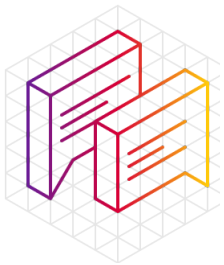
DATASTAX

WHITE PAPER

Systems of Automation



The future of enterprise architecture is agentic

Learn how systems of automation will empower domain experts to scale their abilities and make enterprises work better through AI.

For decades humans have translated and implemented business processes into workflows using code. AI agents, leveraging large language models (LLMs), have the ability to span all three levels: processes, workflows, and code. Agents are equally adept at understanding human concepts and code. They can change altitude—shifting between levels of abstraction—and seamlessly bridge and translate between these levels.

Consequently, enterprise architects must reconsider how agentic capabilities and supporting enterprise services are exposed to these agents. In this evolved software landscape, AI agents unlock the potential of existing service-oriented architectures (SOAs) by dynamically generating functionality according to immediate system needs.

Exposing enterprise-specific services and capabilities within these "systems of automation" takes generative AI beyond conversational interactions to a future where agentic value creation proliferates from the replacement of business processes and operational workflows.

The platform that underlies a system of automation must offer a combination of capabilities:

- assemble context from enterprise data
- support the AI agents that use capabilities to assemble context
- expose a framework that enables the discovery of those capabilities
- enable the development of workflows that use, augment, or replace existing enterprise services
- enable the reuse of these workflows as building blocks of automation

Enterprise architectures get agentic

Systems thinking forms the foundation of enterprise architecture design. A widely accepted framework for understanding the varying rates at which these systems change is "pace layering," a concept introduced by Stewart Brand to describe how different parts of buildings evolve at varying rates. In 2010, Gartner adapted this concept to explain how IT systems change over time. In this model, the *system of record* serves as the stable foundation, evolving slowly like a building's site and structure, while higher layers change more quickly to meet customer needs.

In 2011, Geoffrey Moore coined the term *systems of engagement* to describe these faster-evolving systems that pull data from the system of record but are focused on user experience and collaboration. In 2015, he introduced the concept of *systems of intelligence* to enterprise applications, writing:

> *Systems of intelligence can deliver measurable and lasting competitive differentiation. Their power is in mining engagement in real time to anticipate, influence, and optimize customer*

> *experiences. In their most advanced form, these systems learn both from the user's interactions and external observations, such as an extended network of relationships.*

GenAI applications were poised to redefine systems of intelligence. Their conversational interfaces can participate on-demand with systems of engagement. The introduction of AI agents further advanced the architecture by enabling the system to take action on behalf of the user; this architectural component or capability is more aptly called a system of automation (rather than a system of intelligence).

AI agents can use LLMs to complete tasks. They represent a transformational leap from simple conversational interfaces to dynamic software objects capable of iterating on tasks with and without human intervention. These workflows are managed by systems of automation frameworks, which can interrogate SOA endpoints, write SQL and query databases, perform web and other information retrieval searches, and quickly synthesize collected data into insights, answers, and work products.

Business processes get agentic

Business processes, whether driven by human effort or technology, are workflows composed of various tasks, decisions, and branching points. These business process workflows are carefully structured to manage everything from order processing to customer communications.

Agentic workflows introduce AI agents into these existing processes. Initially, agents will replace specific portions of workflows, automating tasks that were previously manual or rigidly coded. For example, an agent might handle data entry, generate reports, or make routine decisions based on predefined criteria.

As agents become more integrated, they'll begin to manage entire workflows. Processes that once required extensive programming and manual oversight will be executed dynamically by agents. These intelligent agents can generate code on the fly, adapt to new information, and make real-time decisions—all without the need for human intervention.

Agentic workflows are a conceptual extension of existing enterprise workflow software. The key difference is that agents add a layer of intelligence and adaptability to the workflows; they can handle complex tasks more efficiently and adjust processes in response to changing business needs.

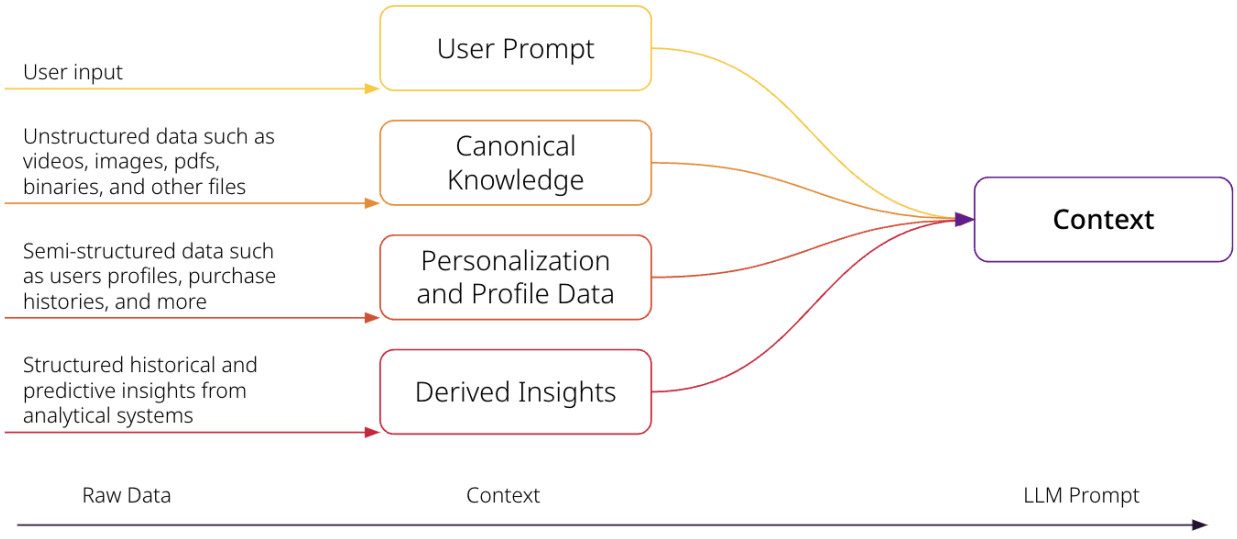
Agentic workflows will become synonymous with traditional workflows in systems like SAP. They will enhance existing processes by making them more dynamic and responsive, ultimately leading to greater efficiency and flexibility in business operations.

GenAI is agentic

ChatGPT's arrival in November 2022 introduced conversational AI to the world, and sparked great excitement about its potential as a problem-solving co-pilot across myriad domains. In addition to a user's prompt, ChatGPT can accept large amounts of text from which the language model can derive an answer it otherwise couldn't address. The critical insight that led to the birth of the agentic model was the introduction of follow-up questions in pursuit of a more detailed context for the prompt. Rather than instruct an LLM to respond to a user's prompt immediately, the LLM is used to construct follow-up questions. The dialogue with the user that gathers the answers to the follow-up questions and adds them to the initial prompt forms the basis of the pattern that drives agent workflows.

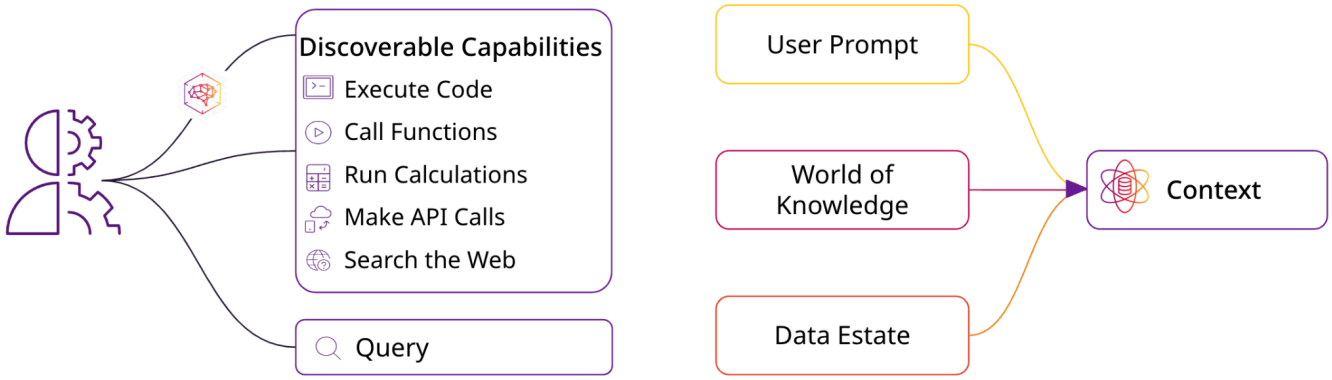
The agentic approach to collaborating with an end user to reach a more detailed context is closely related to agentic tool-use capabilities. Agents can read metadata from various service endpoints and evaluate API documentation, service descriptors, and technical documentation to decide which tool to call and how to call it. The most visible example is when ChatGPT encounters a scenario where the user effectively asks for a summarized web search. ChatGPT extracts keywords from the initial prompt, searches the web, and summarizes the best results for the user. Multiple calls to the language model are made to structure the use of the tool-calling capability, and this agent directs the web search output to a new LLM call, appending the original prompt to the context. Here, an agentic app has automated the human process of performing a web search and pasting the resulting content bodies into a prompt.

GenAI apps depend upon the quality and detail of the context that an LLM uses to generate its output. Context is the data sent to an LLM to produce a grounded, non-hallucinated result.



AI agents can deliver richer context through the use of domain-specific capabilities, which can execute code, call functions, run calculations, make API calls, or search the web to enable them to complete their tasks. While some agentic capabilities can be called directly by the agent, others make iterative calls to an LLM to plan the use of capabilities.

Agentic workflows can quickly generate many language model invocations, and the right choice of model for a particular workflow task will depend on the size and contents of the context. Models with large context windows can derive answers from datasets that are unwieldy for human operators but are slower and significantly more expensive than models with smaller context windows. The long-term utility of a particular agent may depend on how quickly and economically it can make quick decisions or, conversely, if it can fully grasp novel concepts to react to new situations.



Agents

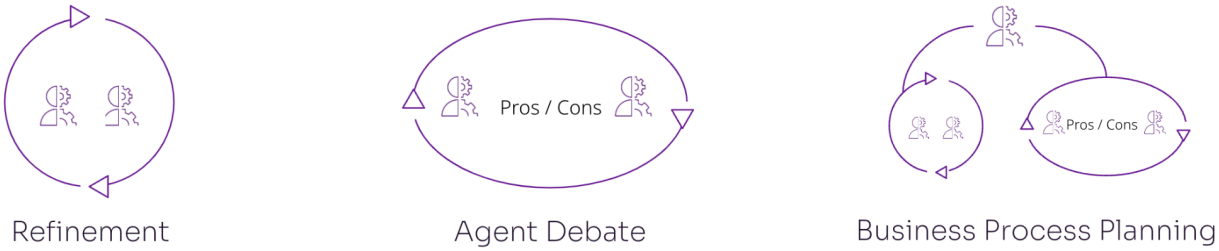
AI agents that use capabilities to assemble context can also use capabilities to complete tasks on the behalf of users. These AI agents can perform several key functions guided by the instructions and information from context:

Tool use - The agent leverages external functions, APIs, or tools to extend its capabilities and perform specific tasks. This includes calling predefined functions or interfacing with external services (like making web requests using cURL or accessing RESTful APIs) to obtain context or execute actions beyond its inherent functionalities.

Decision-making: The agent evaluates available information and selects the most appropriate action to achieve its goals. This involves analyzing context, weighing possible outcomes, and choosing a course of action that aligns with the desired objectives.

Planning—The agent formulates actions or strategies to achieve a specific goal. This involves reasoning about the available context and future states, predicting the outcomes of actions, and selecting the optimal steps to reach the desired outcome.

Decision-making and planning often involve multiple agents working together towards a goal. The agents may seek to refine generated code for correctness, debate whether an agentic decision is biased, or plan the use of other agentic capabilities to complete a task.

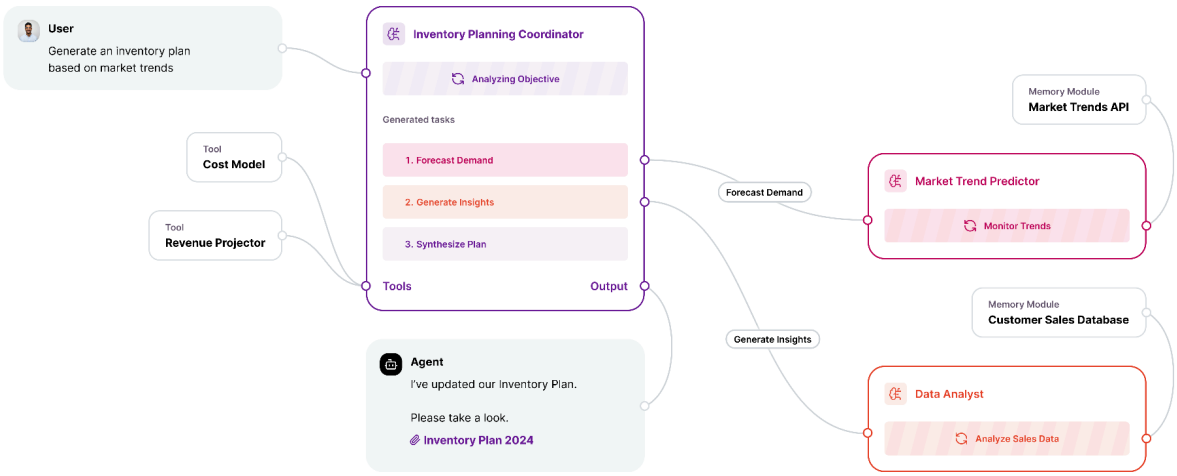


Refinement Agent Debate Business Process Planning

Agentic frameworks

Agentic frameworks enable the discovery of agentic capabilities, enable agents to discover other agents, and enable the development and reuse of agentic workflows by agents and humans alike. The agentic framework further maximizes the value of AI agents by connecting them to enterprise data, external information, and the ever-broadening ecosystem of AI capabilities and retrieval patterns.

The process of context assembly that an agentic workflow ultimately achieves depends on the coordination of many independent tasks, such as calls to external tools, queries to internal data sources and APIs, service-oriented endpoints that are exposed within the framework using service descriptor metadata, and handoffs to complementary agents. This coordination is the responsibility of the framework.



Agentic overachievement

Agentic frameworks lend themselves to visual workflow layout authoring using drag-and-drop AI components. Visual components wired together simplify coordinating the actions of agents from different sources using potentially different models.

Using AI on the scale of agentic systems broadens the scope of stakeholders to any enterprise system. Agentic workflows that visualize the relationships between AI components reduce the risk of miscommunication. More team members can interpret workflow layouts than can write Python code.

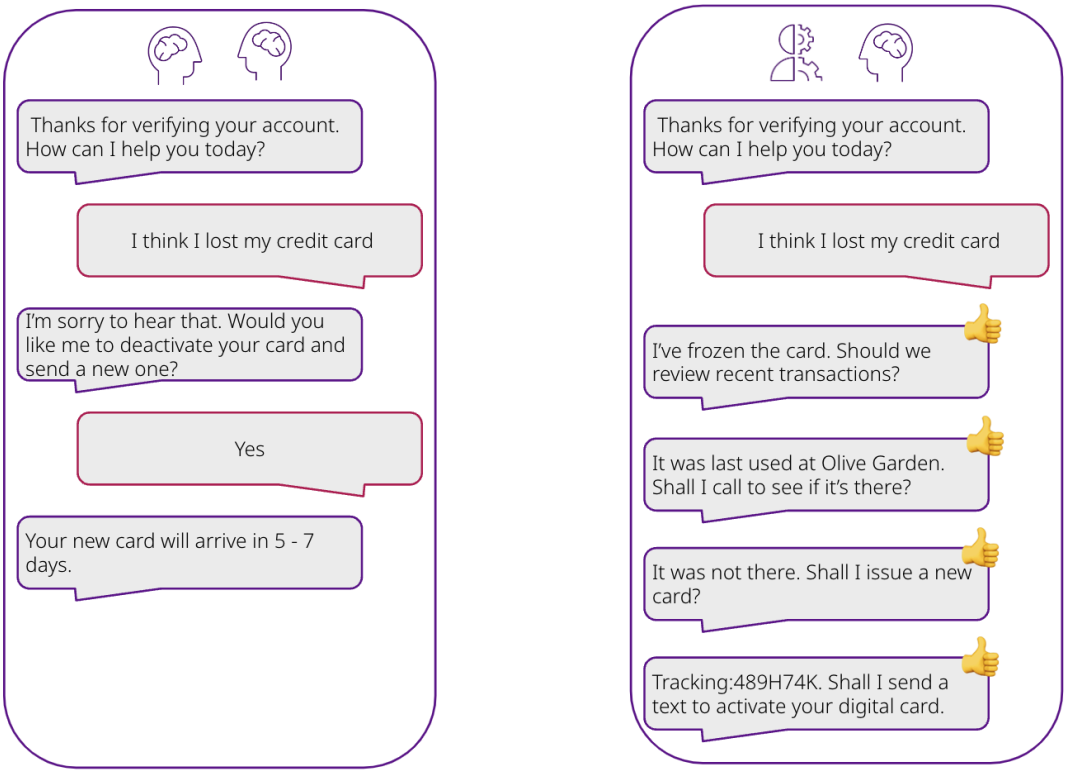
Workflow-centric agentic frameworks also broaden the pool of experts who can contribute know-how to these systems of automation by eliminating programming skills as a barrier. Domain experts can quickly learn to coordinate data sources, tools, and other AI components. While Python developers can easily contribute bespoke agentic components when necessary, complex agentic behavior can be achieved without programming skills.

As a result, the non-programmers most familiar with a set of business processes can orchestrate their automation. The example below considers the experience of a customer losing their credit card and compares a human-to-human interaction with a human-to-agent interaction.

While a positive customer experience is the goal of any customer service interaction, there are rarely enough humans to deliver every customer the highest quality of service possible. Efficiency and completeness have emerged as metrics by which we measure a customer service representative. On the left, the customer's problem was quickly understood, and the appropriate

process was used to rectify the situation. AI agents, being able to scale beyond human limits, can do more for the customer and so improve their experience.

In this example, the AI agent understands the ambiguity in the customer's problem description. The customer is not certain the card is lost and so the AI agent starts with a temporary hold along with a suggestion to review transactions. The agent can even offer to call the restaurant, which is the last place the customer's credit card was used. Unsuccessful, the agent offers to issue a new card. A human agent could do all these things if not for the other ten customers waiting in the queue.



While automation can be introduced to any business process, significant value will be gained from identifying which portions of existing processes are of high impact and high effort—for which an AI agent can deliver high impact with low effort.

Conclusion

Now, agentic workflows bring together enterprise data, AI, and APIs, forming the systems of automation that empower domain experts to scale their abilities and make enterprises work better through AI. Integrating AI agents into enterprise architectures marks a transformative leap in how organizations approach automation and business processes. These agents, empowered by LLMs and agentic frameworks, transcend traditional boundaries by seamlessly operating across processes, workflows, and code.

Adopting agentic workflows promises to enhance efficiency, scalability, and responsiveness across business operations. AI agents can manage entire workflows, handle complex tasks with greater adaptability, and significantly improve customer experiences by providing more personalized and timely interactions. As automation becomes deeply embedded in enterprise systems, organizations that embrace agentic AI will be better positioned to innovate, compete, and deliver exceptional value in an increasingly complex and fast-paced digital landscape.

The future of enterprise architecture is agentic. AI copilots will become pilots, redefining how we engage with data, users, and the evolving digital landscape.

About DataStax

DataStax is the company that powers generative AI applications with real-time, scalable data with production-ready vector data tools that generative AI applications need, and seamless integration with developers' stacks of choice. The Astra DB vector database provides developers with elegant APIs, powerful real-time data pipelines, and complete ecosystem integrations to quickly build and deploy production-level AI applications. With DataStax, any enterprise can mobilize real-time data to quickly build smart, high-growth AI applications at unlimited scale, on any cloud. Hundreds of the world's leading enterprises, including Audi, Bud Financial, Capital One, SkyPoint Cloud, VerSe Innovation, and many more rely on DataStax to deliver real-time AI. Learn more at DataStax.com.

© 2024 DataStax Inc., All Rights Reserved. DataStax is a registered trademark of DataStax, Inc. and its subsidiaries in the United States and/or other countries.