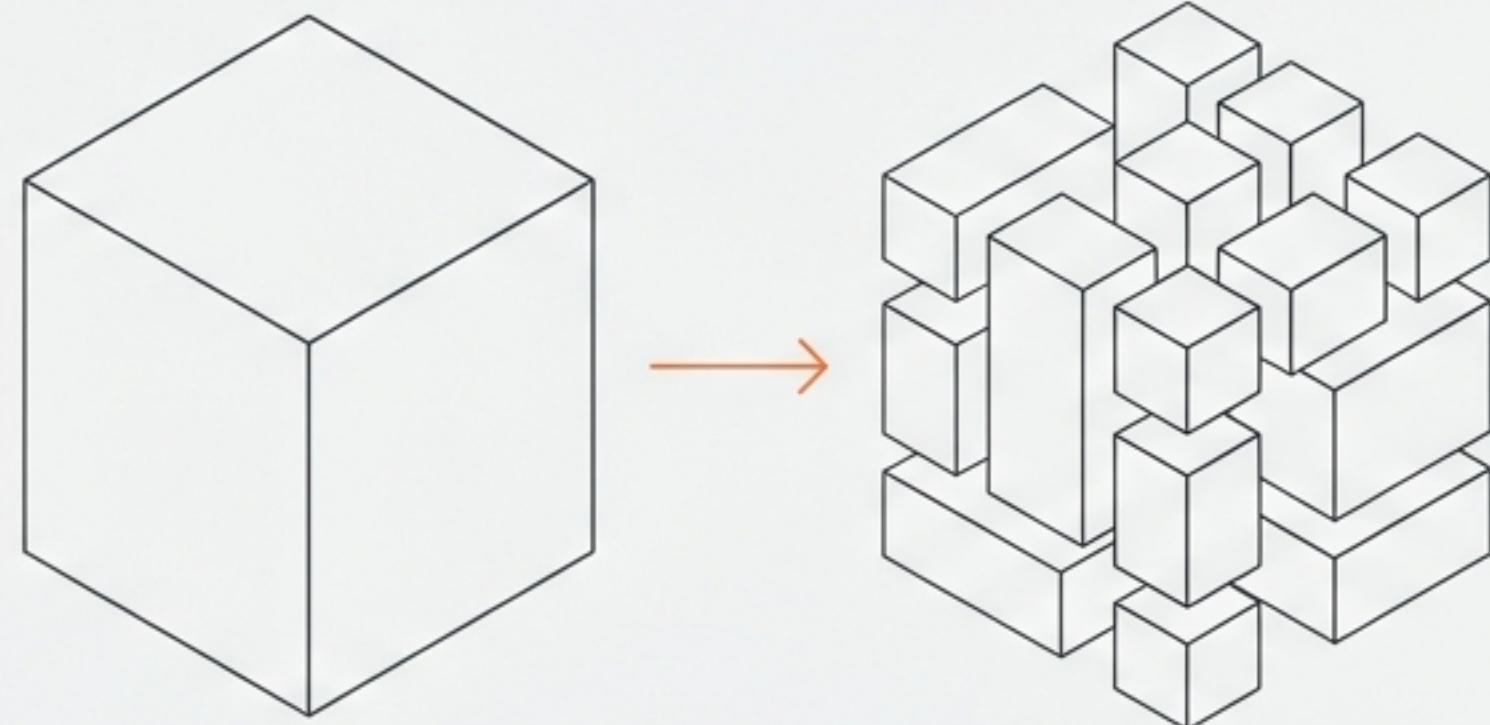


别再构建智能体了， 开始构建技能吧

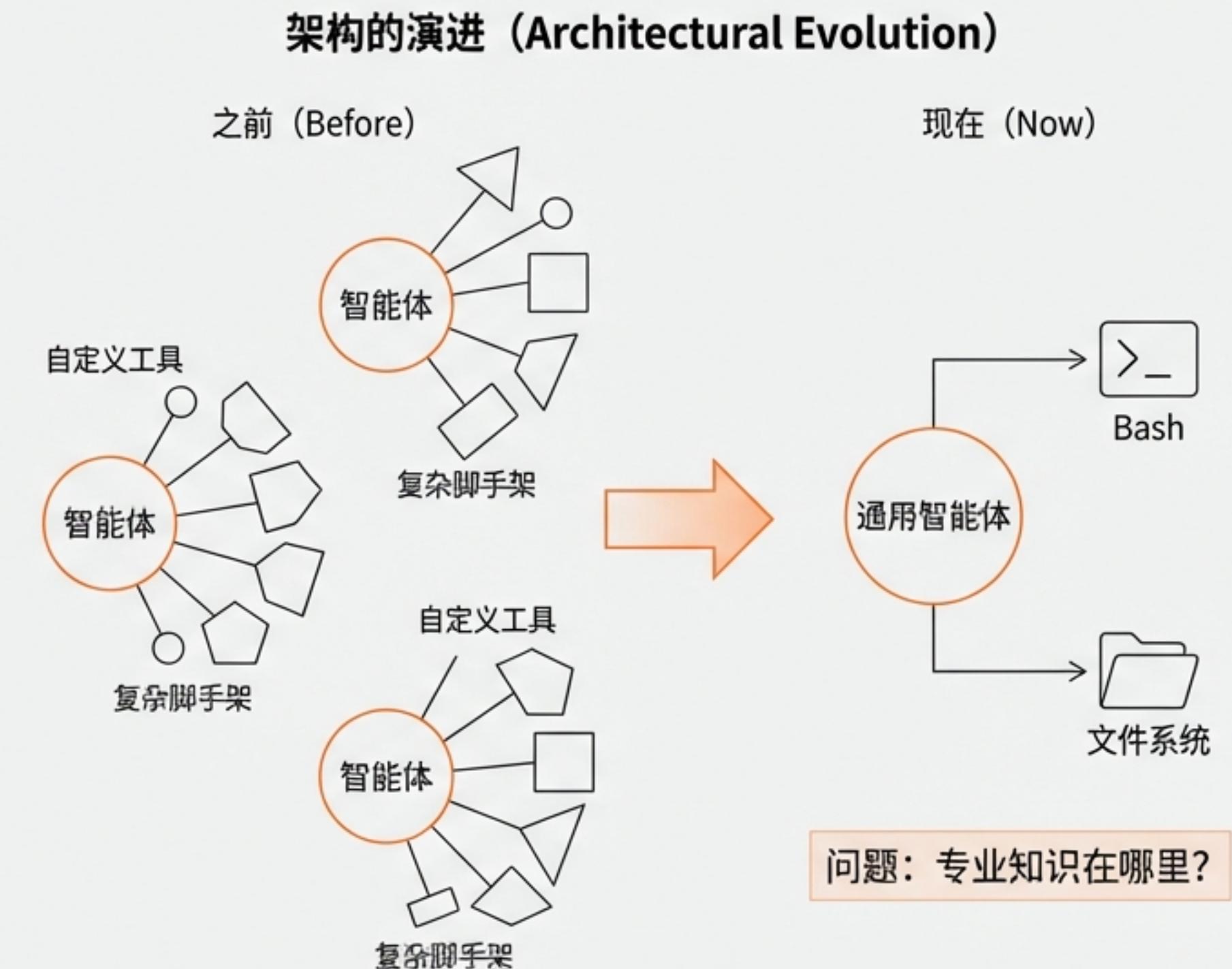
一种为智能体打包和共享
组合式知识的新范式



Anthropic

智能体拥有智慧，但缺乏我们工作所需的专业知识

- 我们已经从探索“什么是智能体”发展到了日常使用智能体的阶段。
- 我们发现了一个新的范式：模型与运行时环境的紧密耦合。我们意识到，**代码是通往数字世界的通用接口**。
- 这种以代码为中心的方法虽然可扩展，但很快就暴露了一个新问题：**领域专业知识的缺失**。



一个核心困境：您希望谁来为您报税？



Mahesh

拥有 300 智商的数学天才
从第一性原理推导税法



Barry

经验丰富的税务专家
遵循可靠流程，确保一致的执行



今天的智能体更像 Mahesh：才华横溢，但缺乏专业知识。
它们无法很好地吸收您的经验，也无法随着时间的推移而学习。

我们的解决方案：智能体技能 (Agent Skills)

技能是打包了组合式程序性知识的有组织的文件集合。

换句话说，它们就是文件夹。

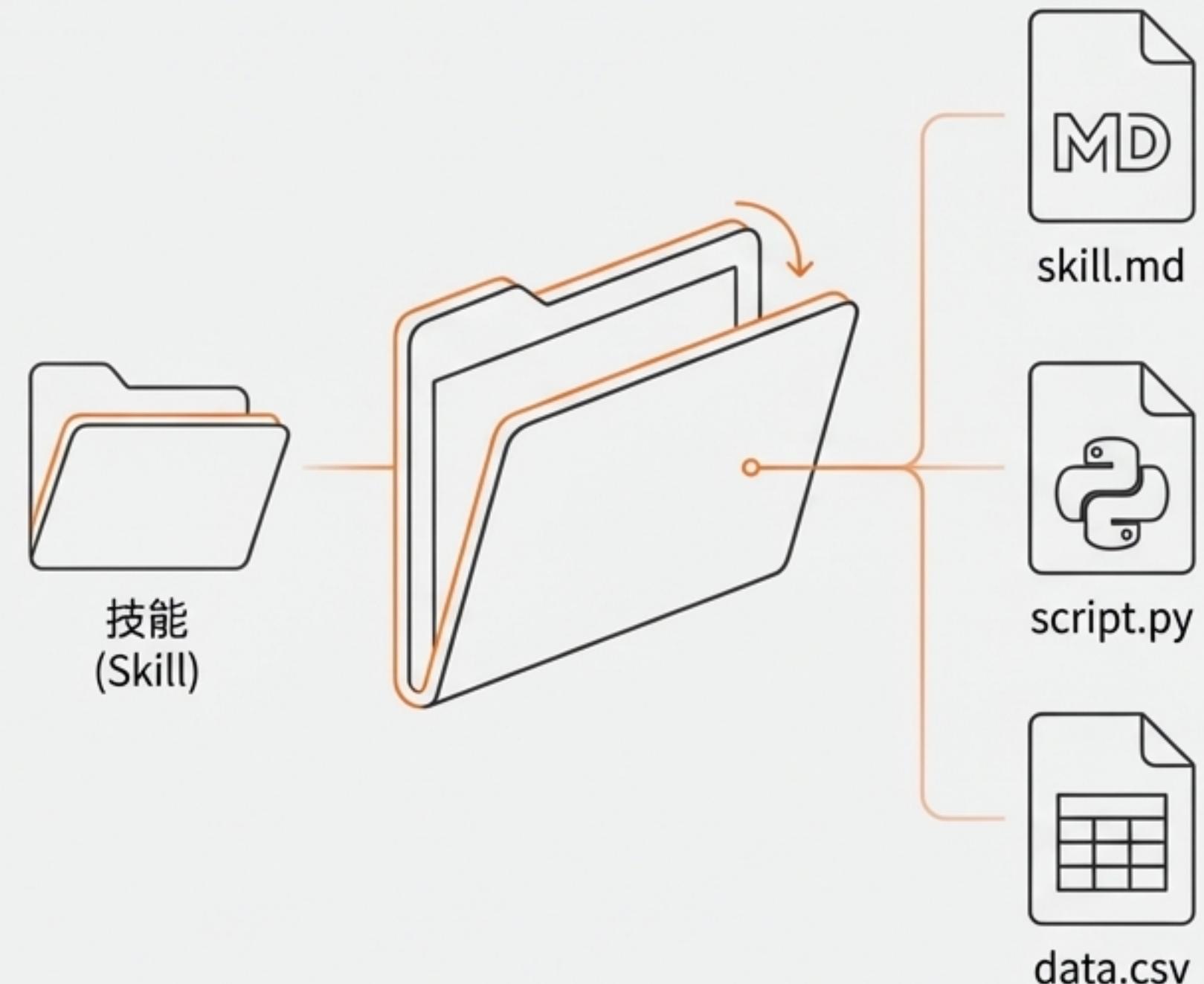
设计哲学



刻意追求简洁：任何拥有电脑的人或智能体都可以创建和使用。



兼容现有工作流：可以用 Git 进行版本控制，放入 Google Drive，或打包分享。



技能如何运作 (1): 将脚本作为工具使用

传统工具的问题

- ✗ 指令模糊不清
- ✗ 模型无法修改工具，陷入困境
- ✗ 始终占用宝贵的上下文窗口

脚本作为工具的优势

- ✓ **自文档化**: 代码本身清晰地表达了其功能。
- ✓ **可修改**: 智能体可以根据需要调整脚本。
- ✓ **按需加载**: 只在需要时才从文件系统加载，节省上下文。

实例

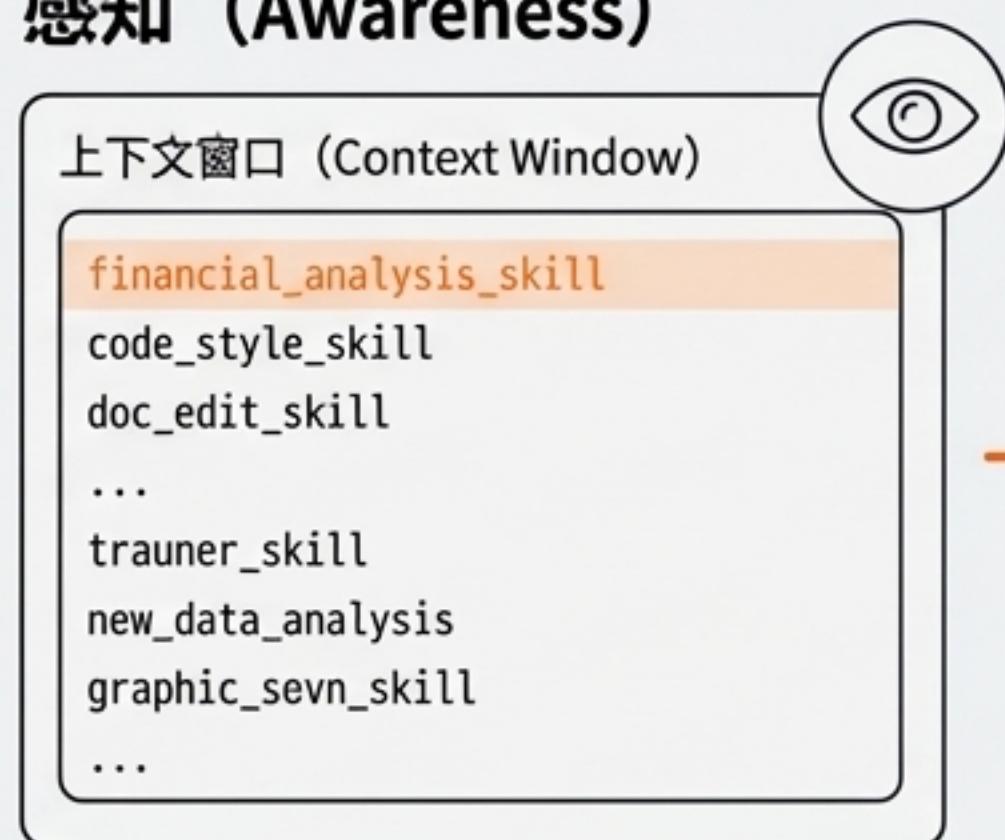
我们观察到 Claude 反复编写一个 Python 脚本来为幻灯片应用样式。我们让它将脚本保存到技能中，供未来的自己使用，从而实现了一致性和高效率。



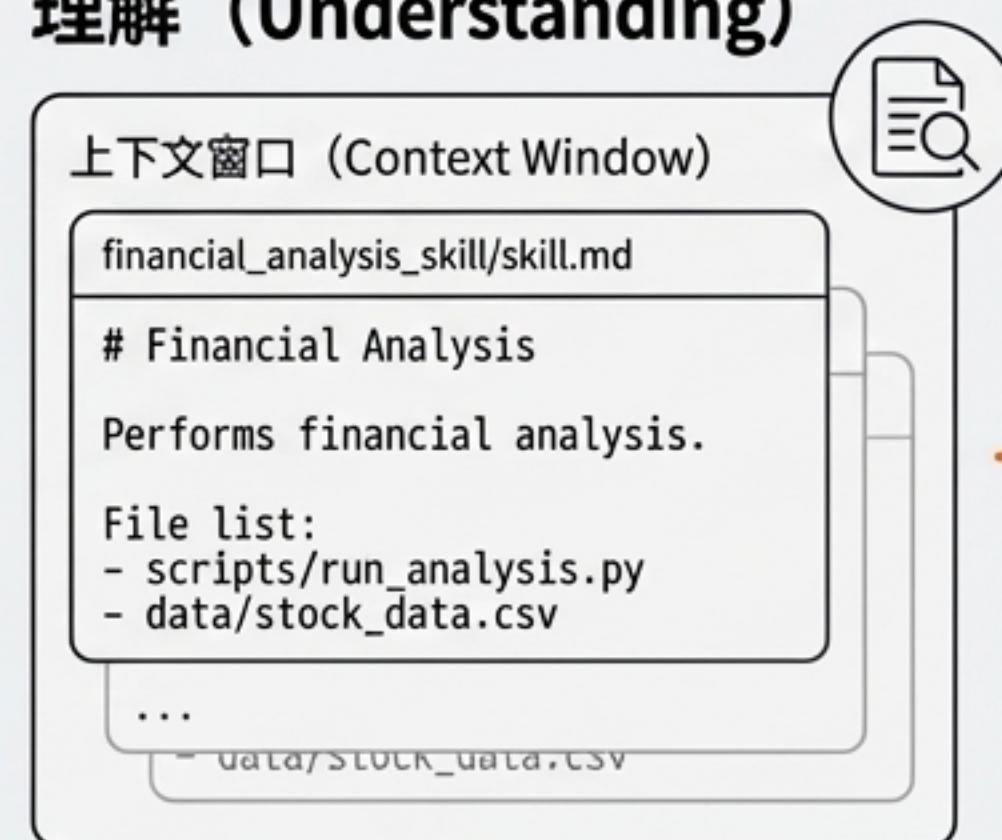
技能如何运作 (2) : 通过渐进式披露保护上下文窗口

为了在上下文中容纳数百个技能并实现真正的组合性，技能在运行时是渐进式披露的。

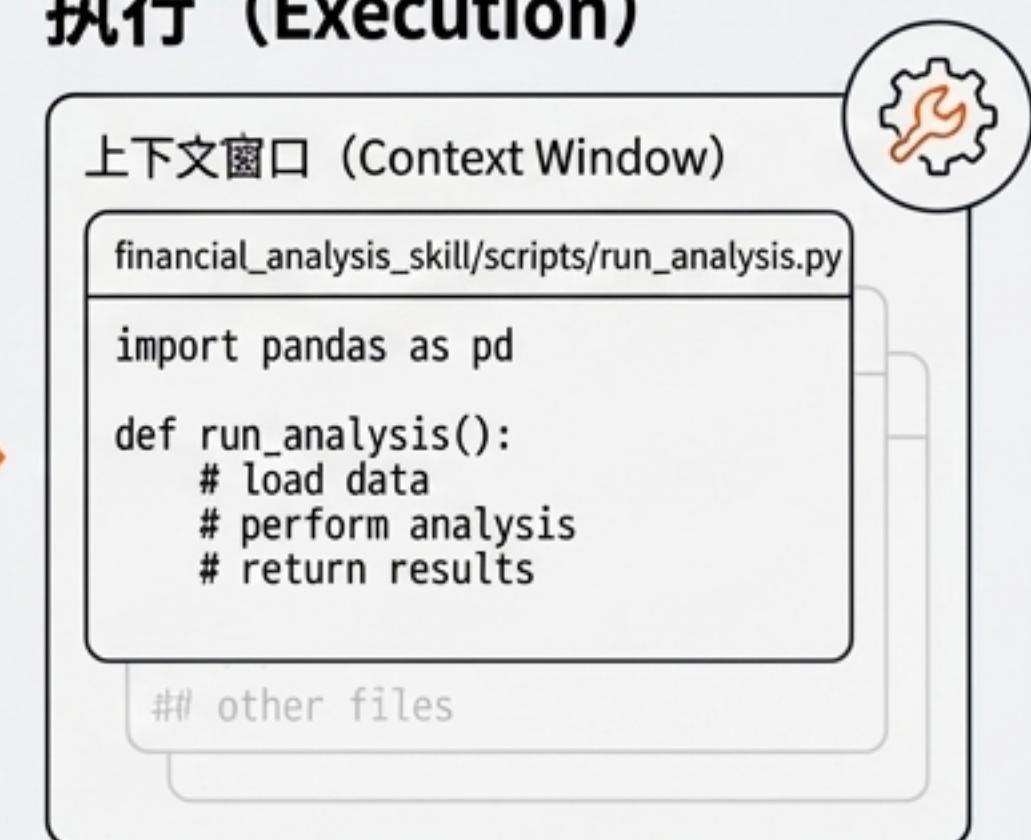
感知 (Awareness)



理解 (Understanding)



执行 (Execution)



模型仅看到技能的元数据 (e.g., `financial_analysis_skill`), 知道技能的存在。

当需要使用技能时，模型读取 `skill.md` 文件，获取核心指令和文件目录。

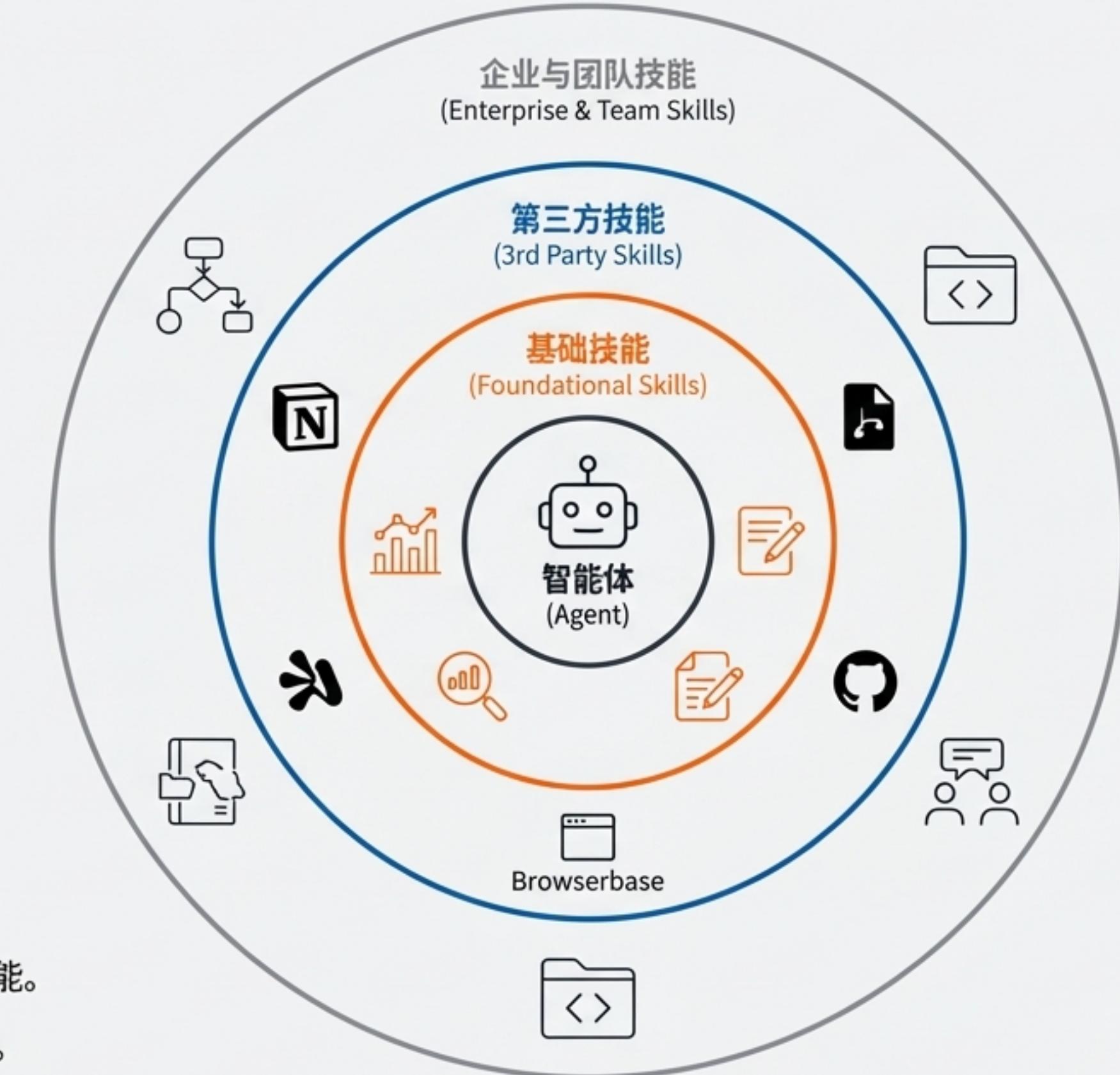
只有在执行特定任务时，模型才会访问文件夹中的具体脚本或文件。

简单的设计催生了蓬勃发展的生态系统

自五周前发布以来，已涌现出 **数千个** 技能。

技能的三种类型

- **基础技能**: 赋予智能体新的通用或领域特定能力。
- **第三方技能**: 由生态系统中的合作伙伴创建，用于增强其产品的功能。
- **企业与团队技能**: 在组织内部构建，用于封装内部知识和最佳实践。



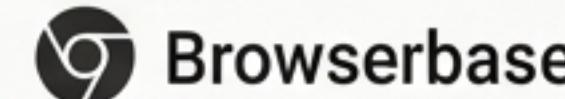
生态系统实践：合作伙伴正在扩展 Claude 的能力

基础与科学研究 (Foundational & Scientific Research)



创建了文档处理技能，使 Claude 能够创建和编辑专业品质的办公文档。

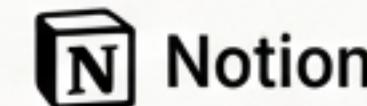
第三方软件集成 (3rd Party Software Integration)



为其开源浏览器自动化工具 Stagehand 构建了技能，使 Claude 能够更有效地浏览网页。



构建了科学技能，赋予 Claude 分析 EHR 数据和使用生物信息学学库等新能力。



发布了一系列技能，帮助 Claude 更好地理解并深度研究您的 Notion 工作区。

企业内部的最大牵引力：将组织知识编码为技能

封装最佳实践（Capturing Best Practices）



世界 500 强公司正在使用技能来教导智能体其独特的组织工作流程和内部软件使用方法。

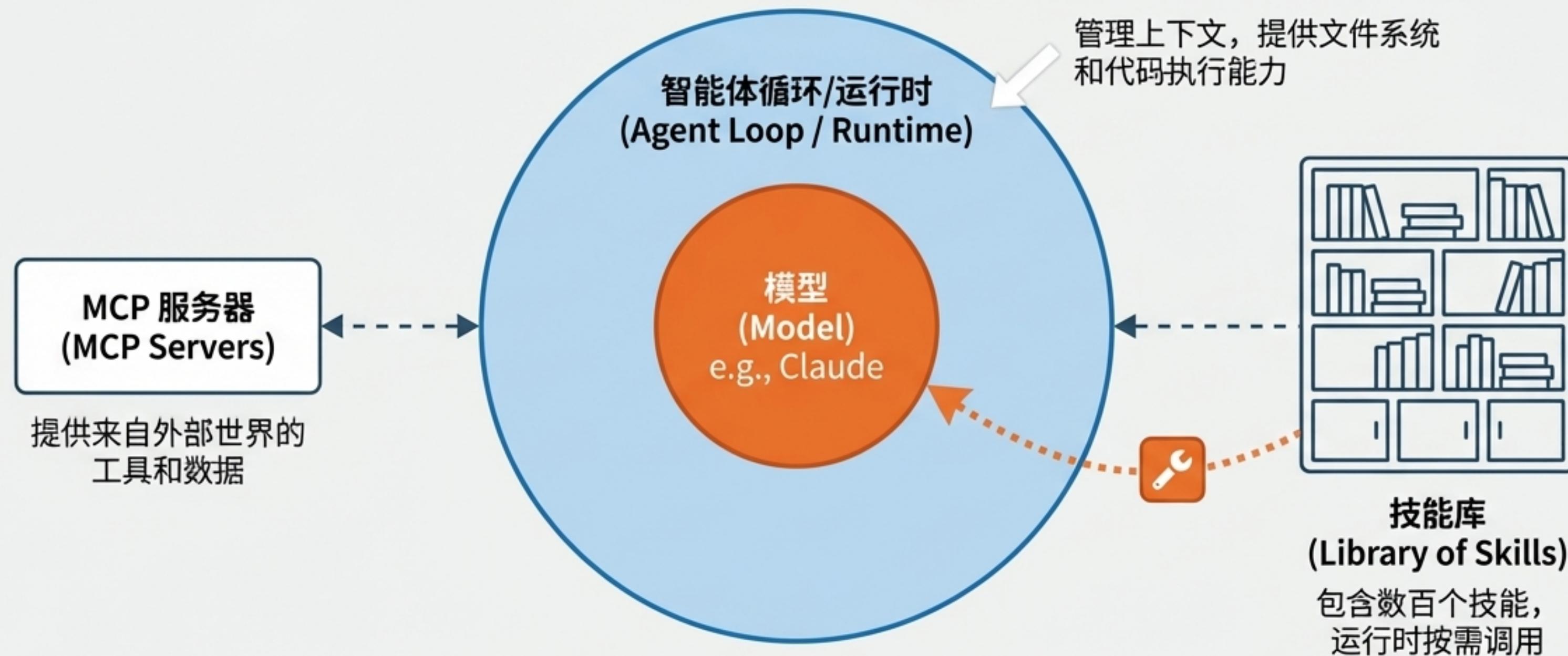
提升开发者生产力（Boosting Developer Productivity）



服务于数万名开发者的团队正在使用技能来部署像 Claude Code 这样的智能体，并教导它们关于代码风格、最佳实践等内部规范。

非技术人员（财务、招聘、法务等）也开始构建技能，这验证了技能降低了扩展通用智能体门槛的初步设想。

新兴的通用智能体架构



在 Anthropic，我们通过为智能体配备特定的 MCP 服务器和技能库，迅速将 Claude 部署到金融服务和生命科学等新垂直领域。

技能的复利效应： 构建集体演化的知识库

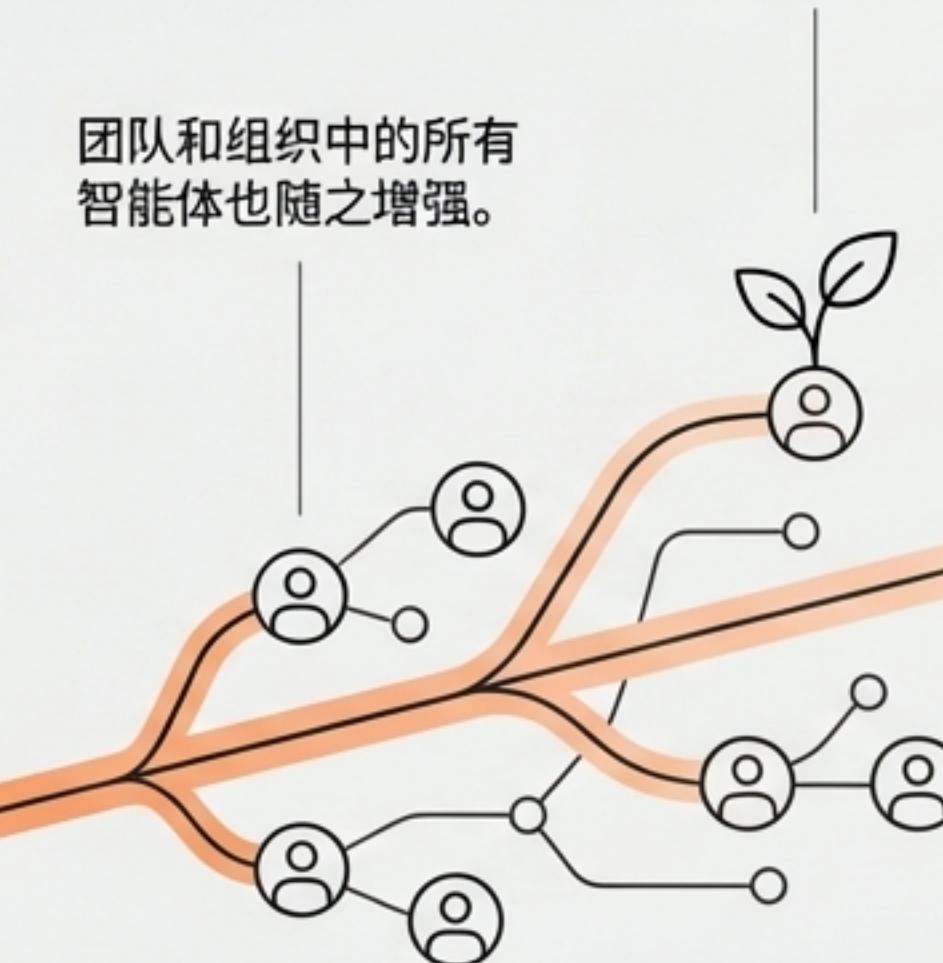
一个由组织内的人和智能体共同策划、
不断演化的能力知识库。



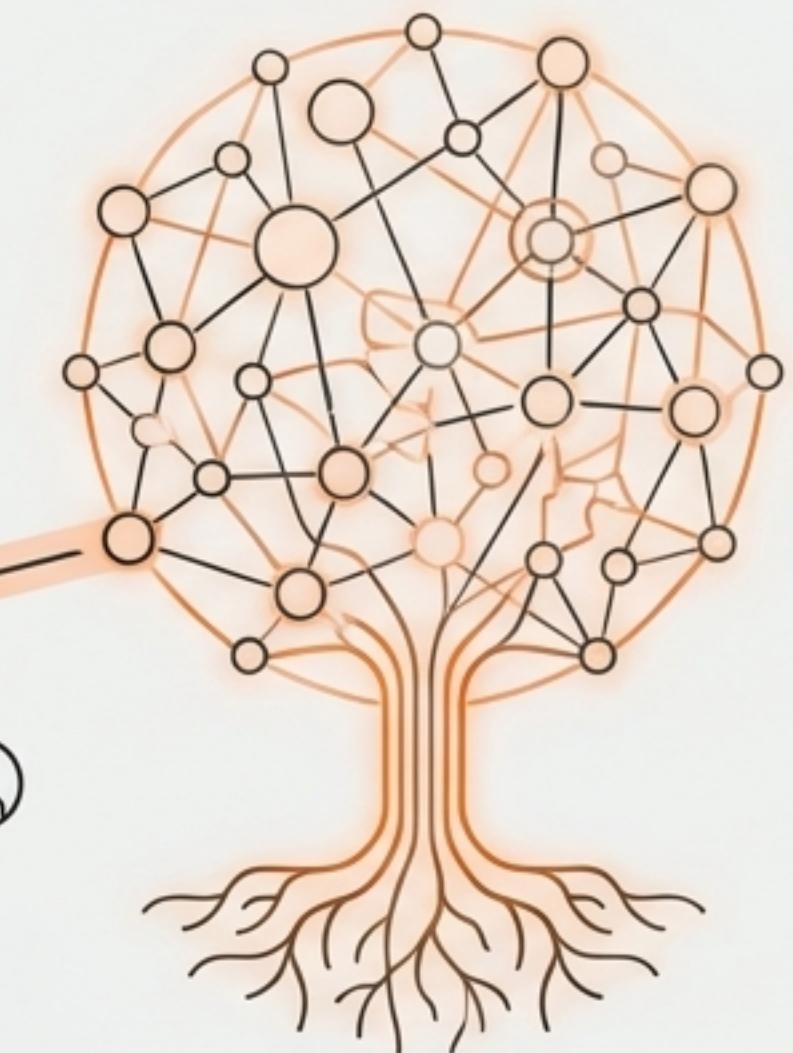
Day 1
Claude

您与智能体互动，提
供反馈和机构知识。

智能体通过技能将这
些知识固化下来，变
得更强。

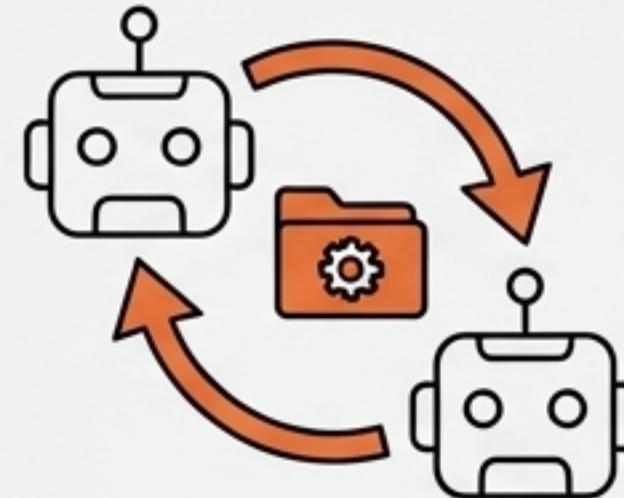


新成员第一天使用的
Claude 就已经了解团
队的工作方式。



Day 30
Claude + Team Knowledge
Claude + 团队知识

解锁真正的持续学习



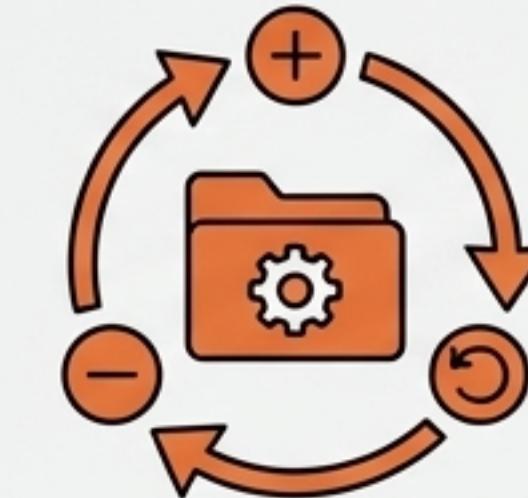
学习的可转移性 (Transferable Learning)

标准化的格式保证了 Claude 写下的任何东西都可以被未来的自己高效使用。



记忆的具体化 (Tangible Memory)

技能使“记忆”的概念变得具体。它们不捕捉一切，只捕捉智能体可在特定任务上使用的程序性知识。



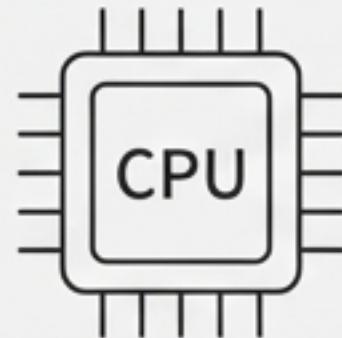
能力的动态演化 (Dynamic Evolution of Capabilities)

Claude 可以即时获取新能力，按需发展它们，并放弃过时的能力。

今天，Claude 已经可以使用我们的“技能创建者”技能为您创建技能。

AI 的演进正在重演计算平台的历史

阶段一

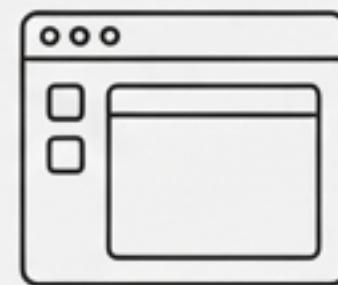


计算平台

CPU (投入巨大, 潜力无限,
但本身用途有限)

人工智能
模型 (Models)
(e.g., Claude)

阶段二

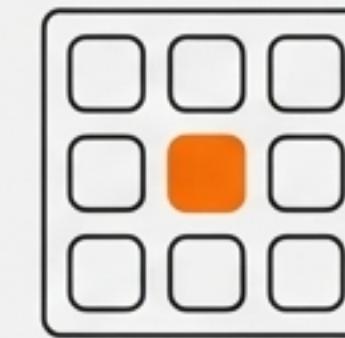


计算平台

OS (通过协调资源和进程,
使处理器更有价值)

人工智能
智能体运行时 (Agent Runtimes)
(通过高效的抽象, 管理模型的
输入输出)

阶段三



计算平台

Software (数百万开发者在
此编码领域知识)

人工智能
技能 (Skills)

技能是 AI 的应用层——是时候开始构建了

- 少数公司构建处理器和操作系统，但数以百万计的开发者构建了应用程序，将领域专业知识和独特视角编码其中，解决了真实世界的问题。
- 我们希望技能能为每个人打开这一层。只需将东西放入文件夹，我们就能发挥创造力，为自己、为彼此、为世界解决具体问题。
- 我们正在趋同于一个通用的智能体架构。
- 我们已经创建了“技能”这个新范式来交付和共享能力。

是时候停止重复构建智能体，开始构建技能了。



github.com/anthropics/clause-agent-sdk