HAVISH M
CONSULTING

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Generative AI Playbook for Leaders



HMC ONE

AI STRATEGY. BUILT TO SCALE.

Generative AI Playbook for Leaders

Havish Madhvapaty and Neha Jha

havishmconsulting.com/downloads

DOWNLOAD ALL OUR

FILES + HANDBOOKS + CONTENT



About Havish M Consulting

Havish M Consulting (HMC) is an award-winning analytics and AI advisory firm and a proud member of NASSCOM, recognized among Siliconindia's Top 10 Data Analytics Companies of 2025. Since its founding, HMC has delivered over 2,500 corporate workshops — including sessions for Boards of Directors and C-suite leadership — across 45+ organizations spanning Energy, CPG, BFSI, Manufacturing, Retail, and Consulting, in 10+ countries across India, Middle East, UK and North America.

At the helm is Havish Madhvapaty, Founder of HMC and a leadership coach to Fortune 500 C-suites and Boards of Directors. He serves as visiting faculty at IIM Kozhikode, SRCC, and Masters' Union, and has taught internationally at NUS Business School and S P Jain School of Global Management. His work has been featured in Businessworld, The Economic Times, Entrepreneur, and Financial Express, and he has authored professional certification content with the Economic Times Masterclass series and for National Stock Exchange NCFM Pro Series. Havish was listed on BusinessWorld 40u40 Education, Analytics Insights 40u40 Analytics, Entrepreneur India and Indian Express Educator of the Year and has received Indian Achievers Award. He has also been nominated by Microsoft as an MVP.

Working alongside him is Neha Jha, Senior AI Consultant, who leads the day-to-day management of GenAI adoption journeys for HMC's enterprise clients. Neha is the connective tissue between HMC's frameworks and their real-world implementation — coordinating readiness surveys, cohort trainings, and rollout logistics across engagements with Amber Enterprises, Bharat Petroleum, RR Kabel, JK Cement and Landmark Group, among others. Her work ensures that HMC's adoption programs don't just launch well but sustain momentum long after the workshop ends.

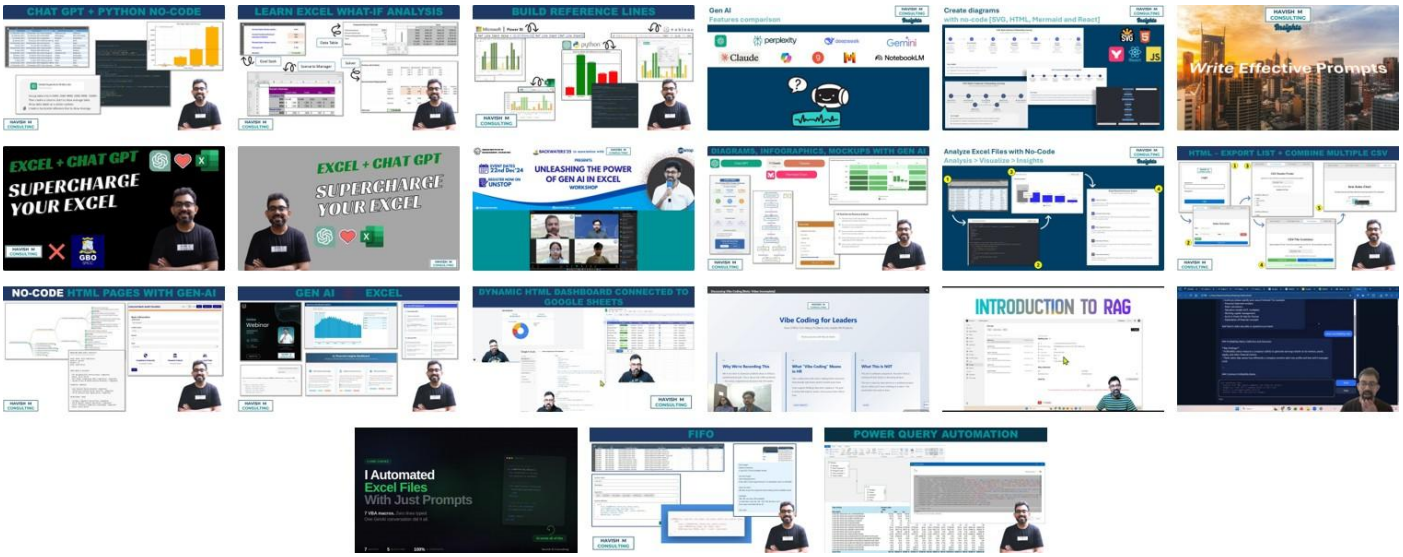
Backing this team is a strong advisory board, including Sanjay Kumar Suri (former CISO, Bharat Petroleum) and Jyotirmoy Bose (former CEO, Bennett Advisory) — bringing decades of enterprise security, governance, and strategic leadership experience to HMC's client engagements.

HMC's Generative AI adoption work sits at the heart of this book. The firm has created Generative AI labs for 8+ large organizations (USD 1 Billion +) guiding leadership teams from foundational awareness through functional training to enterprise-wide safety frameworks with measurable ROI.

YouTube

1600+ Videos covering Generative AI, Prompt Engineering, Agents, Excel Automation / Analytics / Dashboarding, Data Storytelling, No Code ML and more.

https://www.youtube.com/c/havishmadhvapaty



Key Clients


Bharat
Petroleum


amber


Welspun


LANDMARK
GROUP


JKCement


KĀBEL


pnb Housing
Finance Limited


DS GROUP

Client Testimonials



strategy&

Kushal Sinha

Partner @ Strategy&

“Havish adapts to audience requirements and enables the team to pick things up and run themselves. He covers advanced analytics techniques focusing on problem solving techniques.”



Shell

Yogesh Yadav

CFO at Shell Group Entity @ Sprng Energy

“Automation projects done helped us **streamline and accelerate** our workflows and enabled teams to complete **complex tasks faster**.”



IndusInd Bank

Siddharth Sinha

SVP and Head Analytics @ IndusInd Bank

“Wonderful session on GenAI. Really enjoyed”.



Rajeev Jain

SVP Marketing @ DS Group

“I have attended few training sessions from Havish on AI and enjoyed sessions thoroughly. I found him passionate about AI, having thorough knowledge and very sincere in approach. He understands need of trainees and customises sessions accordingly. I along with my team members learned a lot from him about applications of various AI tools.”



Rajwinder Singh

Head Equipment and Procurement LPG @ Bharat Petroleum

“Havish is an excellent trainer in Advanced Excel and Power BI. He is very competent and has the ability to respond quickly to any kind of queries. Had a great learning experience.”

GenAI Journey Snapshot across Clients

Client-anonymised view

Awareness = literacy and leadership buy-in at engagement start

Adoption = depth of embedded usage today

ROI = realised or tracked value

Industry	Sector	Awareness	Adoption	ROI
Wires & Cables	Electrical Equipment / Manufacturing	High at promoter and CHRO level; CIO governance concerns surfaced early	Growing — sanctioned enablement platform live; two flagship workstreams in sales planning and SCM	Medium-High — forecasting cycle compression, freight efficiency gains, reduced shadow-IT usage
Home Textiles, Pipes & Infrastructure	Textiles / Diversified Manufacturing	Strong promoter sponsorship across two distinct business lines	Function-deep rather than org-wide — planning and quality teams first	Medium-High — loom scheduling from days to hours; earlier defect capture; white-space led revenue pitches
Cement	Building Materials / Manufacturing	Concentrated in two business-unit heads; org-wide literacy still early	Executive-only by design — personal fluency before rollout	Medium — leadership self-sufficiency on analysis drafts; internal champions seeded ahead of scale
Retail (Fashion, Home, F&B)	Retail & Consumer	High at Gulf leadership level; category teams following	Mixed — leadership coaching alongside live pricing analytics	Medium-High — margin protection through better-timed markdowns; faster aged-stock clearance
Air Conditioners & Consumer Durables (OEM/ODM)	Consumer Durables / Manufacturing	Promoter-led from day one; baseline readiness survey segmented workforce into quadrant cohorts	Deep — enterprise licences at scale, always-on internal platform across plant, finance, commercial	High — analyst-hours reclaimed on BOM and variance reporting; board-level impact narrative funded Year 2
Steel & Power	Metals & Mining / Energy	Early-stage; discovery-led, building the	Pre-deployment — prioritised use-case pipeline, no	Early — avoided spend on low-value pilots; cross-functional

		business case before commitment	production rollout yet	consensus established
Pharmaceuticals	Healthcare / Life Sciences	Cautious but informed; governance framed the conversation before tooling	Selective — non-regulated functions only, use-case screening gate in place	Medium — faster market intelligence synthesis; reduced external research dependency
Banking	Financial Services	Function-led; strongest in Marketing, building in HR and IT	Broad but shallow — cross-functional training layer, workflow embedding next	Medium — lower agency dependency on first-draft creative; faster HR query turnaround
Trading, Retail & Real Estate	Diversified / Conglomerate	Leadership-driven; AI positioned as a strategy capability, not an IT one	Emerging — strategy and planning cell first, business units to follow	Medium — market scans in hours vs weeks; reduced first-pass consulting spend
Real Estate & Retail	Real Estate / Diversified	Engineering-function led; awareness strongest where estimation pain is felt	Targeted — design and estimation workflows in live use	Medium-High — faster bid-stage estimates; fewer revision cycles; senior engineer capacity freed
Power Transmission Infrastructure (EPC)	Power / Infrastructure	PMO-led; site teams onboarding progressively	Process-embedded — reporting and tracking layer across concurrent projects	Medium-High — multi-site status consolidation from days to hours; earlier delay flagging

Part 1 — Awareness: Understand the Landscape

Article – Creating Value

First, productivity: summarising reports, preparing presentations, drafting emails and converting complex information into decision-ready insights.

Second, knowledge: helping teams search SOPs, research documents, product information and internal knowledge more effectively.

Third, innovation: supporting idea generation, market research, product documentation and faster experimentation.

Fourth, decision-making: analysing business data, identifying patterns and developing scenarios for management review.

Let us begin with a simple question:

Which repetitive activity currently consumes your team's time but does not necessarily require their highest level of expertise?

By the end of the session / engagement / discussion, the objective is that every leader identifies at least 1-3 practical Gen AI use case that can save time, improve quality or accelerate innovation

CREATING VALUE

Gen AI creates value in four places

Four areas where your teams can save time and raise quality.

01

Productivity

Summarise reports, prepare presentations, draft emails and turn complex information into decision-ready insights.

02

Knowledge

Search SOPs, research documents, product information and internal knowledge more effectively.

03

Innovation

Support idea generation, market research, product documentation and faster experimentation.

04

Decision-making

Analyse business data, spot patterns and develop scenarios for management review.

START WITH ONE QUESTION

Which repetitive activity consumes your team's time but does not need their highest level of expertise?

The objective: every person leaves with **1–3 practical Gen AI use cases** that save time, improve quality or accelerate innovation.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Article – AI Terms

Category	Term	Definition	Relevance
Foundational	Artificial Intelligence (AI)	Computer systems performing tasks that typically require human intelligence — reasoning, perception, language, decision-making.	The umbrella term for the entire strategic conversation; sets scope for board discussions and investment cases.
Foundational	Machine Learning (ML)	A subset of AI where systems learn patterns from data rather than following explicit programmed rules.	Most enterprise "AI" initiatives are actually ML; understanding this helps set realistic expectations on data needs.
Foundational	Deep Learning	A type of ML using multi-layered neural networks, effective at complex pattern recognition (images, speech, text).	Powers most modern breakthroughs; explains why compute and data volume drive cost and capability.
Foundational	Neural Network	A model architecture loosely inspired by the brain, made of layers of interconnected nodes that process data.	Underlying structure of nearly all modern AI; useful for interpreting vendor technical claims.
Foundational	Generative AI (GenAI)	AI that creates new content — text, images, code, audio — rather than just classifying or predicting.	The category behind ChatGPT, Claude, etc.; the primary driver of current enterprise AI investment.
Foundational	Large Language Model (LLM)	A GenAI model trained on vast text data to understand and generate human language.	The engine behind most GenAI products; underpins build-vs-buy decisions for AI tools.
Foundational	Small Language Model (SLM)	A compact, lower-parameter language model optimized for efficiency, speed, and on-device or low-cost use.	Enables cheaper, faster, more private deployments for narrow tasks — relevant for cost-sensitive scale-ups.
Foundational	Algorithm	A set of rules or steps a computer follows to solve a problem or complete a task.	Basic vocabulary needed to engage credibly with technical teams and vendors.
Foundational	Model	The trained system that makes predictions or generates outputs, produced by running an algorithm on data.	The "product" of AI development; distinguishing "the model" from "the app" clarifies vendor conversations.
Foundational	Weights / Parameters	The internal numerical values a model adjusts during training to encode learned patterns.	Roughly correlates with model scale and cost; "billions of parameters" is a common vendor sizing metric.

Foundational	Inference	The process of a trained model generating an output for a new input (as opposed to training).	The ongoing, recurring cost of running AI in production — distinct from one-time training cost, matters for budgeting.
Foundational	Natural Language Processing (NLP)	The field of AI focused on enabling computers to understand, interpret, and generate human language.	The technical foundation behind chatbots, document analysis, and text-based automation tools.
Foundational	Computer Vision	The field of AI focused on enabling machines to interpret and process visual information from images or video.	Powers use cases like quality inspection, safety monitoring, and document scanning in manufacturing/retail.
Foundational	Transformer Architecture	A neural network design that processes entire sequences of data at once using "attention," enabling modern LLMs.	The technical breakthrough behind the current AI boom; useful context for why LLMs improved so rapidly since 2017.
Development	Training Data	The dataset used to teach a model patterns and relationships.	Data quality and ownership are competitive assets; "garbage in, garbage out" applies directly to ROI.
Development	Fine-Tuning	Further training a general-purpose model on a smaller, domain-specific dataset to specialize it.	Key lever for customizing AI to your industry/company without building a model from scratch — faster, cheaper than full development.
Development	Prompt Engineering	Crafting inputs (prompts) to guide an AI model toward better, more useful outputs.	Low-cost, high-leverage skill; often the fastest path to value from existing tools, worth investing in workforce training.
Development	RAG (Retrieval-Augmented Generation)	A technique where a model retrieves relevant company/external data before generating a response, improving accuracy.	Central to enterprise GenAI deployments — lets AI answer using your proprietary documents securely, without retraining.
Development	Hallucination	When an AI model generates plausible-sounding but factually incorrect or fabricated information.	The single biggest reputational and operational risk in GenAI deployment; drives the need for human-in-the-loop review.
Development	Context Window	The amount of text/information a model can "see" and consider at once when generating a response.	Determines how much company data (documents, chat history) a tool can process in one interaction — a practical capability constraint.
Development	Token	A chunk of text (roughly a word or part of a word) that models process; usage is often billed per token.	Directly tied to API and licensing costs; relevant to budgeting and vendor negotiations.
Development	Multimodal AI	Models that can process and generate multiple content types — text, images, audio, video — together.	Expands use cases beyond text (e.g., analyzing scanned documents, video, or design mockups), widening the addressable business problem set.

Development	Embedding	A numerical representation of text/data that captures meaning, allowing similarity comparisons between items.	Powers search, recommendations, and RAG systems; helps explain how AI "understands" unstructured company data.
Development	Vector Database	A specialized database that stores embeddings for fast similarity search.	Common infrastructure component in enterprise AI stacks; relevant when evaluating technical architecture proposals.
Development	Reinforcement Learning from Human Feedback (RLHF)	A training method where human ratings of model outputs are used to refine model behavior.	Explains why newer models feel more "aligned" and helpful; relevant to vendor claims about safety and quality.
Development	Zero-shot / Few-shot Learning	A model's ability to perform a task with no examples (zero-shot) or just a few examples (few-shot) provided in the prompt.	Reduces the need for costly custom training; affects how quickly a tool can be adapted to a new task.
Development	Chain-of-Thought Reasoning	A technique where a model breaks a problem into intermediate steps before producing a final answer.	Improves accuracy on complex tasks; explains why some "reasoning" models are slower but more reliable.
Development	Synthetic Data	Artificially generated data used to train or supplement AI models when real data is scarce or sensitive.	Reduces privacy risk and data-scarcity constraints in regulated or data-poor environments.
Development	Model Distillation	Compressing a large, expensive model into a smaller, cheaper one while retaining most of its capability.	A key cost-reduction strategy for deploying AI at scale without sacrificing much performance.
Deployment	AI Agent	An AI system that can autonomously plan and execute multi-step tasks, often using tools or software on a user's behalf.	The next wave beyond chatbots — agents can complete workflows (booking, research, coding), shifting AI from "assistant" to "worker."
Deployment	API (Application Programming Interface)	A technical interface allowing one software system to send requests to and receive data from another.	The standard way companies integrate AI models into existing tools/systems; central to build-vs-buy technical conversations.
Deployment	MCP (Model Context Protocol)	An emerging standard letting AI models securely connect to external tools, data sources, and apps.	Reduces integration cost and vendor lock-in; relevant when evaluating platform strategy and interoperability.
Deployment	Automation vs. Augmentation	Automation replaces a task entirely; augmentation enhances human performance on a task.	Frames the workforce conversation — most near-term enterprise AI value comes from augmentation, not replacement.
Deployment	Copilot	An AI assistant embedded within existing software (e.g., email, spreadsheets) to support a user in real time.	The dominant enterprise adoption pattern — lower change-management burden than standalone AI tools.
Deployment	Model as a Service (MaaS)	Accessing AI models via subscription/API rather than owning the underlying infrastructure.	The default enterprise consumption model; affects cost structure (opex vs. capex) and speed to deployment.

Deployment	Chatbot vs. Agentic AI	A chatbot responds conversationally; agentic AI takes independent action across multiple steps or systems.	Marketing language often blurs this distinction; important for accurately scoping vendor capability claims.
Deployment	Orchestration	Coordinating multiple AI models, tools, or agents to complete a complex, multi-part task.	Increasingly the differentiator between basic AI tools and enterprise-grade AI systems.
Deployment	Latency	The time delay between a request to an AI system and receiving its response.	Affects user experience and viability for real-time use cases like customer service or trading.
Deployment	Sandbox Environment	An isolated testing environment where AI tools can be trialed safely without affecting live systems or data.	Standard best practice for piloting new AI tools before enterprise-wide rollout.
Deployment	Integration Layer / Middleware	Software that connects AI models to existing enterprise systems (ERP, CRM, data warehouses).	Often the largest hidden cost in AI projects — the model is rarely the hard part, integration is.
Deployment	Legacy System Compatibility	The degree to which existing older IT infrastructure can support or integrate with new AI tools.	A common blocker in traditional industries (manufacturing, cement, textiles); shapes realistic implementation timelines.
Governance	AI Bias	Systematic errors in AI outputs that unfairly favor or disadvantage particular groups, often inherited from training data.	Legal, reputational, and regulatory exposure; requires explicit governance in hiring, lending, and customer-facing use cases.
Governance	Explainability / Interpretability	The degree to which humans can understand why an AI model produced a given output.	Critical in regulated industries (finance, healthcare) where decisions must be auditable and defensible.
Governance	Data Privacy	Protections governing how personal or sensitive data is collected, used, and stored by AI systems.	Governs vendor contracts, especially around whether client data is used to train third-party models.
Governance	AI Governance	The policies, roles, and controls an organization puts in place to manage AI risk and ensure responsible use.	Board-level responsibility increasingly expected by regulators and investors; foundational to scaling AI safely.
Governance	Shadow AI	Unsanctioned use of AI tools by employees outside approved, governed channels.	A near-universal, underestimated risk; drives the urgency of clear usage policy and approved-tool lists.
Governance	Guardrails	Technical and policy controls that constrain what an AI system can say or do.	The practical mechanism for translating governance policy into deployed system behavior.
Governance	Human-in-the-Loop	A design principle requiring human review or approval at key points in an AI-driven process.	The primary risk mitigation for high-stakes decisions (legal, medical, financial); a standard board expectation.
Governance	Data Residency / Sovereignty	Legal requirements dictating where data is physically stored and processed, often by country or region.	Directly relevant for Middle East/India operations with cross-border data regulations; a common vendor due-diligence item.

Governance	Model Drift	The gradual degradation of a model's accuracy over time as real-world data diverges from its training data.	Justifies ongoing monitoring budgets — AI systems are not "set and forget."
Governance	Responsible AI	An organizational commitment to fairness, transparency, safety, and accountability in AI deployment.	Increasingly a procurement and brand-trust requirement, not just an ethics exercise.
Governance	IP / Copyright Risk	Legal uncertainty around ownership of AI-generated content and use of copyrighted material in training data.	Directly affects usage policy for AI-generated marketing, code, and client deliverables.
Governance	Model Card	A standardized document describing a model's capabilities, limitations, training data, and intended use.	A useful due-diligence artifact when evaluating and comparing AI vendors.
Governance	Red Teaming	Deliberately probing an AI system for weaknesses, biases, or failure modes before deployment.	A recognized best practice for de-risking AI systems ahead of enterprise-wide rollout.
Governance	Regulatory Compliance (AI-specific)	Adherence to emerging laws governing AI use (e.g., EU AI Act, sector-specific rules).	An evolving compliance landscape that legal and risk teams must track alongside standard data regulation.
Strategy	Foundation Model	A large, general-purpose model (e.g., GPT, Claude, Gemini) trained on broad data, adaptable to many downstream tasks.	Underlies build-vs-buy strategy — few companies build these; most build on top of them.
Strategy	Open Source vs. Closed/Proprietary Models	Open models have publicly available code/weights; closed models are accessed only via a vendor's paid interface.	Shapes cost, control, and customization trade-offs — open models offer flexibility, closed models offer support and reliability.
Strategy	Compute	The processing power (typically specialized chips like GPUs) required to train and run AI models.	The primary cost driver and supply-chain bottleneck in AI; relevant to infrastructure and cloud vendor strategy.
Strategy	AI Maturity Model	A framework assessing an organization's AI capability across stages, from ad hoc experimentation to full integration.	Useful diagnostic for benchmarking your organization's AI adoption stage and setting a realistic roadmap.
Strategy	Total Cost of Ownership (TCO) for AI	The full cost of an AI initiative, including licensing, integration, change management, and ongoing monitoring — not just the tool price.	Prevents underestimating AI project budgets; licensing is often the smallest line item.
Strategy	Change Management (for AI)	The structured process of managing workforce adoption, training, and cultural shift around new AI tools.	The most common point of AI initiative failure is not the technology — it's adoption; this is where leadership attention has highest ROI.
Strategy	Digital Twin	A virtual, data-driven replica of a physical asset, process, or system used for simulation and forecasting.	Common in manufacturing/energy for predictive maintenance and scenario planning — a tangible, high-ROI AI use case.

Strategy	Center of Excellence (CoE)	A dedicated internal team responsible for driving AI strategy, standards, and capability-building across an organization.	A common structural choice for scaling AI beyond isolated pilots into enterprise-wide impact.
Strategy	Proof of Concept (PoC) vs. Production	A PoC tests feasibility on a small scale; production deployment operates at full scale with real users and stakes.	Explains why many "successful" AI pilots never scale — production requires different rigor, cost, and governance.
Strategy	AI Use Case Prioritization	The structured process of ranking potential AI applications by business value versus implementation feasibility.	Prevents scattershot experimentation; ensures AI investment targets highest-impact business problems first.
Strategy	Build vs. Buy vs. Partner	The strategic decision of whether to develop AI in-house, purchase existing tools, or partner with vendors/providers.	The core strategic decision framework for nearly every AI initiative; balances cost, control, and speed.
Strategy	Vendor Lock-in	Dependency on a single AI provider's technology, making it costly or difficult to switch providers later.	A key risk factor in long-term contract negotiations and platform architecture decisions.
Strategy	AGI (Artificial General Intelligence)	A hypothetical future AI with human-level reasoning and adaptability across any intellectual task, not yet achieved.	Distinguishes today's practical AI capability from speculative long-term scenarios — useful for grounding realistic strategy discussions.
Strategy	AI ROI Measurement	Frameworks for quantifying the business value generated by AI initiatives against their cost.	A frequent board-level question; hard to answer without pre-defined baseline metrics before deployment.
Strategy	Workforce Reskilling	Structured training programs to build employee capability in working alongside AI tools.	Determines whether AI investment translates into actual productivity gains rather than idle licenses.
Strategy	AI Adoption Curve	The pattern by which employees or organizations move from awareness to experimentation to embedded daily use.	Helps leaders set realistic timelines and identify where an organization is stalling in the adoption journey.
Strategy	Composable AI Stack	An architecture built from interchangeable AI components (models, data layers, tools) rather than one monolithic system.	Supports flexibility and reduces vendor lock-in as the AI landscape evolves quickly.
Strategy	Data Moat	A durable competitive advantage derived from proprietary data that competitors cannot easily replicate or access.	A key lens for identifying where AI can create defensible, long-term differentiation rather than easily copied features.

Article – Context Window

Every leader who's used ChatGPT or Claude for an extended session has hit the same wall — the model starts ignoring an instruction given twenty minutes ago, loses track of a detail from earlier in the document, or contradicts something it said three prompts back. The instinct is to blame the model's intelligence. The actual cause is almost always simpler: the conversation has outgrown its context window.

What the context window actually is

A context window is the total amount of text — your prompts, the model's replies, any documents or data pasted in — that the model can "see" at one time while generating a response. It isn't memory in the human sense. The model doesn't learn from the conversation or store it anywhere between messages. Each time it generates a reply, it's re-reading the entire visible conversation from scratch, token by token, within a fixed capacity. Once that capacity is full, something has to give.

Think of it less like memory and more like a desk. The model can only work with what's physically laid out in front of it. A bigger desk means more documents stay visible at once. But no desk is infinite, and when a new document comes in and the desk is full, the oldest one gets pushed off the edge — not forgotten exactly, just no longer in view.

Why this matters more than most people realize

This isn't a minor technical footnote — it's the single most common reason AI tools feel unreliable in real business use:

- A long client email thread gets pasted in for analysis, and the model's response ignores the nuance from the first few messages — because by the time it generated the reply, those tokens had scrolled out of the visible window.
- A multi-hour working session drafting a strategy document starts producing inconsistent recommendations — not because the model changed its mind, but because the instructions that shaped its early output are no longer part of what it's reading.
- A long back-and-forth about a dataset leads to the model "forgetting" a constraint stated at the very start, while still perfectly recalling something said two messages ago.

None of this is the model being unreliable in the way a person would be. It's a fixed architectural constraint, and it behaves the same way every time once you understand the rule: what's outside the window simply doesn't exist for that response.

Size varies — and size changes the calculation

Context windows differ sharply by model and vendor — some handle a few thousand words, others handle hundreds of thousands of words' worth of tokens in a single window. A larger window doesn't

eliminate the problem, but it changes what's practical. A 10-page contract fits comfortably in a large window with room to spare for discussion. A 200-page due diligence file might not, regardless of window size, once you add the back-and-forth needed to actually work through it.

This is also why the same task can work well in one tool and fail in another — it's often not a difference in reasoning quality, but a difference in how much of the input the model could actually hold onto while producing the output.

What this means in practice for how work gets structured

A few habits make the difference between fighting the context window and working with it:

- **Front-load what matters.** If something is non-negotiable — a constraint, a format requirement, a piece of data the output depends on — restate it close to where you need it used, rather than trusting it survived from ten messages earlier.
- **Chunk large documents.** Instead of pasting an entire report and asking for a full analysis in one pass, work through it in sections. The output is more reliable, and it's easier to verify against the source.
- **Start fresh deliberately.** When a conversation has run long and responses start feeling inconsistent or repetitive, starting a new session with a tight, well-structured prompt often outperforms continuing to push the old one further.
- **Don't assume persistence across sessions.** A new conversation starts with an empty window — nothing from a previous chat is visible unless it's explicitly carried over or the tool has a separate memory feature layered on top.

The underlying point for leaders

The context window isn't a flaw to work around once and forget about — it's a fixed resource, the same way compute budget or headcount is a fixed resource, and it needs to be managed with the same deliberateness. Treating it as infinite is how teams end up blaming the model for a failure that was actually a workflow design problem. Understanding it precisely — what's in view, what's fallen out of view, and when to reset rather than push on — is what separates AI use that stays reliable at scale from AI use that quietly degrades the longer a session runs.

Context Window. Use It Well.

Every AI model can only read and remember so much at once.
That limit is the context window. Everything in it shapes the answer.

The goal:
Give the model only what the task needs.

Context window, one chat — **Full = model limit**





What it is
All the text an AI can hold in one chat, counted in tokens: prompts, files and replies.



Rule of thumb
1 token ≈ ¾ of a word, so 1,000 words ≈ 1,300 tokens. Long chats lose detail.

How to optimize



One chat per task
Start a new chat for each task and summarize long threads before continuing.



Tools
Each connected tool adds its instructions and results to the window. Turn off the ones you don't need.



Uploaded Excel files
Upload only the sheets and columns the task needs. A full workbook can fill the window on its own.



Projects
Keep shared files and instructions in one project instead of re-uploading them in every chat.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – Awareness to Adoption to ROI

Every Organisation Thinks It Is One Stage Ahead of Where It Is

Ask a leadership team where they are on AI and you will almost always hear the same answer. We are past the basics. We're scaling now. We just need to show the returns.

Then you walk the floor and find managers who have never opened the tool, a pilot that has been running for eleven months, and a finance team that cannot say what any of it has saved because nobody wrote down what it cost before.

There are three stages. Awareness, Adoption, ROI. Nobody skips one. Organisations that believe they have skipped one have simply not noticed that they are still standing in it.

Stage One: Awareness

Awareness is routinely mistaken for exposure. A town hall was held. A newsletter went out. Two hundred licences were purchased. Everyone has heard of the technology, so surely the stage is complete.

Being told is not the same as knowing what to do on Monday morning.

Real awareness has three markers. People can say what the technology does and what it does not do, because a team that cannot separate a chatbot from an agent cannot scope a use case or judge a vendor. Every function can name three tasks inside their own workflow where this applies, because specificity is awareness and enthusiasm is not. And people know what they are allowed to do with company data, because where no safety position exists, the cautious stop and the reckless proceed, and the second group reaches the board first.

The work here takes four to six weeks. Run a baseline before anything goes live, one process at a time, capturing what it costs today in hours, error rate and rework. Put the leadership team in a room and make them fail with the tool rather than watch a demo. Publish a two-page safety position. Set a 90-Day North Star with one named owner and one date.

The gate: a randomly chosen mid-level manager can name one task they now do differently, and one task they have deliberately decided not to use AI for. Both halves matter. The second is what separates an aware organisation from an excited one.

Stage Two: Adoption

This is the longest stage and the one where most organisations quietly stop. It has two halves, and the gap between them is where the money goes to die.

The first half is the pilot. Pilots are easy and almost always succeed. You pick a motivated team, a clean process and a forgiving success metric. Something works. Slides are made.

The second half is scale, and it fails for reasons that have nothing to do with the technology. The process was never documented, so there is nothing to standardise. The workflow depends on one enthusiastic person who has now been promoted. Nobody has budget authority for the licences beyond the pilot. Middle managers were measured on the old outputs the entire time and correctly concluded that the new way was a personal risk with no personal upside.

Pilot purgatory is an organisational design problem wearing a technology costume. The tell is a company with fourteen successful pilots and zero changed processes.

What actually moves an organisation through this stage is unglamorous. Rewrite the standard operating procedure, not just the tool access. Change what managers are measured on, because behaviour follows the scorecard and nothing else. Name an owner for each workflow who has the authority to stop the old version, since running both in parallel forever is how adoption dies politely. Retire the previous method on a stated date.

The gate: the old way of doing the work is switched off, and the organisation has not fallen over. Until something is turned off, nothing has been adopted. You have added a tool, not changed a process.

Stage Three: ROI

ROI is treated as a result. It should be treated as a discipline, and it is a stage rather than a finish line because most organisations reach it and discover they cannot prove anything.

The failure is almost always structural, and it happened months earlier. There was no baseline. Nobody recorded what the process cost before the change, so every claim of saving becomes a negotiation between an optimistic function head and a sceptical CFO. The honest answer is that nobody knows.

The second failure is measuring activity instead of outcome. Licences issued, prompts run, hours of training delivered. These are inputs. They tell you what was spent, not what was returned. A dashboard full of usage statistics is a confession that the outcome was never defined.

The third is attribution. Something improved during a period in which many things changed. A defensible ROI claim requires a stated baseline, a comparison population or period, and an honest note on what else was happening at the time. Senior finance people respect a smaller number with a clear method far more than a large number with a vague one.

Done properly, this stage produces something more valuable than a percentage. It produces the ability to say which use cases are worth repeating and which were expensive theatre, which is the only basis on which anyone should be approving the next round of investment.

The gate: the CFO will repeat your number to someone outside the organisation. Numbers that survive only inside the room that produced them are not returns. They are morale.

The point of naming the stages

The value is not in the diagram. It is in the diagnosis, and diagnosis is uncomfortable because most organisations that describe themselves as scaling are still in Awareness, and most that describe themselves as measuring ROI never left the pilot.

Run the three gate tests before your next AI investment decision. They take an afternoon.

Whichever gate you fail first is where the work is. Everything you spend beyond that point is spent on the wrong stage.

AWARENESS TO ADOPTION TO ROI

Every organization thinks it is one stage ahead

Three stages. Nobody skips one.

01 Awareness

Exposure does not tell people what to do on Monday.

Build capability in 4–6 weeks.
Name workflow use cases, set data safety rules and capture the baseline.

THE GATE

A manager names one task they now do differently and one they deliberately avoid using AI for.

02 Adoption

The gap between a pilot and scale is organisational.

Rewrite the SOP and manager scorecards. Give the workflow an owner and retire the old method on a stated date.

THE GATE

The old way is switched off, and the organisation keeps working.

03 ROI

Returns require evidence beyond usage statistics.

Compare outcomes with the baseline and a comparison period or group. Disclose other changes that affect attribution.

THE GATE

The CFO will repeat your number outside the organisation.

The first gate you fail is where the work belongs.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – China vs US Models

The Strategic Picture

The AI model race has bifurcated into two distinct ecosystems, and enterprise leaders can no longer treat "which model" as a purely technical decision — it's now a strategic, geopolitical, and compliance question.

The US approach is built on closed, heavily-capitalized frontier labs (OpenAI, Anthropic, Google, Meta) competing on raw capability, safety infrastructure, and enterprise trust. Pricing is premium, but so is the assurance layer: audit trails, constitutional safeguards, and enterprise governance tooling designed for regulated industries.

The China approach — led by DeepSeek, Alibaba's Qwen, and Z.AI — has pursued an open-weight, cost-efficiency strategy. These models have closed the capability gap dramatically on reasoning and coding benchmarks while training at a fraction of the compute cost, and critically, many are released as open weights that any enterprise can self-host.

Three Implications for Leaders

1. Cost structures are diverging fast. Open-weight Chinese models let cost-sensitive functions (internal tooling, customer service triage, first-draft content) run on infrastructure you control, often at 10-20x lower inference cost. Reserve frontier closed models for high-stakes reasoning, client-facing outputs, and regulated decisions.

2. Data sovereignty is now a board-level question. Any model touching customer PII, financial data, or IP needs a clear answer to: where does inference happen, who can access logs, and under which jurisdiction's law. This applies equally to US cloud APIs and Chinese self-hosted models — sovereignty risk isn't one-directional.

3. Open-weight doesn't mean vendor-lock-free, but it does mean audit-friendly. Enterprises in manufacturing, energy, and financial services should evaluate open-weight models specifically where explainability and on-premise control matter more than bleeding-edge capability.

The Bottom Line

Don't pick a geography — build a **portfolio**. Route by task sensitivity and cost tolerance, not brand loyalty. Most enterprise GenAI stacks in 2026 will run 2-3 model families simultaneously.

Article – Tool Selection

Tool	Best For
Microsoft Copilot	Safe, enterprise-approved AI — deep M365 integration, good default for regulated orgs
Claude	Enterprise workflows — integrates with Chrome, Excel, PowerPoint; strong reasoning, long documents
Google Gemini	General-purpose assistant — deep Google Workspace integration, strong multimodal (image, video, audio) input
Google AI Studio	Prototyping — testing Gemini models, quick app scaffolding
Perplexity	Research — citations, live web search, document Q&A; also surfaces Chinese models (DeepSeek etc.) well
NotebookLM	Source-grounded research — upload your own documents, get answers tied to those sources; audio overviews for briefings
Hugging Face	Open model discovery — browsing, comparing and evaluating open-source models and datasets
Ollama	Running open-source LLMs locally — full privacy, no API cost, offline use, good for sensitive data you can't send to the cloud
LM Studio	Same local-LLM space as Ollama, but with a friendlier GUI — easier model browsing/switching, better for non-technical experimentation on your own machine
Gamma	Decks — prompt-to-presentation, faster than PowerPoint for internal and training material
Google Stitch	Wireframing — fast UI/UX mockups from prompts
Synthesia	Video — AI avatars, corporate training/explainer videos
ElevenLabs	Audio — voice cloning, narration, realistic TTS
Wispr Flow	Dictation (Speech to Text)
Granola / Fireflies / Otter / Read AI	Meeting intelligence — automatic notes, summaries, action items, CRM sync
n8n / Make / Zapier	Workflow automation — connecting AI to business systems without code

Article – Tool Selection for Marketing

Content Type	Tool	Best For
Images	Midjourney	High-aesthetic, stylized visuals; concept art, campaign key visuals
Images	Adobe Firefly	Commercially-safe images, brand-embedded workflows (Photoshop/Illustrator integration)
Images	Canva (Magic Media)	Quick social graphics, templated brand assets, non-designers
Images	Ideogram	Text-in-image accuracy (posters, logos, typography-heavy creatives)
Video	Higgsfield	Cinematic ad videos with real camera control, character consistency (Soul ID), URL-to-ad generation
Video	Runway (Gen-4)	Fine-grained video editing, VFX, motion control for polished short-form
Video	Sora 2 / Veo 3.1	High-fidelity, realistic text-to-video generation
Video	Kling 3.0	Competitive alternative for cinematic motion at lower credit cost
Video	Canva	Fast templated video edits, captions, quick social cuts
Audio	ElevenLabs	Voiceovers, multilingual dubbing, custom voice cloning
Audio	Suno	Full AI-generated music tracks/jingles for ad backgrounds
Audio	Adobe Podcast	Cleaning up/enhancing recorded voice audio
3D Models	Luma AI (Genie)	Text-to-3D asset generation for product/AR previews
3D Models	Spline AI	Interactive 3D web assets, landing page visuals
3D Models	Meshy	3D models for game/product prototyping
Multi-format	Canva	Brand-consistent images, simple video, presentations in one place
Multi-format	Higgsfield	Marketing Studio: video + image + character assets in one workspace

Article – Multimodal Inputs

Choosing the Right Input Format for GenAI

Modern GenAI tools can accept far more than plain text — PDFs, Excel files, screenshots, CSVs, images, and scanned documents can all be fed in directly. But format choice isn't cosmetic. It determines how accurately the model reads your data, how much it can reason over, and how safely that data is handled. For leaders rolling out AI-assisted workflows, knowing *which* format to use — and why — is as important as knowing the tool itself.

Why format choice matters

Every input format is processed differently under the hood. A native Excel file preserves rows, columns, formulas, and structure that the model can query directly. A PDF may contain that same data as flattened text or, worse, as an ineffectively-rendered table. A screenshot forces the model to *read* the data visually (OCR-style) rather than parse it as structured data — introducing a layer of interpretation, and therefore error, that a native file avoids entirely. Choosing the wrong format doesn't just reduce quality; it can silently introduce mistakes that look plausible but aren't accurate.

A practical decision framework

1. Use native structured files (Excel, CSV) when the task involves numbers.
If you need calculations, aggregations, trend analysis, or anything touching multiple rows/columns, always upload the native file rather than a screenshot or PDF export. The model can parse actual cell values and formulas, rather than guessing numbers from an image — this is the single biggest driver of accuracy in data-heavy tasks.

2. Use PDFs when the source is a formatted document, not a dataset.
Reports, contracts, policy documents, and slide-style content are well suited to PDF, since the model can extract and reason over text and layout. Avoid PDF for tabular data if a native spreadsheet version exists — table extraction from PDFs is often lossy, especially with merged cells or multi-page tables.

3. Use screenshots only when no other format is available, and keep expectations modest.
Screenshots are useful for quick visual context — showing a chart, a UI bug, or a snippet of a larger tool — but they should not be used as the source of truth for any numeric analysis. Treat a screenshot as "show me what you see," not "extract exact figures from this."

4. Use CSVs for large or simple tabular datasets.
CSV is often lighter and faster to process than Excel for very large datasets, and it strips away formatting noise (merged cells, multiple tabs, conditional formatting) that can confuse extraction. If the file is a single flat table with no formulas or formatting dependencies, CSV is usually the cleanest choice.

5. Match the format to the task, not the file you happen to have.
If you only have a screenshot of an Excel sheet, it's often worth taking the extra minute to locate or export the underlying file before uploading — the accuracy gain is usually worth the effort, particularly for anything that will inform a decision or go into a client deliverable.

MULTIMODAL INPUTS

Input format decides GenAI accuracy

Excel, PDF, screenshot or CSV: each is read differently, so pick the one that fits the job.



Excel for anything with numbers

01

The model reads real cell values and formulas, not numbers guessed from an image. The biggest driver of accuracy on data tasks.



PDF for formatted documents

02

Reports, contracts, policies and slides. Avoid for tables if a spreadsheet exists: merged cells and multi-page tables extract poorly.



Screenshots only as a last resort

03

Fine for quick context: a chart, a UI bug, a snippet. Treat it as “show me what you see”, never as the source for exact figures.



CSV for large, flat tables

04

Lighter and faster than Excel, with no merged cells, tabs or conditional formatting. Best for one table with no formulas.

RULE OF THUMB

Match the format to the task, not the file you happen to have. Only got a screenshot? Export the source file first.

Why it matters: a screenshot makes the model read data visually, so it can add errors that **look plausible but are wrong.**

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Part 2 — Adoption: Get it Used

HMC ADOPT-9 AI PERSONA™

The Nine Types of AI Users

Not every employee relates to AI the same way — some avoid it, some overuse it blindly, and only a few use it well. The HMC ADOPT-9™ Persona Grid maps users across two axes: how much they engage with AI, and how much they trust its output.

The riskiest quadrant isn't the loud Skeptic — it's the quiet Passive Truster, who rarely checks what they're told.

The goal for leaders isn't universal adoption; it's moving people toward the AI-Native Leader corner: high engagement, high — but verified — trust.

	Low Trust in Output	**Mid Trust in Output**	**High Trust in Output**
High Engagement	**The Skeptic** Understands AI well enough to distrust it. Points out flaws faster than use cases.	**The Power User** Fluent across tools and use cases. Actively finds new applications for the team.	**The AI-Native Leader** Redesigns workflows around AI, verifies rigorously, coaches others.
Mid Engagement	**The Dabbler** Tried it once for a demo. Curious but hasn't built a habit or seen real payoff.	**The Utility User** Treats AI like a tool, not a partner. Reliable for specific, repeatable tasks.	**The Delegator** Hands off real work to AI and spot-checks outcomes. Balanced reliance.
Low Engagement	**The Avoider** Hasn't opened an AI tool yet. Sees it as a fad or "someone else's job."	**The Cautious Adopter** Uses AI for safe, low-stakes tasks only. Slow to expand scope.	**The Passive Truster** Rarely uses AI, but accepts output without checking. Risky combination.

HMC ADOPT-9™

The Nine Types of AI Users

Not every employee relates to AI the same way — some avoid it, some overuse it blindly, and only a few use it well. The HMC ADOPT-9™ Persona Grid maps users across two axes: how much they engage with AI, and how much they trust its output.

The riskiest quadrant isn't the loud Skeptic — it's the quiet Passive Truster, who rarely checks what they're told.

The goal for leaders isn't universal adoption; it's moving people toward the AI-Native Leader corner: high engagement, high — but verified — trust.

ENGAGEMENT / TRUST	Low Trust in Output	Mid Trust in Output	High Trust in Output
High Engagement	**The Skeptic** Understands AI well enough to distrust it. Points out flaws faster than use cases.	**The Power User** Fluent across tools and use cases. Actively finds new applications for the team.	**The AI-Native Leader** Redesigns workflows around AI, verifies rigorously, coaches others. ★ TARGET
Mid Engagement	**The Dabbler** Tried it once for a demo. Curious but hasn't built a habit or seen real payoff.	**The Utility User** Treats AI like a tool, not a partner. Reliable for specific, repeatable tasks.	**The Delegator** Hands off real work to AI and spot-checks outcomes. Balanced reliance.
Low Engagement	**The Avoider** Hasn't opened an AI tool yet. Sees it as a fad or "someone else's job."	**The Cautious Adopter** Uses AI for safe, low-stakes tasks only. Slow to expand scope.	**The Passive Truster** Rarely uses AI, but accepts output without checking. Risky combination. ⚠ HIGHEST RISK

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

HMC ROLE-TASK-TOOL MATRIX™

Framework for Precision AI Rollout

Most AI rollouts fail on a targeting problem, not a training problem. Leaders issue a blanket license — "everyone gets Claude" — and expect adoption to follow. It doesn't, because a finance analyst, a plant supervisor, and a sales rep don't do the same work, and a generic tool given to all three lands as relevant to none of them.

Adoption isn't a function of access. It's a function of fit — whether the tool maps to a task the person actually owns, this week, with a visible payoff. The HMC ROLE-TASK-TOOL MATRIX™ forces that mapping before a single license is issued.

The Three-Layer Map

Layer	Question It Answers	Failure Mode If Skipped
Role	Who is this person, and what outcome are they accountable for?	Training is generic; content doesn't land as "for me"
Task	Which specific, recurring task inside that role is slow, repetitive, or error-prone?	AI gets used for novelty demos, not real work
Tool	Which AI capability — drafting, analysis, summarization, coding — actually fits that task?	Wrong tool for the job; one bad experience kills further use

Why It Precedes Training

- **Prevents shelf-ware licenses.** A tool with no mapped task is a tool that goes unused by week three — the cost shows up in the license renewal, not the training feedback form.
- **Makes ROI attributable.** When a task is named up front, the time saved on that task is measurable. Blanket rollouts can only ever report anecdotes.
- **Prioritizes training spend.** Not every role needs the same hours. The matrix tells you where 3 hours of training changes behaviour and where 30 minutes is enough.
- **Feeds directly into HMC ADOPT-9™.** Once the right tool is matched to the right task, the persona grid tells you how much coaching versus how much verification that person needs to use it well.

How It's Built

- **Step 1 — Role inventory.** List roles by function, not by title. "Regional sales lead" and "key account sales lead" may need different tasks even with similar titles.

- **Step 2 — Task audit.** For each role, identify the 3–5 tasks that consume the most hours or carry the most error risk. Interview, don't assume — the task leaders think is painful is often not the one employees actually struggle with.
- **Step 3 — Tool assignment.** Match each task to a specific AI capability and a specific workflow (not just "use Claude" — "use Claude to draft the first pass of the weekly variance commentary from the finance dashboard export").
- **Step 4 — Pilot and confirm.** Run the matched pairing with a small group before scaling the license. If it doesn't save time in week one, the mapping is wrong — revisit, don't push harder on adoption messaging.

The matrix isn't a one-time exercise — roles evolve and tasks shift as AI capability improves. Leaders who revisit it quarterly catch the moment a tool has outgrown its original task and can reassign it before the license becomes dead weight.

HMC ROLE-TASK-TOOL MATRIX™

AI adoption follows fit, not access

Map every license to a role, a task that role owns and the AI capability that fits it, before any training starts.

THE THREE-LAYER MAP

Layer	Question it answers	Failure mode if skipped
1 Role	Who is this person, and what outcome are they accountable for?	Training is generic; content doesn't land as "for me"
2 Task	Which recurring task in that role is slow, repetitive or error-prone?	AI gets used for novelty demos, not real work
3 Tool	Which AI capability (drafting, analysis, summarising, coding) fits that task?	Wrong tool; one bad experience kills further use

WHY IT PRECEDES TRAINING

01 **No shelf-ware licenses.** A tool with no mapped task goes unused by week three.

02 **Attributable ROI.** A named task makes time saved measurable; blanket rollouts only yield anecdotes.

03 **Targeted training spend.** Shows where 3 hours change behaviour and where 30 minutes is enough.

04 **Feeds HMC ADOPT-9™.** The persona grid then sets how much coaching versus verification each person needs.

HOW IT'S BUILT

1 Role inventory → **2 Task audit** → **3 Tool match** → **4 Pilot**

List roles by function, not by title.

Find the 3–5 tasks with the most hours or error risk. Interview, don't assume.

Match each task to a capability and a named workflow.

Test with a small group. No time saved in week one means the mapping is wrong.

> **Revisit quarterly.** Tasks shift as AI improves; reassign a tool that has outgrown its task before the license becomes dead weight.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

HMC C.R.A.F.T.S.™

Prompt Architecture Framework

Letter	Element 1	Element 2
C	**Context** — background, data, situation the model needs	**Character** — persona/role the model should assume
R	**Role** — the specific task/job being asked	**Reasoning** — step-by-step thinking shown, or final answer only
A	**Aim** — the specific goal or outcome	**Audience** — who the output is ultimately for
F	**Format** — structure: bullets, table, headers, length	**Flow** — logical sequence the response should follow
T	**Tone** — voice: formal, casual, technical, persuasive	**Test** — review, refine, re-prompt (built-in iteration loop)
S	**Scope** — what's in/out of bounds for the task	**Constraints** — hard limits: length, budget, compliance, format rules

HMC C.R.A.F.T.S.™

A 6-part framework to structure any AI prompt — for consistent, high-quality outputs every time.
Miss one element and the model has to guess.
Nail all six and you get exactly what you asked for — the first time.
Run through Context, Role, Aim, Format, Tone and Scope before you hit enter.

Save this post and make it your go-to prompting checklist.

Letter	Element 1	Element 2
C	Context — background, data, situation the model needs	Character — persona/role the model should assume
R	Role — the specific task/job being asked	Reasoning — step-by-step thinking shown, or final answer only
A	Aim — the specific goal or outcome	Audience — who the output is ultimately for
F	Format — structure: bullets, table, headers, length	Flow — logical sequence the response should follow
T	Tone — voice: formal, casual, technical, persuasive	Test — review, refine, re-prompt (built-in iteration loop)
S	Scope — what's in/out of bounds for the task	Constraints — hard limits: length, budget, compliance, format rules

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – Organizational Adoption

Most GenAI initiatives don't fail because the technology is weak — they fail because adoption is treated as a tech rollout instead of a change program. Leaders who succeed start small: a focused pilot with a willing team, clear success metrics, and fast feedback loops — then scale what works. The bigger risk is skipping straight to "everyone must use AI" without building skills, trust, or workflow fit first. Adoption is won or lost on habits, not licenses purchased.

The leader's real job is sequencing: prove value, build confidence, remove friction, then expand — deliberately, function by function.

Pilot Program — A small, contained trial of AI in one team or use case before wider rollout. Let’s you learn cheaply and fix problems early.

Change Management — The discipline of helping people move from old habits to new ones. The real bottleneck in AI adoption, far more than the tool itself.

Workflow Fit — How well an AI tool slots into existing ways of working. Poor fit is the #1 reason adoption stalls even with good tools.

Champions Network — A group of early, enthusiastic internal users who model and spread adoption informally. More persuasive to peers than top-down mandates.

AI Fluency — A baseline working comfort with AI tools, distinct from technical expertise. The goal for most employees, not deep AI literacy.

Adoption Curve — The typical spread of usage from early adopters to laggards. Helps leaders set realistic timelines instead of expecting instant, uniform uptake.

Friction Points — Practical obstacles to using AI day-to-day (access, trust, unclear use cases). Removing these matters more than more training.

Vanity Metrics — Numbers that look good but don't reflect real impact (e.g. "logins" vs. "hours saved"). Leaders should track outcomes, not activity.

Reskilling — Deliberately building new capabilities in employees whose roles shift due to AI. A retention and morale issue as much as a skills one.

Sponsorship — Visible, active support from senior leadership for AI adoption. Without it, initiatives stay siloed and never reach organization-wide scale.

ORGANIZATIONAL ADOPTION

GenAI adoption is a change program, not a technology rollout

Adoption is won on habits, not licenses purchased. The leader's real job is sequencing.

THE LEADER'S SEQUENCE

1 **Prove value**
Pilot with a willing team

2 **Build confidence**
Clear metrics, fast feedback

3 **Remove friction**
Fix access, trust, use cases

4 **Expand**
Function by function

TEN TERMS EVERY LEADER SHOULD KNOW

01 Pilot program
A contained trial in one team or use case. Learn cheaply, fix problems early.

02 Change management
Moving people from old habits to new ones. The real bottleneck, more than the tool.

03 Workflow fit
How well AI slots into existing work. Poor fit is the #1 reason adoption stalls.

04 Champions network
Early internal users who spread adoption. More persuasive to peers than mandates.

05 AI fluency
Working comfort with AI tools, not technical expertise. The goal for most employees.

06 Adoption curve
Usage spreads from early adopters to laggards. Set realistic timelines.

07 Friction points
Access, trust and unclear use cases. Removing them matters more than more training.

08 Vanity metrics
Logins look good; hours saved show impact. Track outcomes, not activity.

09 Reskilling
Building new capabilities as roles shift. A retention and morale issue too.

10 Sponsorship
Visible senior support. Without it, initiatives stay siloed and never scale.

Start small, then scale what works. Skipping straight to "everyone must use AI" is the bigger risk.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – The 90-Day North Star

Ask a leader how their AI rollout is going three months in, and the answer is almost always a story, not a number — a good demo someone saw, a team that "seems excited," a license count. Not because the initiative failed, but because no one agreed in advance on what would count as progress and when it would be checked.

Without a fixed cadence, progress reporting becomes whoever asks loudest, whenever they happen to ask. The champion reports wins. The skeptic reports the one failure they heard about. Leadership gets neither a trend line nor a decision point — just noise timed to whoever spoke last.

The 90-Day North Star replaces ad hoc updates with three fixed checkpoints — Day 30, Day 60, Day 90 — each asking a different question, in a fixed order. You cannot skip to the ROI question before the adoption question has an honest answer, because the two are not actually the same conversation. Treating them as one is exactly how a rollout stalls quietly for a full quarter before anyone notices.

Day 30 — Adoption Check
The question: are people actually opening the tool? What good looks like: usage frequency by role, not license count. A tool with 100% licensing and 20% weekly use is not "adopted" — it's shelved with a receipt.

Day 60 — Workflow Integration Check
The question: has it moved from a side experiment into the actual way work gets done? What good looks like: the task the tool touches has visibly changed — fewer manual steps, a shorter cycle time, a document that used to take a day now taking an hour. Not "people like it," but "the workflow is different now."

Day 90 — ROI Check
The question: what did it actually deliver? What good looks like: a clear read on time saved, quality improved, or new work made possible — grounded in what changed, not what people felt.

Why the order can't be skipped

ROI without adoption is fiction. A pilot that shows strong time-savings from three power users tells you nothing about what happens when the other twenty people on the team touch it. Day 90 numbers built on Day 30 usage that was never checked are numbers built on sand.

Adoption without integration is a vanity metric. High login counts with no change to the underlying workflow just means people are curious, not that the initiative is working. Day 60 is the checkpoint that separates novelty from habit.

Skipping straight to Day 90 hides where things actually broke. If ROI comes in low, the fixed cadence means you already know whether it was an adoption problem, an integration problem, or a genuine value problem — instead of starting a root-cause investigation from zero, three months late.

How leaders use it

One page, three dates, no exceptions. Every new AI initiative — a new tool, a new team rollout, a new use case — gets the same three checkpoints on the calendar the day it launches, not whenever someone remembers to ask.

Report the miss, not just the win. A Day 30 check that shows low adoption is not a failed report — it's the report doing its job. The initiative gets a correction window before Day 60, instead of a surprise at Day 90.

Use it to decide, not just to report. Each checkpoint should end in a decision — continue, adjust, or stop — not just a status update. A cadence with no decision attached to it is just a calendar reminder.

Most GenAI initiatives do not fail in a single dramatic moment — they fail in the gap between launch enthusiasm and the next time anyone checks. The 90-Day North Star closes that gap by making sure someone checks on schedule, asks the right question for that stage, and has the discipline to act on what the checkpoint actually shows.

HMC 90-DAY NORTH STAR™

Three checkpoints, one fixed order

Replace ad hoc updates with Day 30, 60 and 90 checks, each asking a different question.



DAY 30

Adoption

Are people actually opening the tool?

WHAT GOOD LOOKS LIKE

Weekly usage by role, not license count. 100% licensed at 20% weekly use is shelved.

DAY 60

Integration

Has it become the way work gets done?

WHAT GOOD LOOKS LIKE

The task has visibly changed: fewer steps, shorter cycles, a day's work done in an hour.

DAY 90

ROI

What did it actually deliver?

WHAT GOOD LOOKS LIKE

Time saved, quality improved or new work made possible, grounded in what changed.

WHY THE ORDER CAN'T BE SKIPPED

ROI without adoption is fiction.

Adoption without integration is a vanity metric.

Skipping to Day 90 hides where things broke.

Set all three dates on launch day, and report the miss as well as the win.

End every checkpoint in a decision:

CONTINUE | ADJUST | STOP

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Part 3 — ROI: Prove it

Article – The AI Baseline Document

Why This Exists

You cannot measure an AI initiative's ROI if you never wrote down what "before" looked like. Most rollouts skip this step entirely — they go from vendor pitch to go-live with no ground-truth snapshot of the process being changed. Three months later, someone asks, "did it work?" and the honest answer is "we don't know, because we never measured the starting point."

This document is filled out **before** any AI tool goes live. It takes 30–45 minutes per process and should be completed with the people who actually do the work — not just their manager's assumption of what the work looks like.

One baseline per process/task. If you're rolling out AI across five workflows, fill this out five times.

Section 0 — Identification

Field	Entry
Process / task being baselined	_____
Business owner (accountable for outcome)	_____
IT / technical owner	_____
AI tool being introduced	_____
Team / function	_____
Baseline date	_____
Number of people currently performing this task	_____

Section 1 — Quantitative Baseline

Hard numbers. If you don't have exact figures, use your best honest estimate and mark it "(est.)" — an honest estimate beats a false precision.

Time

Metric	Current State
Average time to complete one instance of this task	_____ (mins/hrs)
Number of times this task is performed per week/month	_____
Total person-hours consumed per month	_____

Number of people involved per instance (handoffs)	_____

Quality / Error

Metric	Current State
Error rate (errors per X instances)	_____
Rework rate (% of instances requiring correction)	_____
Average time to catch and fix an error	_____
Cost of a typical error (rework hours, client impact, compliance exposure)	_____

Cost

Metric	Current State
Fully loaded cost per instance (people time × rate)	_____
Any current tool/vendor cost being replaced	_____
Cost of delay (revenue/opportunity lost while task is in progress)	_____

Speed / Cycle Time

Metric	Current State
Time from task start to final output delivered	_____
Time from output delivered to decision/action taken on it	_____
Number of review/approval cycles before output is considered final	_____

Volume Trend

Metric	Current State
Is task volume growing, flat, or shrinking?	_____
Expected volume in 12 months	_____

Section 2 — Qualitative Baseline

Numbers alone miss the story. Capture this from the people doing the task, in their own words where possible.

Experience of the Work

Question	Response
In one sentence, how does the team describe this task today?	_____
What part of this task do people find most tedious or draining?	_____
What part requires genuine judgment or expertise (i.e., shouldn't be automated away)?	_____
Where does this task currently cause frustration for internal stakeholders or clients?	_____

Risk & Judgment

Question	Response
What's the worst-case outcome if this task is done poorly?	_____
Is there a compliance, legal, or regulatory dimension to this task?	_____
How much of this task currently relies on one person's tacit knowledge (i.e., a single point of failure)?	_____

Stakeholder Perception

Question	Response
How do downstream users/clients currently rate the output of this task (if measured)?	_____
What's the most common complaint about the current process, if any?	_____

Team Readiness (feeds directly into HMC ADOPT-9™)

Question	Response
How would you describe the team's general comfort with AI tools today?	_____
Has this team used any AI tool for this or an adjacent task before?	_____
What's the team's biggest stated concern about introducing AI here?	_____

Section 3 — The One-Line Baseline Statement

Force yourself to compress Sections 1 and 2 into a single sentence. This is the sentence you'll compare against at Day 30/60/90.

"Today, [task] takes ___ hours/week across ___ people, with a ___% error/rework rate, and the team describes it as ___."

Section 4 — Sign-Off

Role	Name	Date
Business owner	_____	_____
Team lead / process owner	_____	_____
AI initiative sponsor	_____	_____

Notes for Use

- **Fill this out before go-live, not after.** A baseline written retroactively is a guess dressed up as data.
- **Re-use the exact same questions at Day 90** (see HMC's 90-Day North Star framework) — the value of this document is entirely in the comparison, not the snapshot alone.
- **Don't skip the qualitative section.** A process can look fine on the numbers and still be quietly hated by the team, or vice versa — a slow process the team trusts completely. Both are real signal.
- **One baseline, one process.** Resist the urge to baseline "the whole department's AI readiness" in one document — it produces vague numbers no one can act on.

HMC TOOLKIT · THE AI BASELINE DOCUMENT

No baseline, no ROI: write down what "before" looked like ahead of go-live

Most rollouts jump from vendor pitch to go-live. Three months later nobody can say whether it worked.

30–45
minutes per process, filled in with the people who do the work

1
baseline per process. Five workflows means five documents

Day 90
re-run the exact same questions and compare

WHAT THE BASELINE CAPTURES

0	**Identification**	Business owner, IT owner, AI tool, team and headcount
1	**Quantitative**	Time, error and rework, cost per instance, cycle time, volume trend. Mark estimates "(est.)"
2	**Qualitative**	What drains people, what needs judgment, risk and tacit knowledge, team readiness for ADOPT-9™
3	**One-line statement**	Sections 1 and 2 compressed into the sentence below
4	**Sign-off**	Business owner, process lead and AI initiative sponsor

THE SENTENCE YOU COMPARE AGAINST AT DAY 30, 60 AND 90

"Today, [task] takes ___ hours/week across ___ people, with a ___% rework rate, and the team describes it as ___."

FOUR RULES OF USE

01
Before go-live. A retroactive baseline is a guess dressed up as data.

02
Same questions. The value is in the comparison, not the snapshot.

03
Keep the qualitative. A process can score well and still be hated.

04
One process each. Department-wide baselines give vague numbers.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

HMC G.A.I.N.S.™

Measuring the ROI of GenAI

Most companies measure GenAI ROI the way they measure software ROI — licenses purchased, logins tracked, hours "saved" on paper. But GenAI's value doesn't show up neatly in a single line item. It shows up as faster decisions, fewer errors, new offerings that weren't possible before, and capacity quietly freed up for higher-value work. Measuring only cost savings misses most of the story — and measuring nothing at all lets pilots die from lack of proof. Leaders need a framework that captures value across the full spread of what GenAI actually changes, not just what's easiest to count.

That's the gap the **HMC GAINS™ Framework** is built for.

The HMC GAINS™ Framework

G — Growth
Revenue-side impact: new products, faster sales cycles, better win rates, upsell driven by AI-assisted insight. Often underestimated because it's indirect.

A — Automation
The classic ROI: hours saved, tasks removed, throughput increased per person. Easiest to measure, but only part of the picture — don't stop here.

I — Improvement
Quality and risk gains: fewer errors, better compliance, more consistent output. Hard to put a number on, but often the most valuable pillar in regulated or high-stakes functions.

N — New Capability
Things the business simply couldn't do before — real-time market scans, personalized content at scale, analysis that used to require a specialist team. This is where GenAI creates option value, not just efficiency.

S — Sustained Adoption
Usage depth and habit formation over time. A tool with high initial use and fast drop-off has near-zero ROI, no matter how good the pilot numbers looked. This pillar is the leading indicator for whether the other four will hold up.

How Leaders Use It

Score each pillar loosely — high / medium / low / not yet measured — for every GenAI initiative, rather than forcing one blended ROI number. A hiring-screening tool might score high on Automation and Improvement but low on Growth; a customer-facing AI assistant might flip that. The GAINS scorecard forces leaders to ask "where is the value actually landing?" instead of defaulting to the one pillar (Automation) that's simplest to justify in a budget meeting.

The real failure mode isn't measuring GenAI ROI badly — it's measuring only the 20% of value that's easy to see, and starving the other 80% of the attention and investment it needs to compound.

HMC G.A.I.N.S.™

Measuring the ROI of GenAI

Most companies measure GenAI ROI the way they measure software ROI — licenses purchased, logins tracked, hours "saved" on paper. GenAI's value doesn't show up neatly in a single line item. It shows up as faster decisions, fewer errors, new offerings that weren't possible before, and capacity quietly freed up for higher-value work. Measuring only cost savings misses most of the story — and measuring nothing at all lets pilots die from lack of proof. Leaders need a framework that captures value across the full spread of what GenAI actually changes, not just what's easiest to count.

That's the gap the HMC GAINS™ Framework is built for.

G — Growth

Revenue-side impact: new products, faster sales cycles, better win rates, upsell driven by AI-assisted insight. Often underestimated because it's indirect.

A — Automation

The classic ROI: hours saved, tasks removed, throughput increased per person. Easiest to measure, but only part of the picture — don't stop here.

I — Improvement

Quality and risk gains: fewer errors, better compliance, more consistent output. Hard to put a number on, but often the most valuable pillar in regulated or high-stakes functions.

N — New Capability

Things the business simply couldn't do before — real-time market scans, personalized content at scale, analysis that used to require a specialist team. This is where GenAI creates option value, not just efficiency.

S — Sustained Adoption

Usage depth and habit formation over time. A tool with high initial use and fast drop-off has near-zero ROI, no matter how good the pilot numbers looked. This pillar is the leading indicator for whether the other four will hold up.

How Leaders Use It

Score each pillar loosely — high / medium / low / not yet measured — for every GenAI initiative, rather than forcing one blended ROI number. A hiring-screening tool might score high on Automation and Improvement but low on Growth; a customer-facing AI assistant might flip that. The GAINS scorecard forces leaders to ask “where is the value actually landing?” instead of defaulting to the one pillar (Automation) that's simplest to justify in a budget meeting.

The real failure mode isn't measuring GenAI ROI badly — it's measuring only the 20% of value that's easy to see, and starving the other 80% of the attention and investment it needs to compound.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Article – What AI is Actually Costing and Creating for Your Business

10 Key Terms for Quick Reference

Term	What It Means
ROI (Return on Investment)	The measurable value gained relative to what was spent
AI Baseline	Your starting performance snapshot before AI is introduced
Productivity Delta	The change in output per person or process after AI adoption
Time-to-Value	How quickly AI deployment starts generating measurable returns
Hard Savings	Direct, quantifiable cost reductions — headcount, hours, materials
Soft Savings	Indirect gains — faster decisions, higher morale, better quality
Error Rate Reduction	The decrease in mistakes or rework after AI enters a workflow
Adoption Rate	The percentage of your team actually using the AI tool consistently
AI Debt	Hidden costs that accumulate when AI is deployed without proper governance
Value Leakage	Value that AI creates but your organization fails to capture or measure

---

Most leaders approve AI investments the same way they buy expensive gym equipment — with great intentions and no plan to measure whether it actually gets used. The problem is not the technology. The problem is that traditional financial frameworks were never built for something that improves over time, behaves differently across teams, and delivers value in ways that do not always show up on a spreadsheet. ROI for AI is not a single number. It is a story told across time, behaviour, and business outcomes.

Before any AI initiative goes live, your job as a leader is to establish a clean baseline. What does the process look like today — the hours consumed, the errors made, the decisions delayed, the customers lost to slow response? Without this snapshot, you are measuring nothing. You are simply hoping. A baseline is not a technical document. It is a business record: here is what normal looks like before we intervene. Every meaningful ROI conversation begins here, not at the vendor pitch, not at the board presentation, but at honest ground-level observation of how work actually happens.

Once AI is deployed, value shows up in two forms — and leaders routinely count only one of them. Hard savings are visible: fewer hours on a task, reduced vendor costs, lower error-related rework. But soft savings are where the compounding begins. Teams make faster decisions. Managers spend less time in status meetings. Customers receive responses that feel faster and more relevant. These gains are real, but they evaporate if no one is tracking them. Build a simple measurement rhythm — weekly for adoption, monthly for productivity delta, quarterly for strategic outcomes. Do not wait for an annual review to find out whether the investment worked.

The most overlooked ROI metric is adoption rate, and it is the one that sits entirely within your control. A sophisticated AI tool used by a fraction of your team returns a fraction of its potential value. If people are not using it, the fault is rarely the technology — it is the change management, the training, the trust, or the clarity of purpose. Every percentage point of adoption is a percentage point of ROI recovered. Leaders who obsess over the tool and ignore the human uptake are measuring the gym equipment, not the fitness.

The Leader's Playbook

Action	**Why It Matters**
Document your baseline before any AI goes live	You cannot measure improvement without knowing where you started
Separate hard savings from soft savings in your tracking	Both are real; treating them the same distorts your picture
Review adoption rates before reviewing financial returns	Low usage is the earliest warning sign of value leakage
Build a simple measurement calendar — weekly, monthly, quarterly	AI ROI is not an event; it is a rhythm that requires consistent attention
Assign a business owner to every AI initiative, not just an IT owner	Technology without accountability produces data, not decisions

What AI Is Actually Costing and Creating for Your Business

- **ROI for AI is a story** told across time, behaviour and business outcomes, not a single number.
- **Set a baseline first:** the hours, errors, delays and lost customers before AI goes live.
- **Count hard and soft savings:** soft gains compound but vanish if no one tracks them.
- **Adoption rate is the metric you control:** every point of adoption recovers a point of ROI.

10 key terms for quick reference

Term	What It Means
ROI (Return on Investment)	The measurable value gained relative to what was spent
AI Baseline	Your starting performance snapshot before AI is introduced
Productivity Delta	The change in output per person or process after AI adoption
Time-to-Value	How quickly AI deployment starts generating measurable returns
Hard Savings	Direct, quantifiable cost reductions — headcount, hours, materials
Soft Savings	Indirect gains — faster decisions, higher morale, better quality
Error Rate Reduction	The decrease in mistakes or rework after AI enters a workflow
Adoption Rate	The percentage of your team actually using the AI tool consistently
AI Debt	Hidden costs that accumulate when AI is deployed without proper governance
Value Leakage	Value that AI creates but your organization fails to capture or measure

The leader's playbook

Action	Why It Matters
Document your baseline before any AI goes live	You cannot measure improvement without knowing where you started
Separate hard savings from soft savings in your tracking	Both are real; treating them the same distorts your picture
Review adoption rates before reviewing financial returns	Low usage is the earliest warning sign of value leakage
Build a simple measurement calendar — weekly, monthly, quarterly	AI ROI is not an event; it is a rhythm that requires consistent attention
Assign a business owner to every AI initiative, not just an IT owner	Technology without accountability produces data, not decisions

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Part 4 — Safety and Governance

HMC S.A.F.E.R. AI™

Five principles to guide responsible AI usage across the organization

Our Guiding Principle: AI is a powerful productivity enabler — but must be used with awareness, caution, and accountability. The S.A.F.E.R. framework gives you a clear mental model to follow.

S — SECURE THE DATA

Never expose what shouldn't leave the organisation

Do not share confidential or client data

Avoid uploading internal documents or financials

Use anonymised or dummy data wherever possible

Prefer enterprise-approved AI tools only

A — ASSESS THE INPUT

Think before you prompt

Is the data sensitive or proprietary?

Can the request be generalised?

Are you sharing more context than necessary?

Would this be safe to share externally?

F — FACT-CHECK THE OUTPUT

AI can be confident — not always correct

Verify facts, numbers, and insights

Cross-check critical outputs with trusted sources

Be cautious of hallucinations (fabricated info)

Don't publish AI output without review

E — EVALUATE RISKS

Understand the consequences before use

Consider legal and compliance implications

Watch for bias or ethical concerns

Assess business impact of incorrect outputs

Escalate uncertainties to IT / Compliance

R — RETAIN HUMAN JUDGMENT
AI assists. Humans decide.

Do not rely on AI for final decisions without validation

Use AI as a co-pilot, not autopilot

Maintain accountability for outcomes

Ensure team members are trained in AI limitations

HMC S.A.F.E.R.™

Five principles to guide responsible AI usage across the organization.
AI is a powerful productivity enabler — but must be used with awareness, caution, and accountability.

The S.A.F.E.R. framework gives you a clear mental model to follow.

S — SECURE THE DATA

Never expose what shouldn't leave the organisation

- Do not share confidential or client data
- Avoid uploading internal documents or financials
- Use anonymised or dummy data wherever possible
- Prefer enterprise-approved AI tools only

A — ASSESS THE INPUT

Think before you prompt

- Is the data sensitive or proprietary?
- Can the request be generalised?
- Are you sharing more context than necessary?
- Would this be safe to share externally?

F — FACT-CHECK THE OUTPUT

AI can be confident — not always correct

- Verify facts, numbers, and insights
- Cross-check critical outputs with trusted sources
- Be cautious of hallucinations (fabricated info)
- Don't publish AI output without review

E — EVALUATE RISKS

Understand the consequences before use

- Consider legal and compliance implications
- Watch for bias or ethical concerns
- Assess business impact of incorrect outputs
- Escalate uncertainties to IT / Compliance

R — RETAIN HUMAN JUDGMENT

AI assists. Humans decide.

- Do not rely on AI for final decisions without validation
- Use AI as a co-pilot, not autopilot
- Maintain accountability for outcomes
- Ensure team members are trained in AI limitations

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Article – Types of AI Challenges

GenAI tools promise speed, scale, and intelligence — and for the most part, they deliver. But like any powerful tool, they come with a set of failure modes that are easy to overlook in a demo and costly to discover in production. For leaders driving adoption across their organization, understanding these risks isn't a technical afterthought — it's a governance responsibility.

 # | Challenge | What it means

1. Inaccurate
Gives wrong or misleading answers, often with the same fluency and confidence as correct ones — making errors hard to spot without review.

2. Hallucinates
Makes up facts, sources, or content that isn't real, which can end up unchecked in reports, decks, or client communication.

3. Irrelevant / Confused
Misses the point or struggles with complex, multi-step questions, sometimes completing only part of a multi-requirement prompt.

4. Outdated
Relies on training data that may no longer reflect current facts, prices, regulations, or product details.

5. Inconsistent
Output quality varies between runs of the same prompt, making it hard to standardize AI-assisted work across a team.

6. Biased
Shows unfair, one-sided, or stereotyped responses reflecting patterns in training data — which can quietly shape hiring copy, reviews, or client-facing content.

7. Data leakage
Sensitive or confidential information entered by employees can be exposed or mishandled, depending on the tool's data policy.

8. Over-reliance
Teams stop verifying outputs, eroding judgment and institutional skill over time — with errors surfacing only after they've caused damage.

9. Cost unpredictability
Usage-based pricing can scale unexpectedly at enterprise volume, especially once adoption spreads beyond a pilot team.

10. IP/copyright exposure
Generated content may unintentionally reproduce copyrighted text, code, or images, creating legal risk for anything published externally.

11. Governance gap
No clear owner exists for approving use cases, monitoring outputs, or taking accountability when something fails.

12. Prompt injection / manipulation
Malicious or hidden inputs — including text buried in uploaded documents — can hijack a model's behaviour or extract confidential context.

13. Shadow AI usage
Employees use unapproved AI tools to work faster, often pasting company data into consumer-grade apps with no enterprise safeguards or visibility.

14. Explainability deficit
Models can't reliably explain why they produced a given answer, making outputs hard to audit in regulated contexts like finance or HR.

15. Vendor/model lock-in
Workflows built around one provider's model or API can be costly to migrate if pricing, policies, or capabilities change down the line.

TYPES OF AI CHALLENGES

15 ways GenAI fails, and why leaders own the risk

Easy to miss in a demo, costly in production. Understanding them is a governance duty, not a technical afterthought.

OUTPUT QUALITY 6

1. **Inaccurate**
   Wrong answers delivered with the same confidence as right ones.
2. **Hallucinates**
   Invents facts or sources that slip into reports and client work.
3. **Irrelevant / confused**
   Misses the point or completes only part of a multi-step prompt.
4. **Outdated**
   Training data lags current facts, prices and regulations.
5. **Inconsistent**
   Same prompt, different quality, so work is hard to standardise.
6. **Biased**
   Stereotyped output that can shape hiring copy and reviews.

SECURITY AND LEGAL 4

7. **Data leakage**
   Confidential data entered by staff can be exposed or mishandled.
10. **IP / copyright exposure**
    Output may reproduce protected text, code or images.
12. **Prompt injection**
    Hidden instructions in inputs hijack the model or extract data.
13. **Shadow AI usage**
    Staff paste company data into unapproved consumer apps.

ORGANISATION AND COST 5

8. **Over-reliance**
   Teams stop verifying, and errors surface only after damage.
9. **Cost unpredictability**
   Usage pricing scales fast once adoption passes the pilot.
11. **Governance gap**
    No owner approves use cases or answers for failures.
14. **Explainability deficit**
    Answers are hard to audit in regulated areas like finance or HR.
15. **Vendor / model lock-in**
    Workflows tied to one model are costly to migrate.

WHAT LEADERS SHOULD DO

Name an owner for every use case, keep a human check on anything that leaves the building, and set a clear data policy **before adoption spreads beyond the pilot.**

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Article – Safety and Data Leakage

As GenAI tools spread across teams, the biggest risk isn't the model — it's what employees paste into it. Confidential client data, financials, source code, or personal information typed into public AI tools can be stored, used for training, or exposed in a breach, outside your control. Leaders must set clear boundaries: which tools are approved, what data classes are off-limits, and how outputs get verified before use. Safety isn't a one-time policy — it's an ongoing habit of awareness, built through training, guardrails, and accountability, so AI adoption speeds up the business without quietly leaking its crown jewels.

1. **Data Leakage** — Sensitive or confidential information unintentionally exposed to an external system. In GenAI, this usually happens via prompts, not hacks.
2. **Prompt Injection** — Malicious instructions hidden in text/documents that trick an AI into ignoring rules or leaking data. A growing risk as AI reads more external content.
3. **Shadow AI** — Employees using unapproved AI tools without IT/security oversight. Creates blind spots leaders can't govern or audit.
4. **Data Residency** — Where your data is physically stored and processed. Matters for compliance with laws like GDPR or India's DPDP Act.
5. **Model Training Opt-Out** — Setting that prevents your inputs from being used to train the provider's future models. Critical for enterprise/business-tier tools.
6. **PII (Personally Identifiable Information)** — Any data that can identify an individual (name, ID, contact details). Should never enter public AI tools unmasked.
7. **Zero Data Retention (ZDR)** — An enterprise agreement where the provider doesn't store your inputs/outputs at all. Highest level of data protection available.
8. **Access Controls** — Rules defining who can use which AI tools, with what data, at what permission level. Prevents uncontrolled or excessive exposure.
9. **Hallucination Risk** — AI generating confident but false information. A safety issue when outputs are trusted and acted on without verification.
10. **Human-in-the-Loop** — Keeping a person responsible for reviewing and approving AI outputs before action. The core safeguard against both leakage and errors.

SAFETY AND DATA LEAKAGE

The biggest GenAI risk is what employees paste in

Client data, financials, code or personal details typed into public tools can be stored, trained on or exposed.



LEADERS SET THREE BOUNDARIES

1. **Approved tools**
   Which AI tools staff may use
2. **Off-limits data**
   Which data classes never go in
3. **Verified outputs**
   How results are checked before use

5 RISKS TO WATCH

- **Data leakage**
  Confidential data exposed to an outside system, usually via prompts, not hacks.
- **Prompt injection**
  Hidden instructions in documents trick the AI into breaking rules or leaking data.
- **Shadow AI**
  Unapproved tools used without IT oversight: blind spots no one can audit.
- **PII exposure**
  Names, IDs or contact details should never enter public tools unmasked.
- **Hallucination risk**
  Confident but false output becomes a safety issue when acted on unchecked.

5 SAFEGUARDS TO SET

- **Human-in-the-loop**
  A named person reviews and approves AI output before action. The core safeguard.
- **Zero data retention**
  The provider stores no inputs or outputs. The highest protection available.
- **Training opt-out**
  Stops your inputs training the provider's models. Essential on business tiers.
- **Access controls**
  Rules on who uses which tools, with what data, at what permission level.
- **Data residency**
  Where data is stored and processed; matters for GDPR and India's DPDP Act.

> **Safety is a habit, not a one-time policy.**
> Build it through training, guardrails and accountability, so AI speeds up the business without leaking its crown jewels.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Article – Compliance

As GenAI adoption scales, data governance isn't optional — here's the landscape leaders should know.

GDPR (EU) governs personal data of EU residents — requires consent, right to erasure, and breach notification within 72 hours. Any AI tool processing EU customer/employee data must comply, regardless of where your company is based.

CCPA/CPRA (California) gives consumers rights to know, delete, and opt out of data sale — increasingly the US benchmark other states mirror.

SOC 2 is an audit standard (not a law) proving a vendor's controls around security, availability, and confidentiality — critical when evaluating AI vendors or cloud tools for enterprise use.

ISO 27001 is the global standard for information security management systems — broader than SOC 2, often required for cross-border enterprise contracts.

HIPAA (US) protects health information handled by covered healthcare entities and their business associates — GenAI vendors processing protected health information (PHI) on their behalf must meet applicable privacy and security requirements and enter into a Business Associate Agreement (BAA).

Emerging additions: ISO 42001 (AI management systems), the EU AI Act (risk-tiered AI regulation), and India's DPDP Act (2023) — increasingly relevant for organizations operating across India.

Practical takeaway: before rolling out any GenAI tool, map what data it touches, which regulations apply by geography, and whether the vendor holds SOC 2/ISO 27001 certification.

DATA GOVERNANCE · THE COMPLIANCE LANDSCAPE

GenAI at scale runs on data governance: know the rules before the rollout

Laws set what you must do with the data. Standards prove a vendor can be trusted with it.

LAWS What you must do with personal data

GDPR
EU · LAW

Consent, right to erasure, breach notice within 72 hours. Applies wherever your company is based.

CCPA / CPRA
CALIFORNIA · LAW

Rights to know, delete and opt out of data sale. The US benchmark other states copy.

HIPAA
US HEALTH · LAW

Protects health data. GenAI vendors handling PHI must sign a Business Associate Agreement.

STANDARDS What a vendor must prove

SOC 2
AUDIT · NOT A LAW

Proves a vendor's security, availability and confidentiality controls. Ask for it before any enterprise AI purchase.

ISO 27001
GLOBAL · CERTIFICATION

Information security management, broader than SOC 2. Often required in cross-border enterprise contracts.

ON THE HORIZON

ISO 42001
Management systems for AI

EU AI Act
Risk-tiered rules for AI systems

India DPDP Act (2023)
Relevant across India

BEFORE ANY GENAI ROLLOUT

1. Map the data
What personal, health or client data does the tool touch?

2. Match the rules
Which regulations apply in each geography you operate in?

3. Check the vendor
Does it hold SOC 2 or ISO 27001 certification?

Compliance is a design input, not a launch-week checklist.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – Levels of Safety

Same model, six different postures. What leaders should actually be deciding when they ask whether AI is safe to use.

"Is it safe?" is the wrong question in a boardroom. The model is identical across every tier. What changes is the contract it runs under — whether your data trains the model, how long it is stored, and what your IT function can see. That makes this a procurement and governance decision, not a technology one.

Anthropic splits its products in two. Claude Free, Pro and Max sit under Consumer Terms. Team, Enterprise, the API and the cloud marketplaces sit under Commercial Terms, where inputs and outputs are not used for model training by default. Almost everything else is detail layered on that single split.

The tiers at a glance

Tier	Terms & training	Data retention	Admin & governance	Right for
Free	Consumer. Chats used for training unless the user switches it off.	5 years if training is on; 30 days if opted out.	None. No SSO, no admin console, no DPA.	Personal experimentation only.
Pro / Max (personal)	Consumer — legally identical to Free. Paying changes usage limits, not data posture.	Same as Free.	None. The employer has no visibility whatsoever.	Individual productivity on non-sensitive material.
Team (2–150 seats)	Commercial. No model training on your content by default.	Standard retention; not configurable.	SSO, domain verification, central billing, connector controls, usage analytics, org-wide skills.	Departments and mid-size firms without hard compliance obligations.
Enterprise (20+ seats)	Commercial. No training by default.	Custom retention controls, customer-managed encryption keys, US-only inference.	Everything in Team plus SCIM, role-based access, audit logs, Compliance API, IP allowlisting, network-level access control, HIPAA-ready with BAA.	Regulated industries and organisation-wide rollout.

API (Claude Platform)	Commercial. No training by default.	Deleted within 30 days by default; Zero Data Retention available by agreement, per organisation.	Workspaces, spend controls, API keys. Governance above the API call is yours to build.	Products, agents and automation you own.
Bedrock / Vertex / Foundry	Commercial, through your cloud provider.	Retained data stays inside your cloud provider's environment.	Inherits the cloud controls and paperwork you already run.	Data-residency requirements and existing cloud commitments.

Also available: Claude Gov for national-security work, and an Education plan for universities — same logic, different paperwork. Commercial tiers sit on SOC 2 Type II, ISO 27001 and ISO 42001, with GDPR and CCPA coverage.

Three things leaders get wrong

1. **Paying is not protecting.** A Pro or Max subscription runs on the same consumer contract as the free tier. No data processing agreement, no admin trail, no way to revoke access when someone leaves. A personal plan used for board papers is an unmanaged vendor relationship.
2. **Zero retention is not zero.** ZDR is negotiated per organisation, not toggled on. Since mid-2026, the most capable frontier models require 30-day retention on every platform, including under ZDR. Content flagged for policy review can be held far longer.
3. **Audit logs are not transcripts.** Enterprise logs record who did what and when — not what was typed. If your incident-response plan assumes a conversation can be reconstructed afterwards, it needs revisiting.

The practical rule

Match the tier to the data, not the headcount. Public and internal-general material: Team is sufficient. Customer, financial or employee data: Enterprise, with retention controls and role-based access. Regulated or health data: Enterprise with a BAA, or the API inside your own cloud boundary.

And whichever tier you buy, the more urgent control is usually the one that stops staff reaching for a personal account — which is precisely what Enterprise network-level access control exists to do. Shadow usage is the real exposure; the licence is only the fix on paper.

A final point worth holding onto: the safest deployment is not the most restrictive one. It is the one people actually use. Lock a tier down past usefulness and the work simply moves to somebody's phone.

AI SAFETY · SAME MODEL, SIX POSTURES

“Is it safe?” is the wrong question: the contract decides, not the model

Training, retention and IT visibility change by tier. That makes this a procurement and governance decision, not a technology one.

TIER	TRAINING	RETENTION	ADMIN & GOVERNANCE	RIGHT FOR
CONSUMER TERMS Free, Pro and Max				
Free	Trains on chats unless switched off	5 years if training on; 30 days if off	None: no SSO, admin console or DPA	Personal experiments only
Pro / Max	Consumer, legally identical to Free	Same as Free	None: employer has no visibility	Non-sensitive solo work
COMMERCIAL TERMS Team, Enterprise, API and cloud marketplaces				
Team	No training by default	Standard, not configurable	SSO, central billing, connector controls, analytics	Departments, mid-size firms
Enterprise	No training by default	Custom retention, own keys, US-only inference	Team + SCIM, RBAC, audit logs, Compliance API, BAA	Regulated, org-wide rollout
API	No training by default	30 days; zero retention by agreement	Workspaces, spend controls; the rest is yours	Products and agents you own
Bedrock / Vertex / Foundry	Via your cloud provider	Stays in your cloud	Inherits your cloud controls	Data residency, cloud commitments

Also: Claude Gov and Education plans. Commercial tiers hold SOC 2 Type II, ISO 27001 and ISO 42001, with GDPR and CCPA coverage.

THREE THINGS LEADERS GET WRONG

Paying is not protecting

Pro and Max run on the Free contract: no DPA, no admin trail, no offboarding.

Zero retention is not zero

ZDR is negotiated per organisation. Frontier models now need 30-day retention, even under ZDR.

Audit logs are not transcripts

Enterprise logs show who did what and when, not what was typed.

THE PRACTICAL RULE Match the tier to the data, not the headcount

PUBLIC AND INTERNAL-GENERAL
→ Team
is sufficient

CUSTOMER, FINANCIAL, EMPLOYEE
→ Enterprise
with retention controls and RBAC

REGULATED OR HEALTH
→ Enterprise + BAA
or the API inside your own cloud

Shadow usage is the real exposure. The safest deployment is the one people actually use.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Part 5 — Architecture: What Leaders are Being Sold

Article – Skills

Most teams' first encounter with GenAI is a single, carefully-written prompt — a good one, refined over a few tries, that produces exactly the output someone needed. The problem shows up a week later, when a colleague needs the same thing and has to reconstruct that prompt from memory, or when the original author leaves and takes the phrasing with them. A good prompt solves a moment. It doesn't solve a workflow.

Skills exist to close that gap.

What a skill actually is

A skill is a packaged, reusable set of instructions — the context, format, steps, and constraints a task needs — saved once and invoked repeatedly, instead of rebuilt from scratch in every conversation. Where a prompt lives in one chat and dies when that chat ends, a skill lives independently of any single conversation: it can be called up by name, shared across a team, and updated centrally when the way the work is done changes.

The practical shift is from "ask well, every time" to "ask once, define well, reuse forever."

Why this distinction matters more than it sounds

Three failure modes show up constantly in teams that rely purely on ad hoc prompting:

- **Inconsistency across people.** Two analysts producing the "same" weekly report end up with different structures, different tone, different level of detail — because each wrote their own prompt, and no one's prompt matches anyone else's exactly.
- **Knowledge walking out the door.** The person who figured out the right way to ask for a particular analysis is also the only one who remembers how they asked for it. Nothing is written down in a form anyone else can run.
- **Drift over time.** Even one person's own prompting style shifts week to week — small wording changes produce small output changes, and six months in, nobody can say why this month's version of a recurring report looks slightly different from January's.

A skill fixes all three at once: the instructions exist in one place, everyone invoking the skill gets the same underlying logic, and updating the skill updates the behavior for everyone who uses it going forward.

What belongs in a skill vs. what doesn't

Not everything needs to become one. The useful dividing line is repetition and stakes, not complexity:

- **Good candidates:** a recurring report format, a review checklist applied every time a certain document type comes in, a company-specific way of structuring an analysis, a workflow with several fixed steps done the same way each time.
- **Poor candidates:** a one-off question, an exploratory brainstorm, anything where the "right" approach is still being figured out. Packaging something into a skill before the underlying process has stabilized just locks in an immature workflow.

The test worth applying: if you've written a similar prompt three or more times with only small variations, that's no longer a prompt — it's an unwritten skill waiting to be formalized.

How skills change the leadership conversation, not just the individual one

For an individual, a skill saves re-typing. For an organization, it does something more structural — it turns tribal knowledge about "how we do this task well" into an asset that survives turnover, scales across a team without retraining everyone from zero, and can be improved deliberately rather than left to drift person by person.

This also reframes where leadership attention should go. Instead of asking "is our team good at prompting?" — a skill that's hard to audit and inconsistent by nature — the better question becomes "which of our recurring tasks have a defined, reusable skill behind them, and which are still running on whatever the individual remembers to type?" The gap between those two numbers is usually a more honest measure of AI maturity than adoption or license counts.

The underlying point

Prompting is a conversation. A skill is infrastructure. Teams that stop at good prompting get good individual moments that don't compound. Teams that convert their best, most-repeated prompts into skills get a capability that keeps paying off — consistent, shareable, and improvable — long after the person who wrote the original prompt has moved on to something else.

GENAI CAPABILITY · SKILLS

A good prompt solves a moment. A skill solves the workflow.

Prompting is a conversation. A skill is infrastructure: saved once, called by name, shared across the team and improved centrally.

Prompt

Ask well, every time

LIVES IN	One chat, gone when it ends
SHARED BY	Whoever remembers the wording
CHANGES	Drifts week to week
RESULT	A moment that doesn't compound

Skill

Define once, reuse forever

LIVES IN	Saved once, called by name
SHARED BY	The whole team, same logic
CHANGES	Updated centrally for everyone
RESULT	A capability that compounds

WHY AD HOC PROMPTING BREAKS

Three failures a skill fixes at once

Inconsistency

Two analysts, two prompts: different structure, tone and detail in the "same" report.

Knowledge walks out

The person who found the right phrasing is the only one who can run it again.

Drift over time

Small wording changes add up. Six months on, nobody can say why the report changed.

WHAT BELONGS IN A SKILL

Repetition and stakes, not complexity

Good candidates

- Recurring report formats
- Review checklists for a document type
- Your company's way of structuring an analysis
- Fixed multi-step workflows

Poor candidates

- One-off questions
- Exploratory brainstorms
- Anything still being figured out: a skill would lock in an immature workflow

3× **Written a similar prompt three times?** It's an unwritten skill. The honest maturity measure is how many recurring tasks have one, not how many licences you hold.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Customize

Skills | Connectors | Plugins | Yours | Discover | Search | Add

Most installed skills

Show all →

Design
Accelerate design workflows — critique, design system management, UX writing,...
by Anthropic · 2.7M installs
Try +

Engineering
Streamline engineering workflows — standups, code review, architecture...
by Anthropic · 2M installs
Try +

Finance
Streamline finance and accounting workflows, from journal entries and...
by Anthropic · 1.8M installs
Try +

Postiz
Social media automation CLI for scheduling posts, managing integrations, uploading...
by Nevo David · 1.8M installs
Try +

PDF Viewer
View, annotate, and sign PDFs in a live interactive viewer. Mark up contracts, fill...
by Anthropic · 1.6M installs
Try +

Sales
Prep calls, summarize and follow up, research accounts, review deals and...
by Anthropic · 1.4M installs
Try +

Categories

Show all 55 ⌄

Engineering	1,158	Code	1,058	Files & documents	758
Designs & media	405	Pipelines & jobs	369	Tickets & tasks	338
APIs	334	Notes & memory	305	AI models & agents	303

Article – API vs MCP

The Strategic Picture

For the last two years, "integrating AI" meant wiring point-to-point API calls between a model and each individual system — one custom connector per tool, per data source, per workflow. That approach doesn't scale. As enterprises move from single AI features to AI agents that need to act across many systems, a new standard has emerged: **MCP (Model Context Protocol)**, an open standard for connecting AI models to tools and data sources.

The API approach is direct, familiar, and battle-tested. Your team writes custom code to call a model, and separately writes custom code to call each business system (CRM, ERP, ticketing, calendars). Every new integration is a new engineering project. It's fully controllable, but the effort compounds linearly — 10 tools means 10 bespoke integrations, each maintained separately.

The MCP approach flips this. Instead of the model's developer writing custom glue code for every tool, systems expose themselves once via an MCP server, and any MCP-compatible AI can discover and use them — no bespoke integration per pairing. Think of it as the difference between building a custom cable for every device versus adopting USB-C: the standard, not the connection, does the work.

Three Implications for Leaders

1. Integration cost inverts. With APIs, cost scales with the number of tool-model pairings. With MCP, a tool builds one server once and every compatible AI system can use it. This matters most for organizations expecting to run multiple AI tools (chat, coding agents, automation) against the same internal systems.

2. Governance shifts from code review to access control. APIs bake permissions into custom code, reviewed case-by-case. MCP servers centralize what a model is allowed to see and do — read-only vs. write access, which systems, which data — making it a policy decision rather than an engineering one. This is easier to audit but requires deliberate access design; a poorly scoped MCP server can expose more than intended.

3. Vendor lock-in decreases, but standardization risk increases. MCP is open and model-agnostic, so switching AI providers doesn't mean rebuilding every integration. The trade-off: the standard is still young, evolving quickly, and not every enterprise system has a mature MCP server yet — some gaps still need direct API work.

The Bottom Line

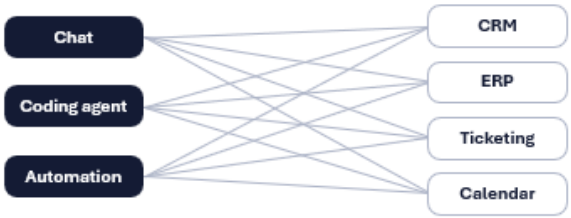
APIs aren't going away — they're the plumbing MCP servers are often built on top of. The leadership decision isn't "API or MCP," it's **where to invest integration effort**: build custom API integrations for one-off, high-control needs, and adopt MCP for anything you expect multiple AI tools to need repeated access to over time.

INTEGRATION · API VS MCP

Point-to-point APIs don't scale to agents. MCP builds each connection once.

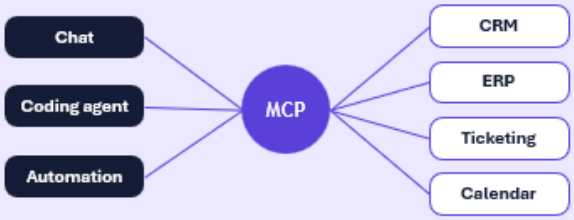
MCP (Model Context Protocol) is an open standard for connecting AI models to tools and data. Think USB-C, not a custom cable for every device.

API point to point



Every tool-model pairing is its own engineering project. 10 tools means 10 integrations, each maintained separately.

MCP build once, reuse everywhere



Each system exposes one MCP server. Any MCP-compatible AI can discover and use it, with no glue code per pairing.

THREE IMPLICATIONS FOR LEADERS

01

Integration cost inverts

With APIs, cost grows with every tool-model pairing. With MCP, a tool builds one server and every compatible AI uses it.

02

Access control, not code

One MCP server sets what a model can see and do: read or write, which systems, which data. Scope it badly and it exposes more.

03

Lock-in falls, risk shifts

Switch AI providers without rebuilding integrations. But MCP is evolving fast, and some systems still need direct API work.

THE BOTTOM LINE

The decision is where to invest integration effort, not API or MCP.

BUILD A CUSTOM API

One-off, high-control needs

ADOPT MCP

Anything several AI tools will need repeated access to over time

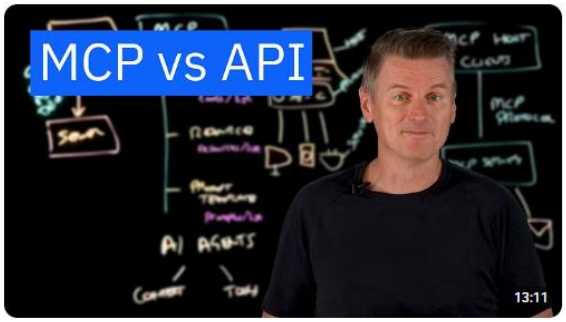
APIs aren't going away. They are the plumbing most MCP servers are built on.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.



MCP vs API: Simplifying AI Agent Integration with External Data

12 lakh • 1 yr ago

IBM Technology

Ready to become a certified Solution Implementer? Register now and use code IBMTechYT20 for 20% off of your exam ...

Subtitles

4 chapters Introduction | What is MCP | Capabilities of MCP | APIs

https://youtu.be/7j1t3UZA1TY?si=Sm1HUbVfSB7nPWrc

Article – Why Agents Matter

Chatbots respond. Agents execute.

A conversational model takes a prompt and returns a completion — a single inference pass, no persistent state, no follow-through. An agent operates in a loop: it decomposes a goal into sub-tasks, selects and calls tools (APIs, functions, databases), observes the result of each action, and decides the next step based on that observation. This loop — often called the **plan-act-observe** cycle — continues autonomously until a termination condition is met (task complete, error threshold hit, or human checkpoint reached).

What's structurally different

- **Tool use / function calling**: Agents can invoke external tools — a SQL query, a web search, an API call, a file write — and incorporate the returned data into their next reasoning step. This is what separates "describing an action" from "taking one."
- **Memory and state**: Agents maintain context across multiple steps (and sometimes across sessions), allowing them to track partial progress on a multi-stage task rather than treating each interaction as stateless.
- **Orchestration**: Complex tasks are often handled by multiple agents with defined roles (a "router" agent that delegates, specialist sub-agents that execute), coordinated by an orchestration layer — this is the architecture behind most production agentic systems today (e.g., Claude Code's sub-agent pattern, or multi-agent frameworks like LangGraph, AutoGen).
- **Self-correction**: Agents can evaluate their own output against a goal or spec (sometimes via a separate "critic" pass) and retry or revise before returning a final result, reducing the need for human review at every step.

Why this matters for business

1. End-to-end task completion, not single-turn output.
Real workflows are multi-step DAGs (directed acyclic graphs) of dependent actions — extract data, transform it, validate it, generate output, distribute it. Agents can traverse this entire graph autonomously. A standard LLM completion handles one node; a human has to manually chain the rest.

2. Reduced orchestration overhead.
The dominant cost in current GenAI deployments isn't inference — it's the human effort of prompt-chaining, context-passing, and manually verifying intermediate outputs. Agentic architectures internalize this orchestration, which is where the productivity gain actually materializes.

3. Native tool integration via APIs and connectors.
Through protocols like **MCP (Model Context Protocol)** or custom function-calling schemas, agents can read/write to CRMs, databases, calendars, and internal systems directly — closing the loop from "recommend an action" to "execute the action" with an audit trail.

4. Encoded, repeatable domain logic.
An agent can be configured (via system prompts, tool access, and guardrails) to replicate a specific

institutional workflow — reconciliation logic, screening criteria, escalation rules — and apply it consistently at volume, without re-specifying the logic each time.

5. A shift in governance surface area.
The core leadership question moves from output quality ("is this answer good?") to **action authorization** ("should this system be allowed to execute this action without a human in the loop?"). This requires defining: which tool calls are auto-approved, which require human-in-the-loop (HITL) confirmation, and what rollback mechanisms exist if an agent takes a wrong action mid-sequence.

The risk

Autonomy is a multiplier on both value and error propagation. A single bad completion from a chatbot is one wrong answer, contained and visible. A bad decision inside an agentic loop can cascade — an incorrect intermediate output becomes the input to the next tool call, compounding before a human ever sees it. This makes guardrails (scoped permissions, action allowlists, confirmation gates for irreversible actions, logging/observability) a precondition for deployment, not a nice-to-have layered on afterward.

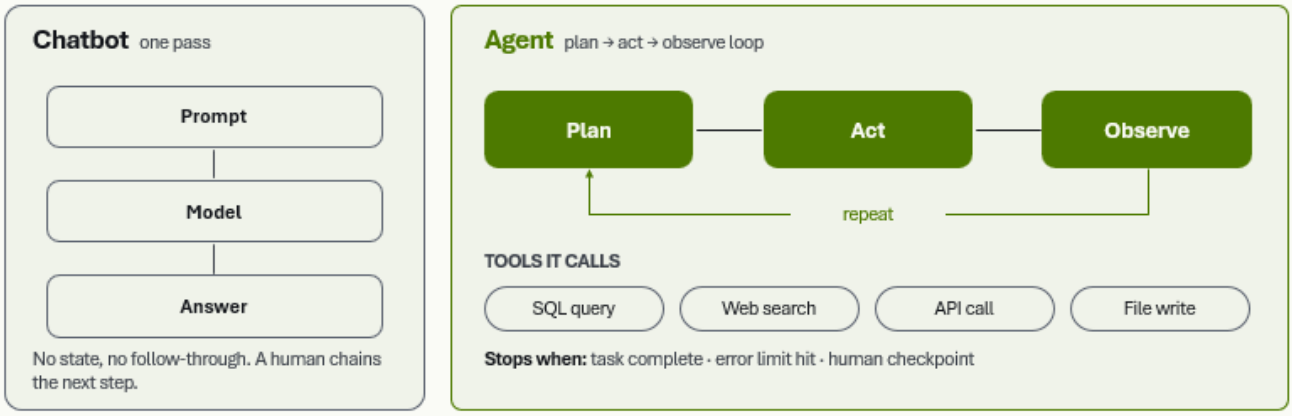
Bottom line

Chatbots are a completion engine. Agents are an execution engine — with tool access, state, and autonomy. The adoption question isn't whether to use agents; it's how fast an organization can build the permissioning and observability infrastructure to run them safely at scale.

AGENTIC AI // WHY AGENTS MATTER

Chatbots respond. Agents execute.

A chatbot returns one completion. An agent runs a loop, plan, act, observe, until the task is done, an error limit is hit or a human checkpoint is reached.



WHAT'S STRUCTURALLY DIFFERENT

Tool use
Calls SQL, search, APIs and files, then reasons on the result.

Memory and state
Tracks partial progress across steps, sometimes sessions.

Orchestration
A router agent hands work to specialist sub-agents.

Self-correction
Checks output against the spec and retries before returning.

WHY IT MATTERS FOR BUSINESS

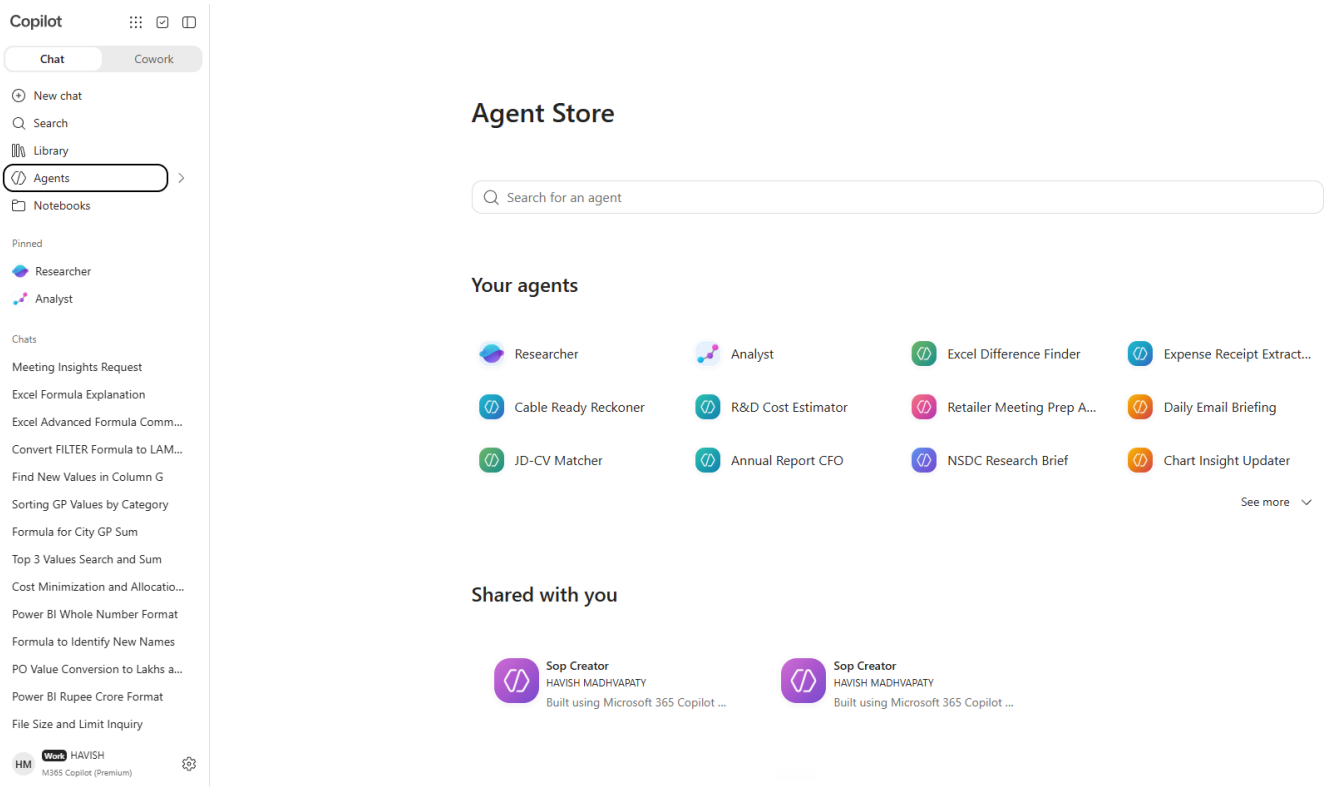
01	**End-to-end tasks** runs the whole extract-to-distribute chain, not one step
02	**Less orchestration** the human prompt-chaining effort moves inside the agent
03	**Native tools** reads and writes CRMs and databases via MCP, with an audit trail
04	**Repeatable logic** applies reconciliation or screening rules the same way at volume
05	**New governance** the question shifts from "is the answer good?" to "may it act?"

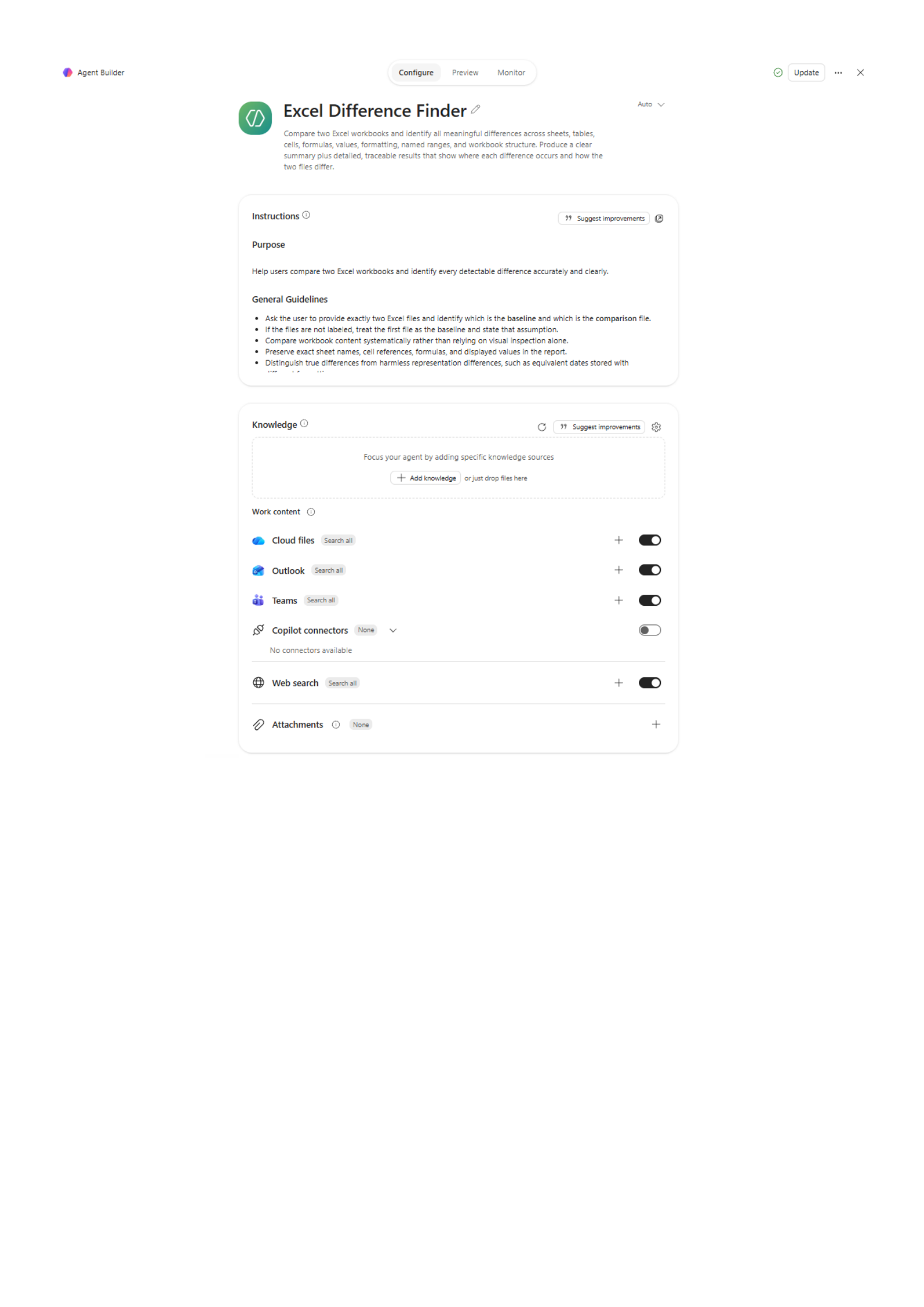
THE RISK **Autonomy multiplies value and errors alike.**
One wrong step becomes the next tool call's input and compounds before a human sees it. Scoped permissions, action allowlists, confirmation gates and logging come before deployment.

The question isn't whether to use agents. It's how fast you can build the permissions and observability to run them safely.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.


Copilot
Chat
Cowork
New chat
Search
Library
Agents
Notebooks
Pinned
Researcher
Analyst
Chats
Meeting Insights Request
Excel Formula Explanation
Excel Advanced Formula Comm...
Convert FILTER Formula to LAM...
Find New Values in Column G
Sorting GP Values by Category
Formula for City GP Sum
Top 3 Values Search and Sum
Cost Minimization and Allocatio...
Power BI Whole Number Format
Formula to Identify New Names
PO Value Conversion to Lakhs a...
Power BI Rupee Crore Format
File Size and Limit Inquiry
HM
Work HAVISH
M365 Copilot (Premium)
Agent Store
Search for an agent
Your agents
Researcher
Analyst
Excel Difference Finder
Expense Receipt Extract...
Cable Ready Reckoner
R&D Cost Estimator
Retailer Meeting Prep A...
Daily Email Briefing
JD-CV Matcher
Annual Report CFO
NSDC Research Brief
Chart Insight Updater
See more
Shared with you
Sop Creator
HAVISH MADHVAPATY
Built using Microsoft 365 Copilot ...
Sop Creator
HAVISH MADHVAPATY
Built using Microsoft 365 Copilot ...


Agent Builder
Configure
Preview
Monitor
Update
Excel Difference Finder
Auto
Compare two Excel workbooks and identify all meaningful differences across sheets, tables, cells, formulas, values, formatting, named ranges, and workbook structure. Produce a clear summary plus detailed, traceable results that show where each difference occurs and how the two files differ.
Instructions
Suggest improvements
Purpose
Help users compare two Excel workbooks and identify every detectable difference accurately and clearly.
General Guidelines
Ask the user to provide exactly two Excel files and identify which is the baseline and which is the comparison file.
If the files are not labeled, treat the first file as the baseline and state that assumption.
Compare workbook content systematically rather than relying on visual inspection alone.
Preserve exact sheet names, cell references, formulas, and displayed values in the report.
Distinguish true differences from harmless representation differences, such as equivalent dates stored with
Knowledge
Suggest improvements
Focus your agent by adding specific knowledge sources
Add knowledge
or just drop files here
Work content
Cloud files
Search all
Outlook
Search all
Teams
Search all
Copilot connectors
None
No connectors available
Web search
Search all
Attachments
None

Article – Agent vs Agentic AI

Agents vs Agentic AI: What Leaders Actually Need to Distinguish

Most enterprise AI conversations in 2026 use "agent" and "agentic AI" interchangeably. They are not the same thing, and the difference determines what you buy, what you govern, and what you can realistically deploy this year.

The distinction

An *agent* is a thing. A model given a defined task, a set of tools, and instructions on how to use them. It drafts the variance commentary, reconciles the invoice batch, screens the CVs.

Agentic is a property. It describes how much of the decision-making the system does for itself. An agentic system is given an outcome rather than a task, and decides its own sequence of steps to get there, correcting as it goes.

Put simply: agent is the noun, agentic is the adjective. The variable that separates them is **who decides the next step** — your process designer, or the system.

The autonomy spectrum

Level 0 — Assistant. You prompt, it responds. You do every step.
Level 1 — Agent. Given a task, it executes a largely fixed path using defined tools.
Level 2 — Agentic. Given a goal, it plans its own path, retries, and decides when the work is done.
Level 3 — Multi-agent. Several agentic systems delegate to and hand off to each other.

Nearly all enterprise value being captured today sits at Level 1. Nearly all vendor marketing describes Level 2 and 3.

Side by side

	Agent (Level 1)	**Agentic (Level 2–3)**
You supply	A task	An outcome
Sequence decided by	Your designer	The system
Memory	Usually per-run	Persistent across steps
Failure mode	Visible, contained	Compounding across steps
Oversight	Check the output	Check the reasoning and every action
Time to value	Weeks	Quarters
Best fit	High-volume, repeatable, bounded	Variable-path, exploratory, iterative

Five questions to ask any vendor

1. Who decides the order of steps — your system, or our workflow designer?
2. What can it change in our systems without a human approving first?
3. Does it carry state across steps, or is each call independent?
4. If it fails at step four of nine, what happens to steps one to three?
5. Does the audit trail show only the output, or the reasoning and every tool call?

If the answers to 1 and 3 are "our designer" and "independent," you are buying an agent with agentic branding. That is often the right purchase — but price and govern it accordingly.

The failure profiles are different, and this is the real issue

An agent fails cheaply. One task, one bad output, one human catches it.

An agentic system fails expensively. An error at step two is not caught — it becomes the input to step three, and the system confidently builds seven more steps on top of it. The output looks complete and internally consistent. That is precisely what makes it hard to catch in review.

This is why agentic deployments demand a different control model. You are no longer reviewing outputs. You are reviewing decisions.

The governance rule: autonomy proportional to reversibility

Do not scale autonomy by confidence in the model. Scale it by the cost of being wrong.

- Reversible, internal, low value → let it run, sample the output.
- Reversible but externally visible → human approves before it leaves the building.
- Irreversible, financial, or contractual → the system proposes, a named human commits.

Most organisations invert this. They pilot agentic AI on the highest-stakes process because that is where the ROI story is, then withdraw the programme when it fails once. Sequence it the other way.

What to do

Start at Level 1 and earn your way up. Deploy bounded agents against processes you have already measured — time taken, error rate, cost per cycle, captured before go-live, one process at a time. Re-run the same numbers at 90 days. Only then consider handing over path decisions.

Assess readiness before autonomy, not after. Team readiness, not model capability, is what breaks agentic pilots. A team that cannot yet review an agent's output has no business reviewing an agentic system's reasoning.

And be direct about the shape of the portfolio. In year one, most enterprises need many narrow agents and one or two carefully scoped agentic pilots. The reverse looks more ambitious in a board deck and delivers materially less.

Bottom line

Agents automate tasks. Agentic AI automates judgement about tasks. The first is a productivity decision. The second is a governance decision. Treat them as the same purchase and you will over-buy autonomy, under-build controls, and stall at the pilot.

AGENTIC AI · AGENT VS AGENTIC

An agent is a thing. Agentic is a property.

Agent is the noun, agentic the adjective. What separates them is **who decides the next step**: your process designer, or the system.

THE AUTONOMY SPECTRUM

Where most vendor marketing sits

Where value is captured today

LEVEL 0 Assistant
You prompt, it responds. You do every step.

LEVEL 1 Agent
Given a task, runs a largely fixed path with defined tools.

LEVEL 2 Agentic
Given a goal, plans its own path, retries, decides when done.

LEVEL 3 Multi-agent
Several agentic systems delegate and hand off to each other.

SIDE BY SIDE

	Agent · Level 1	Agentic · Level 2–3
You supply	A task	An outcome
Sequence decided by	Your designer	The system
Memory	Usually per run	Persistent across steps
Failure mode	Visible, contained	Compounds across steps
Oversight	Check the output	Check the reasoning and every action
Time to value	Weeks	Quarters
Best fit	High-volume, repeatable, bounded	Variable-path, exploratory, iterative

FIVE QUESTIONS TO ASK ANY VENDOR

01 Who decides the order of steps: you or the system?
02 What can it change without human approval?
03 Does it carry state, or is each call independent?
04 If step 4 of 9 fails, what happens to steps 1–3?
05 Does the audit trail show reasoning and every tool call?

SCALE AUTONOMY BY THE COST OF BEING WRONG

Reversible, internal, low value
Let it run, sample the output

Reversible but externally visible
A human approves before it leaves the building

Irreversible, financial or contractual
The system proposes, a named human commits

Agents automate tasks. Agentic AI automates judgement about tasks.

Start at Level 1 on processes you have already measured, re-run the numbers at 90 days, then hand over path decisions. Year one: many narrow agents, one or two scoped agentic pilots.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – Skills vs MCP vs RAG vs Memory

Skills vs MCP vs RAG vs Memory: The Four Layers

These four terms dominate enterprise AI architecture discussions, and they are routinely presented as competing options. They are not alternatives. They answer four different questions, and a serious deployment usually needs all four.

The four questions

RAG — *What does the organisation know?* Retrieval-augmented generation pulls relevant passages from your documents at the moment of the question and hands them to the model. Read-only, corpus-bound.

MCP — *What can it reach and do?* The Model Context Protocol is the plumbing that connects a model to live systems: your CRM, your ticketing tool, your file store. It reads current state and, if permitted, changes it.

Skills — *How do we do this here?* A packaged procedure the model loads when relevant. The steps, the format, the standard, the house style, the things a good analyst knows on their third month that a new one does not.

Memory — *What carries over?* What persists about a user or team across sessions, so context does not have to be re-supplied every time.

Knowledge, access, procedure, continuity. Four layers, four owners, four failure modes.

Side by side

	RAG	**MCP**	**Skills**	**Memory**
Answers	What's true	What it can reach	How we do it	What carries over
Direction	Read-only	Read and write	Instruction	Accumulates
Freshness	As fresh as the index	Live	As fresh as the author	Rolling
Typical owner	IT / knowledge management	IT / security	The function itself	Platform + policy
Effort to stand up	Project	Integration	Document	Setting plus policy

Highest risk	Permission leakage	State change, injection	Silent drift	Privacy, wrong recall

The diagnostic: when output is wrong, which layer failed?

- It invented facts or cited the wrong document → **RAG**. Retrieval, not the model.
- The facts were stale, or it could describe the action but not take it → **MCP**.
- The facts were right but the format, sequence, or standard was wrong → **Skills**.
- It was correct, but you had to re-explain context you had already given twice → **Memory**.

Most enterprises diagnose all four as "the model isn't good enough" and go shopping for a better model. That fixes none of them.

Where the money usually goes wrong

The default enterprise instinct is to start with RAG — "build us a bot that knows our documents." It is the most expensive layer per unit of value in year one. It requires a pipeline, chunking decisions, an evaluation set, and permission inheritance that mirrors your existing access controls. Done badly, it produces a system that confidently answers from the wrong version of a policy, or shows a junior the compensation deck.

Skills are the cheapest layer and the most under-invested. Encoding how your finance team actually writes variance commentary — the sequence, the thresholds, the tone, the exclusions — is a document, not a project. It usually delivers more visible improvement in the first month than a retrieval build delivers in six.

MCP is the highest-leverage and highest-risk layer, because it is the only one that changes something. Read-only connections first. Write access earned, scoped, and logged.

Memory is the layer to leave largely alone at the start. It accrues naturally, and the governance question — what the system is allowed to retain about employees and clients, and for how long — should be settled by policy before it is settled by default.

Three questions to ask any vendor

1. When it answers a question, is it retrieving from a snapshot or reading live? If a snapshot, how old, and who refreshes it?
2. Does it inherit our existing permissions, or does it have its own access model we now have to maintain separately?
3. What can it write to, and what is the log of every write?

The second question is the one most often skipped and most often regretted. A parallel permission model is a permanent operational liability.

The governance rule: separate what it knows from what it can do

Knowledge access and action authority are different grants and should be approved by different people. A system that can read every contract is a confidentiality question. A system that can amend one is a controls question. Bundling them into a single "AI access" approval is how organisations end up with capabilities nobody deliberately signed off on.

What to do

Sequence it: Skills, then read-only MCP, then RAG, then write access, with memory governed by policy from day one. Assign an owner per layer — the function owns its skills, IT owns access, and neither should own both.

And when the next architecture proposal arrives, ask which of the four questions it answers. If the answer is "all of them," it answers none of them well.

Bottom line

RAG is what it knows. MCP is what it can touch. Skills are how it works. Memory is what it retains. Confuse them and you will fund the expensive layer, skip the cheap one, and grant the risky one by accident.

AI ARCHITECTURE · PART 1 OF 2

Four layers, not four options

Skills, MCP, RAG and Memory are often pitched as alternatives. They answer **four different questions**, and each has its own owner and failure mode.

RAG

What does the organisation know?

Pulls passages from your documents at the moment of the question.

DIRECTION
Read-only

FRESHNESS
As fresh as the index

OWNER
IT / knowledge mgmt

EFFORT
A project

HIGHEST RISK
Permission leakage

MCP

What can it reach and do?

Connects the model to live systems: your CRM, tickets and files.

DIRECTION
Read and write

FRESHNESS
Live

OWNER
IT / security

EFFORT
An integration

HIGHEST RISK
State change, injection

Skills

How do we do this here?

A packaged procedure: the steps, format, standard and house style.

DIRECTION
Instruction

FRESHNESS
As fresh as the author

OWNER
The function itself

EFFORT
A document

HIGHEST RISK
Silent drift

Memory

What carries over?

What persists about a user or team across sessions.

DIRECTION
Accumulates

FRESHNESS
Rolling

OWNER
Platform + policy

EFFORT
Setting plus policy

HIGHEST RISK
Privacy, wrong recall

RAG is what it knows. **MCP** is what it can touch.
Skills are how it works. **Memory** is what it retains.

Knowledge, access, procedure, continuity. A serious deployment usually needs all four.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

AI ARCHITECTURE · PART 2 OF 2

Fix the layer, not the model

Most enterprises blame every failure on the model and buy a better one. That fixes none of them.

WHEN THE OUTPUT IS WRONG, WHICH LAYER FAILED?

It invented facts or cited the wrong document	RAG · Retrieval
The facts were stale, or it couldn't take the action	MCP · Access
Facts right, but format, sequence or standard wrong	Skills · Procedure
Correct, but you had to re-explain context twice	Memory · Continuity

WHERE THE MONEY USUALLY GOES WRONG

RAG

Most expensive in year one

Needs a pipeline, chunking, an evaluation set and permissions that mirror yours.

MCP

Highest leverage, highest risk

The only layer that changes something. Read-only first; write access earned and logged.

Skills

Cheapest, most under-invested

A document, not a project. Often more visible gain in month one than RAG in six.

Memory

Leave largely alone at first

Settle by policy what it may keep about people, and for how long.

SEQUENCE IT cheapest and safest first

1 Skills → 2 MCP read-only → 3 RAG → 4 MCP write

Memory governed by policy from day one

ASK ANY VENDOR

01 Snapshot or live? If a snapshot, how old?

02 Does it inherit our permissions, or add its own?

03 What can it write to, and where is the log?

Separate what it knows from what it can do.

Reading a contract is a confidentiality question; amending one is a controls question. Approve them separately.

Ask which of the four questions a proposal answers. If the answer is "all of them", it answers none of them well.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Part 6 — For your Teams

Article – Why Everyone Must Learn Python

For years, learning to code was framed as a career choice — something you did if you wanted to become a software engineer. That framing is now outdated. In the age of AI, Python isn't a specialist skill anymore; it's a literacy — closer to how "using Excel" became a baseline business skill two decades ago. Here's why it matters for everyone, not just technical teams.

1. Python is the on-ramp to coding for non-coders

Python was designed to be readable — its syntax is closer to plain English than almost any other programming language. if revenue > target: print("bonus") reads almost like a sentence. This makes it the natural entry point for someone who has never written a line of code but wants to understand how logic, automation, and AI tools actually work under the hood. You don't need a computer science degree to get functional in Python — you need a few hours and the willingness to experiment.

2. It's the backend language of the AI world

Nearly every major AI framework and model — from TensorFlow and PyTorch to the libraries powering ChatGPT-style tools, data pipelines, and automation scripts — is built in or on top of Python. When you use an AI-powered dashboard, a chatbot, or a "no-code" automation tool, there's a very high chance Python is running underneath it. Understanding even basic Python means understanding the actual machinery of the tools you're now expected to use, adopt, and manage as a leader.

3. Interpretability — knowing what's actually happening

AI tools are increasingly treated as black boxes: you put in a prompt or a file, and something comes out. But when you understand the underlying code — even at a basic level — you gain the ability to *interpret* what's happening: why a model produced a certain output, what assumptions a script made, where a calculation could go wrong. This isn't about writing production code; it's about not being fully dependent on someone else's explanation of what a tool did. Interpretability is what separates informed oversight from blind trust.

4. It closes the "translation gap" between business and technical teams

One of the most common failure points in enterprise AI or data projects is the gap between what a business leader asks for and what a technical team builds — because neither side fully understands the other's constraints. Someone who knows even basic Python can read a script, spot a wrong assumption in logic, or ask a sharper question in a project review, without needing to become the one writing the code. This dramatically improves the quality of oversight and decision-making on any technical initiative.

5. It future-proofs decision-making, not just careers

As AI agents take on more autonomous, multi-step work, the people making decisions about what these systems are allowed to do need enough technical grounding to ask the right questions — about logic, safeguards, and failure points. Leaders who can't read a line of code are entirely dependent on someone else's interpretation of risk. Leaders with even basic Python fluency can push back, verify, and govern more effectively.

6. It builds "vibe coding" fluency — a new baseline skill

With AI-assisted coding tools, you no longer need to write every line yourself — you can describe what you want and iterate with an AI's help. But this only works well if you understand *just enough* Python to read what's being generated, catch errors, and guide the AI effectively. Without that baseline, you're vibe coding blind — accepting output you can't evaluate. With it, you become a far more capable collaborator with the AI itself.

The bottom line

Python isn't asking you to become a software engineer. It's asking you to stop being a passenger in a world increasingly run on code you don't understand. The people who invest even a modest amount of time in Python fluency will be the ones who can genuinely evaluate, question, and direct the AI systems reshaping their industries — rather than simply trusting whatever comes out the other end.

AI LITERACY · WHY EVERYONE SHOULD LEARN PYTHON

Python is the new Excel: a literacy, not a career choice

You don't need to become a software engineer. You need enough to **read, question and direct** the code your AI tools run on.

```
if revenue > target:
  print("bonus")
```

Reads almost like a sentence.
A few hours of practice gets a non-coder functional.

01 The on-ramp for non-coders

Syntax closer to plain English than almost any other language. No computer science degree needed.

02 The backend of the AI world

TensorFlow, PyTorch, data pipelines and most automation tools run on Python underneath.

03 Interpretability

See what a script assumed and where a calculation could go wrong: informed oversight, not blind trust.

04 Closes the translation gap

Read a script, spot a wrong assumption and ask sharper questions in a project review.

05 Future-proofs decisions

As agents act on their own, leaders need enough grounding to question logic, safeguards and failure points.

06 Vibe coding with eyes open

AI can write the code. You need enough Python to read it, catch errors and steer.

Python isn't asking you to become a software engineer. It's asking you to stop being a passenger.

A modest investment in Python fluency is what lets you evaluate, question and direct AI systems, rather than trusting whatever comes out the other end.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – Master Perplexity Research

Most people use Perplexity as a better Google. Type a question, read the answer, move on. That covers about ten percent of what it does.

The bigger use is different. Ask it to build a number, not find one.

Half of what an analyst needs was never published. Market size for a segment you define your own way. What a competitor spends on something they don't disclose. What a process costs at a company that never measured it.

Most people stop there and guess privately. It is better to make the tool guess out loud, show its arithmetic, and rate its own confidence.

Twenty tips, with the prompt wording for each.

Getting numbers that were never published

1. Ask for an estimate, not a fact.
Ask a hard question and Perplexity often replies that the figure is not publicly available. That is a dead end. Tell it upfront that an estimate is acceptable and it will build one.

Prompt
The exact figure is likely not published. Do not tell me it is unavailable. Build the best possible estimate from adjacent public data, state every assumption explicitly, and show the arithmetic.

2. Demand a confidence level on every number.
An estimate with no confidence marker looks exactly like a hard fact. That is how weak numbers reach board decks.

Prompt
For each figure, tag it High, Medium or Low confidence, and give a one-line reason for the grade. High = directly sourced. Medium = derived from sourced inputs. Low = inferred or analogous.

3. Ask for three independent approaches to the same number.
Top-down, bottom-up, analogue. If all three land in the same range, you can use the number. If they are far apart, the question is harder than you assumed.

Prompt
Estimate this three ways using independent methods — top-down from market size, bottom-up from unit economics, and by analogy to a comparable market. Show each separately, then compare. Where they disagree, explain why.

4. Ask for a range, not a single number.
One number implies a precision you do not have. A low case and a high case, each with its logic, holds up better when questioned.

Prompt
Give a plausible low, central and high case. For the low and high, state specifically what would have to be true for that case to hold.

5. Ask for the closest real number available.
Your exact metric may not exist, but something near it usually does. Get that figure and the gap between it and what you asked for.

Prompt
If this exact metric is not published, tell me the closest figure that is actually reported, cite it, and explain precisely how it differs from what I asked for.

6. Give it your own number and ask it to react.
The tool is better at checking a number than producing one from scratch. Put your assumption into the prompt.

Prompt
My working assumption is X. Assess whether that is plausible against available evidence. If it is wrong, tell me the direction and rough magnitude of the error.

7. Ask for the inputs and do the maths yourself.
For anything going to a client, source the inputs and run the calculation yourself. You can then defend every step.

Prompt
Do not calculate the final answer. Give me the three to five input variables this depends on, the current sourced value and date for each, and the formula that connects them.

Controlling what it searches

8. Use Academic mode when the claim must survive scrutiny. It limits results to peer-reviewed papers instead of blog posts and press releases. Use it when someone will ask where the number came from.

9. Use Social mode to find out what users actually say. It pulls from Reddit and similar sites. This is the fastest way to learn how a software product performs after month three, rather than how the vendor describes it.

10. Restrict the source type in the prompt.
Primary documents contain numbers that never appear in articles.

Prompt
Restrict yourself to primary sources — annual reports, regulatory filings, investor presentations, earnings call transcripts and government statistics. No news summaries, no aggregator sites, no vendor marketing pages.

11. Set a recency floor.
Perplexity will mix a 2019 figure with a 2026 figure and present both as current.

Prompt
Prioritise sources from the last 18 months. Label every figure with the year it refers to, not the year it was published. Flag anything older than three years explicitly.

12. Name the geography or you get US data.
Ask a market question without naming a country and you get American figures built on American assumptions. For Indian enterprise work, that error runs through everything after it.

Prompt
Answer for India specifically. Where only global or US data exists, say so, and state whether the Indian figure is likely to be higher or lower and roughly why.

Structuring the answer

13. Specify the format before you ask the question.
Otherwise you spend your time converting prose into a table.

Prompt
Return this as a table with columns: Metric, Value, Year, Source, Confidence. One row per metric. No introduction or conclusion.

14. Cap the length.
Left alone, it writes five hundred words of background you did not ask for.

Prompt
Maximum 150 words. Lead with the answer. No preamble, no summary of what I asked.

15. Ask where the sources disagree.
Perplexity blends everything it finds into one smooth answer. If two credible sources give different numbers, that gap disappears. The gap is often the most useful thing you will learn.

Prompt
Where do credible sources disagree on this? Show the range of published estimates, identify who produced each, and explain what methodological difference accounts for the spread.

16. Ask what would change the answer.
This turns a fixed number into something you can monitor.

Prompt
List the three developments that would most change this conclusion over the next twelve months, and what the leading indicator for each would be.

Catching it when it is wrong

17. Check the citations in a separate prompt.
Perplexity rarely invents a source. The real problem is a genuine source that does not actually say what the answer claims. Ask in a new thread, not the same one.

Prompt
For each numbered claim above, quote the specific line from the cited source that supports it. If no line directly supports the claim, mark it Unsupported.

18. Ask for the opposite case.
Search results favour whoever ranks highest, which is usually the vendor spending most on content. That bias carries straight into the answer.

Prompt
Now argue the opposite position using the strongest available evidence. What is the best case against what you just told me?

19. Use Model Council for decisions.
It runs one query through three frontier models at the same time and shows all three answers together, marking where they agree and where they differ. If the three disagree, the question is genuinely unsettled — not just poorly researched.

20. Ask the same question again in a clean thread.
A long thread picks up your framing and starts agreeing with you. If a number matters, ask it again from scratch, in different words, with no history. The same answer from two separate starting points is worth more than ten follow-up questions in one conversation.

Two workflow habits

Use a Space for recurring work. Set your source rules and output format once instead of retyping them. Anything you research monthly — a client's competitors, a sector's regulatory changes — belongs there.

Separate the two modes. Use fast search for questions that already have an answer. Use Deep Research for questions where the answer has to be built. Using the first for the second is why most people conclude the tool is shallow.

The point

Perplexity is not a source of truth. It is a fast research associate that delivers exactly the standard you ask for and nothing more.

Ask for a number and you get a number. Ask for a number, three ways of reaching it, a confidence grade, and the conditions that would make it wrong — and you get something you can defend in a room.

Most people never set the standard.

AI RESEARCH · MASTER PERPLEXITY · PART 1 OF 2

Ask it to build a number, not find one

Half of what an analyst needs was never published. Don't guess privately: make the tool **guess out loud, show its arithmetic and rate its own confidence.**

1 Ask for an estimate, not a fact

PROMPT *"The exact figure is likely not published. Don't tell me it is unavailable. Build the best estimate from adjacent public data, state every assumption and show the arithmetic."*

2 Demand a confidence level on every number

PROMPT *"Tag each figure High, Medium or Low confidence, with a one-line reason for the grade."*

3 Use three independent approaches

PROMPT *"Estimate this top-down, bottom-up and by analogy. Show each separately, compare them, and explain where they disagree."*

4 Ask for a range, not a single number

PROMPT *"Give a low, central and high case. For the low and high, state what would have to be true."*

5 Ask for the closest real number

PROMPT *"If this metric isn't published, cite the closest reported figure and explain exactly how it differs."*

6 Give it your number to react to

PROMPT *"My working assumption is X. Is it plausible? If wrong, give the direction and rough size of the error."*

7 Get the inputs, do the maths yourself

PROMPT *"Don't calculate the answer. Give me the 3–5 inputs, each with a sourced value and date, and the formula that connects them."*

TAG EVERY NUMBER so a weak estimate never reaches a board deck looking like a fact

High	Directly sourced	Medium	Derived from sourced inputs	Low	Inferred or analogous

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

AI RESEARCH · MASTER PERPLEXITY · PART 2 OF 2

Set the standard, then check its work

Perplexity delivers exactly the standard you ask for and nothing more. Tips 8 to 20 raise it.

CONTROL WHAT IT SEARCHES

08 **Academic mode** peer-reviewed sources when the number must survive scrutiny

09 **Social mode** what users say after month three, not the vendor pitch

10 **Primary sources only** filings, annual reports, earnings calls, government data

11 **Set a recency floor** last 18 months; label each figure with the year it refers to

12 **Name the geography** or you get US data built on US assumptions

STRUCTURE THE ANSWER

13 **Format first** a table of Metric, Value, Year, Source, Confidence

14 **Cap the length** 150 words maximum, lead with the answer

15 **Ask where sources disagree** the spread is often the most useful finding

16 **Ask what would change it** three developments and a leading indicator for each

CATCH IT WHEN IT'S WRONG

17 **Check citations in a new thread** quote the supporting line, or mark it Unsupported

18 **Argue the opposite** the strongest case against what it just told you

19 **Use Model Council** three models, one query; disagreement means truly unsettled

20 **Ask again in a clean thread** the same answer from two starts beats ten follow-ups

TWO WORKFLOW HABITS

Use a Space for recurring work

Set your source rules and output format once. Anything you research monthly belongs there.

Separate fast search from Deep Research

Fast search when the answer exists. Deep Research when the answer has to be built.

Perplexity is not a source of truth. It is a fast research associate that delivers exactly the standard you ask for.

Ask for a number, three routes to it, a confidence grade and what would make it wrong, and you get something you can defend in a room. Most people never set the standard.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Article – Working with Unclean Excel Files

Once you can generate a working HTML dashboard from a prompt, the bottleneck stops being the code. It becomes the file someone drags into it.

Almost every failure we see in these builds traces back to a spreadsheet that was designed to be read by a human and is now being read by a parser. Humans skip the title row, ignore the blank line, understand that the bold row at the bottom is a total. Code does none of this.

Twenty things to get right, on both sides — the file and the parsing logic.

Layout

1. Data starts at A1. No company name in row 1, no logo, no "Report as of March 2025", no blank spacer row. The parser takes the first row it finds as the header. Give it the header.

2. No merged cells anywhere. A merged cell resolves to one value and a series of empty cells. Merged headers are the worst version of this, because the columns underneath end up unnamed and everything downstream is silently misaligned.

3. One table per sheet. Two tables side by side, or stacked with a gap, will be read as one rectangle with a lot of holes. If a sheet has a summary block above the detail, split them.

4. Name the sheet you want. Parsers default to the first sheet, which in most corporate workbooks is a cover page or an instructions tab. Read sheets by name in the code, and fail loudly with a list of available sheets when the name is missing.

5. Hidden and filtered rows still come through. Filters and hidden columns are a display state, not a data state. If someone filtered to Q4 before sending the file, the code still sees all four quarters. Delete, don't hide.

Headers

6. No two columns with the same heading. Most parsers map rows to objects keyed by header. Two columns called "Amount" means one of them silently overwrites the other, and the dashboard will look completely plausible while being wrong.

7. No blank headings. An unnamed column becomes __EMPTY, __EMPTY_1, and so on. Either name it or drop it before upload.

8. One header row, not two. Stacked headers — a merged category row above a detail row — are a human convention. Flatten them into single names like Q1_Revenue, Q1_Units before the file goes anywhere near the parser.

9. Trim and normalise in the code. "Region " and "Region" are different keys, and the trailing space is invisible to whoever created the file. Strip whitespace, standardise case, and build a mapping layer rather than trusting the header string exactly as typed.

Data types

10. Numbers stored as text. The most common single cause of a chart that renders with everything at zero, or a total that concatenates instead of adding. Coerce explicitly and check what fails rather than assuming the type.

11. Currency symbols and separators live inside the cell. ₹1,20,000 is a string. So is $4.5M. Format currency in the display layer, keep the underlying cell as a raw number.

12. Percentages are ambiguous. A cell showing 45% may hold 0.45 or 45 depending on how it was formatted. Decide the convention, state it, and validate against a known total.

13. Dates are the biggest trap. Excel stores them as serial numbers, so 45231 is a date that will parse as a number. Converting through a JavaScript date without controlling for timezone routinely shifts everything by one day. And 03/04/2025 is March or April depending on who made the file. Force an unambiguous format like YYYY-MM-DD at source.

14. Leading zeros disappear. Employee IDs, PIN codes, cost centre codes, account numbers. 007 becomes 7, and your join against another table quietly returns nothing. These are identifiers, not numbers — keep them as text.

Rows and content

15. No total or subtotal rows inside the data. A grand total row will be charted as a data point roughly twice the size of everything else. Totals are something the dashboard calculates, not something it inherits.

16. No blank rows in the middle, no footnotes at the bottom. * Figures exclude the Nagpur plant is not a data row, but it will become one. Strip trailing rows that don't match the column structure.

17. Blank is not zero. An empty cell and a 0 mean different things — one is "no sales", the other is "we don't know". Averages change materially depending on how you treat them. Decide deliberately, and say which you chose on the dashboard itself.

18. Category values must be consistent. North, north, North and NORTH will produce four bars in a chart that should have one. Normalise on read, and surface the distinct values so someone can spot the duplicates.

19. Never use a pivot table or a formula-heavy sheet as the source. Pivots are output, not data. Formula errors — #REF!, #N/A, #DIV/0! — flow straight through as strings and break every numeric operation after them. Paste as values into a clean sheet.

Practical

20. Reference columns by name, never by position. row[3] works until someone inserts a column, at which point the dashboard is wrong rather than broken — which is far more dangerous. Map by header name and throw an explicit error when an expected column is missing.

Two more worth building in as habit rather than rules. Browser-side parsing struggles well before you reach enterprise data volumes, so set a row limit and tell the user what it is instead of letting the tab freeze. And be deliberate about what goes into the file at all — a browser tool is exactly the wrong place to discover that a sheet has salary data or customer names in columns nobody was looking at.

The habit that prevents most of this

One sheet. One table. Starting at A1. One row of unique headers. No merges, no totals, no footnotes, no formatting doing the work of data.

Almost every rule above is a specific instance of that. Most teams get there faster by building a clean extract sheet next to the working file than by trying to reform the working file itself — the operational spreadsheet has a human audience and should keep it.

The parser gets its own copy.

AI BUILDS · WORKING WITH UNCLEAN EXCEL FILES · PART 1 OF 2

Your dashboard is only as clean as the file

Humans skip the title row, ignore the blank line and know the bold row is a total. **A parser does none of this.**

Built for a human

A	B	C	D
Sales Report: March 2025			1
Region	**Q1**		**Amount** 2
North	03/04/25	₹1,20,000	45%
TOTAL		**₹9,80,000**	15
* excl. Nagpur			16

Built for a parser

A	B	C	D
Region	**Date**	**Revenue**	**Margin**
North	2025-03-04	120000	0.45
South	2025-03-04	98500	0.38
East	2025-03-04	131200	0.41
West	2025-03-04	87400	0.36
...			

LAYOUT

01 **Data starts at A1** no company name, logo, "as of" line or spacer row

02 **No merged cells anywhere** merged headers leave the columns beneath unnamed

03 **One table per sheet** two blocks on a sheet read as one rectangle full of holes

04 **Name the sheet you want** the first tab is usually a cover page; fail loudly if missing

05 **Delete, don't hide** hidden and filtered rows still reach the parser

HEADERS

06 **No duplicate headings** two "Amount" columns: one silently overwrites the other

07 **No blank headings** unnamed columns become __EMPTY, __EMPTY_1...

08 **One header row, not two** flatten stacked headers into Q1_Revenue, Q1_Units

09 **Trim and normalise in code** "Region " and "Region" are different keys

DATA TYPES

10 **Numbers stored as text** the top cause of zeros in charts; coerce and check failures

11 **Symbols stay out of the cell** ₹1,20,000 is a string; format in the display layer

12 **Fix the percentage convention** 45% may hold 0.45 or 45; validate against a known total

13 **Dates in YYYY-MM-DD** serial numbers, timezones and 03/04 ambiguity all bite

14 **IDs are text, not numbers** 007 becomes 7 and the join quietly returns nothing

Red numbers on the left sheet match the rules that break. The right sheet is what the parser should receive: one table, one header row, raw values.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

AI BUILDS · WORKING WITH UNCLEAN EXCEL FILES · PART 2 OF 2

Totals are calculated, not inherited

The last six rules are about what sits inside the rows, and how the code reads them. **Wrong is more dangerous than broken.**

ROWS AND CONTENT

15 **No total or subtotal rows** a grand total charts as a bar twice the size of the rest

16 **No blank rows, no footnotes** "* excludes Nagpur" will become a data row

17 **Blank is not zero** "no sales" vs "we don't know"; state your choice on the dashboard

18 **Consistent category values** North, north and NORTH make three bars; normalise on read

19 **Never a pivot or formula sheet** #REF! and #N/A flow through as text; paste as values

PRACTICAL

20 **Reference columns by name** row[3] breaks silently the day someone inserts a column

\+ **Set a row limit** tell users the cap instead of letting the browser tab freeze

\+ **Check what is in the file** salary or customer data in unwatched columns has no place here

THE HABIT THAT PREVENTS MOST OF THIS

Almost every rule is a specific case of these six.

- ✓ One sheet
- ✓ One table
- ✓ Starting at A1
- ✓ One row of unique headers
- ✓ No merges, totals or footnotes
- ✓ No formatting doing the work of data

Build a clean extract sheet next to the working file.

Don't try to reform the operational spreadsheet: it has a human audience and should keep it.
Copy the values into a separate sheet built for code.

The parser gets its own copy.

HMC ONE

AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE

© Havish M Consulting and/or affiliates. All rights reserved.

Article – Safety of HTML

The business case

When you're building dashboards, reports, calculators, or internal utilities through AI-assisted ("vibe") coding, HTML is often the most practical output format — not because it's technically superior in every dimension, but because of what it means for deployment and adoption inside a business.

1. Zero-friction distribution. An HTML file opens in any browser — no installation, no license, no IT approval cycle. You can email it, put it on a shared drive, or drop it into a Teams channel, and the person on the other end just double-clicks and it works. Compare this to a Python script (needs an interpreter and dependencies) or a compiled app (needs OS compatibility, permissions, sometimes admin rights). For a consulting practice or an internal team rolling out a tool to dozens of non-technical stakeholders, this removes almost all deployment friction.

2. Self-contained and portable. A single HTML file can carry its own CSS, JavaScript, and even embedded data or charting libraries. There's nothing to "install" and nothing that breaks because a server went down or a subscription lapsed. This makes it ideal for calculators, one-off dashboards, or utilities that need to outlive a single project engagement — the client can keep using it long after your contract ends.

3. Fast iteration cycles. Because HTML/JS is quick to generate, test, and modify with AI assistance, you can go from "here's what I want" to a working prototype in the same meeting. This is a major advantage when you're co-designing a tool live with a business stakeholder — changes to layout, logic, or branding can be shown immediately.

4. No recurring cost or vendor lock-in. Unlike SaaS dashboard tools, an HTML deliverable has no subscription, no seat licensing, and no dependency on a third-party platform staying in business. For clients who want a tool they fully own, this matters commercially.

5. Branding and professionalism. HTML gives full control over look and feel — logos, color palettes, typography — so deliverables can be styled to match a client's or firm's brand identity, which matters when the output is client-facing.

On data safety: does uploading an Excel file into an HTML tool "go online"?

This depends entirely on *how* the HTML tool is built, and it's worth being precise here rather than making a blanket claim:

- **If the HTML file processes the upload entirely in the browser** (i.e., the JavaScript reads the Excel file locally using something like the SheetJS/xlsx library and does all calculations client-side), then the data never leaves the user's machine. Nothing is transmitted to a server, and there is no network call involved in reading or processing the file. In this case, it's accurate to say the data stays local and private.

If the HTML tool includes any server call — for example, it sends data to an API, calls a backend for AI processing, or logs data to an external service — then the file's contents (or parts of them) *are* going online, and this needs to be disclosed clearly to users.

AI BUILDS · THE SAFETY OF HTML

One HTML file reaches anyone. Where the data goes is the real question.

For AI-built dashboards, reports and calculators, HTML is the most practical format to ship: not because it is technically best, but because of how easily it gets adopted.

WHY HTML FOR VIBE-CODED TOOLS

01
Zero-friction
Opens in any browser. No install, licence or IT approval.

02
Self-contained
CSS, code, data and charts travel in one file.

03
Fast to iterate
Change layout or logic live, in the same meeting.

04
No lock-in
No subscription, no seats. The client owns it.

05
On brand
Full control of logo, colours and type.

DOES UPLOADING AN EXCEL FILE "GO ONLINE"? IT DEPENDS HOW THE TOOL IS BUILT.

Processed in the browser

Excel file .xlsx on your laptop → **Browser** reads and calculates → **Stays local** no network call

The JavaScript reads the file on the user's machine (e.g. with the SheetJS library) and does every calculation client-side.

Nothing is sent to a server.

Accurate to say: the data stays private

Any server call in the code

Excel file .xlsx on your laptop → **Browser** reads and calculates → **Server** API, AI backend or logging

If the tool sends data to an API, calls a backend for AI processing or logs to an external service, the file's contents, or parts of them, **go online.**

Disclose this clearly to every user

"It's HTML" is not a privacy guarantee.

Before an HTML tool goes to clients or colleagues, confirm whether any line of it sends data to a server. If none does, say the data stays on the device. If one does, say what is sent and where.

Be precise, not reassuring.

HMC ONE
AI STRATEGY. BUILT TO SCALE.

Source: HMC ONE
© Havish M Consulting and/or affiliates. All rights reserved.

Article – People to Follow

Name	Best Known For	Why Leaders Should Follow
Allie K. Miller	AI advisor, ex-Amazon/IBM	Named AI Innovator of the Year, with daily content on AI strategy, governance, and implementation for enterprises grounded in real operational experience
Andrej Karpathy	Independent researcher, ex-Tesla/OpenAI	Deep technical expertise with clear communication; cautions that AI agents should be "kept on a leash" rather than left unsupervised
Andrew Ng	Educator, founder of DeepLearning.AI, "The Batch" newsletter	Practical, hands-on approach to AI adoption widely followed by business audiences, not just technologists
Cassie Kozyrkov	Former Chief Decision Scientist, Google	Decision-science framing of AI — useful for governance and judgment questions
Dario Amodei	CEO, Anthropic	Deep credibility and insight into what AI companies believe about safety, deployment, and enterprise risk
Demis Hassabis	CEO, Google DeepMind	Founder-level perspective from one of the frontier labs, shaping where model capability is headed
Emad Mostaque	Founder, Stability AI	Vocal advocate for open, decentralised AI development as an alternative to closed, corporate-controlled AI systems
Ethan Mollick	Wharton professor, "One Useful Thing" Substack	One of the most cited voices for translating AI capability into real workplace use cases
Fei-Fei Li	Co-Director, Stanford HAI	Prioritizes responsible AI over sensationalism; strong voice on human-centered design
Kai-Fu Lee	Chairman/CEO, Sinovation Ventures; author, AI Superpowers	Bridge voice between Western and Chinese AI development, uniquely positioned as both practitioner and investor amid intensifying AI geopolitics
Paul Roetzer	Founder, Marketing AI Institute	Translates frontier AI developments into actionable business guidance without oversimplifying, including case studies on mandatory enterprise AI adoption
Sam Altman	CEO, OpenAI	Company founder shaping the commercial and product direction of the industry
Satya Nadella	CEO, Microsoft	"AI for everyone" vision and massive infrastructure investment have made Microsoft a dominant force in enterprise AI adoption
Sundar Pichai	CEO, Alphabet/Google	Leads one of the world's largest AI research organisations, driving the Gemini model family
Yann LeCun	AI researcher, formerly Chief AI Scientist at Meta	One of the most influential foundational researchers in the field
Yoshua Bengio	Turing Award laureate, AI safety researcher	Operates at the cutting edge of technical research; leading voice on AI safety

Article – Links

Websites

https://ruben.substack.com/

https://venturebeat.com/

https://www.xda-developers.com/

https://menlovc.com/

https://lifearchitect.ai/about-alan/

https://simonwillison.net/

https://magazine.sebastianraschka.com/

https://bytebytego.com/

https://thenewstack.io/

https://linas.substack.com/

YouTube Channels

https://www.youtube.com/@AgenticAI-Foundation

https://www.xda-developers.com/

https://www.youtube.com/ibmtechnology

https://www.youtube.com/@AndrejKarpathy

https://www.youtube.com/@aiDotEngineer

https://www.youtube.com/@theAIsearch

Newsletters

https://www.newsletter.theintelligentworker.com/

https://www.theneurondaily.com/

https://www.therundown.ai/

Gallery



IIM KOZHIKODE



NUS BUSINESS SCHOOL - SINGAPORE



SP JAIN DUBAI



MASTERS' UNION





BHARAT PETROLEUM – KOCHI REFINERY



NUVOCO CEMENT





JINDAL INDIA





BAHRI MAZROEI TRADING COMPANY – DUBAI





LANDMARK GROUP – DUBAI





AMBER ENTERPRISES



youtube.com/c/havishmadhvapaty

havishmconsulting.com