Agentic AI Essentials

Agenda

- Understanding Agentic AI
- Agentic AI Core Components
- Comparing AI Models
- Agent Behavior Patterns
- Agent System Designs
- Single and Multi Agent AI Systems
- Human in the Loop Systems
- Frameworks and Tools Overview
- Ethical and Responsible AI
- Agentic AI Best Practices

Section 1: Understanding Agentic AI

From Passive to Active

Generative AI (Passive): Waits for a prompt, generates text/image, and stops. It's a tool like a highly advanced encyclopedia.

Agentic AI (Active): Pursues goals. It can plan, execute actions, use tools, and iterate to achieve a specific outcome without constant human intervention.

The Evolution of Capabilities

1

Prompt Eng.

Zero-shot & Few-shot querying

2

RAG

Retrieval Augmented Generation (Context)

3

Chains

Deterministic sequences of calls

4

Agents

Autonomous planning & tool use

Section 2: Core Components

The Cognitive Loop

Perception

Reading user inputs, multimodal signals, and environmental feedback.

Brain (LLM)

The reasoning engine that decides "what to do next" based on context.

Planning

Decomposing complex goals into manageable, sequential steps.

Action

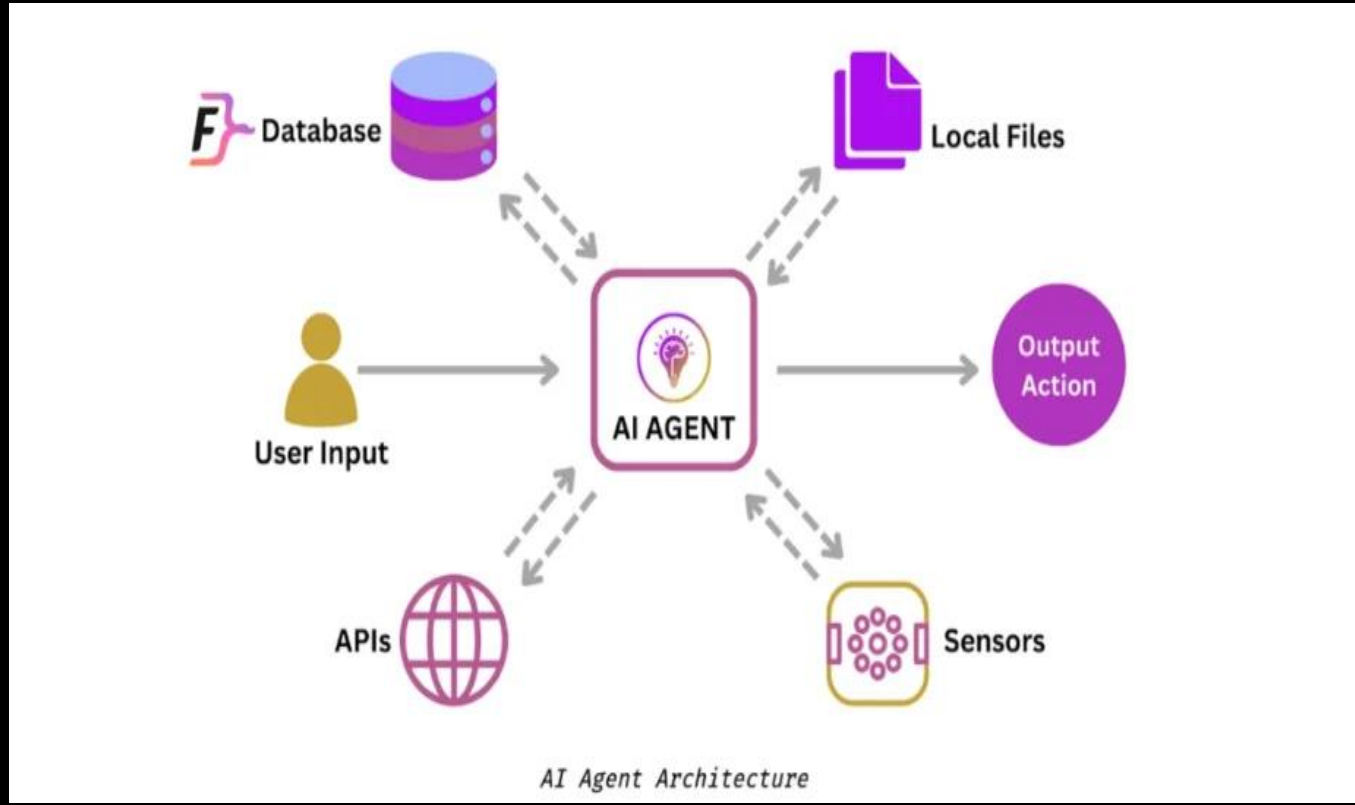
Executing commands via external tools, APIs, or scripts.

Tools & Environment

Extending the Brain

An LLM is trapped in a text box. **Tools** give it hands.

- **Search APIs:** For real-time information.
- **Calculators/Code Interpreters:** For precise math.
- **Database Connectors:** To read/write enterprise data.
- **File Systems:** To create and manage documents.



AI Agent Architecture

Section 3: Comparing AI Models

Standard LLM vs. Agentic AI

Feature	Standard LLM (Chat)	Agentic AI
Interaction	Reactive (Responds to prompt)	Proactive (Pursues objective)
Scope	Conversation & Knowledge	Task Execution & Workflow
Tools	None (or limited built-in)	Unlimited custom integrations
Autonomy	Low (User guides every turn)	High (System navigates obstacles)

Section 4: Agent Behavior Patterns

The ReAct Pattern

Reasoning (Thought)

The model pauses before acting to articulate a plan.
"To answer this question, I first need to find the current stock price of Apple, then compare it to yesterday's close."

Acting (Action)

The model executes the specific step identified in the reasoning phase.
Action: `get_stock_price('AAPL')`

Reflection & Self-Correction

The Feedback Loop

Agents aren't perfect. A critical pattern is **Reflection**, where the agent critiques its own past actions or outputs.

If an API call fails or the result looks wrong, a reflecting agent catches the error, adjusts its plan, and tries again without human help.

Section 5: Agent System Designs

Planning & Decomposition

Complex tasks often confuse LLMs. **Decomposition** breaks a big goal into smaller, solvable sub-tasks.

- **Plan-and-Solve:** Create the entire list of steps first, then execute.
- **Step-by-Step:** Decide only the immediate next step based on the current state.
- **Tree of Thoughts:** Explore multiple possibilities before committing to a path.

Section 6: Single & Multi-Agent Systems

Single Agent Architecture

A single, general-purpose agent equipped with many tools.

Pros: Simple to build, easy to debug, unified context.

Cons: Can get confused with too many tools; struggles with very complex, multi-domain workflows.

Multi-Agent Collaboration

Role Specialization

Assign specific personas (e.g., "Coder," "Reviewer," "Manager"). Specialized prompts reduce hallucinations.

Hierarchical

A "Manager" agent breaks down the task and delegates to worker agents, aggregating their results.

Joint Chat

Agents converse with each other to solve problems, simulating a human brainstorming session.

Section 7: Human in the Loop

Modes of Interaction

Human as Approver

The agent proposes a plan or a sensitive action (like sending an email), and the human must click "Approve" to proceed. Essential for safety.

Human as Helper

When the agent gets stuck or encounters an error it can't fix, it escalates to a human, asking for new instructions or clarification.

Section 8: Frameworks & Tools

Popular Frameworks

LangChain / LangGraph

The industry standard for building graph-based agent workflows with fine-grained control.

CrewAI

High-level framework focused on role-playing agents and multi-agent orchestration.

Microsoft AutoGen

Specializes in conversable agents that can work together to solve tasks via dialogue.

Section 9:
Ethical & Responsible AI

Managing Risks

Autonomous agents introduce new risk vectors beyond standard LLM hallucinations.

- **Infinite Loops:** Agents getting stuck repeating actions, consuming credits.
- **Unintended Consequences:** Agents finding "shortcuts" that are harmful.
- **Data Leakage:** Agents inadvertently sharing sensitive data across tools.

Section 10: Best Practices

Building Reliable Agents

- **Start Simple:** Begin with a deterministic chain before moving to a fully autonomous agent.
- **Define Clear Tools:** Ensure tool descriptions are precise so the LLM knows exactly when to use them.
- **Implement Guardrails:** Use software logic to validate agent outputs before they execute.
- **Tracing & Eval:** Use tools like LangSmith to trace every step of the agent's thought process for debugging.