human resources establish constraints appropriate to the use case. No single function can own the complete Expertise Factory.

Governance should follow authority. An application that summarizes public information for a human analyst should not face the same controls as an agent that changes a customer record, commits funds, or influences an employment decision. The enterprise needs a risk classification that considers data sensitivity, action authority, reversibility, affected population, and the cost of error.

Human oversight must be designed rather than assumed. Requiring a person to click “approve” does not create meaningful review if the person lacks time, context, or incentive to challenge the output. Effective human-in-the-loop design specifies what evidence the reviewer sees, which exceptions require attention, how uncertainty is communicated, and how disagreements are recorded.

Evaluation must also move beyond demonstration accuracy. A useful application can fail because users do not trust it, because it arrives too late in the workflow, because integration costs exceed the recovered value, or because it improves local efficiency while creating downstream work. The correct unit of measurement is the complete business outcome, including model cost, human review, exception handling, maintenance, and risk.

Current enterprise adoption data show rapid expansion in use, but use is not the same as institutional capability. OpenAI's 2025 enterprise report, based on deidentified usage and a survey of 9,000 workers across nearly 100 enterprises, described sharply increasing engagement and reported that workers were using AI across a broader range of tasks.[5] The strategic challenge is to convert that distributed experimentation into governed, repeatable operating advantage.

Northstar therefore creates an application registry. Every production application has an owner, purpose, risk tier, approved data sources, evaluation method, cost measure, review cadence, and retirement condition. Lightweight experiments can proceed quickly in controlled environments. Applications gain authority only as evidence and controls mature. The objective is not centralized control of every idea. It is visible and proportionate stewardship of operating capabilities.

The Expertise Factory in Human Systems

One of the most consequential applications of the Expertise Factory is the management of human capability. Traditional human capital management systems were designed primarily to administer employment transactions and preserve workforce records. AI creates the possibility of a different layer: systems that interpret the relationship among people, work, context, management action, and outcomes over time.

The progression is not from HCM directly to autonomous AI. It moves through distinct stages. Human Capital Management records and administers the workforce. AI-enabled HCM automates and improves familiar processes. Human Capability Intelligence interprets changing workforce conditions and makes uncertainty visible. Human Capability Management uses that intelligence to develop, deploy, and support capability dynamically. The Human Capability Factory turns those insights and management methods into repeatable, governed interventions and applications.

Stage	Primary purpose	Governing question
Human Capital Management	Administer records and transactions	Who is employed and what has occurred?
AI-enabled HCM	Automate and improve existing processes	How can the current process be faster or easier?
Human Capability Intelligence	Interpret people, work, context, and change	What is happening, why, and what remains uncertain?
Human Capability Management	Develop and deploy capability dynamically	What capability is needed and how should it be supported?
Human Capability Factory	Turn intelligence and expertise into repeatable action	How does the organization learn and improve through use?

Figure 6. The workforce progression moves from administration to interpretation, dynamic capability management, and repeatable operating action.

This progression changes the governing question. Traditional HCM asks who is employed, what transactions have occurred, and whether required processes were completed. Human Capability Intelligence asks what is happening, what changed, what context matters, what remains uncertain, and which action is most likely to improve the outcome. It therefore requires longitudinal context, attribution, explicit limits on inference, and meaningful human accountability.

The Human Capability Factory is not a separate enterprise architecture. It is the workforce implementation of the broader Expertise Factory. It combines workforce signals, organizational context, management expertise, trusted policy and evidence, guided workflow, and measured feedback. Its purpose is not to manufacture people. It is to make the organization systematically better at developing, deploying, supporting, and learning from human capability.

The first applications should focus on high-value, reviewable work: manager preparation, employee support, organizational listening, succession risk, workforce planning, role readiness, and the interpretation of longitudinal signals. Applications that materially influence employment, compensation, advancement, discipline, or termination require stronger evidence, rights of correction and appeal, human ownership, and explicit restrictions on machine authority. A companion paper, From Human Capital Management to Human Capability Intelligence, develops this workforce progression in detail.

The strongest objections

A serious thesis must acknowledge where it can fail. The Expertise Factory is not appropriate in every function, and it can become an expensive application mill if selection, governance, and ownership are weak. Several constraints could slow or distort the model.

First, code generation is not production engineering. Mature systems contain hidden dependencies, quality requirements, security constraints, and accumulated context. MIT researchers have emphasized that autonomous software engineering requires progress across many tasks beyond code generation, including system understanding, testing, debugging, and maintenance.[6] The falling cost of prototypes does not eliminate the cost of dependable operation.

Second, tacit expertise cannot always be extracted cleanly. Experts may disagree, rationalize decisions after the fact, or rely on signals that are difficult to observe. Some judgment should remain embodied in human practice rather than formalized prematurely.

Third, more applications can mean more risk. A proliferation of narrow tools can fragment data, confuse users, and create unowned dependencies. The thesis only holds if lifecycle governance matures alongside creation.

Fourth, continuous advisory relationships can become expensive. A managed application that requires constant expert intervention may simply recreate consulting labor behind a new interface. Providers and clients must distinguish scalable expertise from permanently customized service.

Finally, incumbents will adapt. Large software platforms will absorb many common application patterns and provide safer integrated alternatives. The opportunity for purpose-specific applications will concentrate where the organization’s context, workflow, or expertise is genuinely distinctive.

These objections refine rather than invalidate the thesis. The relevant shift is not from professional engineering to effortless generation, or from platforms to infinite bespoke software. It is from a world in which only large standardized needs justify applications to one in which many more contextual needs can be addressed—provided the enterprise applies judgment and discipline.

6. Strategic Implications

The Expertise Factory will affect different leaders in different ways, but the central strategic response is the same: build the organizational capacity to turn a valuable problem and the human expertise surrounding it into a governed capability repeatedly.

For chief executives, the issue is operating design. AI should not be delegated solely as a technology program. The chief executive must decide where faster learning, better judgment, or greater expert capacity can change the company's position. The highest-value opportunities may cross departmental boundaries and require changes in decision rights, incentives, or management practice.

For chief information officers, the challenge is to enable controlled adaptability. The technology organization must provide shared identity, connectors, model access, evaluation, logging, and cost visibility without becoming a queue that suppresses experimentation. The goal is a paved road: business teams can move quickly because safe patterns already exist.

For functional leaders, the task is to identify valuable expertise. They should look for recurring decisions, repeated information assembly, expert bottlenecks, inconsistent handling of exceptions, and important work that remains trapped in email or memory. The first application should be narrow enough to measure and important enough to matter.

For consulting firms, the strategic decision is whether to remain upstream of implementation or move into the operating layer. Firms that do move will need product discipline, technical capability, and recurring-service economics. They will also need to protect independence; the obligation to operate an application should not prevent the advisor from recommending its retirement.

For investors, the category suggests a class of firms between software and services. Their economics will depend on how much of each engagement can become reusable without stripping away the client-specific expertise that creates value. The strongest models may combine proprietary methods, shared infrastructure, specialized data, and senior advisory relationships.

For boards, the central governance question is not whether the enterprise uses AI. It is whether management can identify, measure, and control the growing set of applications that influence decisions and actions. Boards should ask how authority is assigned, how outcomes are evaluated, how data use is governed, and how the organization avoids both reckless autonomy and paralyzing caution.

A practical starting sequence

Select one important outcome. Observe the real work. Identify the strongest expertise. Map the required evidence. Design the human role. Build the smallest useful application. Measure the complete outcome. Expand only through evidence.

A near-term agenda

Organizations do not need to predict the final architecture of artificial intelligence. They need a repeatable method for learning. Over the next twelve months, an enterprise can establish the foundation by choosing a small number of high-value use cases, creating a proportionate governance model, building shared technical patterns, and measuring outcomes rather than activity.

Northstar begins with three applications: quotation preparation, regulatory monitoring, and manager preparation. Each has a named business owner and a defined baseline. The company does not begin with autonomy. It begins with information assembly and advice, where errors are visible and reversible. As evidence accumulates, selected actions can be delegated under explicit limits.

The company also identifies what not to build. Generic drafting and summarization are provided through enterprise tools rather than duplicated. Low-value personal automations remain user-controlled and do not

enter the production registry. Applications involving employment decisions, safety, or binding financial commitments receive additional review. This negative discipline is as important as invention.

Conclusion: Building the Expertise Factory

The first era of enterprise computing was organized around systems. The software-as-a-service era was organized around standardized products. The emerging era will increasingly be organized around problems, human expertise, and outcomes. The organizations that benefit will be those that build an Expertise Factory: a repeatable system for discovering valuable knowledge, translating it into useful capabilities, governing those capabilities, and improving them through evidence.

Applications will not disappear as durable assets, and large platforms will remain foundational. But a growing share of enterprise differentiation will come from purpose-specific capabilities created around the organization's own decisions, history, language, customers, employees, and methods of work.

This changes where value resides. Code and model access remain necessary, but they become less differentiating. The scarce assets are the ability to recognize a problem worth solving, discover how the organization truly works, capture the judgment of capable people, connect reliable evidence, design accountable action, and improve the capability through use.

It also changes consulting. The report and presentation remain useful, but they are no longer the natural endpoint. Advice can enter the workflow, remain connected to changing information, and improve through measured outcomes. The advisor becomes accountable not only for explaining what should happen, but for helping the client build the Expertise Factory through which it repeatedly can.

The organizations that succeed will not be those that produce the largest number of applications. They will be those that develop judgment about which applications deserve to exist, discipline in how they are operated, and speed in adapting them as conditions change.

Advantage will belong not to the organization that owns the most software, but to the one with the strongest Expertise Factory: the ability to turn human expertise into scalable, governed operating capability.

About the Author

Dave Watkins is an entrepreneur and technology executive whose work focuses on artificial intelligence, organizational knowledge, workforce intelligence, and purpose-built applications. He developed the Expertise Factory framework to explain how changes in the economics of software creation alter the relationship among consulting, enterprise technology, human expertise, and operational performance. His work explores how organizations can turn what their people know into what the enterprise can repeatedly do.

Notes and Sources

1. Zheyuan (Kevin) Cui et al., "The Effects of Generative AI on High-Skilled Work: Evidence from Three Field Experiments with Software Developers," Management Science, published online February 27, 2026. The combined experiments included 4,867 developers and found a 26.08 percent increase in completed tasks for developers using an AI coding assistant.

2. Joel Becker et al., “Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity,” 2025. In a randomized controlled trial involving 16 experienced developers and 246 tasks, access to AI tools increased completion time by 19 percent.

3. Jeffrey S. Allen, “Monitoring AI Adoption in the U.S. Economy,” Board of Governors of the Federal Reserve System, April 3, 2026.

4. Gartner, “How EA and Application Teams Collaborate on SaaS Sprawl,” August 22, 2025.

5. OpenAI, “The State of Enterprise AI 2025,” based on deidentified enterprise usage data and a survey of 9,000 workers across nearly 100 enterprises.

6. MIT CSAIL, “Can AI Really Code? Study Maps the Roadblocks to Autonomous Software Engineering,” MIT News, July 16, 2025.