PEOPLE IN THE LOOP

The Lexicon

The AI words that get used loosely, each one explained next to the word people actually confuse it with.

People in the Loop

Nina Thomas First edition

hirepitl.com

Contents

Every entry here is a comparison, because the real question is almost never "what does this word mean" on its own. It is "how is this different from the thing next to it". These are the same comparisons published in the Library room, bound so you can keep them.

01	Workflow vs. Automation vs. Agent	3
02	Human in the loop vs. Full automation	4
03	AI agent vs. AI assistant	5
04	Claude Skills vs. MCP	6
05	Claude Team vs. Claude Enterprise	7
06	LLM vs. Chatbot	8
07	API vs. App	9
08	Claude chat vs. Claude Cowork	10
09	Claude vs. Copilot	11

01

Workflow vs. Automation vs. Agent

Workflow: The sequence of steps a task actually follows Automation: A fixed version of that sequence, no person needed Agent: The steps inside it that genuinely need judgment

A workflow is the sequence of steps a task actually follows, whether or not software is involved. An automation runs a fixed version of that sequence without a person, best for steps that are well-defined and repeatable. An agent handles the steps inside that sequence that require judgment, deciding what to do based on what it finds, not following a fixed script.

This is the mistake worth naming directly: not every problem is an agent problem. A lot of what teams actually need is a clearly mapped workflow and a well-built automation for the repeatable parts, with an agent doing only the piece that genuinely needs judgment. Reaching for an agent everywhere adds cost and fragility where a plain automation would do the job better.

02

Human in the loop vs. Full automation

Human in the loop: A person stays close enough to check it and correct it Full automation: That checkpoint is removed entirely

Human in the loop means a person stays close enough to an AI system's work to check it, correct it, and decide when it can run on its own. Full automation removes that checkpoint entirely. Neither is universally right, the question is always which specific steps in a process still need a person, and which don't.

This is the actual method behind everything written here, not a tagline. Any build worth trusting starts by mapping exactly where a human needs to stay in the loop and where the work is safe to hand fully to an agent or automation.

03

AI agent vs. AI assistant

AI agent: Given a goal, takes multiple steps toward it on its own AI assistant: Responds to what you ask, one request at a time

An assistant responds to what you ask it, one request at a time, inside a conversation you're actively driving. An agent is given a goal and takes multiple steps toward it on its own, using tools, deciding what to do next, and checking in only when it hits a decision a human actually needs to make.

Most teams start with assistant-style use (asking Claude or Copilot questions) and only later move to agent-style use (Claude doing a multi-step task with a defined goal). Jumping straight to agents without the assistant habit already built rarely works. That order matters more than the tooling does.

04

Claude Skills vs. MCP

Claude Skills: Teach Claude your team's know-how MCP: Give Claude access to your team's systems

Claude Skills are packaged instructions, examples, and sometimes files that teach Claude how to do a specific task the way your team actually does it, so it doesn't need re-explaining every time. MCP (Model Context Protocol) is different: it's how Claude connects to outside tools and data, like a calendar, a CRM, or a codebase.

Put simply: Skills teach Claude your team's know-how. MCP gives Claude access to your team's actual systems. Most real implementations use both, a Skill for how to do the work, MCP connectors for where the work's data lives.

05

Claude Team vs. Claude Enterprise

Claude Team: Serves most teams under 20 to 30 people well Claude Enterprise: For a specific governance requirement

Most teams under 20-30 people are well served by Claude Team. Claude Enterprise matters when there's a specific governance requirement, SSO, audit logs, data residency, not because of headcount alone.

06

LLM vs. Chatbot

LLM: The underlying engine. Claude is one Chatbot: One way of putting an LLM in front of a person

An LLM, a large language model, is the underlying engine, trained on huge amounts of text to predict and generate language. Claude is an LLM. A chatbot is one specific way of putting an LLM in front of a person: a chat window you type into. The LLM can also power a lot of things that don't look like a chat window at all, an agent taking action in your systems, a search feature, a document reviewer.

Worth knowing going in: when adoption stalls, it's often because a team only ever met the chatbot version and never saw what the same LLM can do wired into their actual tools. That gap is worth naming before anybody blames the tool.

07

API vs. App

API: Two pieces of software talking directly App: Something a person opens and clicks through

An app is something a person opens and clicks through. An API, application programming interface, is how two pieces of software talk to each other directly, no person clicking anything. When Claude reads a support ticket, drafts a reply, and logs it in your CRM without anyone opening a chat window, that's the Claude API doing the work behind the scenes, not the Claude app.

This is usually the real fork in a build: does this need a person typing into a chat window, or does it need two systems talking to each other automatically. Most of what actually saves a team time is the second one.

08

Claude chat vs. Claude Cowork

Claude chat: One conversation, back and forth Claude Cowork: A task with several steps and real work product

Claude chat is one conversation at a time, you ask, Claude answers, you keep going back and forth. Cowork is built for a task that needs several steps and real work product at the end, research, drafts, files, something you hand off rather than read and close.

A useful rule of thumb: if you'd normally ask a smart colleague a quick question, that's chat. If you'd normally hand a colleague a whole task and check back later, that's Cowork. Most teams start in chat because it's the obvious front door, and only find Cowork once they've hit the ceiling of what one back-and-forth conversation can carry.

09

Claude vs. Copilot

Claude: Anthropic's assistant, strong at reasoning and real work Copilot: Microsoft's assistant, wired straight into Office

Both are AI agents built on a large language model, not the model itself. Copilot's edge is where it already lives: inside Word, Excel, Outlook, and Teams, so a team already living in Microsoft 365 gets it with the least friction. Claude's edge is longer, more careful reasoning and Claude Code and Cowork, built for real multi-step work rather than an assist-as-you-type layer over an existing document.

Most teams do not pick one and drop the other. The real question is which tasks belong to which: quick edits inside a document a team already lives in usually go to Copilot, and anything that needs judgment across several steps, research, a build, a governance review, usually goes to Claude.