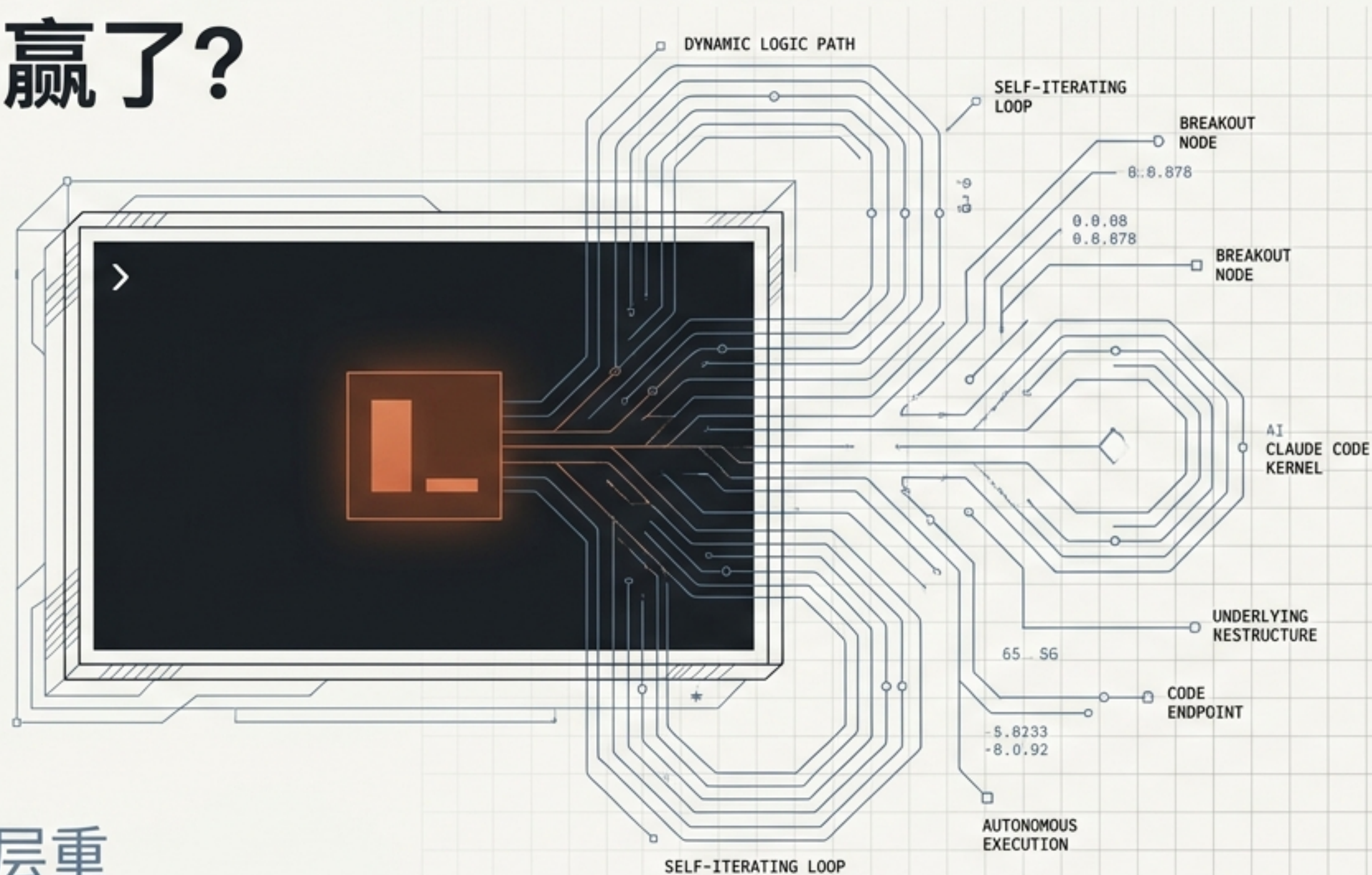
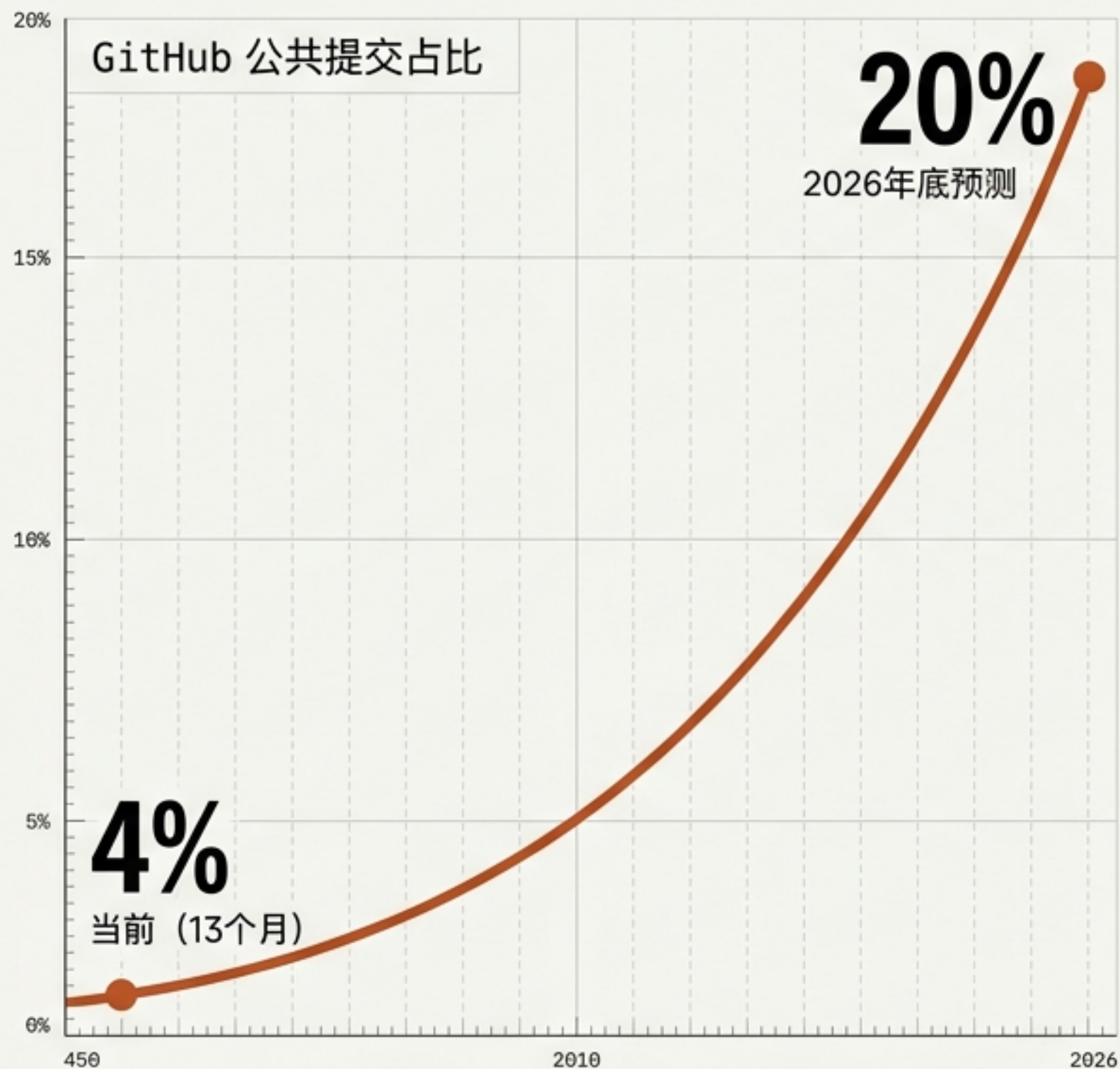


AI Agent 的真正拐点：为什么偏偏是 Claude Code 赢了？



一份关于代码终结与底层重
构的深度洞察报告

渗透率极速飙升：从开源社区到火星探测



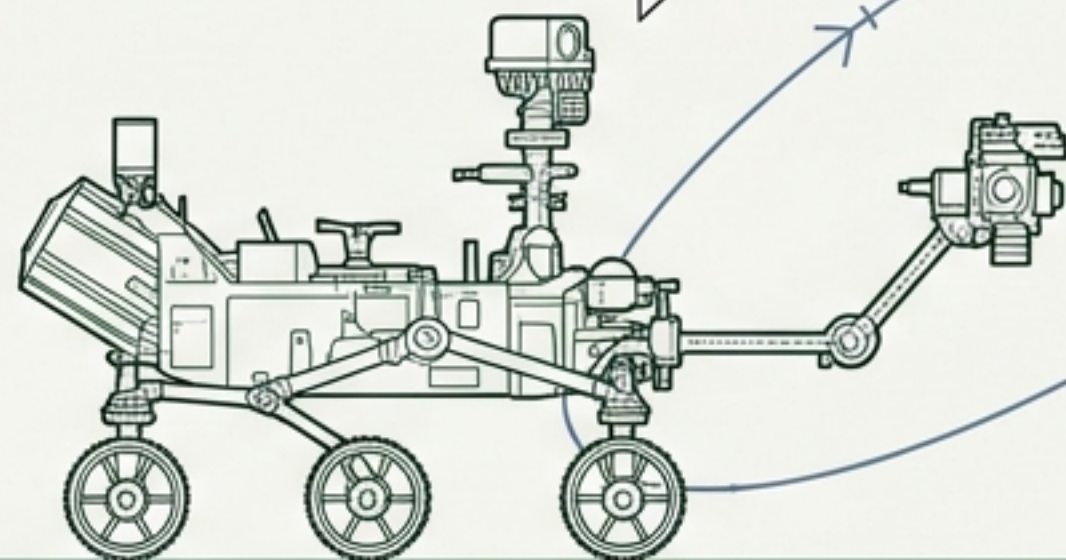
70%

作为首选模
型的创业公司
比例

(Mercury 数据)

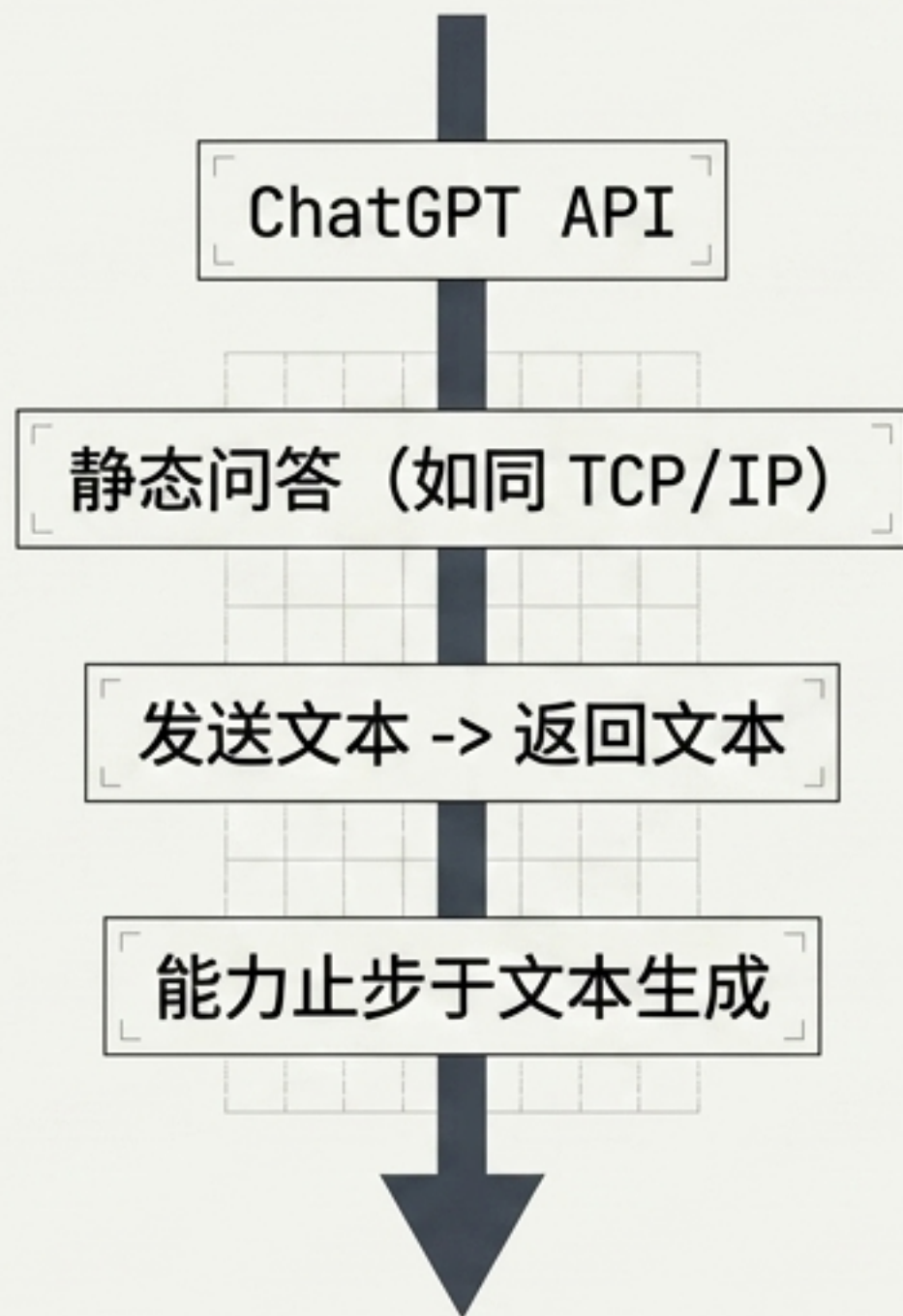
这是最酷的事。—— Boris Cherny

NASA 火星探测器 (Perseverance) 轨道计算

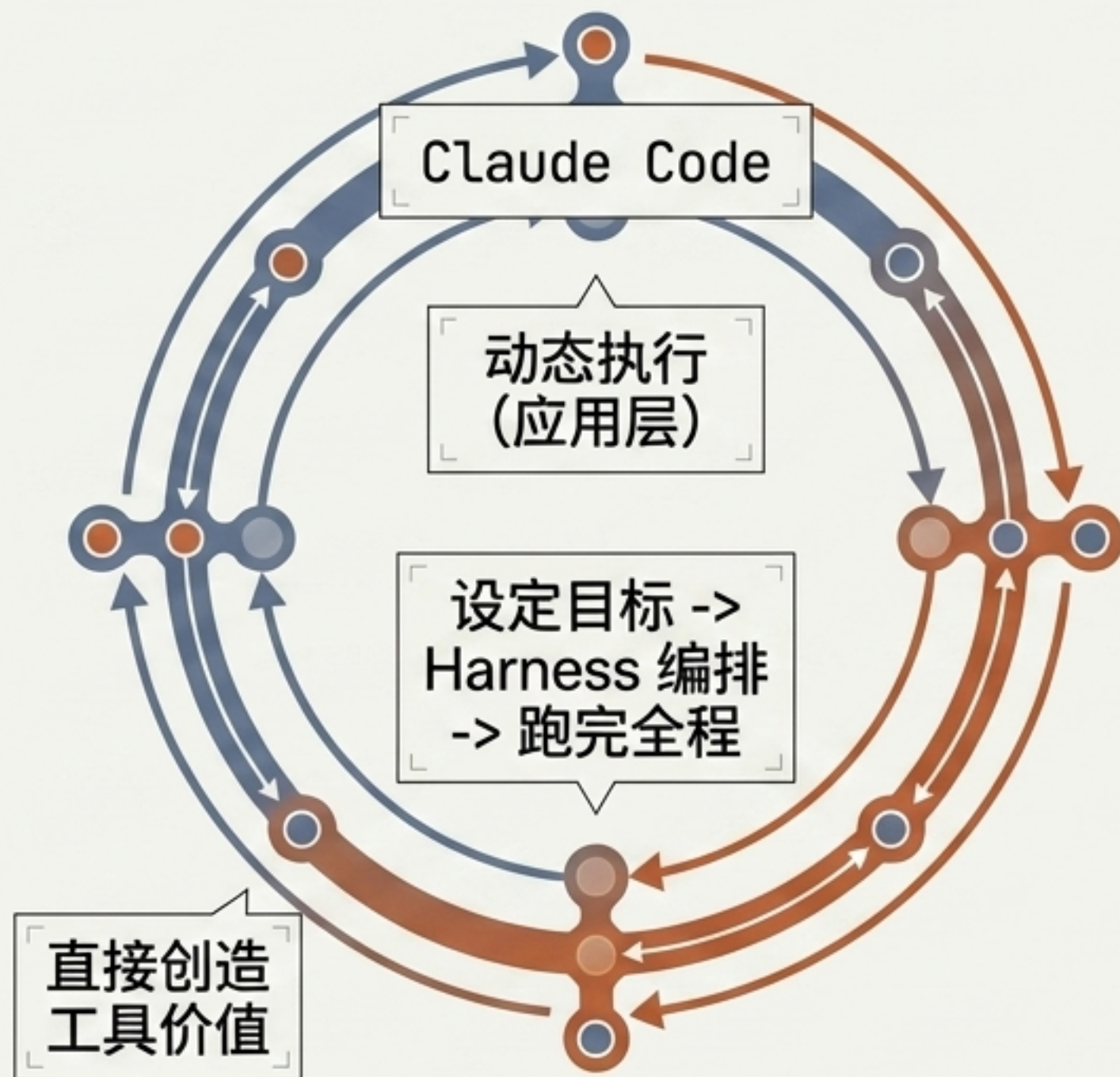


AI 时代的 Web 2.0 时刻

旧范式：Web 1.0



新范式：Web 2.0



意料之外的涌现：模型主动寻求工具

2024年9月原型

TypeScript 终端 + Bash 工具

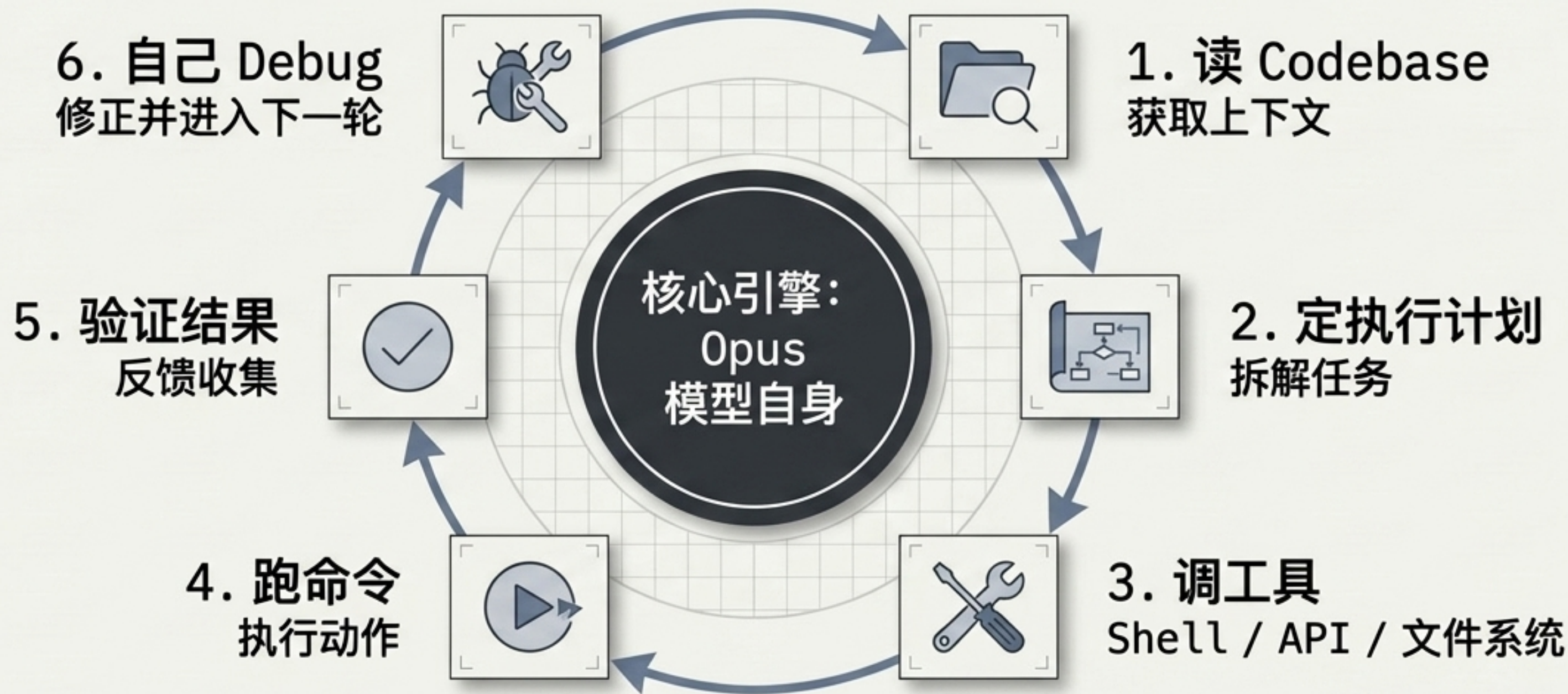
User: 现在在听什么歌?

Sonnet 3.5 (自主决策):

-
- ```
graph TD; 1((1. 编写 AppleScript)) --> 2((2. 控制 Mac 系统)); 1 --> 3((3. 查询音乐播放器)); 3 --> 4((4. 返回播放结果));
```
1. 编写 AppleScript
  2. 控制 Mac 系统
  3. 查询音乐播放器
  4. 返回播放结果

那是第一次**真正有 AGI 的感觉**：这个模型只是想要使用工具。这就是它想做的全部事情。

# 底层重构：从「文本生成」到「任务编排」飞轮

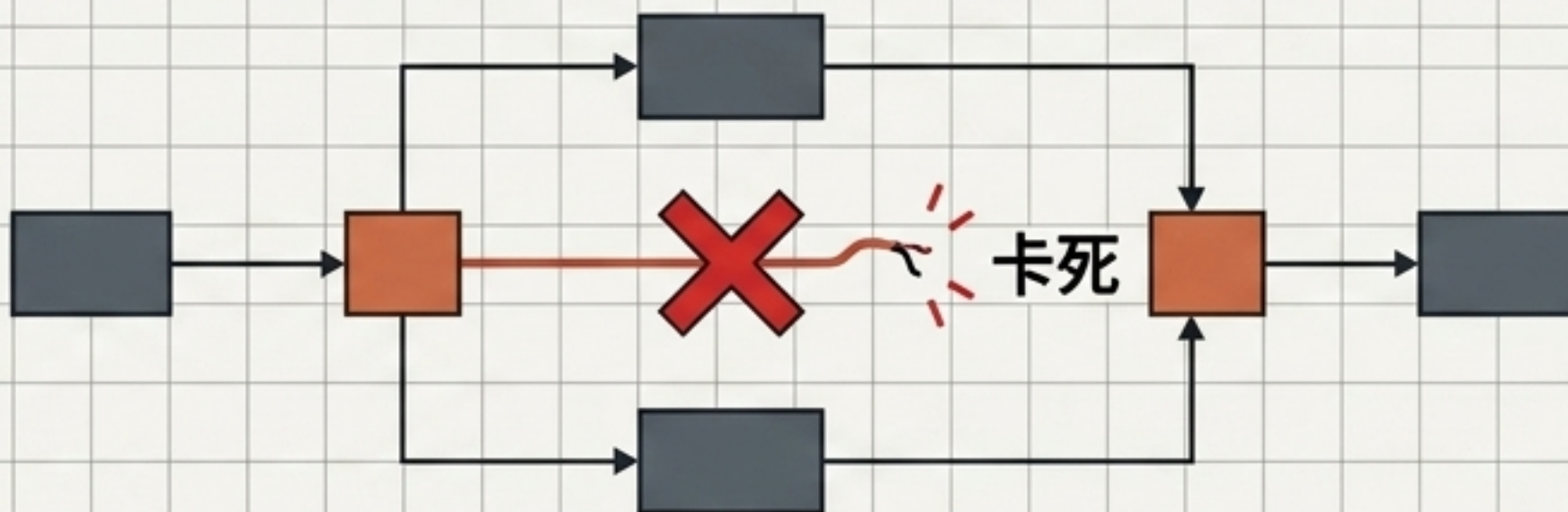


能力上限不再是预设逻辑，而是模型指数级增长的判断力。

# 打破预设规则的绝对边界

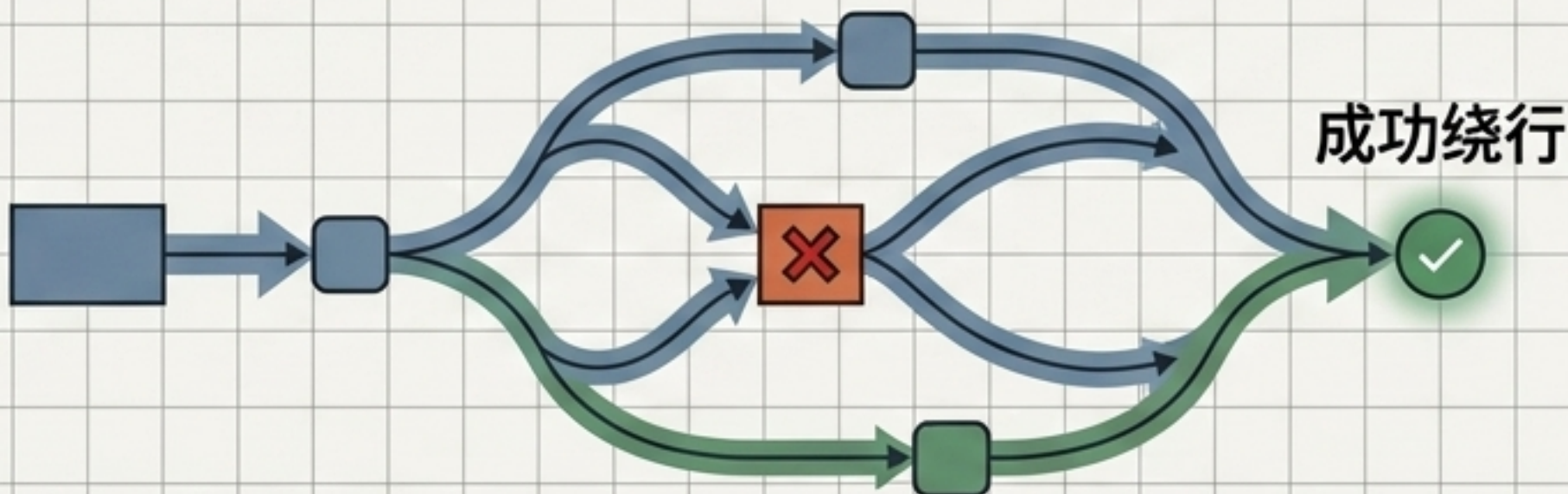
## 传统自动化 (CI/CD, RPA, 脚本)

- 规则写死
- 依赖条件判断 (if-else)
- 遇到意外即刻卡死崩盘



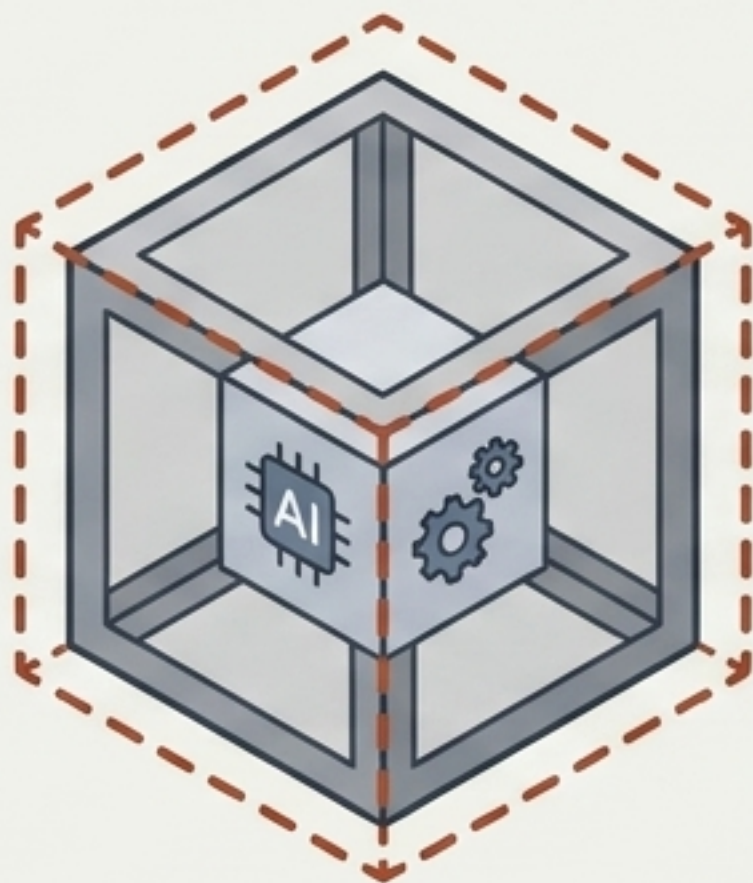
## Agentic 范式 (Claude Code)

- 理解原始意图
- 动态规划路径
- 一条路不通自动换一条路重试



# 反直觉的选择：为什么是“简陋”的终端？

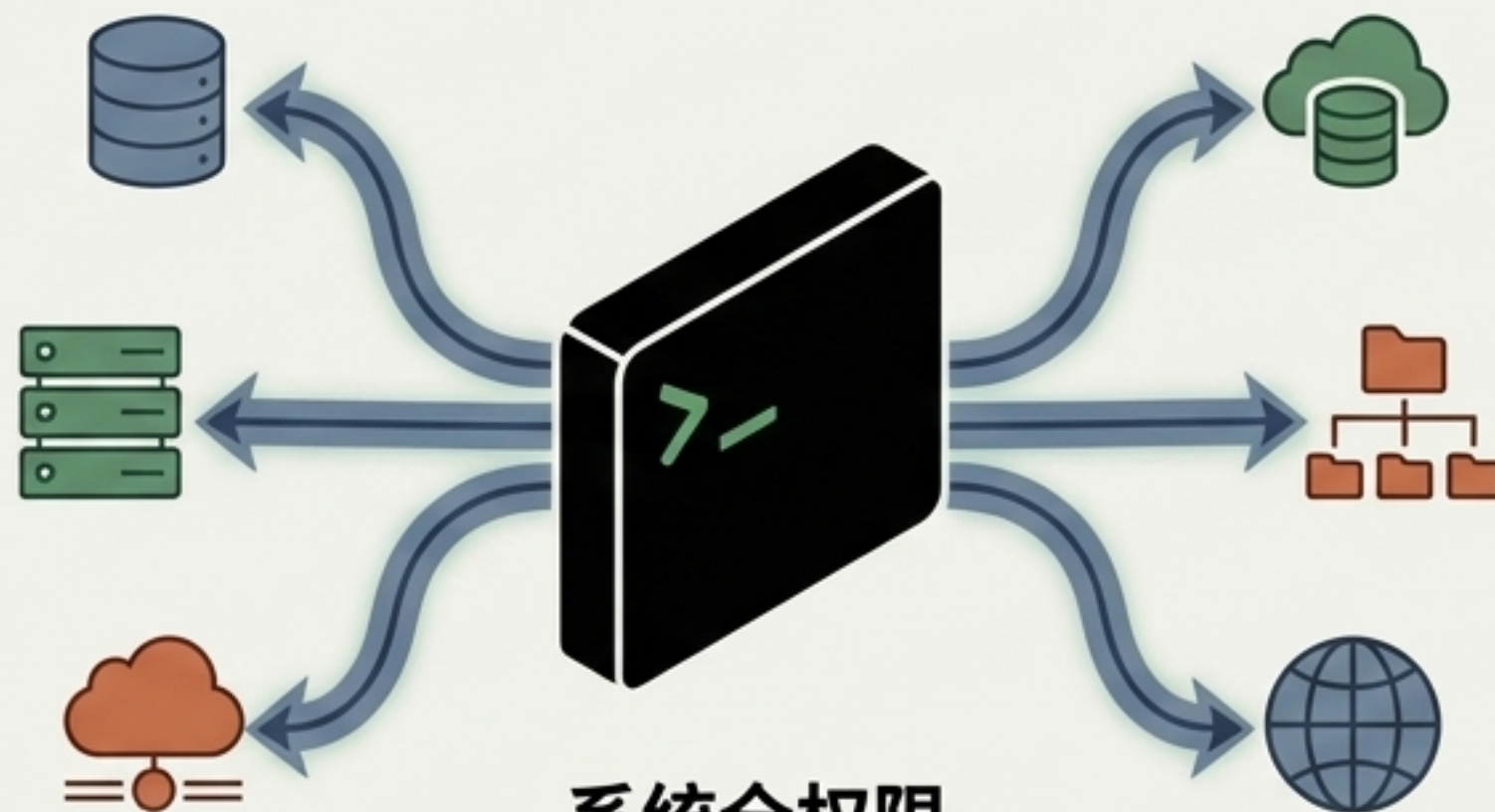
## IDE 插件 (如 Copilot/Cursor)



### 受限沙箱

- 本质是“人在写代码，AI 搭把手”

## Terminal 终端 (Claude Code)

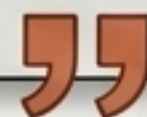


### 系统全权限

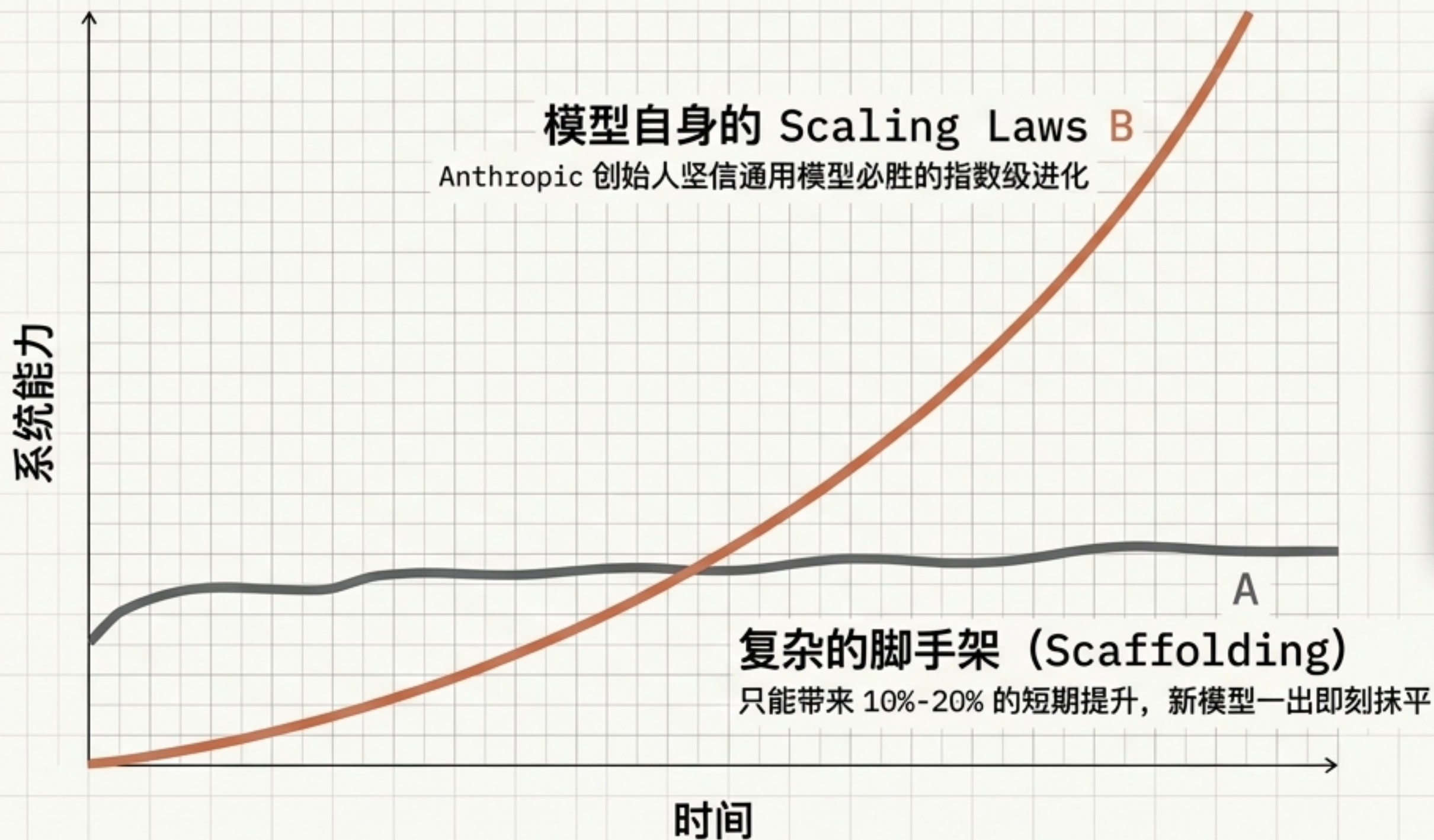
- 极高耐久度
- 跑 shell、部署服务、AI 跑全流程



DAU 图表完全垂直上升，你们在强制工程师使用吗？—— Dario Amodei



# 核心信条：为 6 个月后的模型设计产品



## The Bitter Lesson

“Claude Code 没有任何一行代码是六个月前写的。”

放弃过度工程，将全部筹码押注于模型的自我进化。

# 自我强化的创造循环：Claude 制造 Claude

100%

彻底颠覆

- 100% 由 AI 编写
- 卸载 IDE，全天候仅用 Claude Code (Opus 4.5)
- 每日提交 10-30 个 PR

4 人 10 天

团队效率

- 开发全新功能 Cowork 的惊人周期

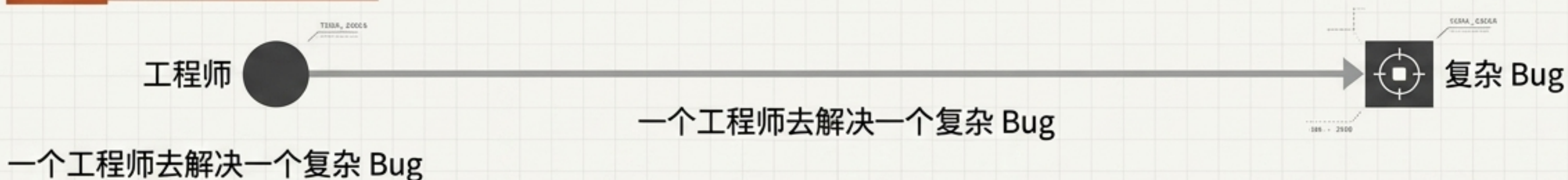
30 分钟

极速上线

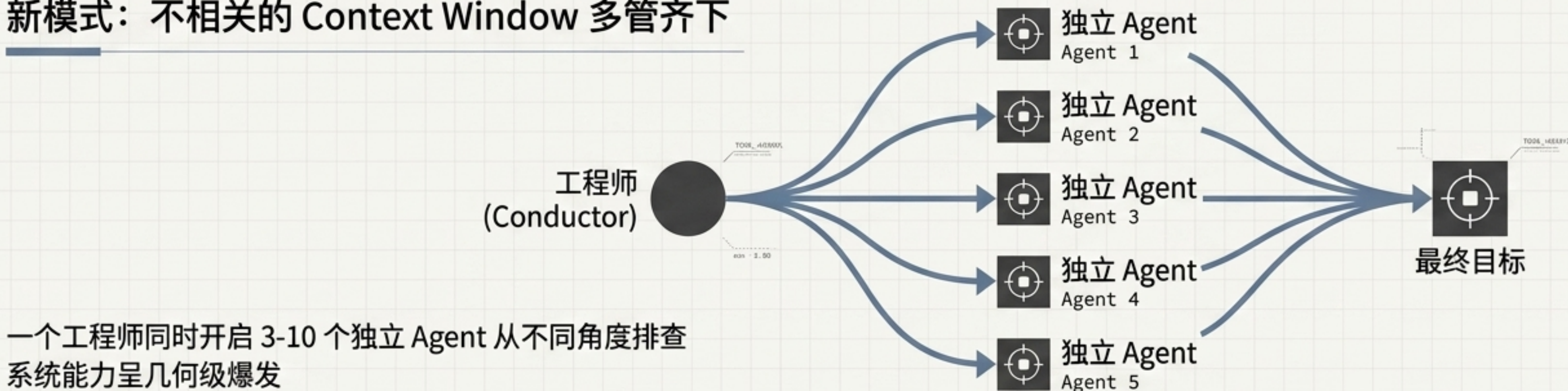
- 30 分钟上线
- 写出 Plan Mode。只需在 Prompt 加入“请先不要写代码”

# 全新的 workflows: 并行 Agent 的降维打击

## 旧模式: 单线程排队



## 新模式: 不相关的 Context Window 多管齐下



# 生产力数据的质变：1000倍工程师的诞生

一个 Anthropic 工程师现在的平均生产力，是巅峰期 Google 工程师的 1000 倍。

—— Steve Yegge (前 Google 工程师)

Meta 传统模式

耗时一年优化，仅提升数个百分点

Anthropic + Claude Code (150%)

生产力瞬间跃升 150%

# 行业共识：手写代码时代的彻底落幕



**Andrej Karpathy**

明显感到手动编写代码的能力在退化...代码生成和解读在大脑里已是两种不同能力。



**Ryan Dahl (NodeJS 作者)**

人类编写代码的时代已经结束。



**DHH (Rails 作者)**

手写 Ruby 代码如今已是奢侈体验，这门手艺或许很快会失传。

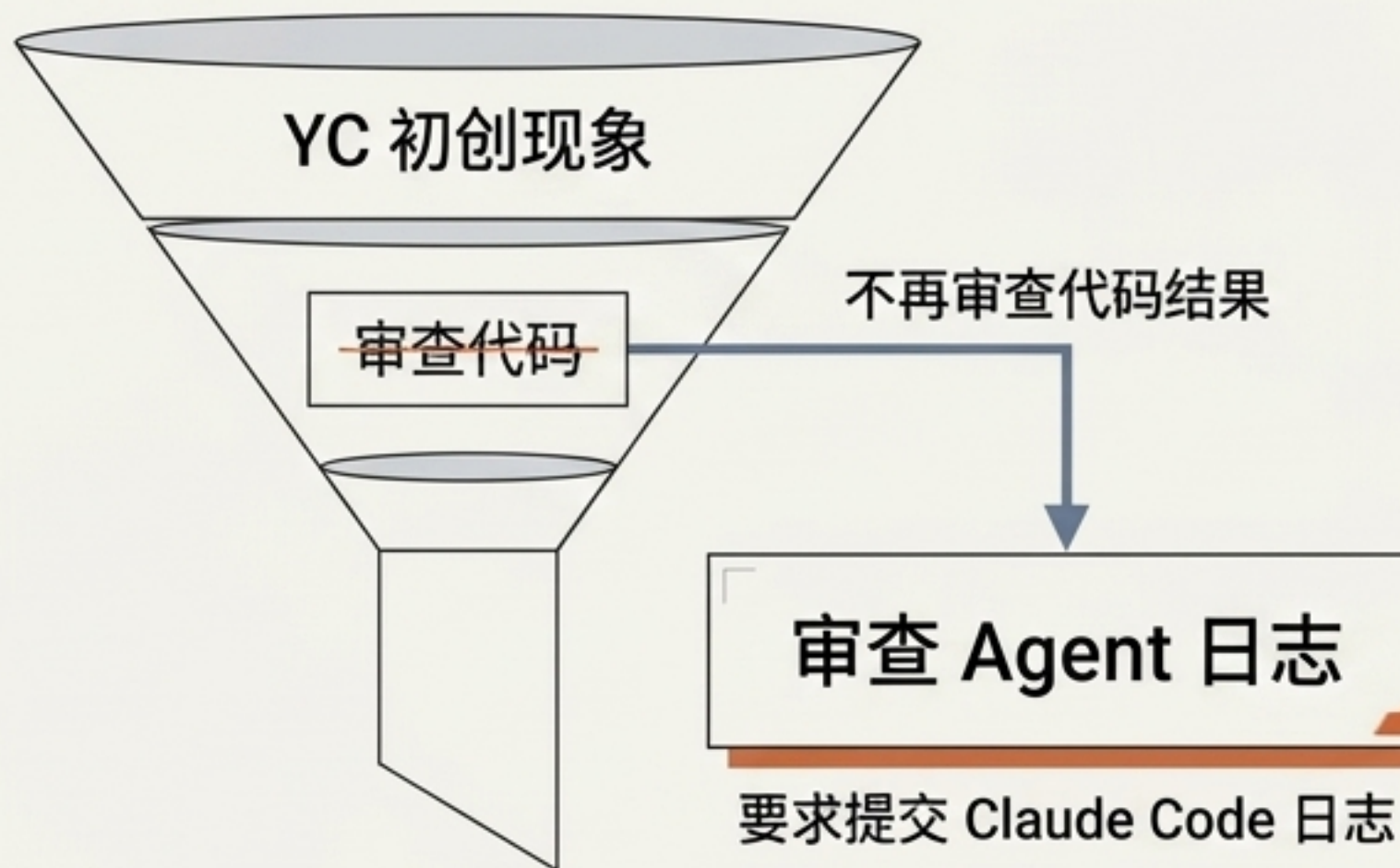


**Linus Torvalds**

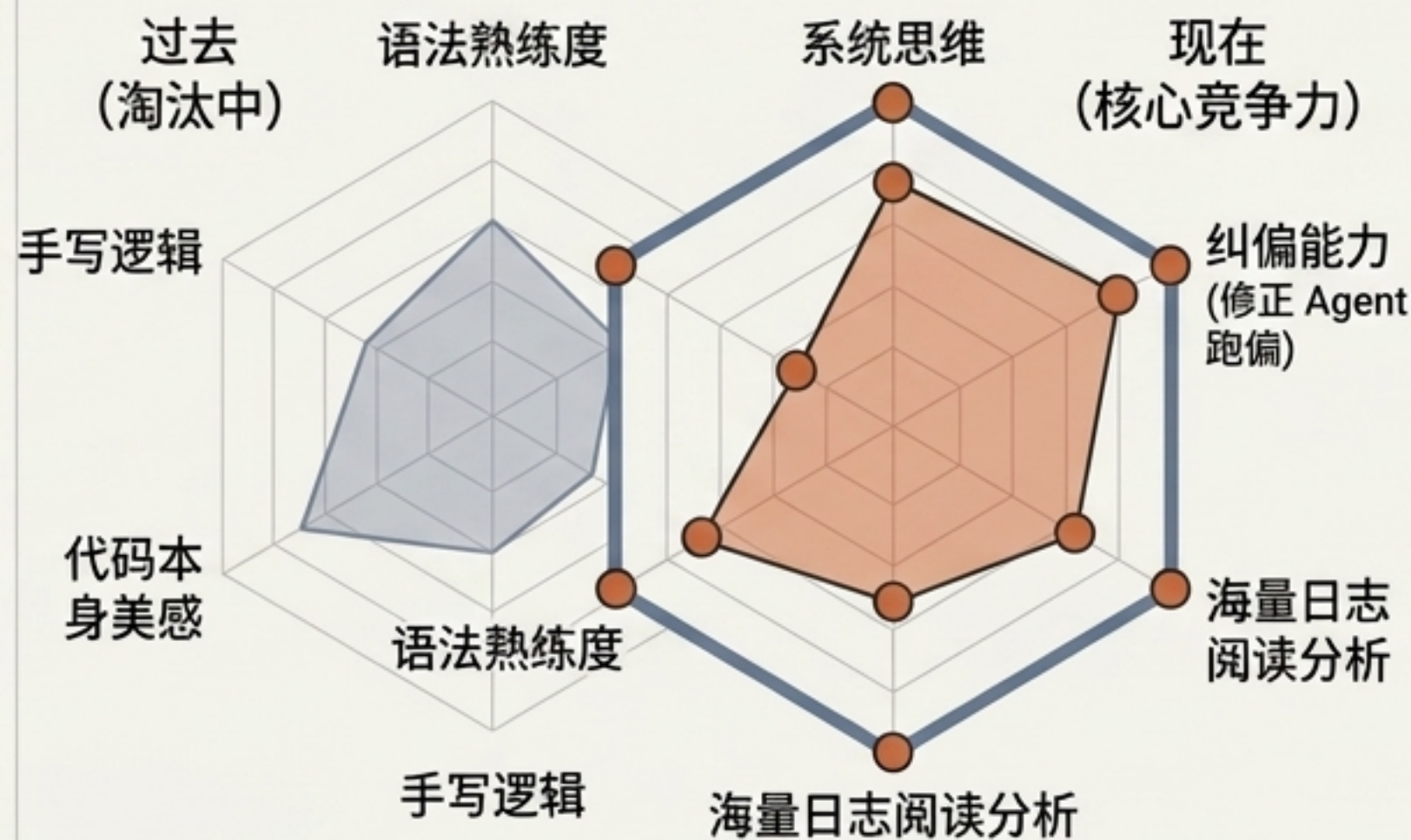
已全面转向 AI 辅助编程。

# 人的重构：从 Software Engineer 到 Builder

## 筛选标准转变（YC 初创现象）



## 技能重构矩阵



结论：你不再是写代码的工人，而是驾驭 Agent 的产品经理（PM）和构建者（Builder）。

# 终局视界：重构的下界与上界

