Healthy AI First, Episode 11: Practice 3, Fable 5.2 on Low, Measured, Not Assumed

Sources and evidence for this episode

The third practice episode shows how I measured Claude Fable 5.2 at the effort level low against high, and summarizes eight prompt changes from Anthropic's guide. The only figures of our own are the measurements of the six runs on 17 November 2026. All other figures come from Anthropic's testing and are labelled as such in the episode.

All links checked on 17 November 2026.

What we mention in this episode

Setting effort

- **Claim:** The effort level sets how much the model thinks, how many tool calls it makes, how long a turn takes and how many tokens it uses. Fable 5.2 has five levels; the API default is high. Anthropic advises starting at high and moving up or down against your own evals.
  Source: Anthropic, Prompting Claude Fable 5.2 (section "Calibrate effort"); Anthropic, Effort.
  Link: https://platform.claude.com/docs/en/build-with-claude/prompt-engineering/prompting-claude-fable-5-2 , https://platform.claude.com/docs/en/build-with-claude/effort
  Context: Vendor documentation as of 17 November 2026.
- **Claim:** In the API, the level goes in the field `output_config.effort`. In Claude Code, there is the `--effort` flag at launch and `/effort` in a session; the status line shows the current level.
  Source: Anthropic, Effort; Claude Code, model configuration; own check in Claude Code 2.1.326.
  Link: https://code.claude.com/docs/en/model-config
  Context: The episode deliberately does not name the default in Claude Code, because the sources disagree.
- **Claim:** In Anthropic's testing, Fable 5.2 at low scored close to what its predecessor reached at higher effort. At high and above, the model thinks before almost every reply. Per-message effort (beta) keeps the prompt cache.
  Source: Anthropic, Prompting Claude Fable 5.2.
  Context: Anthropic's own testing, not re-measured by us.

The measurement

- **Claim:** A real bug fix from our client portal (15 files including tests), reset to the state before the fix. Three runs each at low and high with Claude Code headless (`claude -p --model claude-fable-5-2 --effort <level> --output-format json`), project settings only. Prompt: ticket text plus one sentence about tests and type check. Graded with the 84 tests of the real fix plus the type check; before the fix, 5 of the 84 fail.
  Source: Own measurement (17 November 2026).
  Context: One task, three runs per level, a well-specified ticket. A vague task and medium were not tested. The tests check behavior, not code structure. All six runs ran at the same time. Cost is the API list price.

- **Claim:** All six runs passed. low on average 5:28 minutes, 4.01 US dollars, about 31,900 output tokens, 92 to 98 turns. high on average 14:35 minutes, 10.17 US dollars, about 80,400 output tokens, 133 to 171 turns.
  Source: Own measurement (17 November 2026), individual values in the show notes.
  Context: Time is the wall-clock time of the whole call.
- **Claim:** On long, multistep tasks, Fable sometimes ends a turn before the work is done; more often at low and medium.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Finish the whole task").
  Context: Our runs took minutes, not hours.

Eight prompt changes

- **Claim:** Finish-the-task text: in Anthropic's testing at medium, about half as many sessions with unfinished work, about 30 percent more output tokens; endings that list what remains were not removed.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Finish the whole task").
- **Claim:** A line like "answer directly without deliberating" no longer stops the model from thinking at medium and above.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Calibrate effort", "Review prompts and harnesses designed for earlier models").
- **Claim:** An approval rule for "sending messages to people outside this conversation" made the model hold back about half of its routine requests in simulated tasks.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Approvals").
- **Claim:** Ownership sentence: duplicated code consolidated in about 75 instead of under 10 percent of runs, about 70 to 80 percent more output tokens, plus unrequested changes.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Say who owns the code").
  Context: Anthropic's testing on small coding tasks.
- **Claim:** Progress updates come back as thinking blocks, empty at the default setting; set `thinking.display` to `"updates"` (beta) or `"summarized"`. With the system-prompt sentence, a message before the first tool call in every run, about 20 percent more output tokens.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Keep users informed during long turns").
- **Claim:** How much Fable delegates depends on the harness. In Anthropic's testing at high, a longer, cost-focused tool description roughly halved the number of subagents.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Set expectations for subagents").
- **Claim:** Caution piles up; "never" is treated as absolute; a general "act more freely" is too vague. Hard style rules make reports choppy.
  Source: Anthropic, Prompting Claude Fable 5.2 ("Keep caution from piling up", "Make reports easy to follow").

The harness

- **Claim:** Changing the system prompt or tools in a resumed conversation can fail the prefix check (error 400). Harness code that edits earlier turns invalidates thinking blocks. A token budget shown in the prompt changes spending; in Anthropic's testing at high and xhigh, removing a large budget lowered cost with equivalent performance.

Source: Anthropic, Prompting Claude Fable 5.2 (note block, "Review prompts and harnesses designed for earlier models").

Audit

- **Claim:** `/claude-api prompt-audit` checks prompts, skills, CLAUDE.md and agent files for text written for older models and returns a report with file, line and a proposed diff, without changing anything.
  Source: A Claude Code tool.
  Context: Not part of Anthropic's prompting guide.

Guide

- Our guide with all prompt texts: https://www.linkedin.com/feed/update/urn:li:activity:7528489810681044992/

Healthy AI First is AI-researched and AI-generated, narrated in an AI version of my voice and reviewed by me (Björn).