Multi-agent AI system
Building Effective Agents

Personal Studies

André Vieira da Silva

Friday 19th December, 2025

1 Introduction

Artificial Intelligence (AI) agents represent a shift from static, rule-based software to systems capable of reasoning, acting, and adapting in dynamic environments. This article provides a complete introduction to AI agents, covering their core concepts, architectures, types, applications, and challenges [1].

2 What Are AI Agents?

An **AI agent** is a software system that can perceive an environment, make decisions, and take actions to achieve a specific goal. Unlike traditional programs, AI agents are not limited to predefined execution paths; instead, they adapt their behavior based on context, feedback, and objectives.

Modern AI agents are often powered by Large Language Models (LLMs) and extended with tools, memory, and feedback loops.

3 Agents vs Traditional Software

Agents are particularly useful when it is not possible to anticipate all execution steps in advance.

Aspect	Traditional Software	AI Agents
Behavior	Fully predefined	Adaptive and dynamic
Decision-making	Rule-based	Model-driven reasoning
Flexibility	Low	High
Handling uncertainty	Poor	Strong
Learning from context	No	Yes

Table 1: Comparison between traditional software and AI agents

4 Core Components of an AI Agent

4.1 Goal

Every agent operates with a clear objective, such as answering a question, analyzing data, or completing a workflow. Without a well-defined goal, effective reasoning is not possible.

4.2 Perception

Agents receive input from various sources, including user prompts, databases, documents, APIs, or sensor data. These inputs define the current state of the environment.

4.3 Reasoning and Planning

Reasoning allows the agent to determine what actions should be taken next. Planning involves decomposing a goal into smaller steps and selecting appropriate strategies and tools.

4.4 Actions

Agents act by executing code, calling APIs, querying databases, retrieving documents, or generating reports. Actions produce observable effects in the environment.

4.5 Memory

Memory enables consistency and learning. Common forms include:

- Short-term memory: task context and conversation state.
- Long-term memory: historical interactions and learned information.
- Retrieval memory: external knowledge accessed via vector databases (RAG).

4.6 Feedback and Observation

After acting, the agent observes the outcome, evaluates success or failure, and adjusts its strategy if necessary. This feedback loop enables self-correction.

5 The Agent Loop

Most AI agents operate within a continuous loop:

`Goal` → `Plan` → `Act` → `Observe` → `Evaluate` → `Adjust`

The loop repeats until the goal is achieved or a termination condition is met.

6 Types of AI Agents

6.1 Reactive Agents

Reactive agents respond directly to inputs without memory or planning. They are simple but limited in capability.

6.2 Deliberative Agents

Deliberative agents maintain internal models of the environment and can plan multiple steps ahead. Most LLM-based agents fall into this category.

6.3 Tool-Using Agents

These agents extend reasoning capabilities by interacting with external tools such as APIs, databases, or code execution environments.

6.4 Learning Agents

Learning agents improve their performance over time using feedback or reinforcement mechanisms.

6.5 Multi-Agent Systems

Multi-agent systems consist of multiple agents working collaboratively or competitively. Each agent typically has a specialized role, increasing scalability and robustness.

7 Multi-Agent Systems

In a multi-agent system, responsibilities are distributed among agents. Typical roles include:

- Planner Agent: task decomposition and strategy definition.
- Executor Agent: performs actions and tool calls.
- Data Agent: handles data ingestion and transformation.
- Analyst Agent: performs statistical or analytical tasks.
- Critic Agent: validates outputs and detects errors.

This structure mirrors human teams and improves system reliability.

8 When to Use AI Agents

8.1 Appropriate Use Cases

AI agents are suitable when tasks are complex, open-ended, or require iterative reasoning and adaptation.

8.2 When Not to Use Agents

Agents should be avoided when workflows are simple, deterministic, or highly cost-sensitive.

9 Applications

AI agents are used in a wide range of domains, including data analytics, software engineering, finance, customer support, robotics, and scientific research.

10 Challenges and Risks

Key challenges include hallucinations, evaluation difficulty, computational cost, lack of transparency, and security risks. Effective agent systems require validation, monitoring, and clear operational constraints.

11 Future Directions

AI agents are evolving toward greater autonomy, improved safety, standardized evaluation, and deeper integration with real-world systems. They are expected to become a foundational component of next-generation intelligent software.

12 Conclusion

AI agents are not merely advanced chatbots. They are goal-driven systems that plan, act, observe, and adapt using reasoning, tools, and memory. Understanding AI agents is essential for building scalable, reliable, and intelligent AI-powered applications.

---

13 Building AI Agents in Python for Investment Business Models

Referências

1. Anthropic. *Building Effective Agents* Engineering Blog. Anthropic. https://www.anthropic.com/engineering/building-effective-agents (2025).