



从单智能体走向群体智能： SkillClaw 让 Agent 实现协同进化

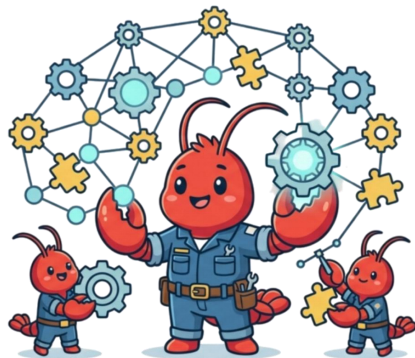
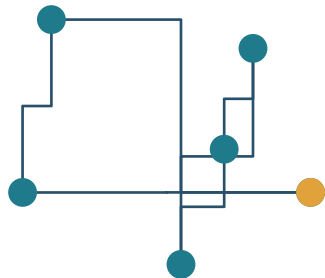
Let Skills Evolve Collectively with an Agentic Evolver

让经验在群体里、在 Skill 层持续演进，关于群体进化的一次探索

SkillClaw 团队

2026.07.03

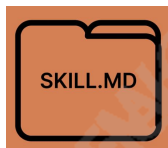
AgenticAiCon • 自进化智能体研讨会



MOTIVATION

群体 / Cowork Agent 现状

从共享助手到 agent cowork buddy



Skills

- 能力封装
- 可复用、可共享



OpenClaw

- 长期运行、自主规划
- 私人助手



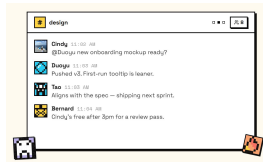
Hermes

- Skill驱动持续进化
- 学习经验、自动升级



Multica

- Agent作为预定义员工
- 看板、任务分派



Raft

- 频道式协作
- 实时对话

@Claude

Claude Tag

- 频道中全员共享
- 记忆层面连接用户群体

2025

2026

个体能力阶段

专注于单个Agent能力构建、经验积累

自进化阶段

Agent具备学习、进化能力

群体管理阶段

管理多个异构Agent协同工作

生态协同阶段

Agent群体生态化

经验“沉淀”与“共享”之困

问题一：经验难沉淀



关键
瓶颈



问题二：经验难共享



如何让经验在群体里获得复利？

共享了 Agent，能力却没有一起变强。多个用户、多台设备，反复在同一个坑上摔跤。



同一个坑，被每个人各踩一遍。 今天这个 agent 学会了，没有沉淀成可复用的经验，别人从零再来。

在 **Skill 层**，可以复用从Agent在反馈中的经验，持续进化

将经验从 Agent 个体会话中提取出来，在 skill 层进行沉淀、复用

Agent群体行为沉淀为 Skill

Skill A

Skill B

Skill C

Skill D

整个群体复用，经验在组织层面积累。

Skill: 承载经验的载体

可复用 + 可验证 + 可分发, 沉淀交互获得的环境反馈, 天然适合作为「进化」和「共享」的载体。

1

好的行为 pattern

将「怎么把事做好」的概念写成可复用的 SOP

2

可验证

能在真实环境里跑, 判断到底有没有用

3

可分发

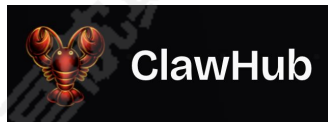
一处沉淀, 同步给所有 agent 和人

4

独立于模型权重之外

在模型权重之外承载行为方法

开源 Skill 平台应运而生, 但形式仍为「静态」, 无法通过 Agent 交互本身进化



SkillHub



面向群体的 Skill 进化：SkillClaw

SkillClaw 让群体在面对多种不同任务时，把经验沉淀、复用起来。

做multi-session的跨用户、跨时间的多源数据汇总

从聚合好的同类任务中提炼出可复用的经验，在 skill 层面形成可复用范式

在上线、共享前需要经过安全、有效性评估以及校验，以及后续的治理



SkillClaw 的定位

经验范围 × 共享形态 四象限：定位不同 Agent 系统

上下文 / 记忆共享 ↑

同步 Multi-Agent 编排

当前任务中进行分工、规划、协作
共享的经验不会跨越任务

Agent Swarm / Claude Code subagents



共享上下文 / Memory

@Claude

将偏好、规范、知识通过记忆留存
并通过记忆隔离

Claude Tags

← 当前任务内使用

跨任务长期沉淀 →

个体 Agent

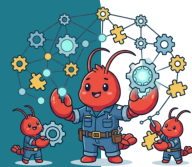
一个 Agent 使用工具解决当前任务
经验停留在个体

Hermes Agent / OpenClaw



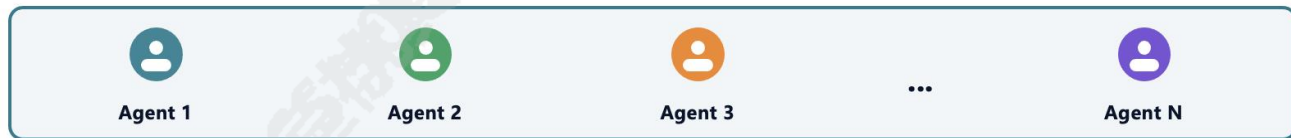
Skill 群体进化

多个 Agent / 多次执行的轨迹被聚合
进行长期沉淀，形成可复用、可演化的 Skill
SkillClaw



经验 / Skill 共享 ↓

SkillClaw 架构总览



记录完整的 Agent 内容
将命中的 skill 进行加载、注入
为不同 Agent 客户端提供
统一的 Skill 生成与同步入口



记录所有的 Agent 群体 session + 进化后的 skill
群体历史、与群体经验进行汇聚



负责从共享存储中挖掘 agent
交互记录, 发现可复用范式,
进化并持续改进已有 Skill

● ① Evidence

1. 进化信号来源



● 信号 1 | 人类反馈 ✓



当 Agent 执行结束后，人类的一次修正或拒绝往往可以代表对当前执行的不满意，而确认或结束则大概率代表 Agent 的有效执行

● 信号 2 | 环境反馈



多次的错误如编译失败、权限不足、甚至是在 Agent 自身的 self-verify 中最终的结果不符合预期，都可以反应当前执行的缺陷；



当不同 session 命中了同一 skill，相当于构造了大量天然的消融，便于直接分析生效、失效原因，基于此实现实例级 skill 进化

2. Skill 进化管线

成功 session 沉淀行之有效的行为模式；失败 session 提炼亟待修正的问题模式。

方式1: 静态pipeline

流程由固定的三阶段形式组成：摘要 → 总结 → 执行，优势是稳定，能够结构化的处理进化流程



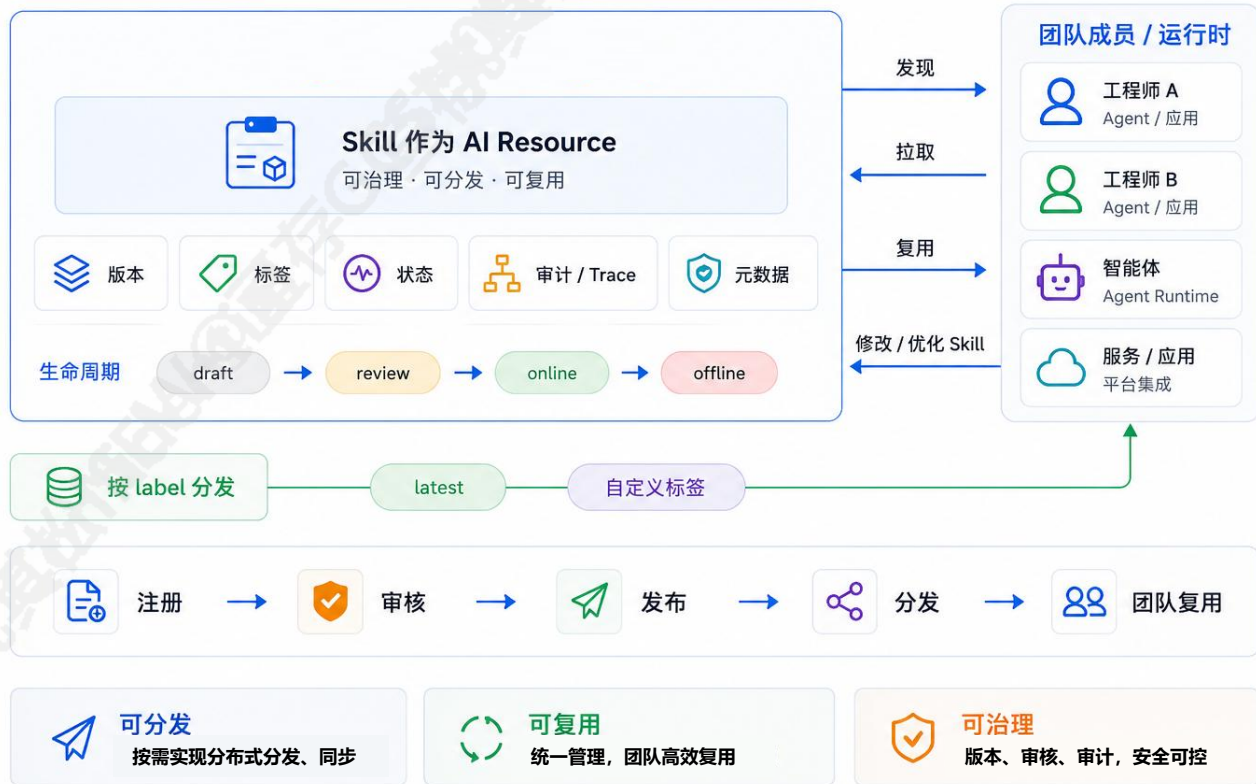
方式2: Agentic Evolver

Agentic Evolver 让在 Harness 中的 Agent 自主决定对聚合 Session 的沉淀去留，能够适应不同、非结构化的 skill 场景，当 skill 本身面临重构、合并、或边界条件时能够通过 Agent 自主决策



3. Skill 进化后如何管理

把 Skill 从个人经验文件，变成可分发、复用、治理的 AI 资产



进化端自动校验

Critic Agent 自动生成测试用例，在模拟环境中进行自检

用户端的真实验证

在用户侧闲时进行分发、重放，在真实场景中评估效果

构成了模拟到真实的验证闭环
确保进化后的 skill 经过检验

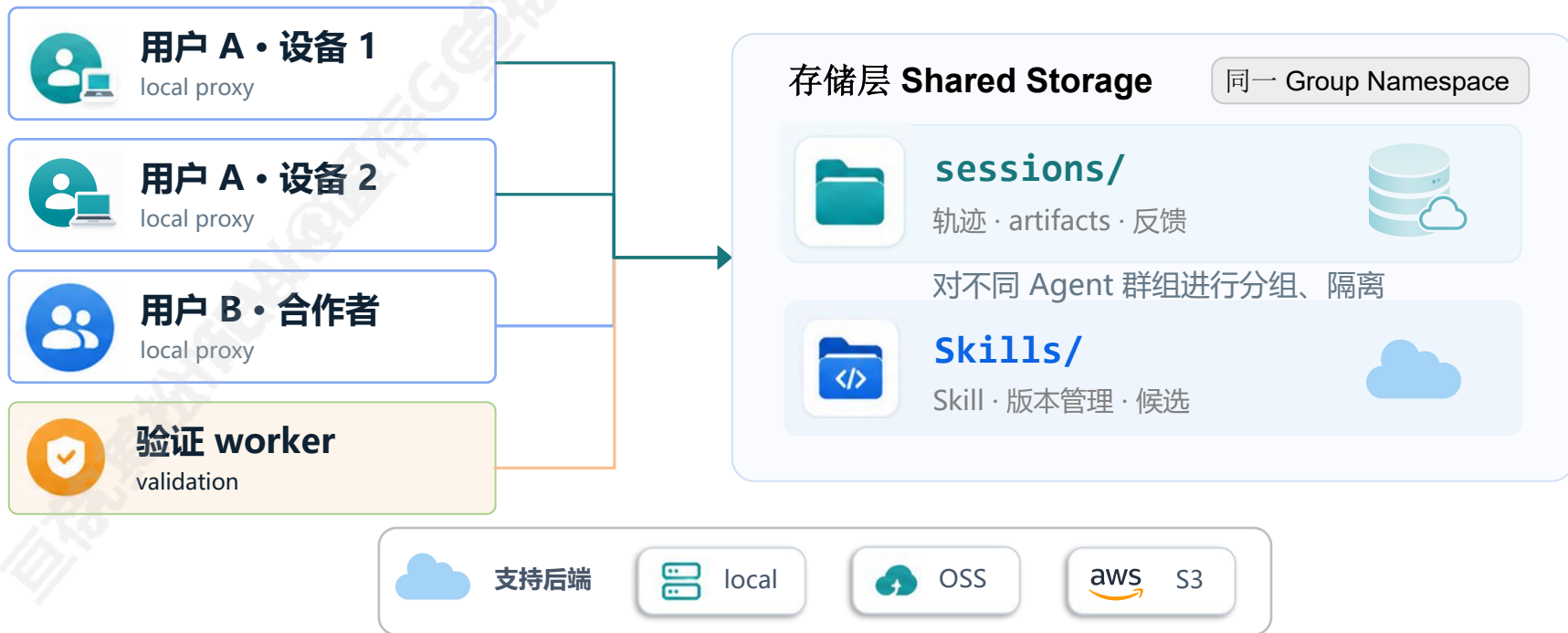
Client Proxy: 低侵入接入

用户照常和 agent 聊天，只要把 agent 的 API endpoint 指向本地 proxy，SkillClaw 在后台记录、注入、同步。



跨用户 / 设备，分布式管理

接入 SkillClaw Proxy 的 Agent 会通过共享存储层进行交互，共享存储中保存两类数据



团队级 Skill 治理

和 Nacos 合作落地：把 Skill 当作 AI 资产纳入统一的注册中心进行管理、治理、分发，让 skill 具备完整的生命周期

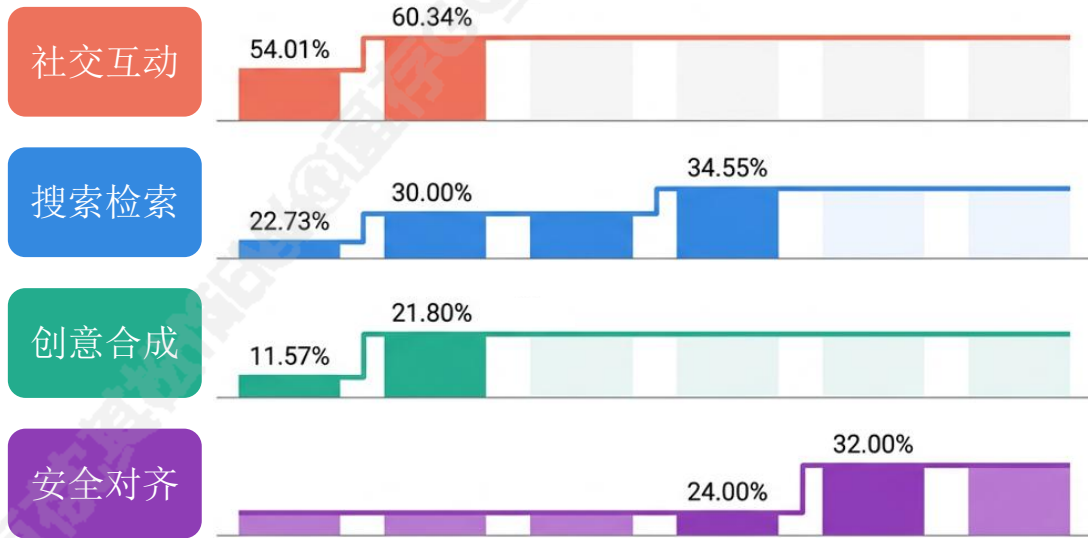


● 实验结果

WildClawBench 实验结果

在真实环境 agent 评测进行评测，基于环境反馈信号的多轮群体进化带来持续提升

多轮群体进化



实验发现

- **多轮**进化能够带来稳定提升
- 通过**环境反馈**获得的监督信号能够在真实任务中持续优化 Skill
- 任务类型来看，**过程经验更易沉淀**（工具调用、环境适配、错误恢复等），依赖推理的任务收益更小

评测基于真实环境的端到端 agent benchmark (WildClawBench)，60 个真实世界 Agent 任务

● 总结与展望

SkillClaw: 从单智能体走向群体智能



总结: SkillClaw 的核心价值

01

Skill 是群体智慧的沉淀

将一次性交互结果转化为可复用、可验证、可分发的能力资产



02

SkillClaw 构建进化闭环

聚合 → 归因 → 进化 → 验证 → 治理, 让 Skill 在真实反馈中持续变强



03

从个体交互走向群体交互

Agent 在群体范围内共享、验证并迭代行为模式, 实现持续自我进化



展望: 迈向规模化的 Agent Skill 生态



更大规模的真实交互

更多 Agent / 用户 / 设备 / 任务场景



更长期的持续演化

跨天 / 跨任务 / 跨环境的反馈积累与更新



更可靠的 Skill 治理

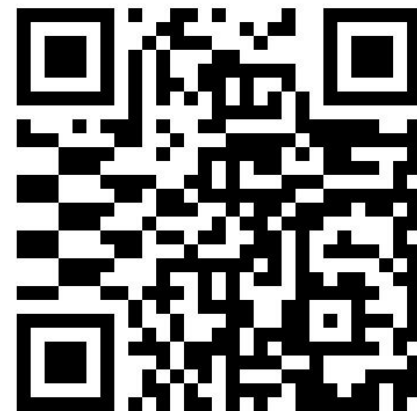
验证 / 版本管理 / 分布式同步 / 可信分发



SkillClaw: Let Skills Evolve Collectively with an Agentic Evolver

Github: <https://github.com/AMAP-ML/SkillClaw>

Paper: <https://arxiv.org/abs/2604.08377>



SkillClaw

· DreamX Team

Thanks