ELLM

The eLLM external API

The eLLM external API allows systems to interact with ELLM using an OpenAI-compatible interface, providing options for document-grounded answers, plain chat, and automatic model selection. It supports authentication, streaming, and tool integration.

AUTHOR
Paul Flanders

PUBLISHED
8 Jun 2026

DOCUMENT
InfoHub — Knowledge, news, and updates from EssingtonITS

The eLLM external API

Paul Flanders · 8 Jun 2026

What this product is for

Sometimes you want another system, not a person, to use ELLM: an automation that drafts replies, a chatbot, or a tool that looks information up. The external API provides a standard, OpenAI-compatible way for those systems to send questions and receive answers, while still respecting your access rules.

Main features

- An OpenAI-compatible interface, so existing tools and clients can point at ELLM with little change.
- Authentication by API key, with each key scoped to groups for access control.
- A choice of modes: document-grounded answers, plain chat, or automatic model selection.
- The ability to target a specific model, or apply an organisation skill.
- Optional streaming of answers, and optional tool/function calling for agent-style clients.

How to use it

Getting set up

1. Ask an administrator to create an API key and scope it to the right groups (see the API keys article).

2. Point your OpenAI-compatible client at ELLM's API address, ending in /v1.
3. Use the API key as the client's credential.

Choosing how ELLM answers (the model field)

In these clients there is a "model" field. With ELLM, this field is not a real model name; it chooses how your question is handled:

- **ellm-rag:** a document-grounded answer using the default model, with citations.
- **ellm-chat:** a plain answer with no document grounding.
- **ellm-auto:** a document-grounded answer where ELLM automatically picks the best model.
- **A specific model name:** target a particular registered model, with document grounding.
- **A combined form:** for example a plain-chat answer from a specific model, giving you full control.

Your administrator sets which model the basic modes use by default. You can ask the API which options your key may use, and it will list the available modes and models.

Holding a conversation

The API is stateless for external callers: eLLM does not remember your previous turns. To continue a conversation, your client sends the earlier messages along with each new question. The most recent user message is treated as the current question, and any earlier ones as history.

Streaming and tools

- **Streaming:** you can request the answer as a stream, which some clients prefer.
- **Tool calling:** agent-style clients can attach their own tools. In plain-chat mode, ELLM passes the tools to the model and returns its tool requests, which your client then runs

and sends back. In the document-grounded modes, attached tools are ignored and you always get a grounded, cited answer.

- **Skills:** you can apply an organisation skill by its short name to shape the response.

Common tasks

- **Connect an automation tool:** point it at the API address with an API key and choose a mode.
- **Get grounded, cited answers:** use ellm-rag or ellm-auto.
- **Use ELLM as a plain assistant:** use ellm-chat.
- **Let ELLM pick the model:** use ellm-auto.
- **Build an agent that runs its own tools:** use plain-chat mode and attach tools.

Things to know

- A key can only reach the documents its groups are allowed to see, exactly like a person.
- External callers are stateless: send prior turns with each request to continue a conversation.
- The document-grounded modes always ground and cite, even if a client attaches tools; only plain-chat mode passes tools through to the model.
- Every API request is recorded in the audit log, identified by the key.
- Because the API is OpenAI-compatible, most existing clients work by simply changing the address and key.
- Powering AI coding tools (Cline, and soon Codex and Claude Code) is a separate, dedicated path with its own code key and ellm-code mode see Coding with eLLM.

Troubleshooting

- **My client is rejected:** check the API key is valid and not revoked, and that the address ends in /v1.
- **I get plain answers when I wanted grounded ones:** you are likely using ellm-chat. Switch to ellm-rag or ellm-auto.
- **My tools are being ignored:** tool passthrough only works in plain-chat mode. The grounded modes ignore attached tools.
- **A service can't see expected documents:** its key needs to be mapped to the right groups.
- **Streaming fails on one endpoint:** not every endpoint supports streaming; check which one your client uses.

Frequently asked questions

Is this the same as OpenAI's API?

It is compatible with it, so OpenAI-style clients work by changing the address and key. The "model" field, however, selects an eLLM mode rather than a real model name.

How does access control work?

The API key's groups decide what it can see, just like a person's groups.

Does ELLM remember my conversation?

Not for API callers. Send earlier messages with each request to keep context.

Can I make eLLM choose the model for me?

Yes. Use ellm-auto and eLLM picks the best available model.

Summary

The external API lets other software use eLLM through a familiar, OpenAI-compatible interface, secured by group-scoped API keys. The "model" field chooses how a question is handled (grounded, plain, automatic, or a specific model) and optional streaming, tools, and skills give connected services flexibility while your access rules and audit record stay in force.

> Need assistance navigating the complexities of eLLM? EssingtonITS offers expert guidance and tailored IT solutions to help you succeed. Visit EssingtonITS.co.uk for comprehensive support.