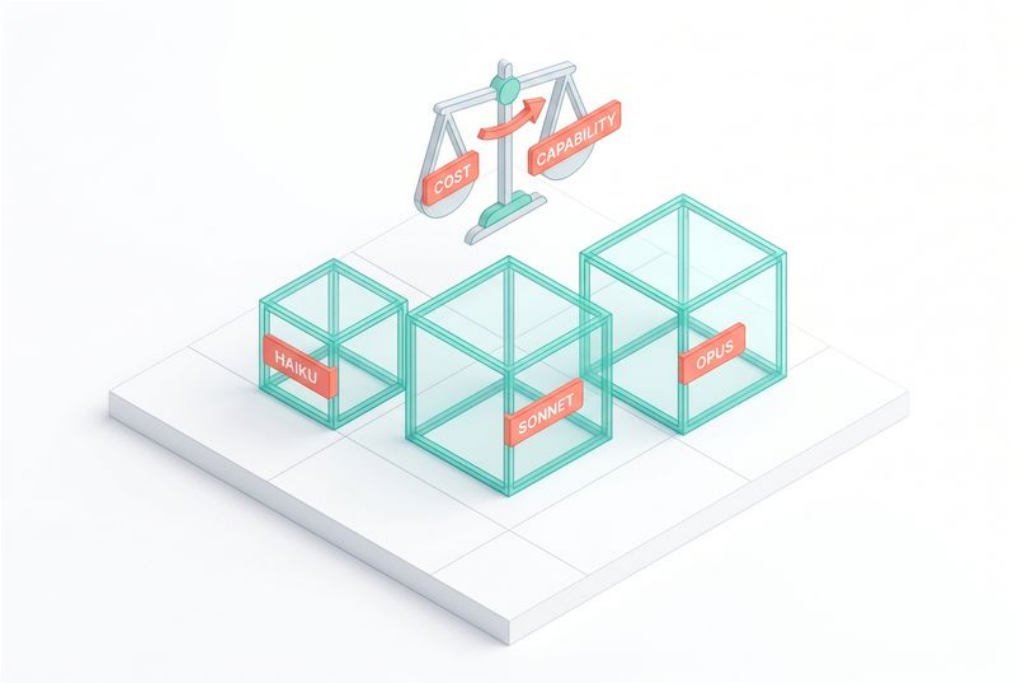
Picking the Right Model

Haiku, Sonnet, Opus. Cost vs capability vs latency, with a decision tree.



Three lanes

- **Haiku** — cheap one-shots, classification, routing, simple extraction
- **Sonnet** — production traffic, daily engineering tasks, the default
- **Opus** — long-horizon agentic work, ambiguous tasks, code that requires deep context

Decision questions

- Will the agent run for more than 30 minutes? Opus is usually worth the cost
- Is the task structured and bounded? Sonnet is enough
- Is the task a classification or a one-shot? Haiku is the right call
- Is the work production traffic with thousands of users? Cache aggressively, route to the smallest model that passes evals

Three gotchas

- Eval noise: small differences between runs are usually noise, not signal
- Infrastructure overhead: latency budgets often dominate model choice
- Cost saturation: prompt caching at 1/10th list price changes the math

From "Claude Code, Definitive Guide for 2026" — Chapter 2. Source: Picking the right model talk, Code with Claude London 2026.