

Lesson 13 — The LLM & Agent Era

What changed 2022–2026 — LLMs, RAG, agents, MCP, the new product landscape.

IT Landscape

Module 4

High-volatility — as of 2026

Where we are in 2026

The most disruptive software shift since the smartphone — happening over three years.

- **Nov 2022** — ChatGPT released to the public.
- **By 2026** — AI assistants are embedded in IDEs, browsers, the OS, customer-service stacks, and a growing share of search.
- **Workflows are being rewritten** — coding, writing, support, research, design.

Snapshot — every number, every brand, and every pricing figure in this lesson has a “best before” date.

What is an LLM?

*A **Large Language Model** is a giant neural network trained to **predict the next word**.*

- **That's it.** One simple objective — applied at extreme scale.
 - **Scale** — trillions of words of training data, hundreds of billions of parameters.
 - **Out of that** emerges reasoning, translation, code generation, summarization, conversation.
 - It is exactly Lesson 12's "model" idea — with a particular shape (**transformer**) and a particular signal (next-word prediction).
-

The big players (as of 2026)

Lab	Family	Known for
Anthropic	Claude (Sonnet, Opus, Haiku)	Frontier reasoning, safety focus, long context
OpenAI	GPT family	Broadest consumer reach (ChatGPT)
Google DeepMind	Gemini family	Multi-modal, integrated with Search & Workspace
Meta	Llama family	Open-weights leader
Others	DeepSeek, Mistral, Alibaba Qwen, xAI Grok	Frontier or open-weights challengers

Names and rankings move quarterly — as of 2026.

Closed vs open weights (as of 2026)

Closed-weight

Claude, GPT-4-class, Gemini Pro.

- Best frontier capability.
- Accessed via the vendor's API.
- You do *not* run the model yourself.

Open-weight

Llama, Qwen, Mistral, DeepSeek.

- You can download and run yourself.
 - Great for privacy, cost, customization.
 - Capability gap to frontier is much narrower than in 2023.
-

What a prompt looks like

"You are a helpful assistant. Summarize the following email in two sentences and suggest one reply."

- A **prompt** is just text.
 - **Prompt engineering** — the craft of writing prompts that get good answers reliably.
 - **System prompt** (instructions, persona) vs **user prompt** (the actual question).
 - The model sees them concatenated, as one stream of text.
-

The context window

The context window is the model's RAM.

- Everything in one request — system prompt + history + documents + the new question — must fit.
 - As of 2026, common windows are **200K–2M tokens** (~150–1500 pages).
 - Anything beyond the window is invisible — the model “forgets” it.
 - **Pricing** happens here too — vendors charge per million input and output tokens (ballpark single dollars per million as of 2026; verify before quoting).
-

RAG — Retrieval-Augmented Generation

LLMs only know what they were trained on — their training has a **cutoff date**.

To answer questions about your private documents (or anything newer):

- **Retrieve** — pull the most relevant chunks from a database.
- **Augment** — paste them into the prompt.
- **Generate** — let the LLM answer using both.

The retrieval step typically uses a **vector database** (Lesson 07).

Most enterprise “ask our docs” chatbots are RAG under the hood.

Tools and function calling

A modern LLM can be given a list of **tools** — functions it may request:

- `search_web(query)`
- `send_email(to, subject, body)`
- `query_database(sql)`

The model decides one would help, asks the host program to call it, the result comes back into the conversation, and the model continues.

*This is how LLMs reach beyond their training data and **take action** in the world.*

Agents — LLMs in a loop

*An **agent** is an LLM that calls tools, looks at the result, and decides what to do next — in a loop.*

Not magic — a loop. Examples (as of 2026):

- **Cursor, Windsurf** — IDE agents that edit your code.
- **Claude Code** — terminal agent that runs commands and reviews their output.
- **Devin** — brands itself as an autonomous software engineer.
- **Operator** — web-browsing agent.
- **Salesforce AgentForce, Microsoft Copilot Studio** — enterprise platforms.

Each one lets the LLM iterate until the task is done — or it gives up.

MCP — the agent's USB-C (as of 2026)

Every agent needs to plug into your tools — Slack, Gmail, the filesystem, a database, the internal wiki.

MCP (Model Context Protocol) — introduced by Anthropic in late 2024 — is the open standard for how agents talk to those tools.

Clients that speak MCP (as of 2026):

- **Claude Desktop, Claude Code, Cursor, Windsurf, Cline.**
- VS Code (via extension) and a growing list of others.

*Write one tool as an **MCP server** — plug it into any compatible client. This platform has a whole **MCP Servers** course.*

The new product landscape (as of 2026)

Category	Names you'll hear
Chat assistants	Claude, ChatGPT, Gemini, Copilot, Grok
Coding	Cursor, Claude Code, GitHub Copilot, Windsurf, Cody
Search	Perplexity, Google AI Overviews, Brave Leo
Browsing & action	Operator, Comet, MS Copilot
Voice	Advanced voice models from each major lab
Image / video	Midjourney, DALL·E 3+, Stable Diffusion, Sora, Veo
Open-source orchestration	LangChain, LlamaIndex, LiteLLM

Every name on this slide is on `refresh-checklist.md`.

Recap

- An **LLM** is just a giant next-word predictor — at extreme scale.
- **RAG** and **tools** let it touch your data and act in the world.
- An **agent** is that loop, with permissions.

Coming up: Lesson 14 — who does what in IT, the open-source ecosystem, and which course on this platform to take next based on what you've enjoyed.

