

Cheat Sheet: Temporal for AI

Temporal has an answer for every one of the key elements of an AI application or agent. Use this cheat sheet to map AI capabilities to Temporal features, so you can build production-ready AI applications.

Key element in AI	How Temporal addresses it
We stitch together a bunch of steps into chains, graphs or agentic loops .	This is a Temporal <u>Workflow</u> . Workflows are equally well-suited to designs that structure steps at design time (first diagram) as those that are dynamic (second diagram).
We use Large Language Models (LLMs).	You invoke these through Temporal <u>Activities</u> , delivering resilience out of the box.
We invoke tools, craft prompts , and access resources .	You invoke these through Temporal <u>Activities</u> , delivering resilience out of the box.
We also invoke MCP servers via MCP clients .	When tools are MCP servers, the MCP client is implemented within an Activity.
We implement MCP servers .	These are implemented as Temporal Workflows and Activities.
AI applications, especially AI agents, are responsible for providing memory .	You just manage your application state in variables in your Workflow. As an added bonus, with Temporal, that state is durable.
We use checkpointing to keep from having to rerun steps if a process crashes.	Temporal delivers this implicitly through its event sourcing and state management architecture. You never have to think about checkpointing.
We must allow for humans in the loop .	This is achieved through <u>Temporal Signals & Updates</u> , and <u>Temporal Queries</u> .
AI applications are often long-running .	In addition to its event sourcing and state management foundation, Temporal handles long-running processes through its <u>Worker</u> architecture.

