



AI透镜系列研究

*AI Coding*

**非共识报告**



# AI Coding：超越软件的编程范式革命

长久以来，编程被定义为一种严谨的、逻辑驱动的活动，是将人类意图转化为机器可执行的、确定性指令的过程。然而，AI正在颠覆这一核心定义，将编程从“Coding”这一动作，提升到“表达意图”和“实现愿景”的更高维度。

1995年的比尔·盖茨在《未来之路》中预言“信息高速公路”的到来，在那个时间节点，很难想象今天每个人都能随时随地连接全世界。我们现在又正站在了新的转折点上，这不仅仅是技术更迭或者工具升级，软件开发正经历一场范式级变革，更是对三个底层问题的重新定义：

- 什么是编程？
- 谁能编程？
- 数字世界将如何被构建和消费？

和信息高速公路不同的是，这场范式变革正在以惊人速度和共识，从想象走向现实。在AI Ascent 2025上，红杉资本把AI Coding的进行时和未来时放在了更具有前哨视角的位置，预测我们正在步入“丰饶时代”，代码是第一个被颠覆的市场，而这场颠覆将成为“丰饶时代”的预演。并且认为Coding Agent的持续演进值得拭目以待，因为这不仅会重塑整个软件产业，也将成为未来其他行业AI化进程的重要先兆。

这意味着当我们对产品、模型、价值、投融资、商业化、竞争策略等多层镀膜的认知镜头，对准飞速演化的AI Coding现场，从微软、谷歌、Meta、亚马逊、Salesforce等领先企业的内部调研数据和CEO访谈，到招聘数据到开发者、创建者的广泛调研，再转到对AI Coding生态中核心创新企业的系统分析，以及深入创始人或核心构建者的近150个深度访谈洞察时，也在用一个独特的“AI透镜”看向软件行业的演进和未来行业的AI化进程。

编程范式的重构

“丰饶时代”的预演

增长和变革最快领域

所有行业AI化进程先兆

在数据透镜之下，AI Coding从消费者端到企业端，渗透率、采用率和影响力都在飞速地跨越边界。这种断层式的领先优势，在融资市场上得到了最直接的体现，但更令人印象深刻的是，这个行业在收入增长方面创造的奇迹。

AI创新视角下，AI Coding公司正在以极小的团队规模、极快的发展速度达到千万、过亿美元甚至5亿美元的年度经常性收入（ARR），冲击大公司的增长和商业模式。AI转型视角下，编程是企业内部AI变革发生最快、效果最为显著的应用方向。从微软Github Copilot开始，AI Coding已成为融资规模仅次于基础大模型、增长最迅猛的AI应用方向之一，甚至出现了成立仅3年的百亿美元公司。爆发式增长的背后，反映的是产业生态的系统性重构。

但是当我们把透镜对准产品、价值和商业化竞争时，我们又能发现在AI Coding大方向的共识下，从数据到认知，再到行动等七个方向，都存在着巨大的非共识。

非共识01：AI Coding的最佳产品形态是什么？——本地 VS 云端

非共识02：AI Coding产品选择什么模型？——自研 VS 第三方

非共识03：AI Coding给用户的价值有多大？——提效 VS 降效

非共识04：AI Coding产品理想付费模式是什么？——固定 VS 按需

非共识05：大企业推进AI Coding应用的态度？——激进 VS 渐进

非共识06：AI Coding对组织发展的影响？——裁员 VS 扩张

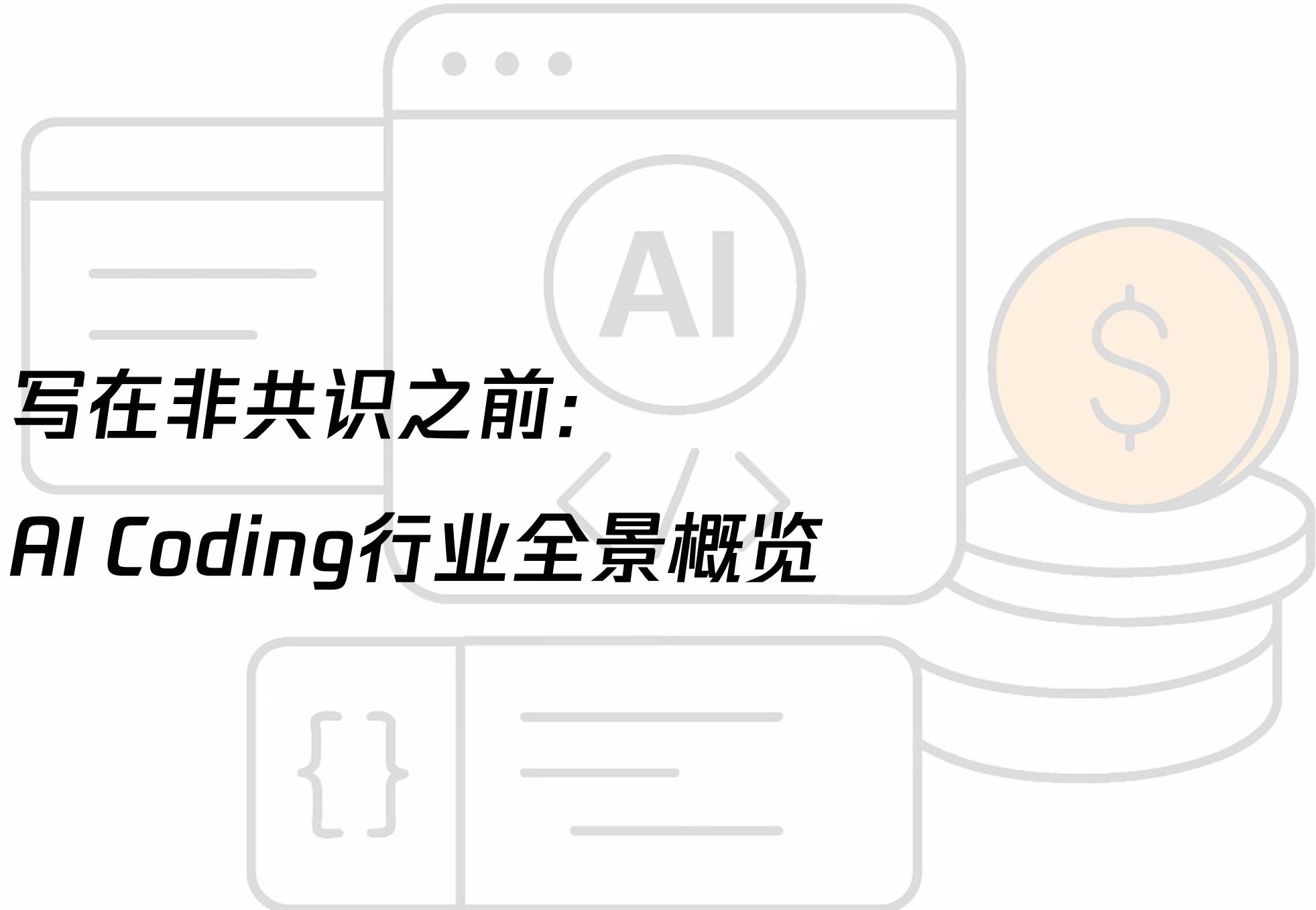
非共识07：AI Coding的未来市场格局是怎样？——专业 VS 普惠

如果说市场数据描绘了“正在发生什么”，那么这些非共识，则将指向“为什么会这样”以及“未来将走向何方”，以及在“丰饶时代”的大共识下，还有哪些产品及技术方面的共识有待达成。



# 对AI Coding领域的全面调研

- 微软、谷歌、Meta、亚马逊等头部企业的调研数据
- Indeed招聘数据、GitHub年度开发者调研
- AI Coding主要产品分析，包括Cursor、Replit、bolt.new、devin、windsurf、Github、Claude code、V0、Open AI Codex、Lovable、Augment、Magic.dev、Poolside AI、Reflection AI
- 相关产品工具创始人的近150个访谈



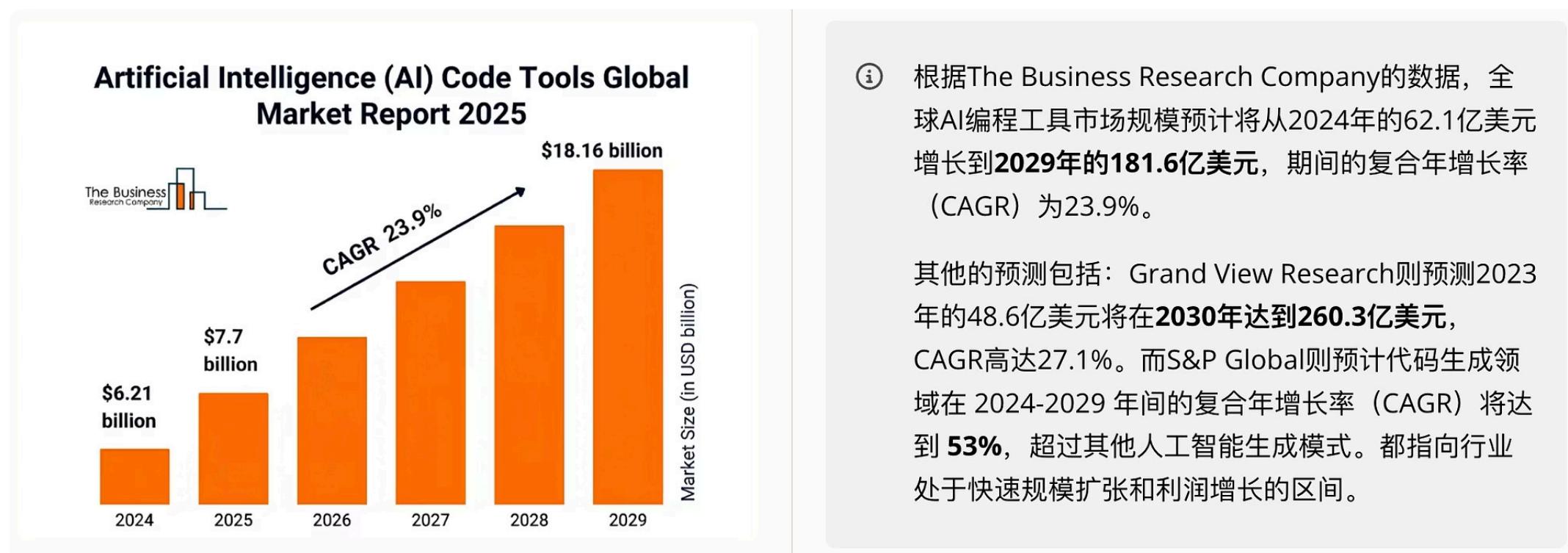
# 写在非共识之前： *AI Coding* 行业全景概览

# 概览01: AI Coding在C端的渗透率仅次于写作支持，以47%排名第二

为了准确了解AI工具在消费者群体中的实际应用情况，知名风险投资机构Menlo Ventures与专业市场调研公司Morning Consult于2025年4月联合开展了一项大规模调研。该调研针对5031名美国成年人，深入探讨了消费者在过去六个月中如何使用AI和其他工具完成特定活动，以及他们对这些技术的态度和接受度。



在消费者端，AI 渗透率最高的活动中，编程位居第二，渗透率达到47%，也就是接近一半的受访者已经在日常生活或工作中的编程都已经开始采用AI，广泛接受度仅次于写作支持。47%的渗透率表明AI编程工具跨越了早期采用者的边界，进入主流市场接受阶段，反映的是产品易用性和实用价值的显著提升。市场成熟度的指标和用户群体的扩展，直接指向了行业的整体规模的爆发增长。



## 概览02: AI Coding在B端是落地最快和影响力最大的AI应用方向

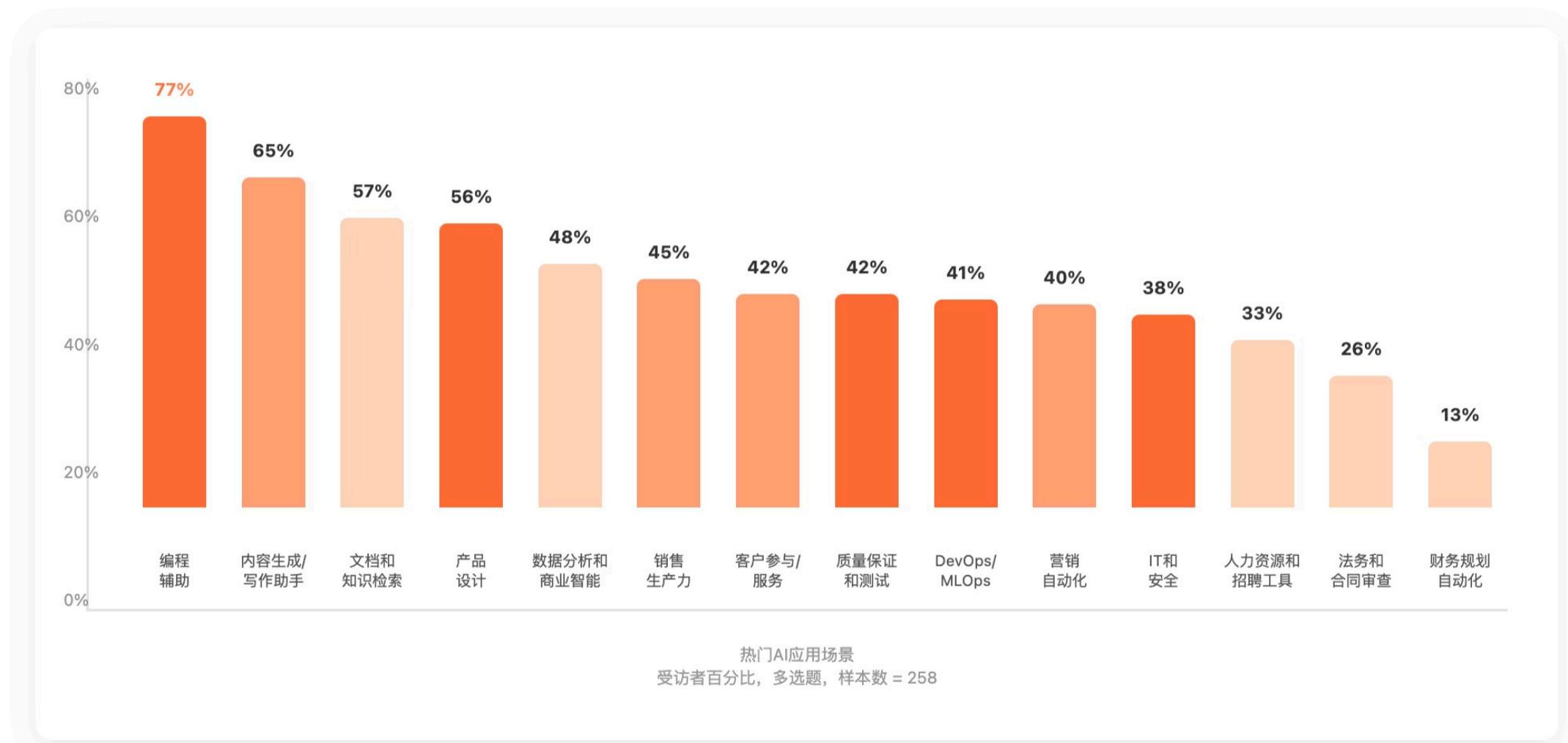
在企业内部的AI用例和转型采用上，通过对多个权威机构调研数据的综合分析，我们发现AICoding在企业级AI应用中展现出了显著的领先优势，不仅仅是落地最快、最佳实践最多、影响力最大的行业方向，而且一直保持断层式领先的优势。这种领先地位得到了不同时间、不同样本、不同调研机构的一致验证。

Menlo Ventures在2024年9月，对美国600名拥有50名或以上员工的企业IT决策者进行的调查，数据显示企业的AI落地51%都集中在代码生成上，远超客服聊天机器人的31%的数据。

### 热门AI应用场景：按普及度排名

研发和销售营销部门的AI应用领先，而综合管理部门相对滞后

■ 研发部门 (R&D) ■ 销售营销 (S&M) ■ 综合管理 (G&A)



The Builder's Playbook-2025 State of AI Report

ICONIQ作为硅谷的头部成长期投资基金，2025年4月对300位软件公司构建人工智能产品的执行高管进行了调查。最新数据显示，编程辅助在企业落地中的普及度数据已经达到了77%，影响力也以65%的比例断层领先，呈现广泛使用又效果显著的特点，是当前最成熟的AI企业落地方向。除了编程辅助之外，AI应用普及度排名中，和编程相关的产品设计、QA测试和DevOps等AI进展也较为突出，几乎都达到了超过40%进展，R&D部门已经初步形成了完整的AI工作流，从设计→开发→测试→部署。

# 概览03: AI Coding企业收入增长快, ARR记录屡被刷新

2022年6月Github Copilot 正式上线，3年时间内AI Coding领域出现了几乎前所未有的增长率，屡破年度经常性收入（ARR）记录。



Cursor从100万到1亿美元，用时12个月，打破了行业记录，到2025年6月已突破5亿美元。Bolt.new5个月内突破了 5000 万美元 ARR，Lovable7个月内达到7500万美元ARR。Replit2025年6月宣布ARR突破1亿美元，而2024年底时仅为 1000 万美元，不到 6 个月内实现了10倍增长。Vercel在6月也宣布ARR突破 2 亿美元，15 个月内实现收入翻倍。

**CURSOR**

5亿美元  
打破历史记录

**GitHub Copilot**

4亿美元  
年增长4倍

**replit**

1亿美元  
6个月10倍增长

**lovable**

7500万美元  
7个月时间

## 概览04：AI Coding领域独角兽扎堆，估值高吸引投资者广泛关注

在消费者和企业采用端的进展，以及收入的突飞猛进，让AI Coding成为生成式AI浪潮最具商业价值和投资回报的赛道之一，出现融资热潮。赛道中，B轮过亿美元融资成标配，独角兽扎堆出现，出现成立仅三年的百亿美元公司。

2025年6月，Cursor的母公司Anysphere完成了AI编程领域史上最大的单轮融资—9亿美元C轮融资，估值高达99亿美元。Cognition AI创下了一个月内估值从3.5亿美元飙升到20亿美元的记录。

融资狂飙目前还在继续，7月17日，Lovable宣布完成2亿美元融资，估值18亿美元，超过此前的1亿美元15亿估值的融资预期。最近有报道称Cursor正在接洽180亿-200亿美元的新一轮融资。Replit也正在洽谈新一轮约2亿美元融资，预计融资后估值将接近30亿美元。

公司名称	轮次	融资时间	金额	估值	创立时间
Anysphere (Cursor)	C轮	2025年6月	9亿美元	99亿美元	2022
Poolside AI	B轮	2024年10月	5亿美元	30亿美元	2023
Cognition AI (Devin)	A+轮	2024年4月	1.75亿美元	20亿美元	2023
Windsurf	并购	进行中	24亿美元	24亿美元	2021
Lovable	A 轮	2025 年7月	2亿美元	18亿美元	2023
Replit	B+轮	2025年2月	9740万美元	11.6亿美元	2016
Reflection AI	A 轮	2025年3月	1.05 亿美元	5.55 亿美元	2024
Augment	B轮	2024年4月	2.27亿美元	9.77亿美元	2022
Bolt.new	B 轮	2025年1月	1.055 亿美元	未披露	2017
Magic	B轮	2024年8月	3.2亿美元	未披露	2022

截止到2025年7月

今年行业还出现了两起并购案例，分别是Windsurf和Base44。OpenAI、Google DeepMind、Cognition AI在Windsurf的收购中出现戏剧性的走向，最终Google以24亿美元收购Windsurf部分高管并授权其技术，Cognition AI收购Windsurf IDE的完整所有权和剩余的Windsurf团队和技术资产。Base44作为一个仅有6个月历史的产品，被Wix以8000万美元收购，也在行业引发巨大讨论。

非共识01

*AI Coding的最佳产品形态是什么？*

——本地 VS 云端

# AI Coding 产品形态象限图

1

本地配置，随时随地  
有网即用

2

打通云资源全栈开发

3

调试预览分享协作丝  
滑

云端 AI 开发产品，迎合 Vibe Coding，致力人人可用



1

开发者最熟悉

2

使用最灵活

3

不强依赖网络和云服  
务

本地 AI 开发工具，贴近开发者、解放生产力

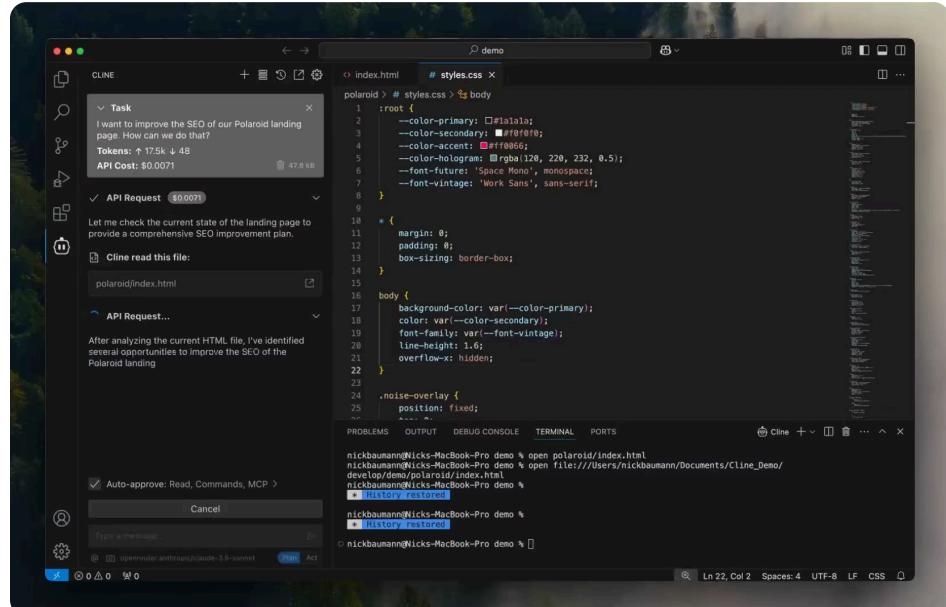
# 类型01: IDE&插件，从专业中来，到专业中去

从 VS Code 这一最流行的代码编辑器/IDE出发，发展出插件和派生两类 AI Coding 产品。

## 插件 (Extension)

作为 VS Code 的补充，根植于其生态之上，提供 AI 功能：

- 早期以代码补全插件为主，如 GitHub Copilot、Codium (Windsurf 前身)
- 现多支持 Agent 模式，如 GitHub Copilot Agent Mode、Cline 等



## 派生 (Fork)

当插件 API 约束了功能开发时，部分玩家选择基于 VS Code 开源版本改造，来实现更自定义的 AI 功能实现：

- Cursor
- Windsurf

A screenshot of the Windsurf AI development environment. It shows an IDE window with a Rust file (lib.rs) containing code for a Merkle tree. A floating window displays AI-generated thoughts and code completion suggestions. The status bar indicates the AI has thought for 10 seconds and provides a link to restore the checkpoint. Another floating window shows a plan to update constructors to handle depth.

```
impl MerkleTree {
    pub fn write_log_file() {
        Ok(())
    }

    pub fn get_path_depth(
        &self,
        absolute_path: &Path,
    ) -> Result<usize, anyhow::Error> {
        match self.find_path_ref(&absolute_path)? {
            MerkleRef::Branch(branch: MerkleBranchKey) => {
                if let Some(branch_node: &MerkleBranch) = self.nodes.branches.get(key: branch) {
                    Ok(branch_node.depth)
                } else {
                    Err(anyhow!("Branch not found"))
                }
            }
            MerkleRef::File(file: MerkleFileKey) => {
                if let Some(file_node: &MerkleFile) = self.nodes.files.get(key: file) {
                    Ok(file_node.depth)
                } else {
                    Err(anyhow!("File not found"))
                }
            }
        }
    }
}
```

- ① 本地 AI 开发工具必须要本地安装，使用本地资源和工具链等完成构建、编译、调试等工作，存在显著的上手门槛

# 类型02：命令行工具CLI，进可攻退可守通用潜力股

*Claude Code* 通过工程技巧让 *Claude* 在终端环境中充分发挥

- 当前口碑最佳的 Coding Agent
- 两核心负责人短暂被 Cursor 挖走又回归 Anthropic



单周写代码近 2 亿行，81% 开发者选择自主执行



命令行工具共享终端环境的所有可能性，意味着可以让 AI 模型自主决定查阅哪些资料、编辑哪些文件、执行什么操作，已经体现出一定通用性，如 Anthropic 团队就曾分享内部各岗位都在用 Claude Code 协助工作：

大厂开源跟进：

- OpenAI Codex CLI 轻量开源复现



The report title is "How Anthropic teams use Claude Code". It includes a section on "Anthropic's internal teams are transforming their workflows with Claude Code, enabling developers and non-technical staff to tackle complex projects, automate tasks, and bridge skill gaps that previously limited their productivity." A table of contents on the right lists various departmental uses of Claude Code, each with a page number:

Claude Code for data infrastructure	3
Claude Code for product development	5
Claude Code for security engineering	7
Claude Code for inference	9
Claude Code for data science and visualization	11
Claude Code for API	13
Claude Code for growth marketing	15
Claude Code for product design	17
Claude Code for RL engineering	19
Claude Code for legal	21

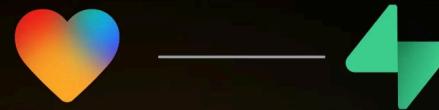
- Google Gemini CLI 开源+免费推广，尝试追赶

# 类型03: *Vibe Coding* 产品，普通人说话做软件

赶上2025年初的这波“氛围编程”热潮，一众云端开发产品纷纷上新AI功能，吸引普通用户“说话”就能打造自己的产品，如：Lovable、Bolt.new、Replit等。迎合vibe coding，此类产品陆续推出了诸多适用普通用户的特色功能：

打通后端数据库，实现全栈开发：

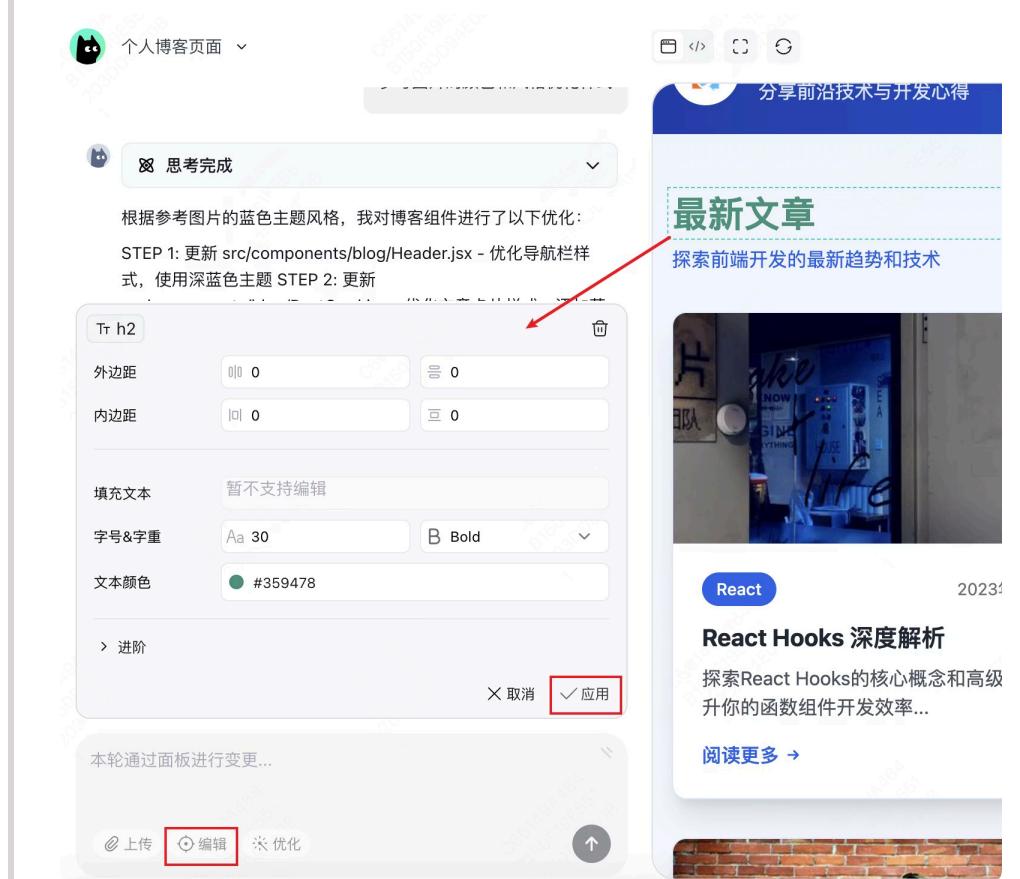
How Lovable's Supabase Integration Changed the Game



Author: Alex at Lovable

Four months ago, we shipped a game-changing native integration with **Supabase**.

可视化修改，实现相对精细的控制：

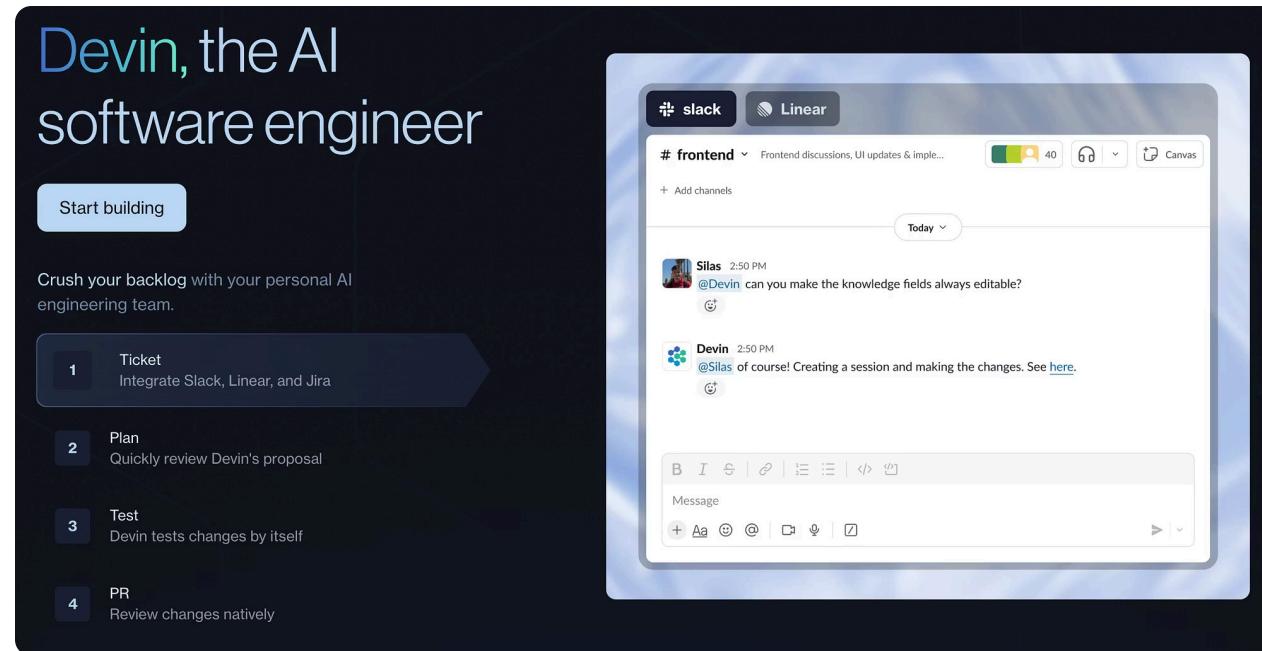


① 此类产品同时也面临难以胜任生产级系统、存在安全隐患等诸多质疑

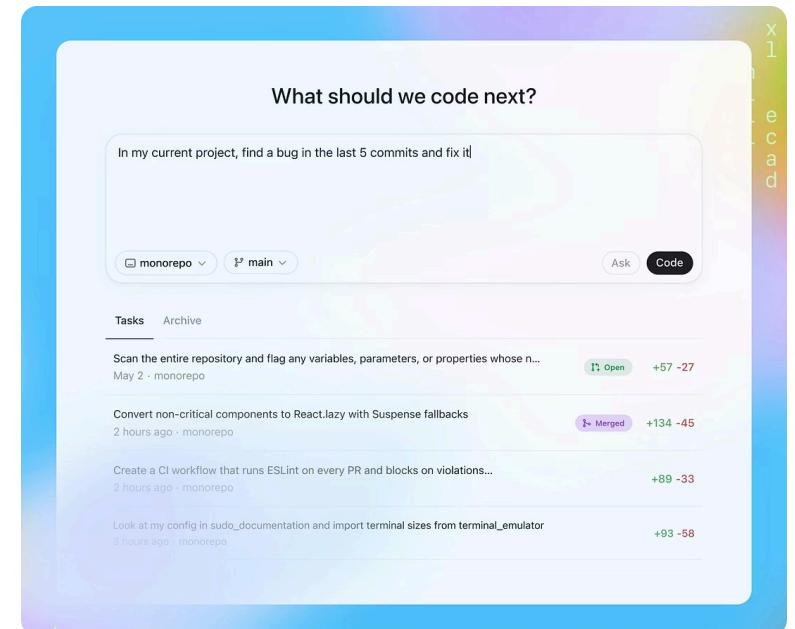
# 类型04：异步 Coding Agent, 7×24 的云端程序员

此类产品更突出 Agent 解决实际开发任务的能力，相对更面向专业场景，同时云端异步的属性让其可以更丝滑、更自主地接入现有开发工作流，如：

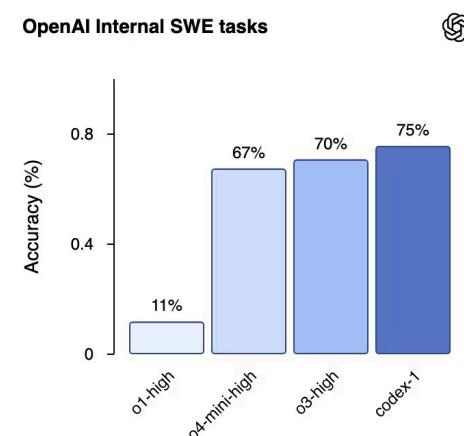
## Devin by Cognition



## Codex by OpenAI



这类异步 Coding Agent 常要面对已有中大型代码仓库，且要集成进较为专业的开发工作流，因此往往需要更专业化的模型，比如 OpenAI Codex 背后就是 codex-1 这款模型，基于 OpenAI o3、面向开发场景精调而来。



# 尽管产品形态不一，但功能交集渐密

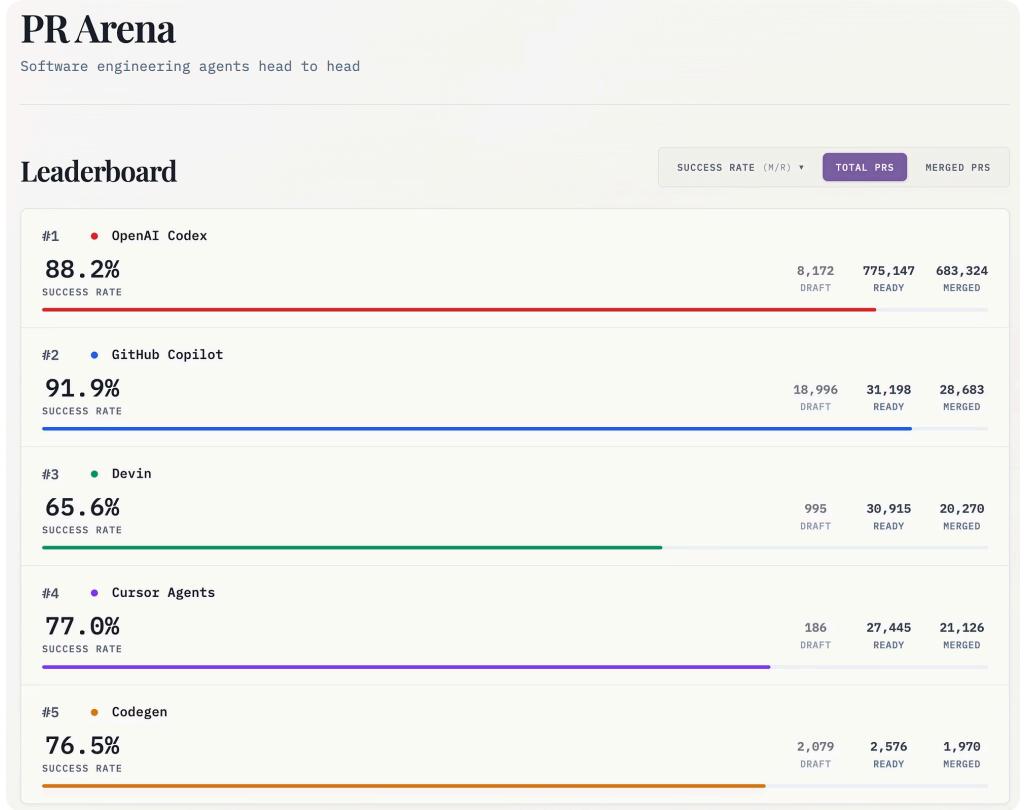
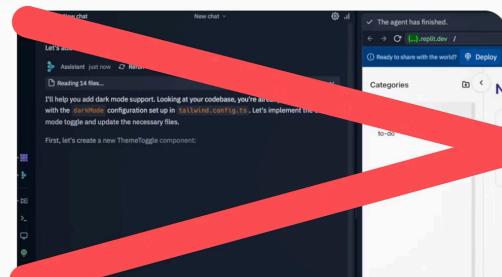
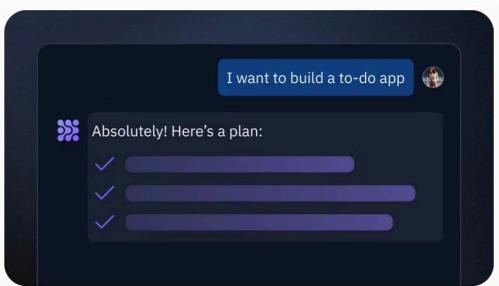
- MCP 生态支持：
  - MCP 成为事实标准，各类 AI Coding 产品全面接入，以拓展工具、资源等，让 AI 完成更多事。
- Agent 模式必不可少：
  - Cursor、GitHub Copilot 等产品都走了对话/Chat → 编辑/Edit → 智能体/Agent 模式的历程
  - Lovable、Replit 等也从助理 (Copilot/Assistant) 模式全面切换至 Agent 模式
  - 随着模型规划能力持续提升，AI Coding 的自主化程度 (Agenticness) 会越来越高
- 多平台 bot 形态：
  - Devin 面世后不久就增添了 Slack、Linear 等集成，可以在团队原有工作流中召唤 Devin bot 指派任务
  - Cursor、Codex、Claude Code 等也在主体软件外纷纷推出 GitHub、Slack 等平台 bot 应用，可针对需求或 bug 自主完成任务

**Replit Agent**

- Bring your projects to life, from ideas to working prototypes
- The best tool for ANYONE — both technical & non-technical creators
- See an app or website that inspires you? Simply screenshot, upload, and Agent will build it

**Replit Assistant**

- Iterate on working prototypes to take your projects to the next level
- Quick updates without breaking flow
- Implement new features in natural language to add features or make changes to your app (no more copying & pasting code)



非共识02

*AI Coding*产品选择什么模型？

——自研 VS 第三方

# AI Coding 产品高度依赖模型能力

多位AI Coding创始人在访谈中都明确提到，Claude 3.5 Sonnet的出现，标志着AI辅助编码进入了一个新的阶段，它让AI Coding的潜力真正得以释放，从"有趣的实验"到"可计算的商业价值"，并成为众多AI编码产品实现产品市场契合（PMF）的临界点，也几乎所有产品指数级增长的开始。



## Replit CEO

变革性的突破发生在6月24日，当时Sonnet 3.5发布了。Sonnet 3.5有两个重要特性：长上下文和连贯性。

## Cursor CEO

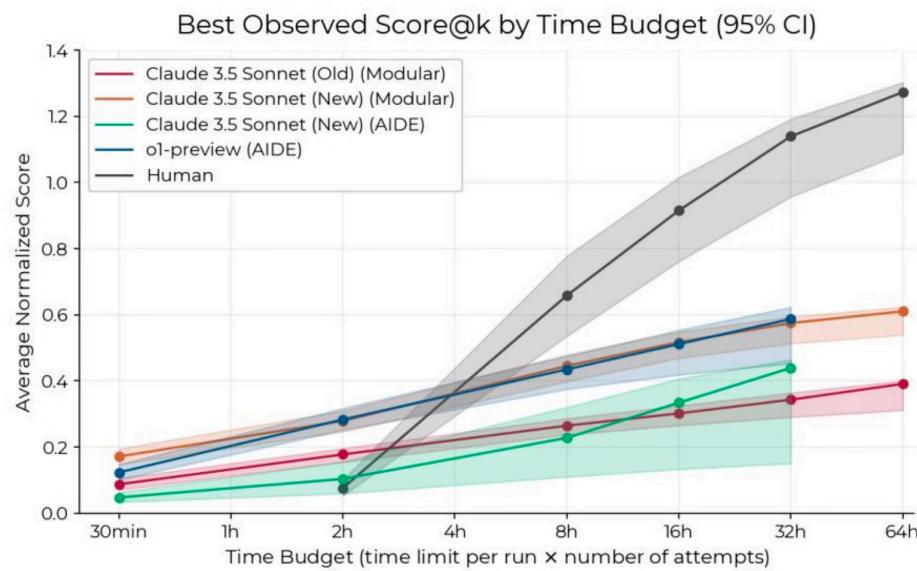
Cursor是靠Claude 3.5 Sonnet实现产品市场契合的。直到模型足够好，对人们编程方式的愿景才真正落地。

## Bolt CEO

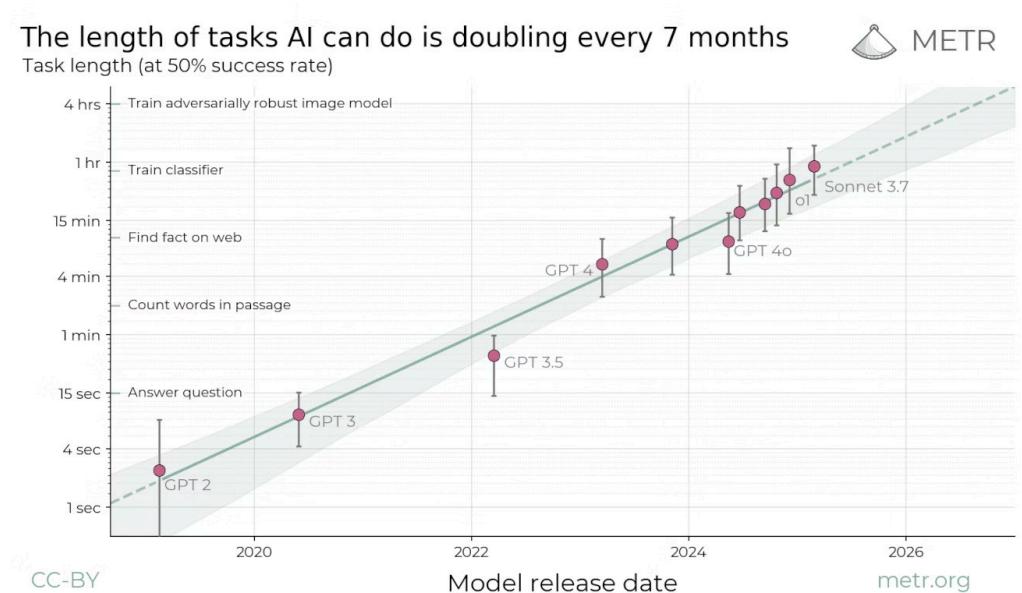
Sonnet一上线，快速增长就立即开始了，是推动整个领域所有产品发展的核心动力，所有产品都在3.5发布后呈现指数级增长。

## Windsurf

Claude 3.5 Sonnet的出现，让Windsurf能够“大幅度地更Agentic”，从而在市场上获得了显著份额。



METR 研究显示在 2 个小时内的编程任务中，**Claude 3.5 Sonnet 和 o1-preview 表现均优于人类专家**。AI 编程速度能以超越人类 10 倍速度生成并测试各种方案。



另外METR最新研究显示，在过去 6 年中，通用前沿模型代理能够以 50% 的可靠性自主完成的任务长度（以人类专业人员花费的时间来衡量）大约每 7 个月翻一番。

# 创业企业AI Coding产品的模型选择

正是因为产品高度依赖于模型的进展，如何设置自己的模型策略，对于AI Coding公司，而言是极度重要的战略选择。

深度自研模型的公司中，既有通用大模型ChatGPT、Claude、Gemini对于代码的重度押注。也有以AGI和代码模型为目标及核心的Magic、Poolside。Anthropic，认为编码是AI能力最重要的先行指标。Magic和Poolside的创始人的观点是为了实现AGI或者全自动化开发，达到达到“99.9%”或更高的准确率，不拥有自己的基础模型是完全不合理的。典型的观点是“软件统治着我们世界的一切，并且软件有效地支撑着现代社会，软件是最重要的一一个领域，甚至比通用目的更重要。”

在自研模型的战略中，还有一种选择是用开源模型做专业化训练。Augment首席执行官 Scott Dietzen表示他们对所用的开源模型进行了大量的专业化训练，使其在从软件中挖掘额外洞察力方面表现出色。Reflection.ai专注于进行深度的“后训练”。

## 自研模型+第三方模型

Cursor

Winsurf

Replit

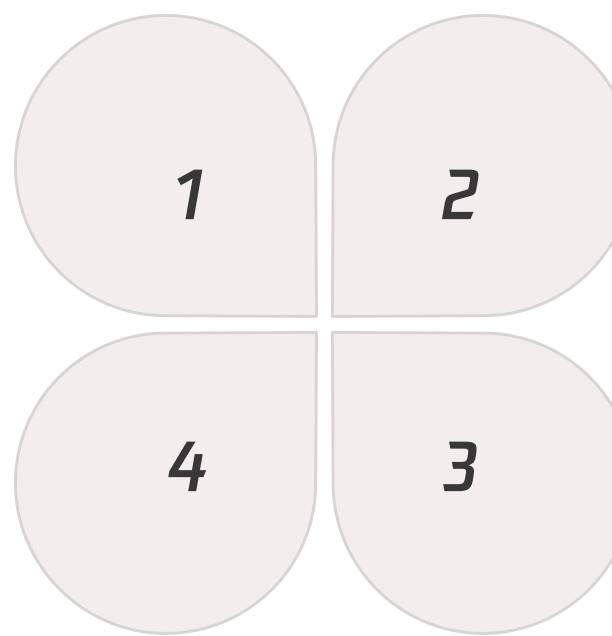
1

2

## 开源模型的专业化训练

Augment

Reflection.ai



## 深度自研模型

Open AI、Claude、Gemini

Magic

Poolside

## 第三方多元模型

Lovable、Bolt.new、Vercel及  
v0

Cognition

Github

纯粹地使用第三方多元模型的公司有Lovable、Bolt.new、Vercel及v0、Github等。代表性的观点是目前的客户需求在于“尽可能多和最好的智能”，而满足这一需求，使用外部的前沿模型更为高效。

也有从选择使用第三方模型到自研投入混合策略，或者预计要自研投入的公司，代表性的是Cursor、Winsurf。Lovable也表示DeepSeek 等开放权重模型“更容易进行低级别控制并训练自己的模型”，“预计我们很快就会这样做”。混合模型的另一种做法是replit，Replit Agent主要使用的是来自Anthropic的Sonnet模型，内部训练的Replit Ghostwriter 模型参数量为 30 亿，主要用于自动补全的产品。

Cursor的联合创始人Aman Sanger对于混合策略的看法是，虽然在产品初期未预料到需要自研模型，但随着团队规模扩大、专业知识积累和数据量的增长，自建模型变得越来越有影响力。“现在Cursor的每一个魔力时刻都以某种方式涉及定制模型，主要用于对低延迟和低成本有严格要求的场景，以及那些预训练模型可能无法覆盖的、拥有大量独特数据的领域”。

# 头部大厂AI Coding产品的模型选择

同样的模型策略选择差异，也出现在科技巨头的AI Coding 产品上，路线包括自有模型、自有+第三方模型的混合模式、及完全的第三方模型。但是无论目前的模型策略选择如何，科技巨头的偏向还是从模型到产品的闭环。Meta从 OpenAI、Anthropic、Google挖来顶尖人才组建了“Meta超级智能实验室”(MSL)，媒体报道称苹果在内部讨论收购 AI 独角兽公司 Perplexity和另一家基础模型公司Mistral。

## 自有模型

### 谷歌

基于Gemini的"Goose"模型，专门针对谷歌内部的工程知识和代码库进行了优化。

### Salesforce

### CodeGenie

基于专有大语言模型，专门针对代码生成优化

## 自有+第三方模型

### 亚马逊

### Q Developer

可以选择自有Titan系列模型和多家第三方模型。

### Meta

### Devmate AI

整合Claude等第三方模型，用于复杂任务。

### Meta

### Metamate

基于自有模型，处理简单任务。

### 腾讯 Codebuddy

### 字节 Trea

混合自有模型和Deepseek

## 第三方模型

### 微软 GitHub Copilot

一直使用外部模型，但是从单一依赖OpenAI转向多模型支持策略。

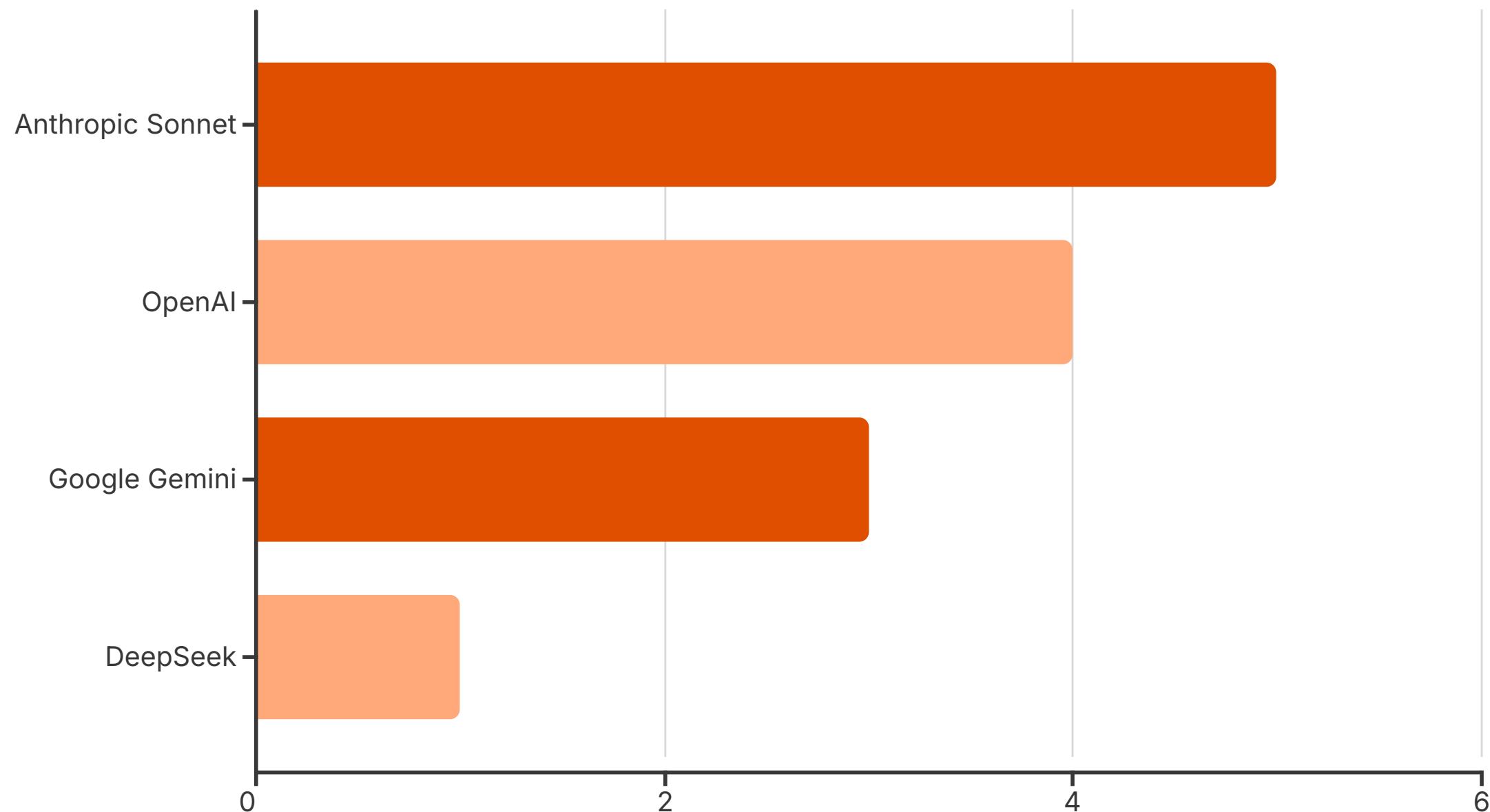
### 苹果

与Anthropic合作开发基于Claude的内部AI编程平台

# AI Coding 产品的模型选择综合排序

在选择第三方模型的时候，Anthropic的Sonnet模型普遍被选为最佳，认为其在速度、代码编辑能力、代码处理量、长上下文和编码能力方面“总体最佳”。

多模型策略成为行业的一个普遍走向。如何更好地落地多模型策略，也有利用“智能路由”来决定使用哪个模型，或者生成、调试、Agent与代码补全等不同环节或者不同产品使用不同模型的策略差异。



## Vercel

主要使用OpenAI、Gemini 和 Claude等模型。所有的评估中 Llama的质量还不够好，Gemini 发展得极其迅速，也关注 Mistral，尽管目前尚未深度使用。

## Cursor

主要使用Sonnet模型，部分团队成员也偏好GPT-4o (o3) 或o1 用于推理密集型任务，以及DeepSeek的R1模型。DeepSeek模型因其知识渊博、智能且运行成本低，已被Cursor用自己的推理服务达8到12个月。

## Lovable

使用所有 OpenAI 模型、Google Gemini，以及 Claude 模型作为编写代码的主要工作核心，利用“智能路由”来决定使用哪个模型。

## Replit Agent

主要使用的是 Sonnet 模型，它是“最擅长编码的模型”。此外，Replit 也使用OpenAI 的模型，因为其架构是“多智能体系统”，不同的模型具有不同的能力。

## Bolt.new

基于现有领先的AI模型，特别是Sonnet系列模型，构建Agent。在调试方面，Bolt也采用了Gemini 模型来增强推理能力。目前已支持多模型，尽管用户对此并不知情。

非共识 03

*AI Coding*给用户的价值有多大？

——提效 VS 降效

# 九成开发者使用AI Coding，近六成用户每天使用

从2024年开始，从第三方机构到产业链上下游都对开发者群体的AI Coding状态和效果进行了调研。Stack Overflow 2024年7月发布的开发者调查收集了来自185个国家超过65,000名技术人员的见解，76%的受访者表示今年已在开发流程中使用或计划使用AI工具。

2025年5月，Jellyfish对645名来自不同工程岗位（包括个人贡献者、经理和高管）的全职专业人士进行了调查。受访者来自各种规模的组织，小至不足10人的小型团队，大至拥有500多名工程师的大型企业。调查显示目前有**90%**的工程团队已在工作流程中集成并使用AI编码工具，大约48%的受访者表示他们正在使用两种或更多AI编码工具。

QODO.ai也在2025年对609名来自不同规模企业和地区的开发者进行了调查，**显示59.8%的开发者每天使用AI编程产品，AI生成或显著影响的代码比例超过40%**。Stack Overflow 2024年的调研数据中，顶尖组织的开发者日常/每周的AI使用率比例才达到60-70%，一年时间内这个数据成为了行业平均水平。



## 进入工作流

90%的开发者已在工作流程中集成并使用AI编码工具  
(Jellyfish)

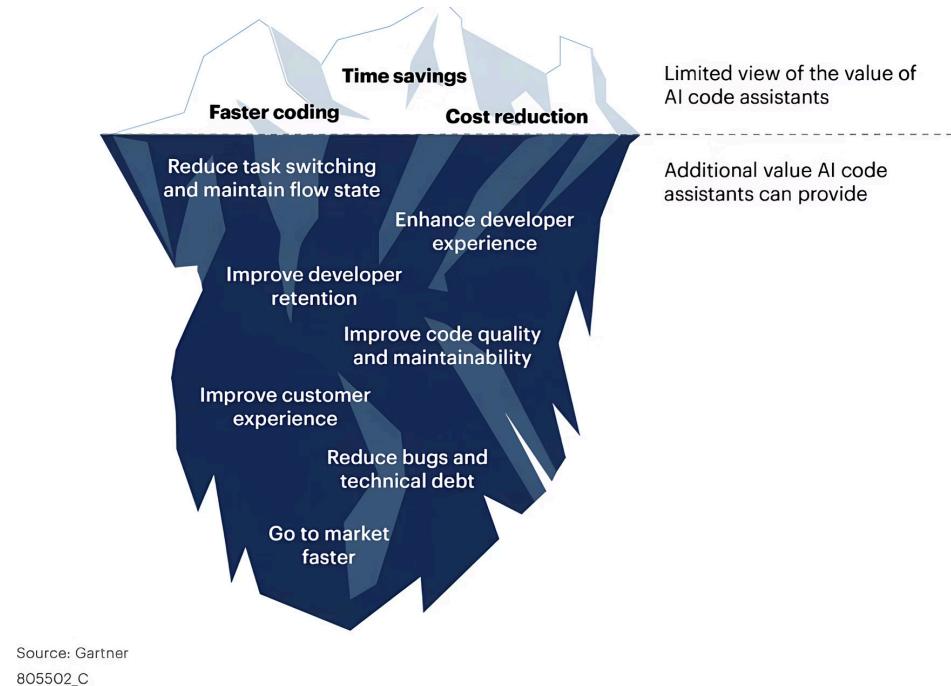
## 每天使用

每天使用AI编程工具  
(QODO.ai)

## AI生成或显著影响

由AI生成或者显著影响的代码比例超过40% (QODO.ai)

# AI Coding 总体提效显著，但存在降效质疑



Source: Gartner  
805502\_C

Gartner.

Gartner认为评估AI Coding 价值的时候，水面上的"显性价值"是节省时间、编码更快和降低成本。

在水面下包含了更深层的系统性影响：开发者体验层面，提升开发者体验、保持心流状态、提高开发者留存；产品质量层面，提升代码质量和可维护性，减少bug和技术债务；业务影响层面，改善客户体验，更快产品上市。

但是关于AI Coding对于专业开发者是提效还是降效，无论是从行业领袖的观点，还是数据统计，都存在巨大的冲突和非共识。



## 提效

**Perplexity CEO Srinivas:** 工程原型开发时间已从“三到四天缩短到仅仅一小时”。

**吴恩达:** 写生产级代码时，用 AI 助手可能能提升 30% 到 50% 的效率。写原型验证代码时，效率提升完全不是 50%，而是至少 10 倍。

**微软CEO Satya Nadella:** 在Python项目中AI生成代码的效果最为突出，C++等底层语言则相对较弱。

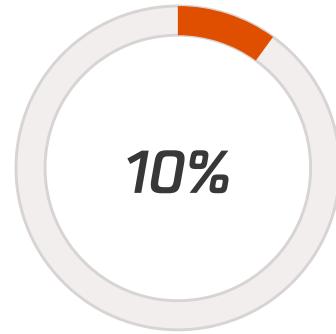


## 降效

**GitHub CEO Thomas Dohmke:** AI 工具可能会降低经验丰富的工程师的工作效率。将编码知识转化为自然语言指令可能比直接编码耗时更长。

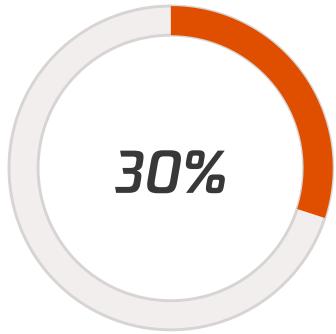
**OpenAI 联合创始人 Greg Brockman :**人类将只剩下“编码中不那么有趣的部分”。与创造性编码相比，代码审查和部署任务“一点也不好玩”

# 大公司采用评估提效10%-30%，节省开发工作量



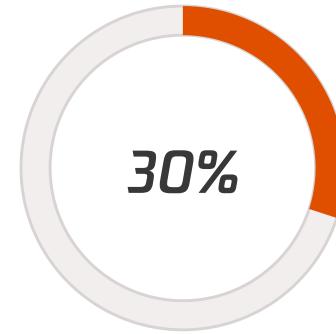
## 工程速度

谷歌CEO Sundar Pichai: AI的使用使工程速度（即特定时间段内完成的软件工作量）估计提高了 10%。



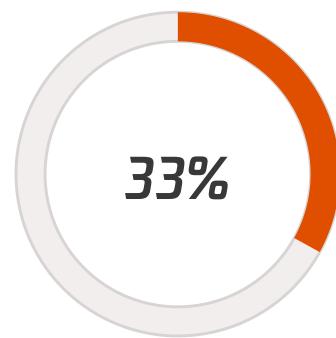
## 接受率

微软CEO Satya Nadella: AI代码接受率目前在30%到40%之间，还在不断上升。



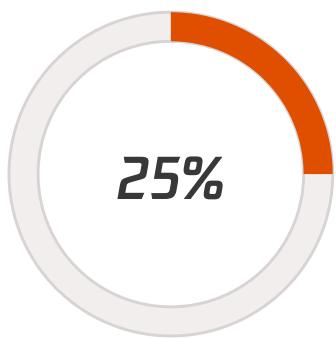
## 生产力提升

Salesforce 首席执行官 Marc Benioff: AI 带来了30%生产力提升。CodeGenie让内部开发者平均每周节省125分钟。



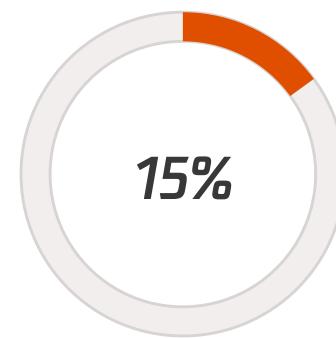
## 代码更改建议

谷歌CEO Sundar Pichai: 开发者在每三次代码更改中，就有一次是接受了AI的建议。



## 代码审查

多邻国反馈，提升了25%的效率，尤其是生成样板代码。代码审查平均周转时间下降67%。

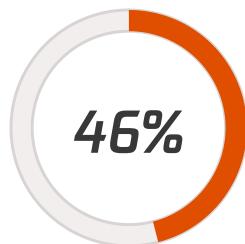


## PR 合并率

埃森哲拉取请求（PR）数量增长8.69%，PR 合并率提升 15%，成功构建率增加 84%

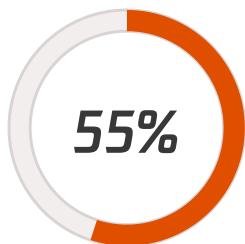
# 个人开发者报告提效明显，但代码质量提升有限

作为目前市场占有率排名第一的Github Copilot，Github Copilot Usage Data Statistics (2025)对2025年Copilot用户的使用情况调查，不仅调查了时间、速度和成本的显性价值，也对隐形价值的开发者体验、产品质量、业务影响进行了统计，统计的结果全方位支持了AI Coding产品的显性和隐性价值提升。



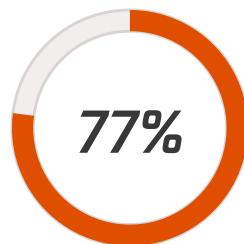
## 代码生成

平均46%的用户代码由  
Copilot 生成



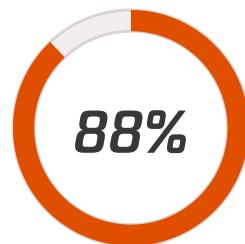
## 完成速度

对照测试中，使用  
Copilot 的开发者完成  
任务速度快了 55%。



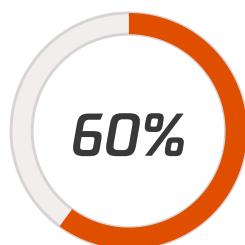
## 代码审查

代码审查速度提升了  
15%，近 70% 的审查意  
见被采纳



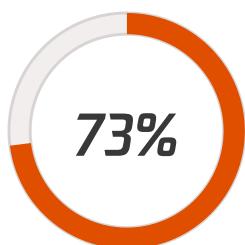
## 代码保留

Copilot生成的代码中有  
88% 字符会被保留在最  
终版本中



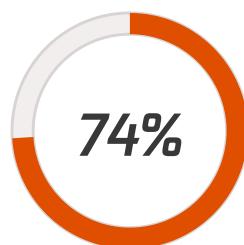
## 满意度

60%至 75%的用户表  
示 Copilot 让他们对工  
作更满意，减少挫败感



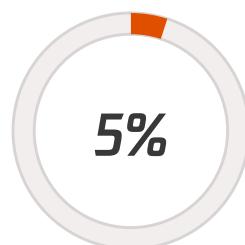
## 保持心流状态

73%的开发者反馈使用  
Copilot时能够更长时间  
地保持“专注”或心流状  
态。



## 更高价值任务

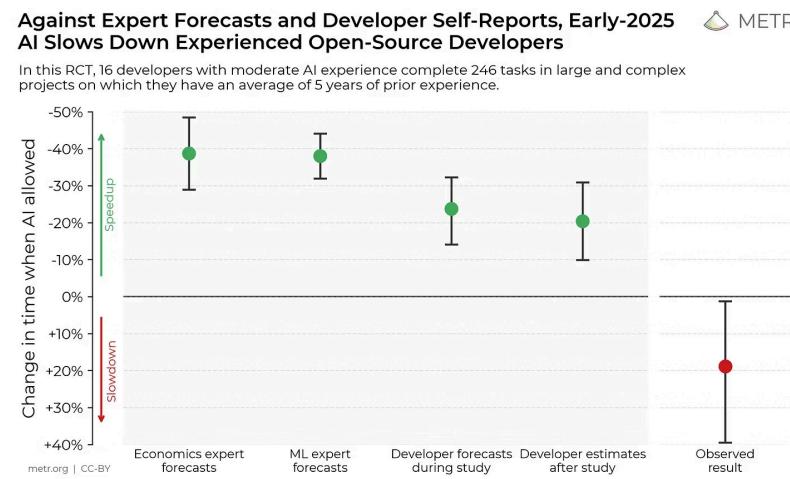
能够将精力从重复性工  
作转移到创造性任务  
上。



## 质量和批准率

Githut2024年调研显  
示，代码批准可能性高  
5%，可读性提升  
3.62%，可靠性提升  
2.94%

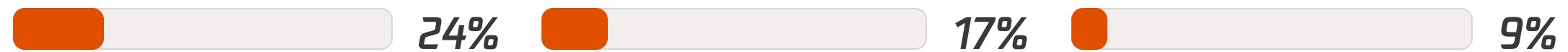
# 部分研究显示开发者提效体感有误，甚至有降效可能



2025 年春季进行的METR实验，从各大开源项目中招募了 16 名开发者，针对 246 项任务，在真实工作环境下进行了货真价实的随机对照试验。生产力反而下降了 19%。甚至连研究参与者们自己也感到惊讶：在体感上，他们认为 AI 让自己的生产力提高了 24%。

2024年11月发布的DORA 报告也给到了类似的结论，近3000名技术和相关岗位的专业人士参与，大多数参与者拥有10-25年的工作经验。这项调查也发现开发者自我报告的时间节省与PR吞吐量指标之间存在脱节。75%的受访者表示，在过去三个月中，使用AI工具后他们的生产力得到了提升，但交付指标却呈现出负面趋势，并且开发者调研称自己有价值时间的明显减少。

METR实验对于时间，也有更加详细的统计。主动编程、阅读与搜索的时间都有下降。但是审查AI输出、与AI进行提示和交互、等待AI模型的响应的时间成为新增时间投入。



## 主动编程

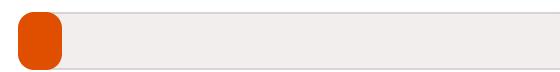
从37%降至24%，编程时间减少 35%。

## 阅读与搜索

从 24% 降至 17%，信息搜索时间减少 30%。

## 审查AI输出

新增了 9% 的时间用于审查AI生成内容。



## 与AI交互

新增了 8% 的时间用于与AI进行提示和交互。

## 等待AI响应

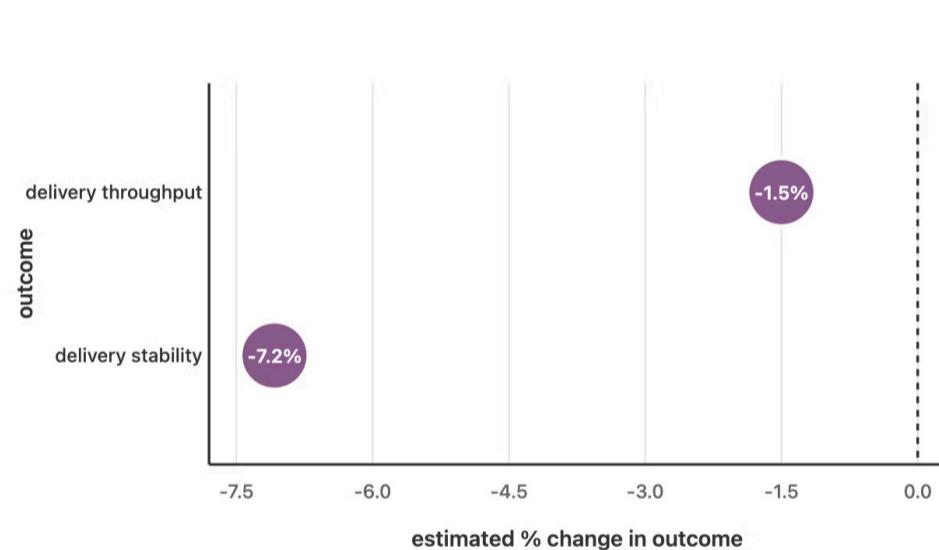
新增了 4% 的时间用于等待AI模型的响应。

# 部分开发者报告AI Coding降低了代码质量，对开发有负面影响

来自于DORA、Stack Overflow和QODO.ai的开发者调研，对于软件交付绩效、准确性、复杂任务和代码质量的负面影响或者争议进行了支持。DORA2024年调查数据显示，AI采用可能对软件交付性能产生负面影响。随着AI采用率的增加，交付吞吐量估计下降了1.5%，交付稳定性估计减少了7.2%。

Stack Overflow2024年的调查中，43%受访者对AI准确性感到满意，31%持怀疑态度。同时，开发者对于AI Coding 的担忧，主要集中在信息误导、数据归属和算法偏见以及复杂任务处理上。近半数（45%）的专业开发者认为 AI 工具在处理复杂任务方面表现糟糕或非常糟糕。

QODO.ai的2025开发者调查报告显示，21.4%的开发者认为AI降低了代码质量，其中10.7%选择了显著下降。



## 信息误导

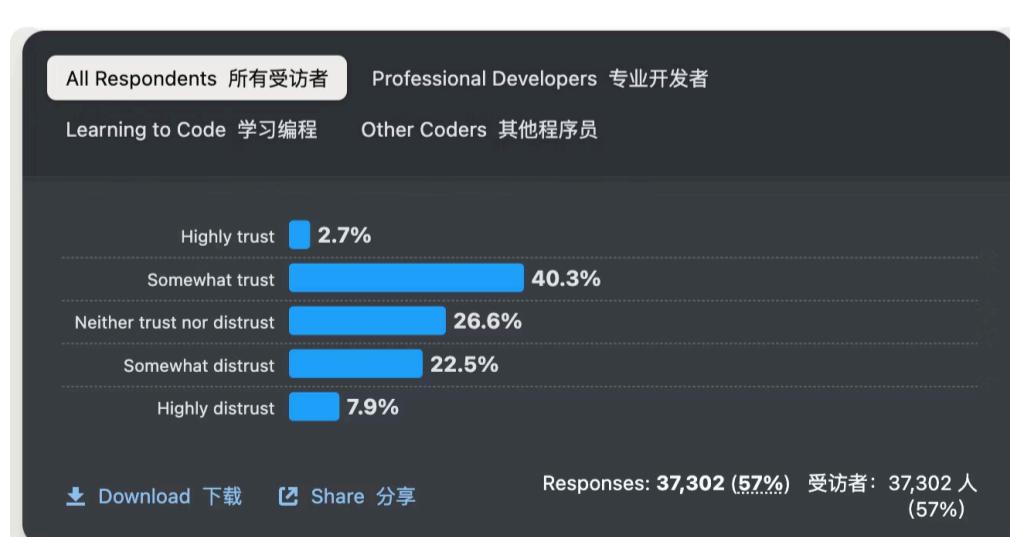
79%的开发者担心AI有传播虚假信息的潜力。



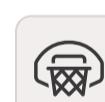
## 数据归属

65%的开发者对数据来源的正确归属表示担忧。

from: DORA 2024年



数据来源：Stack Overflow2024年



## 算法偏见

50%的开发者担心AI输出中可能存在的偏见会扭曲视角。



## 复杂任务处理

45%的开发者认为AI工具在处理复杂的编程任务时力不从心。



数据来源：Stack Overflow2024年

## 非共识04

*AI Coding*产品理想付费模式是什么？

——固定 VS 按需

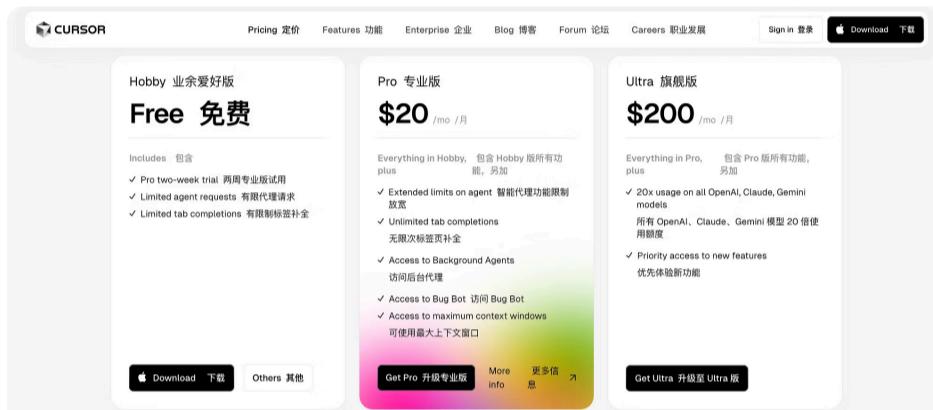
# AI Coding的四种付费模式

基础的付费价格设置，行业存在高度共识，20美元是大部分AI Coding 产品的主要起步价格，也是市场对于AI产品的主流心理价位。最激进的Devin，2025年4月也将500美元的起始价格，大幅下调至每月20美元起。主打企业级市场的Augment 也只是将起步价格定为了50美金。

付费模式上，订阅制目前还是主流，大部分产品的核心付费模式是按月或按年订阅，提供不同功能和资源限制的套餐。但这并非是传统的固定月费或者席位的订阅制。

## 订阅制固定月费

按月／年支付固定费用，获得大部分功能与额度，超额单独计费。



代表产品：GitHub Copilot、Cursor、Replit Core、Claude Code

## 消息交互计费

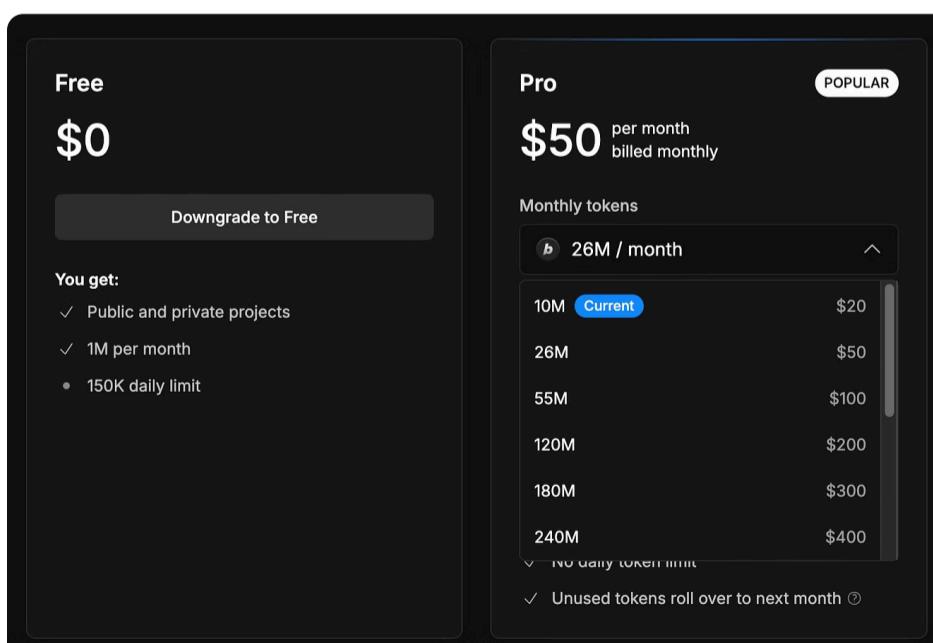
基于用户消息的交互计费，一条消息 = 一次计费。

	\$0/month	\$50/month	\$100/month	\$250/month	Custom
COMMUNITY	50	600	1500	4500	Unlimited
ADDITIONAL USER MESSAGES	\$10/100	\$10/100	\$10/100	\$10/100	Custom
CHAT & AGENTS	✓	✓	✓	✓	✓
CONTEXT ENGINE	✓	✓	✓	✓	✓
HCI & NATIVE TOOLS	✓	✓	✓	✓	✓
NEXT EDITS & COMPLETIONS	✓	✓	✓	✓	✓
TEAM MANAGEMENT	✗	✓	✓	✓	✓
SOC 2 TYPE II	✓	✓	✓	✓	✓
SUPPORT	Community	Community	Community & Email	Community & Email	Dedicated
AI TRAINING	Allowed	Never	Never	Never	Never
	Install now	Install now	Install now	Install now	Contact sales

代表产品：Augment Code、Lovable.dev

## 基于Token/Credit计费

预先购买一定数量的 Token/Credit，使用即扣减，额度用尽可续购。



代表产品：Bolt.new、Windsurf、V0、Magic.dev

## 按需付费

按照实际Token/任务/资源/时长计费，按照任务复杂度进行收费。

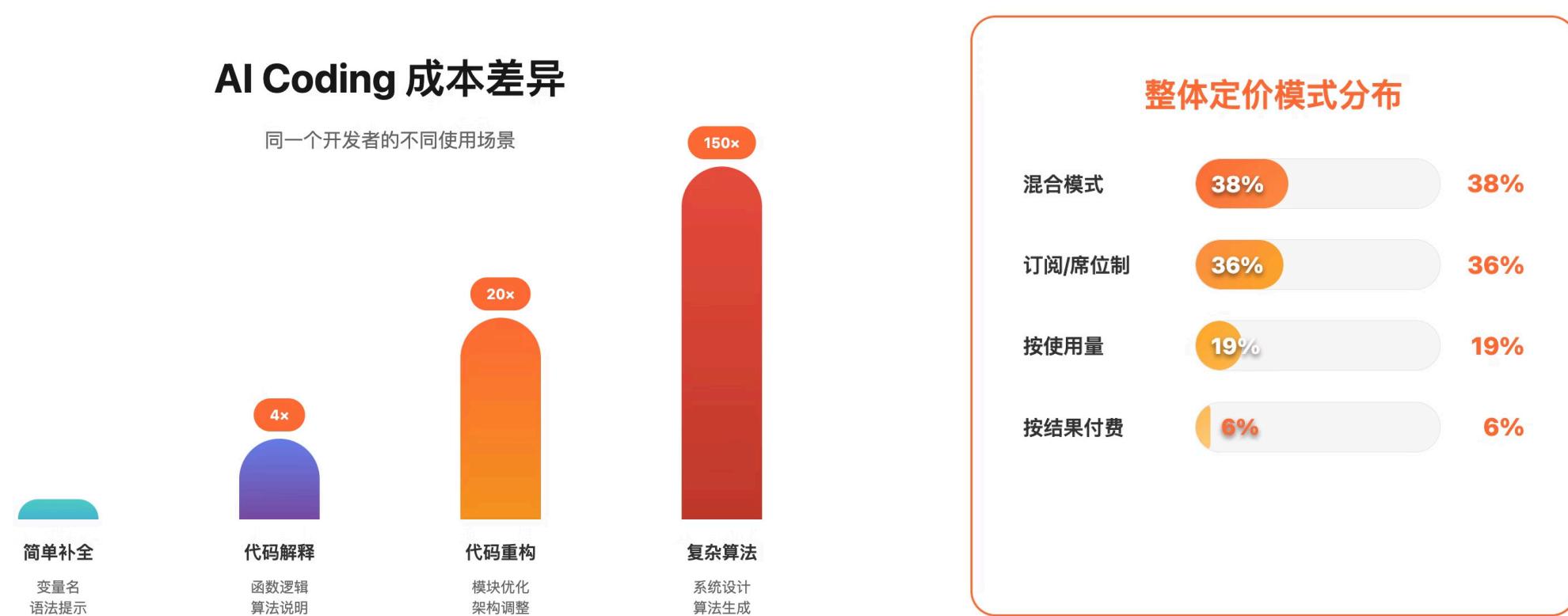
Core 核心功能	Team 团队版	Enterprise 企业版
Pay as you go, starting at \$20 按需付费，起步 20 美元	\$500/month 每月 500 美元	Custom pricing 定制价格
Get started 开始使用	Get started 开始使用	Contact us 联系我们
Includes: 包含内容：	Everything in Core, plus: 包含核心版所有功能，外加：	Everything in Team, plus: 包含团队版所有功能，外加：
Key capabilities: 核心功能：	Key capabilities: 核心功能：	Key capabilities: 核心功能：
<ul style="list-style-type: none"><li>Autonomous task completion 自主任务完成</li><li>Devin IDE</li><li>Ask Devin 向 Devin 提问</li><li>Devin Wiki Devin 维基</li><li>Learns over time 持续学习进步</li></ul>	<ul style="list-style-type: none"><li>Access to early feature releases and research previews 抢先体验新功能和研究预览</li></ul>	<ul style="list-style-type: none"><li>Access to Devin Enterprise 访问 Devin 企业版</li><li>Access to Custom Devins 可定制 Devin 的功能扩展</li></ul>
Usage: 使用量：	Usage: 使用量：	Security: 安全防护：
<ul style="list-style-type: none"><li>Unlimited users 用户数量无限制</li><li>Share and collaborate 共享与协作</li><li>Up to 10 concurrent Devin sessions 最多同时运行 10 个 Devin 会话</li><li>No monthly commitment, pay-as-you-go plan 无需月度承诺，按需付费方案</li><li>Auto-reload settings for on-demand consumption 自动充值设置满足按需使用</li></ul>	<ul style="list-style-type: none"><li>Unlimited concurrent sessions 无限并发会话</li><li>250 ACUs included monthly 每月包含 250 个 ACU</li></ul>	<ul style="list-style-type: none"><li>Deploy in your virtual private cloud (VPC) 部署在您的虚拟私有云 (VPC)</li><li>SAML/OIDC SSO - SAML/OIDC 单点登录</li><li>Centralized enterprise admin controls 集中式企业管理控制</li><li>Teamspace isolation 团队空间隔离</li></ul>
Account Support: 账户支持：	Account Support: 账户支持：	Account Support: 账户支持：
<ul style="list-style-type: none"><li>Dedicated Slack Connect channel for support 专属 Slack Connect 通道提供支持</li><li>Optional onboarding call with the Cognition team 可选加入认知团队的入门指导电话</li></ul>	<ul style="list-style-type: none"><li>Dedicated Slack Connect channel for support 专属 Slack Connect 通道提供支持</li><li>Optional onboarding call with the Cognition team 可选加入认知团队的入门指导电话</li></ul>	<ul style="list-style-type: none"><li>Dedicated account team 专属客户团队</li><li>Custom terms 定制条款</li><li>Centralized billing and usage analytics across multiple Devin organizations 跨多个 Devin 组织的集中计费和使用分析</li></ul>

代表产品：Devin、Replit Agent、OpenAI Codex

# 由于API费用“不可控”，订阅制付费模式受挑战

硅谷一线投资机构ICONIQ，2025年4月对300位软件公司构建人工智能产品的执行高管进行了调查，数据显示，70%公司面临API费用控制难题。传统SaaS的固定订阅模式在面对AI的高变动成本时出现了结构性问题，这种API费用的“不可控性”正在重塑整个AI行业的商业逻辑。

传统SaaS可以随着用户增长和产品成熟，实现低边际成本 + 规模效应 + 高利润率。但AI产品面临成本随使用量线性增长、价值难以量化，以及容易出现“重度用户亏损，轻度用户流失”的特征。数据显示，37%AI公司正在探索按照ROI导向定价和混合订阅+按量付费模式的新付费模式，计划对付费模式进调整。混合模式已经超过了订阅/席位制成为了最主流的AI产品定价模式



这个挑战和变化同样发生在AI Coding 行业，而且AI Coding 相比其他AI领域更加复杂，整体行业呈现重度用户多、使用频率高、Token消耗大、成本波动大的特点。

在AI Coding产品定价上，可以看到多种付费模式的交织和融合，今年多个产品的付费模式变动背后的风波，也是整个AI行业的预演。

# 差异化的免费服务和不同维度的Agent计费模式

AI Coding目前发布的产品，都提供了免费层级或试用期，以降低用户入门门槛，吸引用户体验产品。但是虽然都是免费层级，不同的企业也采用了不同做法。

## 专业版试用+功能限制

以cursor为代表，提供一定时间的专业版试用。免费版限制模型、功能等。

## 同等功能，数量限制

Bolt.new、Lovable.dev 等，都是给予免费用户同等权利，但是在 token 和 Credit 或者消息条数上进行限制。

## 隐私性区别

Augment Code，明确指出免费版会用于AI训练，但付费版不会。

另外Agent 计费已不再是简单的“token × 模型单价”，更应考虑“任务复杂度”、“执行时长”、“流程环节”多维度成本，所以在Agent 的收费模式的设计上，也存在着不同的策略。

### 1 统一且同等计费模式

Agent 和 Chat 同样的计费模式，是目前的主流。

### 2 统一但非同等计费模式

Lovable在消息交互扣费的统一标准下选择模式，原有的 Chat 模式是每发送一次 AI 消息或编辑请求，消耗 1 credit。但是 Agent Mode 率先引入“复杂度定价”，小改动比 1 credit 便宜，大型重构或多轮搜索则更贵，官方强调“更贴合真实工作量”。

### 3 单独付费体系

Replit给Replit Core和 Replit Agent设计了不同付费模式。Replit Core采用标准固定订阅模式，享受固定费用下的完整开发环境，包括计算资源、存储空间和协作功能。

Replit Agent采用了“基于工作量”(Effort-Based)定价模式，根据AI执行任务的实际复杂度和工作量来计费。

# 订阅+按需的混合模式成为主流，计费单元抽象化

随着AI模型的使用量化（如Token、信用点、用户消息、ACU等），混合按量付费的订阅制模式变得越来越常见。哪怕是最传统的订阅制，也变更成订阅套餐包含基础额度，超出部分按量计费。**基于token/credit或者基于消息交互的预购消耗模式已经逐渐成为AI Coding 的主流，同时也有按需付费的探索。**

Cursor、Claude Code、GitHub Copilot都在2025年5月和6月，都对付费模式进行了调整。变动最大且用户反弹最大的调整，来自于Cursor，Cursor宣布于6月16日从固定请求限制转向\$20算力池模式。Claude Code则是静默收紧，重度用户遭遇严格的使用限制。GitHub选择渐进调整，通过引入Premium Request，Agent Mode/Code Review 等高级调用走 Premium Bucket计费，用尽后进行加购。

Cursor	旧模式	新模式（自6月16日）
收费基础	500次“Fast Requests”+无限“Slow Requests”	\$20/月算力池（按API费率扣费），Auto模式无限制；推出\$200 Ultra ( $\approx$ 10,000 Fast)
超额处理	Fast 请求达限停止	超出\$20后按API单价计费；Auto切换仍不限使用

另外计费的基础单元走向抽象化，也是当前AI Coding 产品的共同走向。早期的AI Coding 产品常以模型的调用次数作为付费基元，Bolt.new、Devin、lovable 等产品，都开始用自定义的token、credit、checkpoint、ACU (Agent Compute Unit)等计费基元。目的是让开发者易预算，也防止直接暴露模型单位成本。

非共识05

大企业推进AI Coding应用的态度？

——激进 VS 漸进

# 从30%-90%，企业内部AI代码占比将持续提升



## 微软

Satya Nadella称微软代码库**30%**由AI编写，约20%-30%的项目代码很可能完全由AI自动编写。



## 谷歌

Sundar Pichai2025年Q1财报电话会议上表示：谷歌超过**30%**的新代码由AI生成，6个月提升了5%。



## Meta

Mark Zuckerberg 预测：2026年AI将承担**50%**的编程工作。AI将成为开发的主力，未来开发者角色会发生根本性变化。



## *Anthropic*

Dario Amodei2025年3月做出了最激进的预测：“我们认为我们将在三到六个月内达到AI编写**90%**代码的状态。在12个月内，我们可能会处于**AI编写基本上所有代码**的世界中。”

# 从强制使用到进入绩效，激进公司加快推动落地



## 强制使用 AI

Perplexity 已“强制要求”所有员工至少使用一种 AI 编码工具，通常是 Cursor 或 GitHub Copilot，或两者的结合。

## 人员缩减但产出不变

亚马逊目前已经要求在人员缩减和交付周期加快的情况下，保持相同的代码产出量，迫使程序员不得不依赖AI生成代码来维持生产力。

## 进入员工绩效评估

Microsoft Developer Division 部分团队已在2025年中在年度绩效回顾中考量员工AI工具实际使用情况，下一周期将整体正式纳入公司绩效评估体系。

微软之外，也有多家公司选择同样的做法：Shopify直接把会不会用AI写进绩效考核，多邻国计划将AI应用能力纳入招聘考核，并将AI使用效能计入绩效评估，阿里巴巴所有部门已被告知，2025年的绩效将通过如何利用AI促进增长来评估。

2025年5月，Jellyfish对来自各种规模的组织645名来自不同工程岗位的全职专业人士进行了调查，调查数据显示在公司态度、授权、安全审查和正式使用政策上也存在一定的差距。

**49.4%**

### 自下而上的试验

49.4% 的公司鼓励开发者主动尝试 AI 工具

**41.7%**

### 自上而下的授权

41.7% 的组织为首选工具提供企业级授权

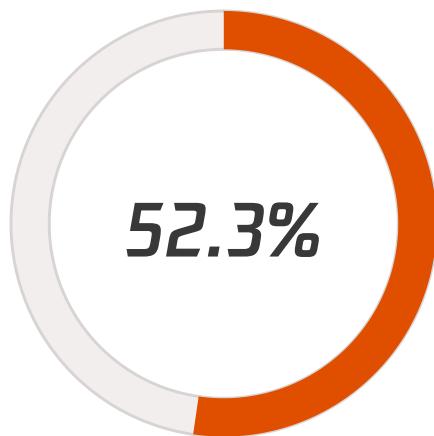
**22.5%**

### 正式使用政策

22.5% 的公司已发布针对 AI 编码工具的正式使用政策

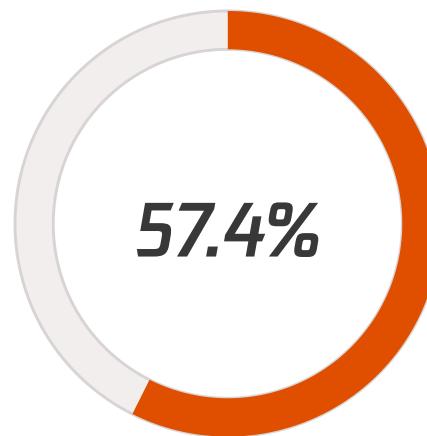
# 大企业也在积极推进非开发部门的AI Coding应用

Jellyfish的调研报告也揭示了一个新的趋势，那就是目前已经有52.3%的非工程部门成员利用AI进行实验、原型设计或编写代码。DevOps 研究与评估 (DORA) 报告也建议，最大的生产力提升来自于将非用户转化为周期性用户，而不是提高现有用户的使用频率。



非工程师的AI使用情况

利用AI进行实验、原型设计或编写代码



工程师的支持度

支持非工程师使用AI编程工具的工程师比例

AI Coding 创业公司，作为先行者，目前都采用了鼓励非开发部门使用AI Coding，并且将其作为产品和团队的最佳实践。

## 赋能非技术人员自主构建

AI Coding工具使非技术背景的员工能够快速构建实用工具和应用程序，极大地提升了团队的灵活性和效率。

- **Replit**: 人力资源部门员工三天内构建了比SaaS更好的组织结构图工具；没有技术背景的员工在一周内搭建了数据管理仪表板。
- **v0**: 销售工程副总裁利用v0创建工具帮助客户理解产品，销售团队通过实时协作缩短销售周期，基础设施工程师利用v0创建精美可视化效果。

## 企业内部推广与文化融合

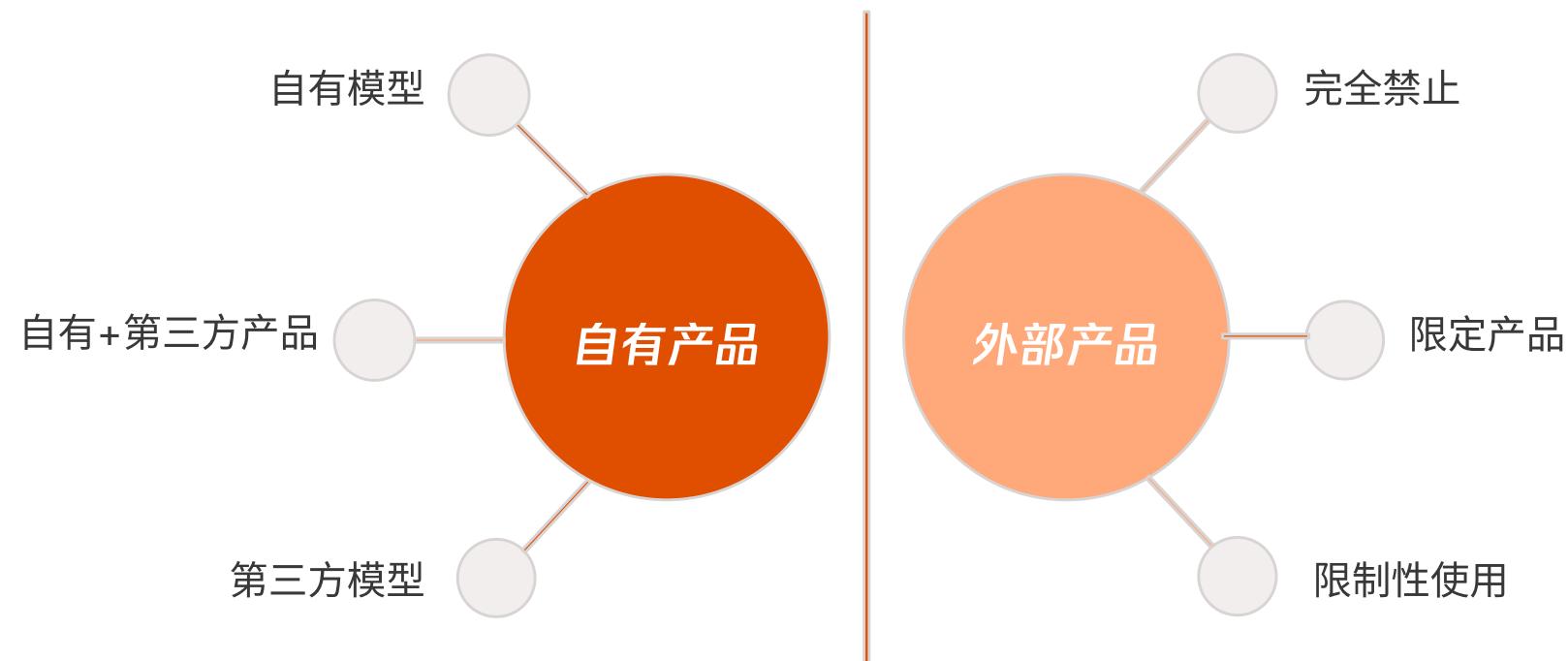
领先的AI Coding公司积极鼓励甚至强制非开发部门使用AI工具，将其融入企业文化的工作流程，实现“狗粮式”开发并推动全员AI化。

- **Anthropic**: 鼓励非技术部门试用Claude Code，秉持“狗粮式”发展理念，让所有员工体验AI能力。Bolt.new也高度鼓励非技术背景团队参与AI编码。
- **GitHub**: Copilot和AI工具的使用已成为企业文化的一部分，并在有意义的地方强制推行，与Git和GitHub的普及程度相当。

# 基于安全考虑和模型能力，大企业对是否使用外部工具有不同策略

出于安全性、竞争以及行业趋势等视角，头部公司打造自己的AI Coding 产品已经成为了普遍的做法，也都选择基于自己公司的代码库对模型进行微调。但是关于模型的选择、是否允许使用外部AI编程产品存在众多非共识并且动态调整。

Meta和Amazon都是一个典型的员工推动的策略改变。在亚马逊的员工投票中， Cursor远超其他编程软件。Meta的Metamate在使用中能力不足，迫使Meta推出新的产品，分别处理简单任务和复杂任务。



## 是否允许使用外部AI编程工具

微软允许员工使用符合特定安全要求的外部 AI 工具，目前员工可以使用编程助手 Replit。

亚马逊正在考虑部署 Cursor 的 AI 工具，但需要解决“几个高优先级的安全问题”。

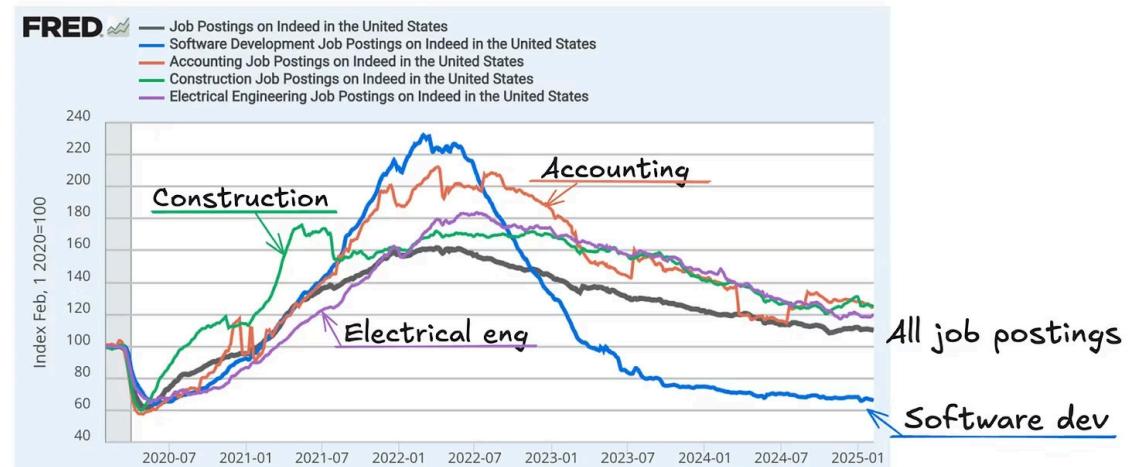
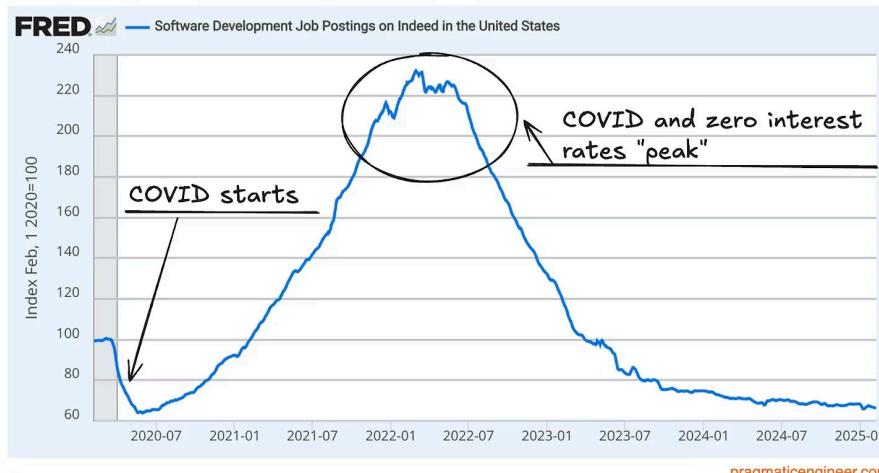
头部领先大公司之外，Jellyfish2025年5月的调查数据显示，35.6%的公司在允许使用 AI 编码工具之前会进行信息安全审查，在安全和能力的选择中，可以看到另一种值得关注的状态是安全措施和意识的缺位。

非共识06

*AI Coding*对组织发展的影响？

——裁员 VS 扩张

# 软件开发岗位需求锐减，初级和高级工程师需求分化



20%

40%

## 初级工程师岗位

根据Indeed数据，自2022年达到峰值以来，软件开发岗位的招聘需求呈现断崖式下跌。当前，软件开发岗位的发布数量仅为**2020年1月水平的65%**，这意味着相比疫情前减少了35%。相较于2022年中期的历史高点，职位空缺数量锐减了3.5倍；而与去年同期相比，招聘信息量也下降了8%。

根据CompTIA的调查，自2023年1月以来，面向初级软件工程师的空缺职位比例从近30%下降到略高于20%。与此同时，要求七年或以上经验的职位发布量增加，占开放职位的近40%，高于之前的30%多一点。

## 高级工程师

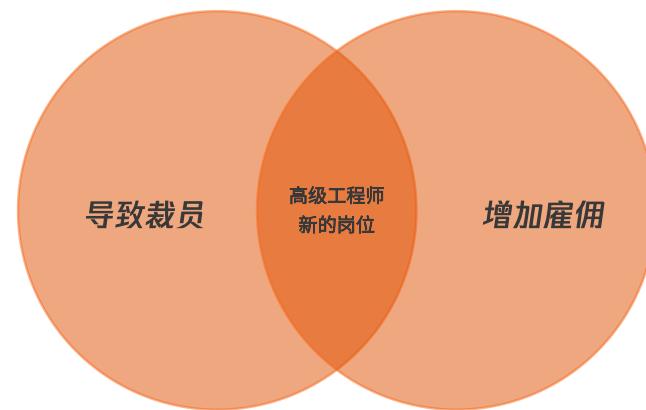
# 提效背景下的裁员浪潮，也在孕育新可能

## 导致裁员

**Amazon CEO**-随着AI广泛应用，公司整体员工数量将会减少。不仅是为提高效率，也是为适应未来工作模式，预计将导致更多的裁员和工作方式的根本转变。

**Stack Overflow CEO**-AI直接削减了我们的人力需求和组织规模。我们在2024-2025年裁掉了近30%工程与支持团队。

**Salesforce**在2025年进行了大规模裁员，目的是为AI计划腾出资源和预算空间。



## 增加雇佣

**GitHub CEO**-AI让大部分代码实现自动化，企业用同样的钱，可以雇佣更多懂AI协作的人，把新想法更快变成产品。

**Netlify CEO**-企业现在可以用更少的资金雇佣更多程序员，从而着手处理那些以前因成本过高而无法启动的软件项目。

**TabNine 创始人**-同一预算下，你不需要砍团队，而是能雇佣更多执行力强、愿意学习AI工具的人。

</>



## 传统模式

过去企业面临的选择：要么购买昂贵的SaaS产品，要么投入巨资聘请熟练的开发人员来构建定制化软件。

## 新经济模式

成本构成随之改变，企业现在可以用更少的资金雇佣更多程序员，从而着手处理那些以前因成本过高而无法启动的软件项目。

# 团队精简、生产提效与人才重构的变化趋势

## 团队规模精简，实现高增长

- **Lovable**: 最初仅由「15人」组成，随后略增至「18人」(其中至少12人从事代码编写)，在极短时间内实现惊人增长。
- **Verce**: v0团队仅有「不到10人」的规模，却能实现惊人的增长和影响。
- **Bolt**: 团队规模约「20到25人」，高度精简。
- **Cursor**: 团队从2023年初的「30人」增长到2024年中的「接近60人」，但对于其规模和影响力而言仍是“小型的”，体现“精益求精”和“产品驱动”。

## 生产力实现指数级飞跃

- **Kevin Scott (微软CTO)** : 预测未来五年内，95%的代码将由AI生成。一个由「10名工程师」组成的团队，借助AI工具，就能完成过去需要「100名工程师」才能完成的工作。
- **Garry Tan (Y Combinator CEO)**: 预计AI辅助编程将帮助「10名工程师」的团队完成过去「100名工程师」的工作量。
- **Eric Steinberger (Magic)**: 预期技术能使开发者生产力实现「超越2倍的提升」，并认为「第一次AI效率翻倍应该在今年完成」。

## 人才结构重塑与招聘新标准

- **Replit**: 强调「不重学历重能力」，「只招聘最优秀的黑客」，寻找「未经定价的人才」和「原始天赋」，相信「谁都可以在Replit上学习编程、构建初创公司」。
- **Bolt**: 高度鼓励非技术背景的团队成员和用户参与AI编码。
- **Cursor**: 公司内部拥有“高比例的工程、研究和设计人才”，总人数更少但工程师密度高。
- **Poolside**: 团队规模“超过一百人”，主要成员分布在“应用研究”领域，并拥有一支规模“可观的市场推广团队”。

# 个人开发者和10人以下团队，正在成为创新主力

24%

*y Combinator*

2025年数据显示，孵化项目中**24%**初创企业，  
95%代码由AI生成。团队平均规模3.2人，实现年  
均300万美元营收。

70%

*Gartner*

Gartner预测，到**2027年70%**的软件创新将源自  
10人以下团队

63%

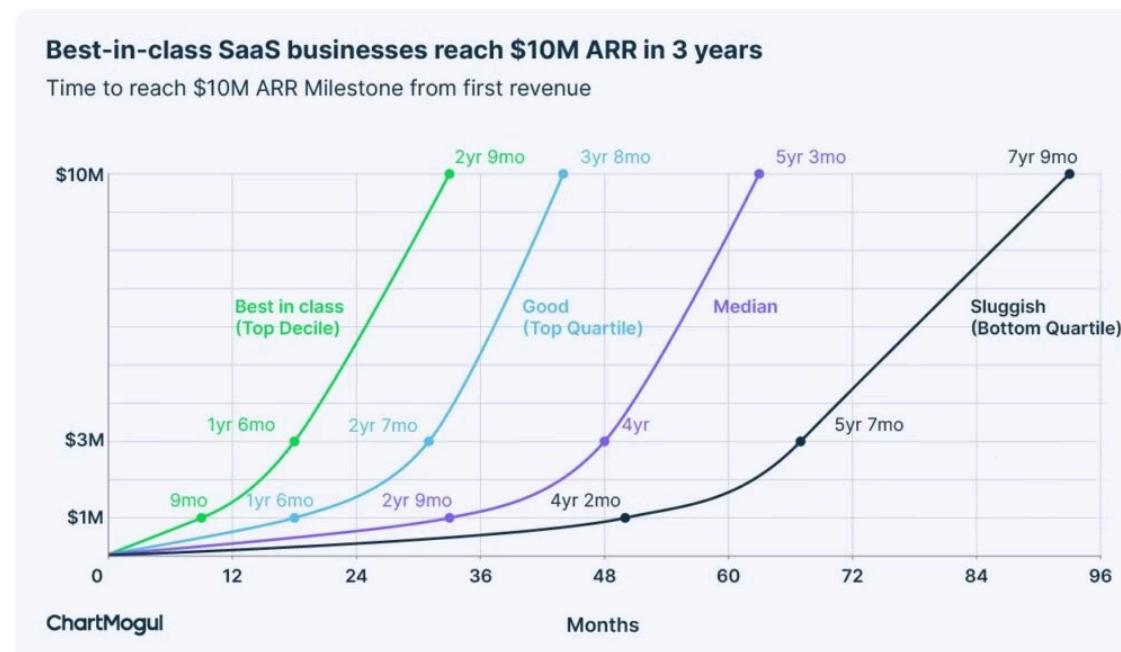
*Hugging Face*

Hugging Face社区数据显示，2025年个人开发者  
贡献的AI模型占比达**63%**

83%

*GitHub*

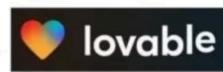
2025年全球个人开发者仓库数量同比增长  
**217%**。其中**83%**的项目使用**AI**辅助编程工具。



0 to \$100M ARR in 21 months w/ 20 people



0 to \$20M ARR in 2 months w/ 15 people



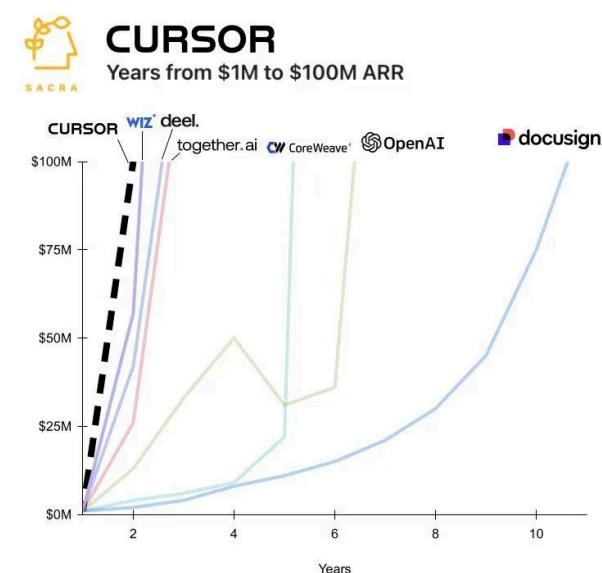
0 to \$10M ARR in 2 months w/ 15 people

**1000万ARR规则被改写：从最快3年到15人2个月实现。** ARR达1000万美元是一个标志，代表公司已经拥有了可重复的客户获取机制，可靠且高效地获取客户，并确保成功。但是在这轮AI Coding 公司和上一任SaaS公司之间产生了巨大的不同。

# 产品自增长、社区传播、零营销预算的AI原生团队显现

**Cursor**的打法极具AI时代特色：极致产品自增长、低价高频、社区驱动，依靠开发者口碑和网络效应，短时间内实现爆发式增长。代表AI时代“产品力+社区自增长”的新范式。

- 团队变化-极致人效：**实现1亿收入时，仅有20位员工，每位员工产生500万ARR。
- 组织变化-无传统部门：**几乎全员为工程师或AI研究员。**无传统销售、市场、HR部门，2025年才开始补充更多产品、运营和支持岗位**，支撑更大规模的用户和企业客户需求。
- 销售模式变化-社区即销售：**零营销投入，完全依赖开发者社区口碑和产品自传播
- 客户变化-大规模个人开发者客户：**收入由36万名个人开发者贡献。从个人开发者向企业客户自然扩展
- 收入模式变化-低价高频：**ACV仅\$276，远低于其他SaaS企业。



公司	业务方向	1M到100M ARR用时	团队人数	人效	客户数	ACV (平均合同金额)	客户结构	收入驱动逻辑
Cursor	AI Coding	12个月	12人	500万	36万	\$276	个人开发者为主	海量开发者自然转化
Wiz	企业级安全 SaaS	18个月	480人	约20万	260	38.4万美元	大型企业客户	聚焦大型企业安全需求
Deel	全球化HR SaaS	20个月	2000人	约5万	1800	5.5万美元	中大型企业客户	服务跨国人力资源
Ramp	企业支出管理	24个月	约500人	20万	5000	2万美元	中小企业	覆盖广泛中小企业

非共识07

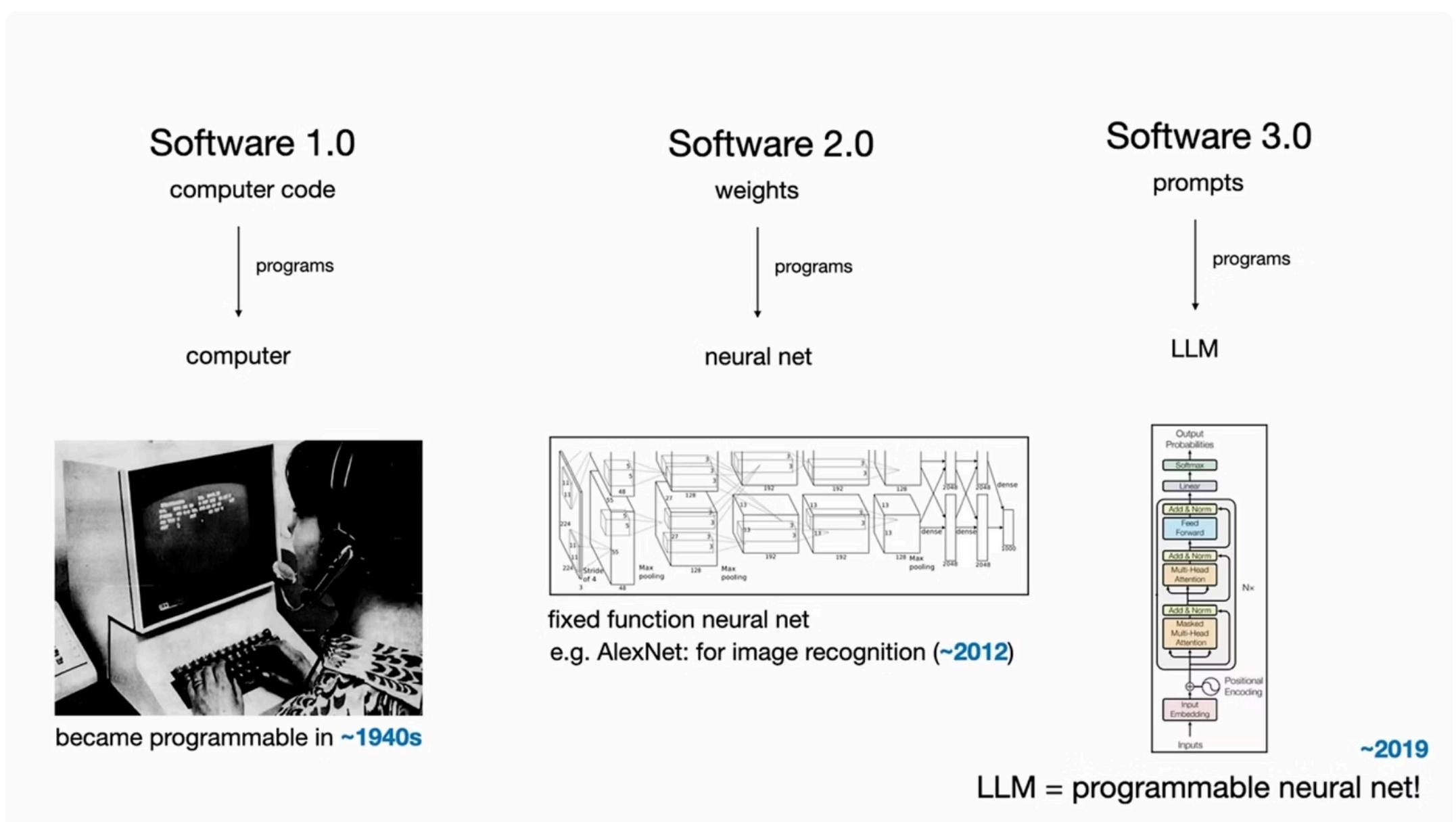
*AI Coding*的未来市场格局是怎样？

——专业 VS 普惠

# 代码≠编程，意图将成为编程的核心驱动力

AI会改变什么？传统意义上的编程，往往意味着在集成开发环境（IDE）中逐行编写、调试和维护特定编程语言的语法。然而，AI的介入正在模糊这一界限。

代码将不再等同于编程本身，而编程正以前所未有的速度迈向高度抽象化，其中自然语言正成为新的编程范式。源代码本身将成为一种“产物”，而不再是最重要的交互单位，意图将成为新的核心，这种转变的深层原因在于，AI模型在生成可运行、甚至美观的代码方面，其可靠性已经迈过了边界。这使得开发者能够跳过大量繁琐的实现细节，将重心从低层级的语法实现，提升到更高层级的“意图表达”和“问题分解”。



## 软件1.0

传统的编程方式，您为计算机编写的代码。这是我们熟悉的GitHub上的代码库，所有这些都是已编写的计算机执行数字任务的指令。

## 软件2.0

基本上是神经网络，特别是神经网络的权重。您不是直接编写这些代码，更多的是调整数据集，然后运行优化器来创建神经网络的参数。

## 软件3.0

神经网络通过大型语言模型变得可编程。您的提示现在是编程大型语言模型的程序，这些提示是用英语编写的，这是一种非常有趣的编程语言。

# AI Coding正在从代码生成走向产品定义

好的产品，代码只是其中的实现手段。调研显示，工程师实际上放在写代码的时间只有20%左右。如果仅仅只是解决代码的实现效率，AI Coding只是在开放流程的一个环节进行了提效。真正具有变革意义的是AI能够理解用户需求，将意图转化为完整的产品方案，从而在产品设计和功能定义层面提供支持，推动整个开发流程的升级。这也是目前所有AI Coding产品的演进路线。



## 传统编程

以代码实现为核心  
关注技术细节与功能逻辑



## 范式转变

从“实现”到“定义”  
意图成为新的交互语言



## 意图驱动

理解设计美学与商业目标  
触及产品定义的深层逻辑



### *Cursor: PRD到代码*

AI旨在消除编辑器中所有“低熵”操作，让编程更像编写“伪代码”，直接从产品需求文档（PRD）生成代码。



### *Replit: 构建自动化*

愿景是让开发者专注于“构建”，将环境配置、依赖安装、部署等繁琐工作交给AI处理。



### *Lovable: 可视化编辑*

定位为“个人AI软件工程师”，允许用户通过可视化方式直接编辑界面元素，修改会“深入到代码库中”。



### *Devin: 自主AI工程师*

被认为是首个“AI软件工程师”，能够自主执行复杂开发任务，包括阅读文档、编写代码、调试和部署应用。



### *Vercel: 理解深层意图*

AI不仅能理解功能需求，还能理解设计美学和市场营销的“意图”，例如根据目标受众生成营销文案。

1

#### 功能需求

理解技术实现要求

2

#### 设计美学

洞察视觉与交互意图

3

#### 市场营销

理解商业与用户心理

4

#### 产品定义

整合深层逻辑

# 开发者不会被完全取代，但需要成为技术专管和通才

尽管Agent的潜力巨大，但“人机协作”的哲学依然是主流。大多数创始人认为，AI不会完全取代开发者，但会重塑他们的核心职责。

Poolside的Jason Warner的观点是AI在短期内并不会完全取代软件开发者的角色，而是更多地替代开发过程中的某些任务。Hansen Wang & Alexander Embiricos (OpenAI Codex Team)说“他们不确定Codex对专业软件工程师的数量影响是增多还是减少，但目前来看，效率提升非常显著，头部用户每天提交10+个PR。”

对于专业开发者的未来职业道路。AI Coding的创业者在大的共识下面，有一些小的不同看法。微软CEO萨蒂亚·纳德拉认为未来的工程师将更像“技术主管”，管理“自己的Agent团队”。Replit创始人Amjad Masad观点更加直接和鲜明，一个是往底层走，一个是往用户走。



如果你想成为程序员并拥有工作保障，我认为你必须选择尽可能接近底层的东西。比如，NASA不会让GPT生成的JavaScript来运行火箭，特斯拉汽车也不会。另一个方向是你要真正擅长成为产品人员，成为设计师、创造者、产品经理。在软件领域，对创意生成者的需求将大幅增加，因为执行将不再是瓶颈，想法才是。

我认为全栈开发者是最危险的，因为这是GitHub上最常见的，这是AI最擅长的，这已经在某种程度上被自动化了。

-Replit创始人Amjad Masad

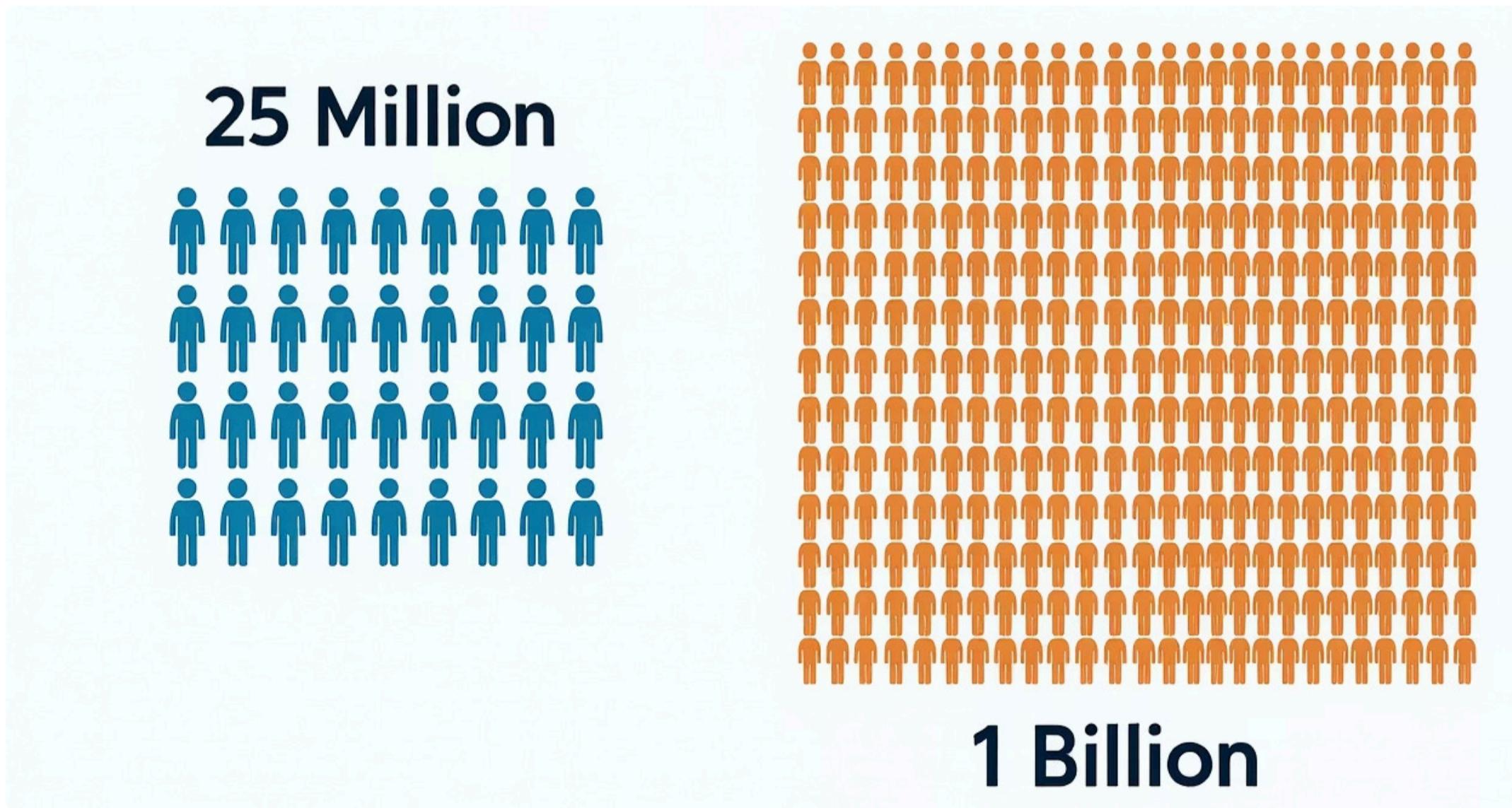
关于第二个成为通才的方向，Bolt的Eric Simons的观察也能加以佐证，那些日常与开发者打交道的产品经理和创业者，正是最擅长使用AI构建工具的人。

另外，几乎所有人都强调人类的最终审查和决策判断不可或缺。开发者需要像“教练”一样，指导、评估、纠正AI的产出，并将其推向预期的高质量标准。Alexander Embiricos & Hansen Wang (OpenAI Codex Team)将与AI代理协作比作管理团队。“AI虽然提高了代码产出，但也带来了新的“产品管理”挑战，例如“特性过剩”和产品臃肿。这要求产品团队和工程师重新思考产品卫生和架构管理实践，从“能做多少”转向“应该做多少”。

# 从“2500万”到“10亿”，消费者重新成为创造者

谁来构建，谁将成为构建者主流？另一种观点是，非开发者会成为软件构建者的主流。就像非专业人士成为短视频行业的主流生产者一样。A16Z的合伙人、增长黑客的代表性人物Andrew Chen 觉得未来的软件的构建者，属于时间富翁而非专业富翁，那些有时间的人，例如学生和年轻人会成为开发的主力。

关于编程现阶段只是极少数人的能力和特权，Eric Simons (Bolt CEO) 和 Anton Osika (Lovable CEO) 给了一个具体的数字和比例支持：“全球有约2500万开发者，只有0.23% 的人口懂得编程”。从2500万迈向“10亿”，不仅仅是一个数量上的增长目标，更是意味着软件的边界将极大地拓展，软件会成为每个人都能使用的表达工具。



而AI时代，把10亿开发者从愿望变成愿景，成为众多公司创始人的行动。GitHub作为全球最大的开发者平台，已经将“培育10亿开发者”设定为其核心愿景。Replit联合创始人兼CEO Amjad Masad 说他相信最终会有十亿开发者。已经发生的事情是AI的引入已经使得Vercel的注册用户数量翻了一番。

除了数量上的变化，更重要是人群画像的变化。Bolt.new CEO Eric Simons 指出，他们平台上的大多数用户（60-70%）并非传统意义上的开发者。非技术背景的用户正在大规模涌入编程领域。Bolt、Lovable、Replit谈到典型用户包括设计师、学生、健身教练、销售人员、11岁的小女孩、运营人员、教育企业家、老师等。v0的Guillermo Rauch认为“AI将成为软件的同义词”，未来“如果你能梦想它，你就能发布它”。

# 丰饶时代的预演



# 丰饶时代，品味将成为核心竞争力

过去，软件开发的门槛主要在于掌握具体的编程语言、框架和工具，即“能否实现”功能。然而，随着AI技术的飞速发展，这一门槛正被大幅降低。当代码生成变得普遍且简单，区分优秀产品和开发者的关键要素就不再是“能否实现”，而是“做得有多好”，人类的“品味”和“方向设定”变得极其关键。

品味将成为核心竞争力，是所有创始人和产品负责人的共识，但是关于品味到底是什么？这里存在两种不同的观点，一种是认为品味不仅仅是审美，更是更加底层的知道构建什么，以及复杂系统的理解和解决方案。另一些则更多地把品味作为艺术和感觉，那些灵感一闪和人性深度的复杂需求。

## 呈现“正确想法”

未来工程的关键将从“严谨性”转向“品味”，即对软件如何运作和呈现的“正确想法”。

未来会越来越无缝转换—你想要构建什么、希望所有功能如何运作、期待它呈现怎样的外观，都能毫不费力地实现。“审美”这个说法可能不太准确，关键在于对构建内容有正确的构思”。

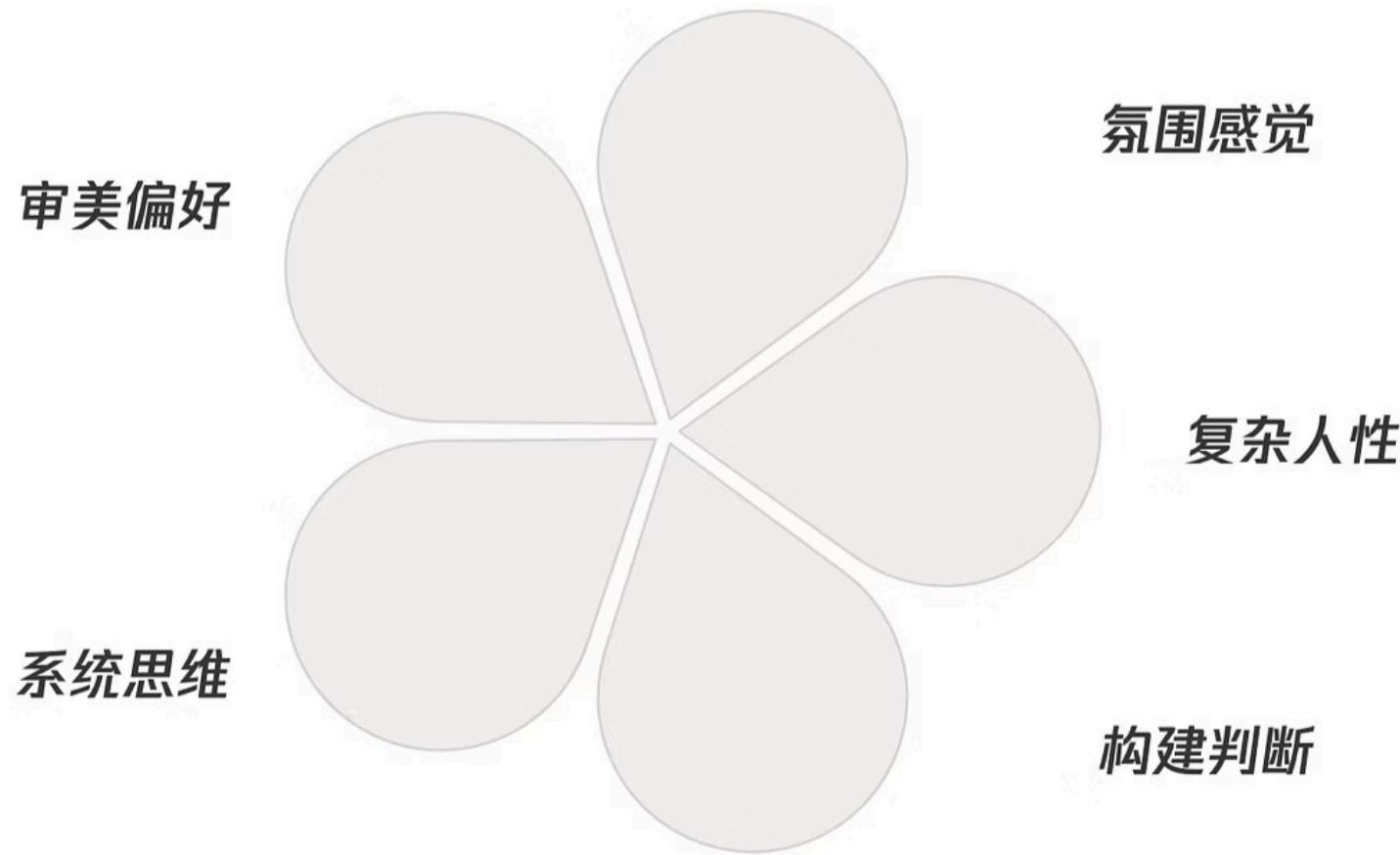
——Michael Truell (Cursor)

## 拥有问题意识

AI编程已经让大家变得更快了，但很多核心问题仍然存在：“理解非常复杂的系统，在整个公司的背景下工作，思考你正在构建的产品或你正在做的工作，以及理解“好吧，我们想要解决的问题是什么？我们想要如何解决这些问题？我们想要构建的解决方案到底是什么？我们将要做出的所有这些关键决策和权衡是什么？”

——Scott Wu (Cognition AI )

另一些创始人，更多地是从艺术、感觉、灵感、直觉去理解品味的核心竞争力。Amjad Masad (Replit) 提到Paul Graham对编程语言的看法，认为它更像是一门艺术，而不是一门科学。这些作品是审美艺术品，而不仅仅是功能性艺术品。Thomas Dohmke (GitHub) 也将软件开发描述为“创意工作，是一种艺术，非常类似于艺术家所做的事情”。



Anton Osika (Lovable) 认为即使AI能生成大量代码，但人类的“品味、经验和直觉”在最终审查和打磨阶段仍不可或缺。“他分享了一个案例：当AI代理陷入困境时，他通过一句看似随意的“看下历史记录”提示，使AI想到了正确的方向，这种直觉判断是项重要能力。

Guillermo Rauch (Vercel) 走的更远，他坦言，机器可以非常廉价地构建东西时，你不再关心‘如何’完成，而是关心最终结果的‘感觉’如何。这种带有审美倾向、让人感觉“这样就是好”且具有社会效应的事情，需要人成为凌驾于机器之上的“氛围掌控者”。

Amjad Masad (Replit) 的解释是初始的灵光一闪、洞察人性深处的复杂需求，仍源于人类心智。AI可以执行任务，但它不具备对社会、文化和人类情感的深层理解，而这些是构建真正有吸引力、有市场价值产品的基础。

# 氛围编程从原型走向生产级应用的可能

“Vibe Coding”这个概念由前Open AI联合创始人之一、特斯拉人工智能总监Andrej Karpathy提出，他将其描述为“一种全新的编程方式，你完全沉浸于氛围之中，拥抱指数级增长，甚至忘记了代码的存在。”

关于Vibe Coding 仅仅在个人兴趣项目上发挥作用，还是逐渐走向生产级，业内存在巨大非共识，并且这个答案也会直接影响到，仅有个人微型应用会爆发，还是会让产业级的定制化软件也进入丰饶时代。



不再下载App，而是构建你自己的个性化应用

每个消费者都被赋能成为创造者



Karpathy自己的看法是“对于一次性的周末项目来说，这还不错”。Michael Tru (Cursor): 认为“Vibe Coding”目前无法应用于长期维护的代码库，因为缺乏对底层细节的理解会导致许多问题。

Guillermo Rauch (Vercel)的观点相对中立且具洞察力。他认为“vibe coding”将“吞噬”大部分前端开发工作，因为AI在生成React、Tailwind和Next.js等前端代码方面表现出色。然而，在处理底层基础设施或需要高度稳定、安全、可预测性的场景时，仍然需要人类工程师的高度谨慎和专业。

向生产可用性演进的观点也有不少支持者。Repl.it 和 Lovable 都表示，他们的技术突破正在解决“当项目变得复杂时AI会卡住”的问题，使得用自然语言就能完成95%的软件价值，并能连接后端、用户认证、数据存储等功能。

Thomas Dohmke (GitHub CEO)认为我们正在回归那种状态，即每个消费者都被赋能成为一个创造者。你不需要从App Store下载App，而是要构建你自己的个性化应用，这些应用连接到我们所谓的‘代理网络’——服务器、API、应用程序编程接口、信息存储、数据库、支付网关、甚至模型本身和全球的GPU集群，以实现所有这些场景。这种转变类似于消费者使用手机摄像头进行创作，未来的设备将使我们能够创造任何想创造的东西”。

# 定制微型应用的爆发时刻即将到来

除了让无法创造的人能够制作软件之外，时间和费用效率提升也是正在发生的事情，v0的Guillermo Rauch提到了一个案例：“一位用户用300美元在Bolt上构建了一个CRM系统，而传统代理机构的报价是3万美元，耗时6个月”。

在AI coding 创业者中，普遍共识的是“软件定义世界”的范畴将进一步扩大，从专业领域渗透到个人、小微企业乃至日常生活中。每个人都可以成为自己需求的“软件设计师”，催生大量个性化和微型应用。

Amjad Masad (Replit CEO) 的观点非常具有代表性：“软件会无处不在，定制化成为常态，每个人都可为自己的特定需求或小众问题定制软件，例如：为家庭成员计算零花钱的App、优化外勤人员路线的Agent、或根据健身计划生成Web App等。这里的边界跨越，在Amjad Masad看来是，“把构建应用会比Google 搜索更快”作为AI代码生成系统的终极测试”。

OpenAI Codex Team的Hansen Wang, Alexander Embiricos的看法也类似：软件越容易编写，我们就能拥有越多软件。现在手机上大部分应用都是大型团队为数百万用户构建，很少有应用是为个人定制的。”

	复式记账法	AI 编码工具
信息对称性	标准化账务记录，提供全面财务透明度	自然语言描述即可生成代码，人人皆可“看懂”
门槛消除	会计知识门槛降低，普通商人也能做簿记	编程门槛降低，非技术人员可快速开发软件
规模效应	促进跨区域商业扩张与资本流动	释放全球创新潜力，促进软件爆发式增长
决策支持	提供实时、准确的财务数据供决策使用	AI 实时生成代码，加速产品迭代与决策

Bolt CEO Eric Simons则从产业市场机遇的角度强调，全球90%以上的公司和个人因技术门槛而无法构建软件，AI的出现将解锁一个“亿万级”的巨大市场，促成“产品和软件的大爆发”。

Amjad Masad分享了一个百年老企业西尔斯家居服务部的案例：他们的非技术运营团队，在耗时六个月完成Cobol代码（一种诞生于艾森豪威尔时代的编程语言，通常运行在大型机上，其维护和现代化是许多金融机构和大型企业面临的巨大痛点）迁移后，直接跳过了整整一代软件，没有选择采购或开发传统的SaaS或ERP系统，而是开始使用Replit创建AI代理来管理业务。例如利用AI代理来优化外勤服务人员的日常路线，帮助他们每天醒来就能获得最优路线，以赚取最多收入并服务最多客户。

这预示着技术史上前所未有的平台转移。这种普及将催生大量过去无法想象的新产品、新公司和商业模式，例如“一人十亿美元公司”的潜力。正如Guillermo Rauch所说，AI有可能导致Google等巨头的“解绑”，为垂直领域的专业AI产品带来巨大机会。

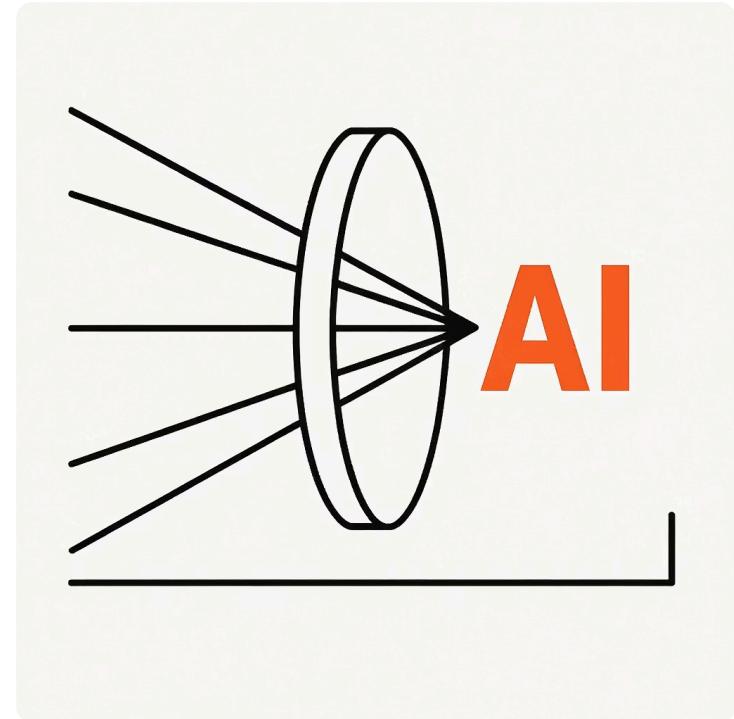
# AI 透镜研究系列简介

透镜不产光，却能重新排布光线。

它让入射的光折射、聚焦、放大，把遥远的星体拉近视网膜，把微弱的纹理放大至可辨，把色散的杂斑过滤成纯粹的信号。也在几何偏折里提醒我们，视角的改变同样可能带来失真与盲区。

透镜既是工具，也是一种立场和视角—用不同曲率、口径与镀膜看世界，让信息穿过不同的"介质"重新折射，呈现出新的方向与意义。

「AI 透镜」是一枚多层镀膜的认知镜头，对准飞速演化的人工智能现场，排布光线-融资背后的产品真实价值、追踪调研AI公司的组织创新变革、记录和放大人与AI的协作实践，从新的镜头看见 AI 细节与趋势。用系统性的观察和分析，穿透噪声，最终落到可见可用的洞察之面。



研究团队：余一、曹士圯、袁晓辉

腾讯研究院 出品 2025.07.24