RAINBIRD

SKILL OVERVIEW · AUGUST 2026

RAKE

The Rainbird Agentic Knowledge Engineer

SECTION 01

What it is

The Rainbird Agentic Knowledge Engineer is an automated build tool for constructing Rainbird decision graphs that act as AI agents you can trust. They can automate complex decisions in any regulated domain, ensuring that outcomes are precisely anchored to your own policy and institutional knowledge, deterministic and fully auditable.

SECTION 02

How it works

RAKE turns a written brief and your source documents into a live, tested Rainbird agent that can automate decisions that are deterministic, auditable and 100% grounded in your policy. This is in stark contrast with LLM-agents which are probabilistic and opaque. To achieve this, RAKE researches the domain, ingests your documents, designs the decision model, encodes the rules, runs adversarial tests, compiles against the real Rainbird engine and, where you choose, consult your subject-matter experts along the way. You describe the end-state; RAKE does the knowledge engineering.

A typical build takes 30 to 60 minutes and runs on the server, so you can close the tab and come back.

SECTION 03

Core capabilities

Researches for you.	Reads your documents and, where permitted, researches public sources to establish the requirements behind a policy or regulation. Switch external research off for a fully internal build.
Builds traceable graphs.	Encodes requirements as a complete graph in Rainbird's knowledge representation language RBLang, with a requirement-to-rule coverage map, so you can confirm everything you briefed was captured.
Extends and repairs existing graphs.	Each revision iterates on the previous graph, reads the project's accumulated knowledge and delivers the next version. Give it feedback, changes in policy or new knowledge to get a new version.
Keeps humans in the loop.	Optional SME consultation sends experts a private review portal (no account needed) with a document to challenge and annotate. Optional test review shows the planned test scenarios so experts can change or add extra evals they want tested.
Tests against the real engine.	Every build validates and tests on a live Rainbird environment, so delivered test results are engine-exact, not guessed. Zero syntax errors.
Tunes itself.	A Simplify stage removes redundant rules and over-fitting, re-running every test to prove behaviour is unchanged. Optional performance tuning speeds up queries and reports before and after.
Ships a queryable endpoint.	Every project publishes a stable MCP endpoint serving the current version, so you can query the graph in plain language from Claude or any MCP client, and it keeps working across rebuilds and rollbacks.

SECTION 04

Outputs

01	The live graph in your Rainbird account, versioned within the project, with one-click rollback to any earlier version
02	The RBLang XML, ready for Rainbird Studio
03	A Rainbird Studio test suite generated from live runs, ready to import
04	Documentation: the logic and decisions, the coverage map, which relationship to query, and ready-to-use prompts
05	An MCP endpoint with a per-client connection guide (REST API also available)
06	A performance report, if tuning is switched on

SECTION 05

Business value

RAKE compresses the journey from captured knowledge to a validated, auditable decision model from weeks of specialist effort to around one hour, while enforcing the quality bar production systems require: traceability, expert review, engine-verified test coverage and instant rollback.

SECTION 06

Getting the best out of it

Invest in the brief. It is the single most important input. Describe the decision to be made and the real-world logic behind it, and direct the research: “deeply research the HMRC guidance on the SRT”. RAKE will do the legwork for you, but you need to tell it clearly what you want.

State the hard lines. Call out anything that must never happen, for example “never approve loans while a credit check is unknown”.

Give numbers. Rules, thresholds, bands and edge cases in plain terms beat vague descriptions.

Separate known from asked. Say which facts will come from a system-of-record and which the graph should ask for. This tells the agent what to inject rather than ask.

Attach the evidence. Policies, guides and spreadsheets, referenced in the brief, are read as source material. If you include your success criteria, the graph will be delivered once these have been met.

Describe your experts precisely. The agent consults SMEs selectively, only where a design query sits squarely in their stated expertise, so be specific about descriptions of SME knowledge so RAKE can determine when they should be called on.

Use test review for the nasty cases. Add any boundary scenarios you know from experience at design stage. The agent modifies the graph to ensure tests pass as part of it's build cycle.

Steer, don't restart. If a build drifts, `Interrupt` and redirect it in chat, or attach missing documents mid-build. If it stops short, `Continue this build` resumes from the same workspace, on a different model if you prefer.

Keep one project per decision. If the graph that you've built has issues, update the project rather than start fresh. New builds are automatically seeded with everything already learned, and the MCP endpoint never changes.

Pick a strong model. Use the most capable model available to you (Opus or Fable class) for best outcomes.

SECTION 07

Requirements

01	Access by invitation: an administrator sends a private sign-up link
02	A Rainbird API key (free accounts at app.rainbird.ai/signup)
03	An Anthropic API key, unless your organisation runs builds on AWS Bedrock, in which case you can bring your own Bedrock credentials.

SECTION 08

Availability

Closed beta, by invitation from Rainbird. Sign up for our waiting list at rainbird.ai/rake.

SECTION 09

Support

Community support via the Rainbird Forum, linked from within RAKE.

SECTION 10 · GOOD TO KNOW

One build runs per project at a time; email tells you when a build finishes or is waiting on you.

Treat an MCP endpoint URL like a password: pause serving or rotate the address instantly if required.

This is a research preview. If something looks wrong, interrupting the build and clarifying usually gets it back on track.

DECISION AUTOMATION

FOR REGULATED WORK.