

What is AI agent planning?

What is AI agent planning?

AI agent planning refers to the process by which an artificial intelligence (AI) agent determines a sequence of actions to achieve a specific goal. It involves decision-making, goal prioritization and action sequencing, often using various planning algorithms and frameworks.

AI agent planning is a [module common to many types of agents](#) that exists alongside other modules such as perception, reasoning, decision-making, action, memory, communication and learning. Planning works in conjunction with these other modules to help ensure that agents achieve outcomes desired by their designers.

Not all agents can plan. Unlike simple reactive agents that respond immediately to inputs, planning agents anticipate future states and generate a structured action plan before execution. This makes AI planning essential for [automation](#) tasks that require multistep decision-making, optimization and adaptability.

Industry newsletter

The latest AI trends, brought to you by experts

Get curated insights on the most important—and intriguing—AI news. Subscribe to our weekly Think newsletter. See the [IBM Privacy Statement](#).

We use your email to validate you are who you say you are, to create your IBMid, and to contact you for account related matters.

Business email

Your subscription will be delivered in English. You will find an unsubscribe link in every newsletter. You can manage your subscriptions or unsubscribe [here](#). Refer to our [IBM Privacy Statement](#) for more information.

Your subscription will be delivered in English. You will find an unsubscribe link in every newsletter. You can manage your subscriptions or unsubscribe [here](#). Refer to our [IBM Privacy Statement](#) for more information.

How AI agent planning works

Advances in [large language models](#) (LLMs) such as OpenAI's [GPT](#) and related techniques involving [machine learning algorithms](#) resulted in the [generative AI](#) (gen AI) boom of recent years, and further advancements have led to the emerging field of autonomous agents.

By integrating tools, APIs, hardware interfaces and other external resources, agentic AI systems are increasingly autonomous, capable of real-time decision-making and adept at problem-solving across various use cases.

Complex agents can't act without making a decision, and they can't make good decisions without first making a plan. Agentic planning consists of several key components that work together to encourage optimal decision-making.

Goal definition

The first and most critical step in AI planning is defining a clear objective. The goal serves as the guiding principle for the agent's decision-making process, determining the end state it seeks to achieve. Goals can either be static, remaining unchanged throughout the planning process, or dynamic, adjusting based on environmental conditions or user interactions.

For instance, a self-driving car might have a goal of reaching a specific destination efficiently while adhering to safety regulations. Without a well-defined goal, an agent would lack direction, leading to erratic or inefficient behavior.

If the goal is complex, agentic [AI models](#) will break it down into smaller, more manageable sub-goals in a process called task decomposition. This allows the system to focus on complex tasks in a hierarchical manner.

LLMs play a vital role in task decomposition, breaking down a high-level goal into smaller subtasks and then executing those subtasks through various steps. For instance, a user might ask a [chatbot](#) with a natural language prompt to plan a trip.

The agent would first decompose the task into components such as booking flights, finding hotels and planning an itinerary. Once decomposed, the agent can use [application programming interfaces \(APIs\)](#) to fetch real-time data, check pricing and even suggest destinations.

State representation

To plan effectively, an agent must have a structured understanding of its environment. This understanding is achieved through state representation, which models the current conditions, constraints and contextual factors that influence decision-making.

Agents have some built-in knowledge from their training data or [datasets](#) representing previous interactions, but perception is required for agents to have a real-time understanding of their environment. Agents collect data through sensory input, allowing it to model its environment, along with user input and data describing its own internal state.

The complexity of state representation varies depending on the task. For example, in a chess game, the state includes the position of all pieces on the board, while in a robotic navigation system, the state might involve spatial coordinates, obstacles and terrain conditions.

The accuracy of state representation directly impacts an agent's ability to make informed decisions, as it determines how well the agent can predict the outcomes of its actions.

Action sequencing

Once the agent has established its goal and assessed its environment, it must determine a sequence of actions that will transition it from its current state to the desired goal state. This process, known as action sequencing, involves structuring a logical and efficient set of steps that the agent must follow.

The agent needs to identify potential actions, reduce that list to optimal actions, prioritize them and identifying dependencies between actions and conditional steps based on potential changes in the environment. The agent might allocate resources to each step in the sequence, or schedule actions based on environmental constraints.

For example, a robotic vacuum cleaner needs to decide the most effective path to clean a room, ensuring it covers all necessary areas without unnecessary repetition. If the sequence of actions is not well planned, the AI agent might take inefficient or redundant steps, leading to wasted resources and increased execution time.

The ReAct framework is a methodology used in AI for handling dynamic decision-making. In the ReAct framework, reasoning refers to the cognitive process where the agent determines what actions or strategies are required to achieve a specific goal.

This phase is similar to the planning phase in agentic AI, where the agent generates a sequence of steps to solve a problem or fulfill a task. Other emerging frameworks include ReWOO, RAISE and Reflexion, each of which has its own strengths and weaknesses.

Optimization and evaluation

AI planning often involves selecting the most optimal path to achieving a goal, especially when multiple options are available. Optimization helps ensure that an agent's chosen sequence of actions is the most efficient, cost-effective or otherwise beneficial given the circumstances. This process often requires evaluating different factors such as time, resource consumption, risks and potential rewards.

For example, a warehouse robot tasked with retrieving items must determine the shortest and safest route to avoid collisions and reduce operational time. Without proper optimization, AI agents might execute plans that are functional but suboptimal, leading to inefficiencies. Several methods can be used to optimize decision-making, including:

Heuristic search

Heuristic search algorithms help agents find optimal solutions by estimating the best path toward a goal. These algorithms rely on heuristic functions—mathematical estimates of how close a given state is to the desired goal. Heuristic searches are particularly effective for structured environments where agents need to find optimal paths quickly.

Reinforcement learning

Reinforcement learning enables agents to optimize planning through trial and error, learning which sequences of actions lead to the best outcomes over time. An agent interacts with an environment, receives feedback in the form of rewards or penalties, and refines its strategies accordingly.

Probabilistic planning

In real-world scenarios, AI agents often operate in uncertain environments where outcomes are not deterministic. Probabilistic planning methods account for uncertainty by evaluating multiple possible outcomes and selecting actions with the highest expected utility.

Collaboration

Single agent planning is one thing, but in a [multiagent system](#), AI agents must work autonomously while interacting with each other to achieve individual or collective goals.

The planning process for AI agents in a multiagent system is more complex than for a single agent because agents must not only plan their own actions but also consider the actions of other agents and how their decisions interact with those of others.

Depending on the [agentic architecture](#), each agent in the system typically has its own individual goals, which might involve accomplishing specific tasks or maximizing a reward function. In many multiagent systems, agents need to work together to achieve shared goals.

These goals could be defined by an overarching system or emerge from the agents' interactions. Agents need mechanisms to communicate and align their goals, especially in cooperative scenarios. This could be done through explicit messaging, shared task definitions or implicit coordination.

Planning in multiagent systems can be centralized, where a single entity or controller—likely an LLM agent—generates the plan for the entire system.

Each agent receives instructions or plans from this central authority. It can also be decentralized, where agents generate their own plans but work collaboratively to help ensure that they align with each other and contribute to global objectives, often requiring communication and negotiation.

This collaborative decision-making process enhances efficiency, reduces [biases](#) in task execution, helps to avoid [hallucinations](#) through cross-validation and consensus-building and encourages the agents to work toward a common goal.

After planning

The phases in agentic [AI workflows](#) do not always occur in a strict step-by-step linear fashion. While these phases are often distinct in conceptualization, in practice, they are frequently interleaved or iterative, depending on the nature of the task and the complexity of the environment in which the agent operates.

AI solutions can differ depending on their design, but in a typical [agentic workflow](#), the next phase after planning is action execution, where the agent carries out the actions defined in the plan. This involves performing tasks and interacting with external systems or knowledge bases with [retrieval augmented generation \(RAG\)](#), tool use and function calling ([tool calling](#)).

Building [AI agents](#) for these capabilities might involve [LangChain](#). Python scripts, JSON data structures and other programmatic tools enhance the AI's ability to make decisions.

After executing plans, some agents can use memory to learn from their experiences and iterate their behavior accordingly.

In dynamic environments, the planning process must be adaptive. Agents continuously receive feedback about the environment and other agents' actions and must adjust their plans accordingly. This might involve revising goals, adjusting action sequences, or adapting to new agents entering or leaving the system.

When an agent detects that its current plan is no longer feasible (for example, due to a conflict with another agent or a change in the environment), it might engage in replanning to adjust its strategy.

Agents can adjust their strategies using [chain of thought](#) reasoning, a process where they reflect on the steps needed to reach their objective before taking action.