



TRAE Agent 架构演进： 从 Workflow 到 Agentic Loop

张聪

字节跳动 / TRAE 架构师



📍 北京

QCon

全球软件开发大会

会议时间：4月10-12日

- 大模型赋能 AI Ops
- 越挫越勇的大前端
- 云上业务架构演进
- 多模态大模型及应用
- 海外 AI 应用创新实践

📍 北京

AICon

全球人工智能开发与应用大会

会议时间：6月27-28日

- 端侧智能
- AI Agent
- 多模态大模型
- 金融+大模型

📍 上海

QCon

全球软件开发大会

会议时间：10月23-25日

- AI Agent
- 大模型训练推理
- 端侧 AI
- 搜推深度融合
- 数智金融

4月

5月

6月

8月

10月

12月

📍 上海

AICon

全球人工智能开发与应用大会

会议时间：5月23-24日

- 通用大模型
- AI Agent
- 垂直领域应用
- 模型可解释性

📍 深圳

AICon

全球人工智能开发与应用大会

会议时间：8月22-23日

- 模型效率与部署
- 多模态大模型
- 大模型安全
- 智能硬件

📍 北京

AICon

全球人工智能开发与应用大会

会议时间：12月19-20日

- 通用大模型
- 智能硬件
- LM Ops
- 具身智能

极客邦科技 2025 年会议规划

促进软件开发及相关领域知识与创新的传播



参会咨询



查看会议

个人简介



工作经历:

- 2012 年毕业, 12 年大规模流量基础设施系统相关工作经验, 精通 CDN 及高性能网络数据平面架构设计。
- 目前在字节跳动 TRAE, 致力于 AI Coding 方向 Agent 工程能力的建设。

技术变迁:



Email: timebug.info@gmail.com

目录

01 业界趋势

02 工程实践

03 案例分享

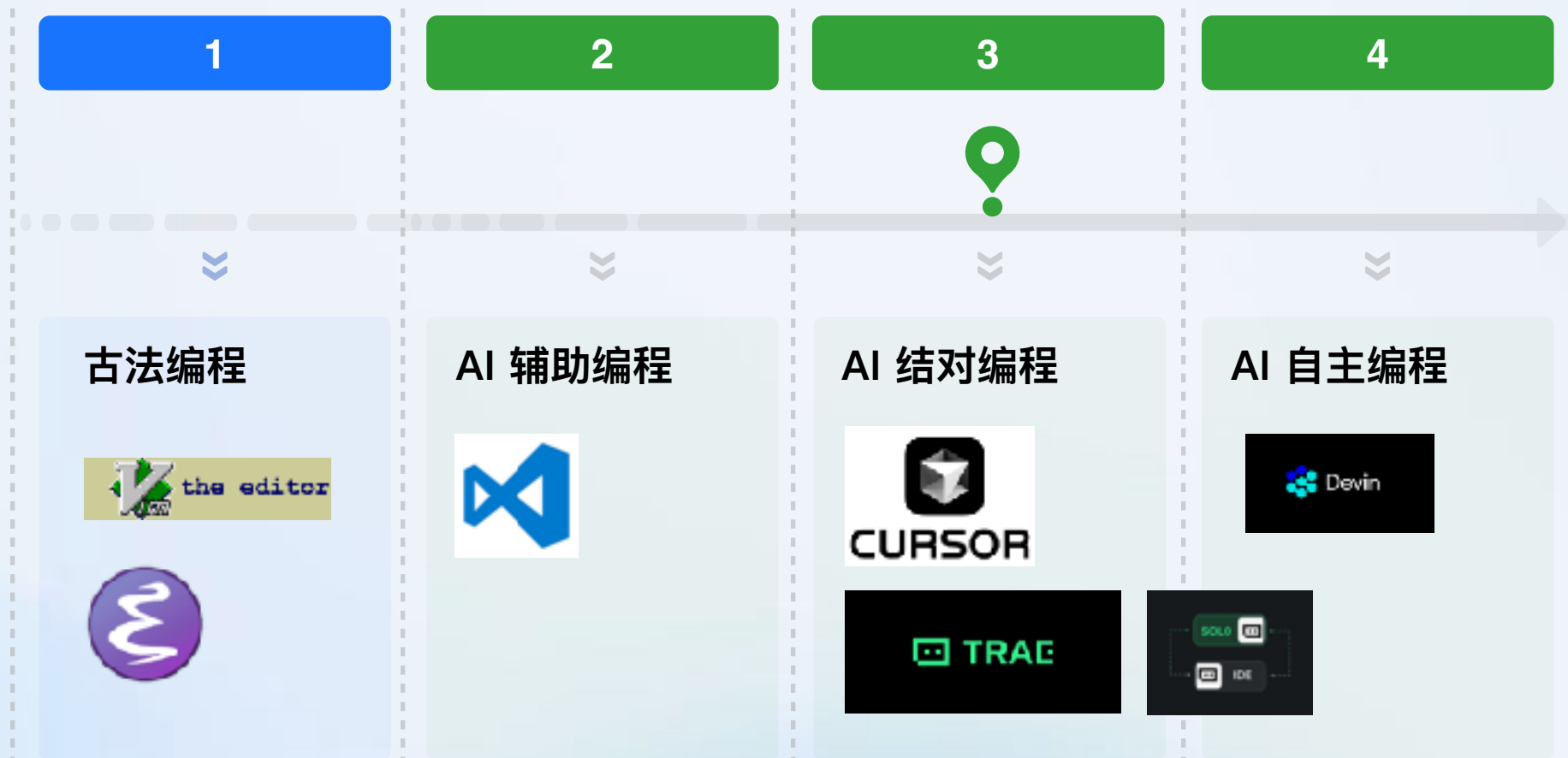
04 未来展望

01 业界趋势

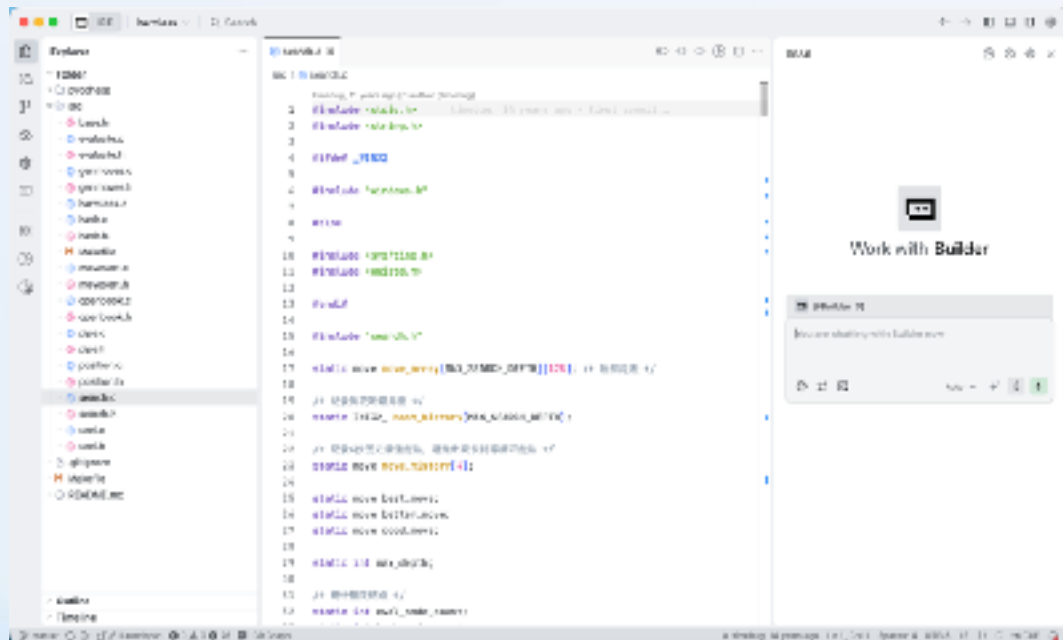
AI Coding 发展过程中 IDE 形态的变化

AI Coding 发展过程中 IDE 形态的变化

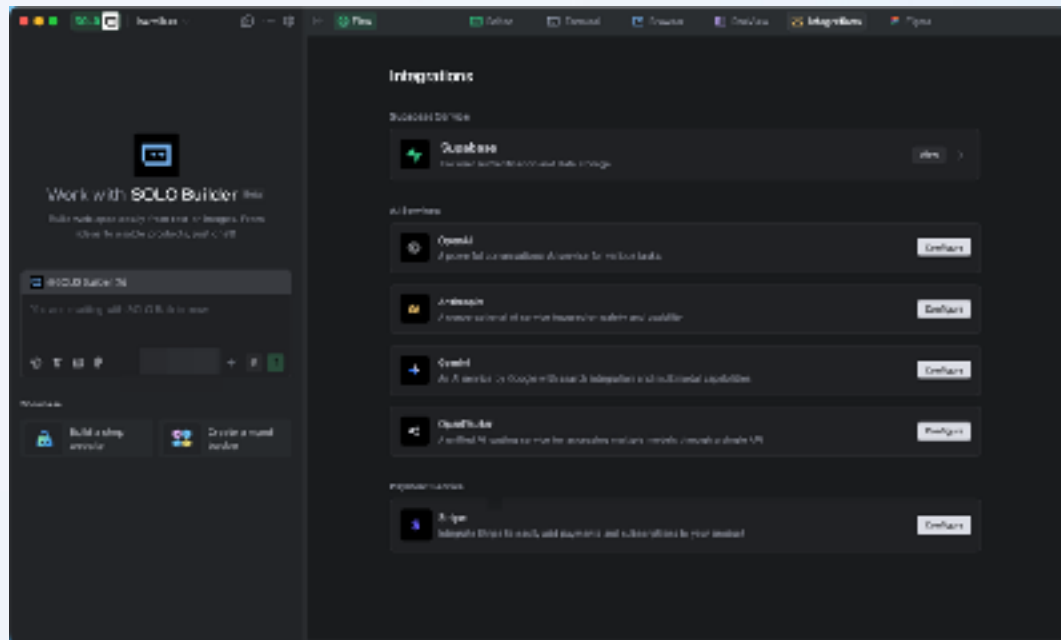
从辅助编程、结对编程到自主编程，AI 正在重塑软件开发流程



关于 TRAE



IDE 模式

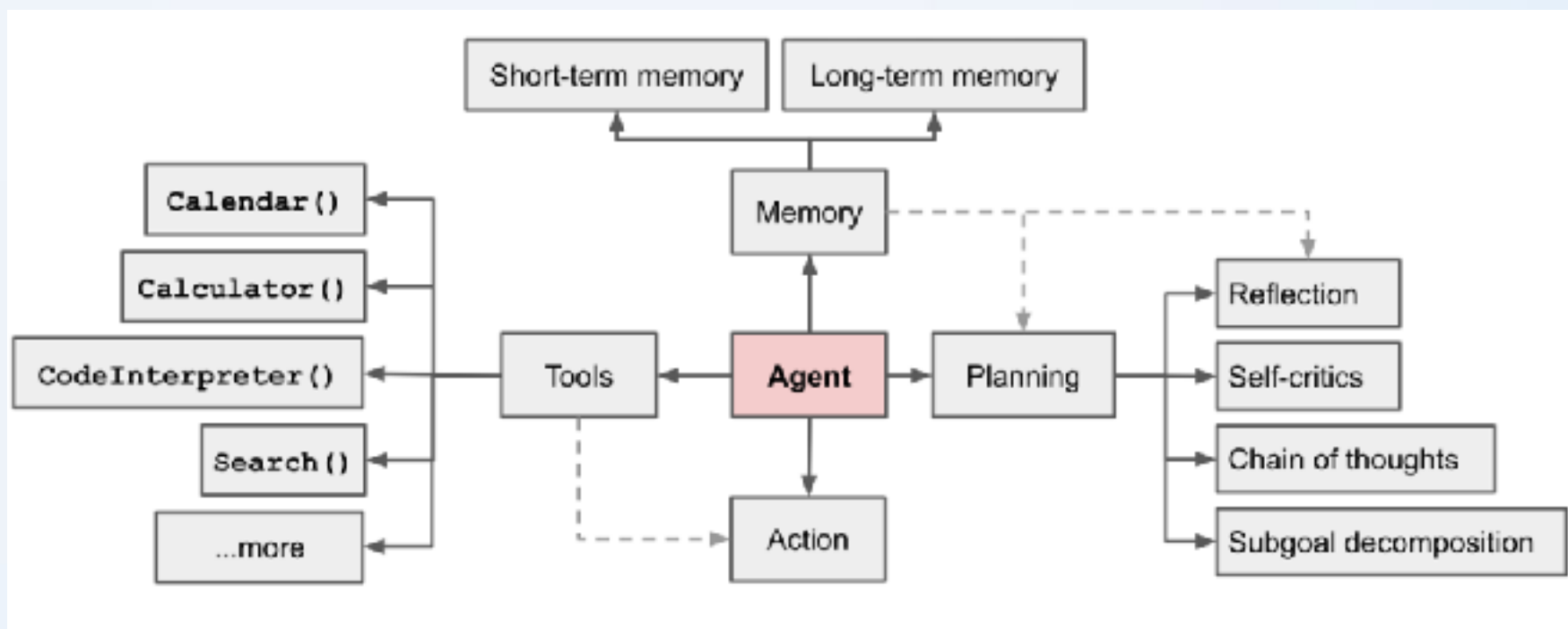
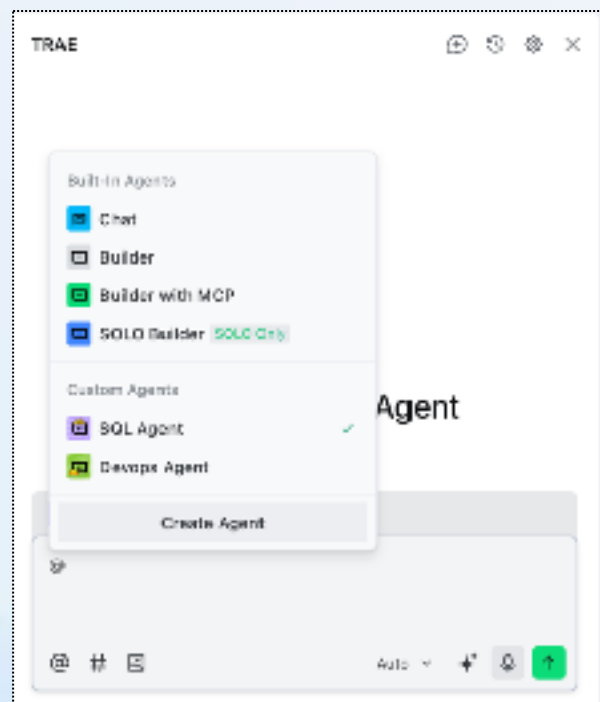


SOLO 模式

02 工程实践

TRAE Agent 架构演进

关于 AI Agent



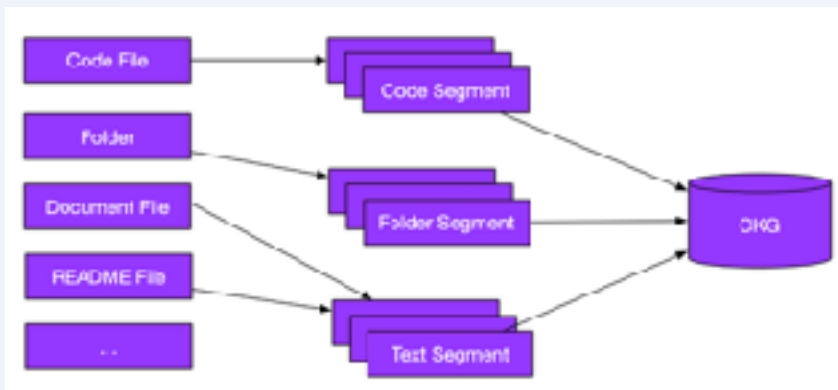
TRAE Agent 架构演进 – Side Chat

AI 辅助理解代码和分析问题



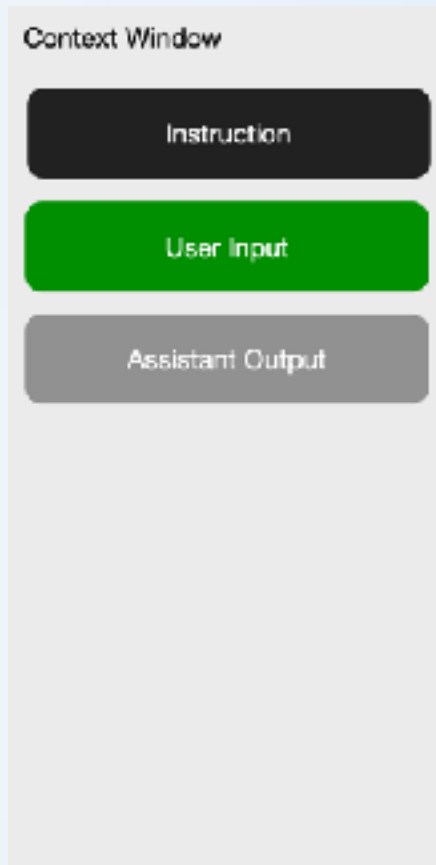
Context References

- 代码文件
- 目录结构
- Code Knowledge Graph



Fast Apply

- 一键合并代码
- Search Replace 工程替换 + 小模型兜底



TRAE Agent 架构演进 – Agent 1.0

Let @Agent #Context deliver "user query"



Workflow: Proposal → Code RAG → Plan/ToolCall Loop

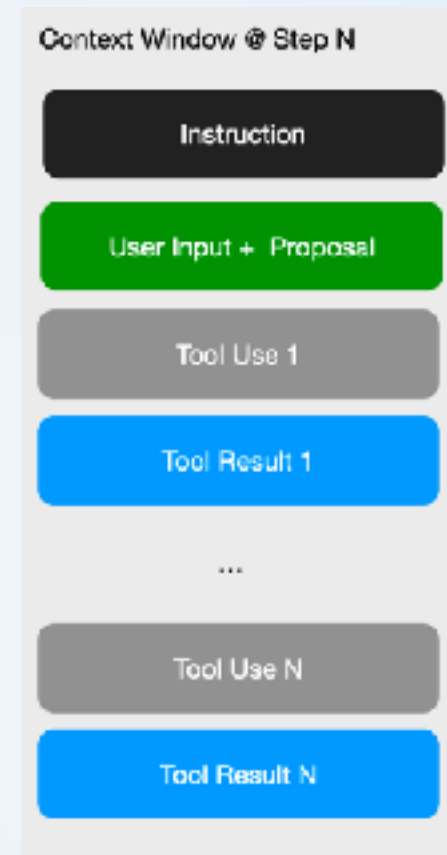
- 携带历史对话信息，引导模型产出需求的解决思路提案
- RAG 召回项目索引信息补充到上下文，帮助模型更好理解项目
- 结合原始需求、解决思路提案以及项目相关上下文信息，引导模型给出规划和工具调用，并持续此过程直到任务结束

Tools

- 内置工具：文件系统/终端/预览/搜索等
- MCP 集成

Context

- #Workspace #Code #File #Folder #Doc #Web



TRAE Agent 架构演进 – Agent 2.0

随着模型变强，去掉部分固化的 Workflow，完全交由模型驱动

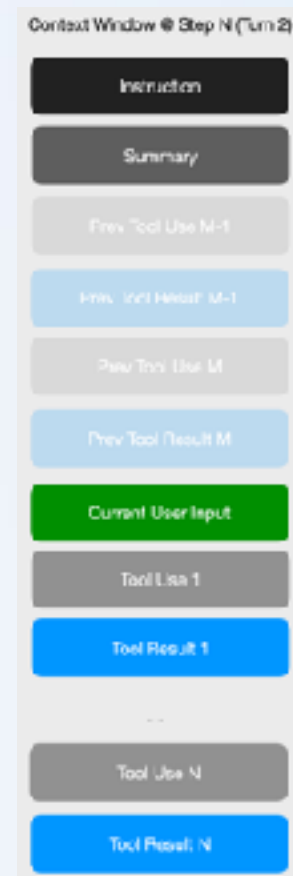


More Agentic Loop

- 去掉 Proposal 流程，让模型结合包括历史在内的上下文信息直接驱动
- 引入 Search Codebase Tool，代替固定的 Code RAG 流程，实现 Agentic RAG

More Seamless Context Flow

- 利用模型对历史上下文进行总结压缩，代替工程裁剪，保留更多有效信息
- 更大的上下文窗口，并利用滑动窗口机制由新到旧进行动态拼接
- 完善 Long-term Memory 机制，让跨多轮会话的重要信息不被遗忘



TRA E Agent 架构演进 – Agent 2.0

More Agentic != More Controllable



TO-DO List & Repeated Reminder

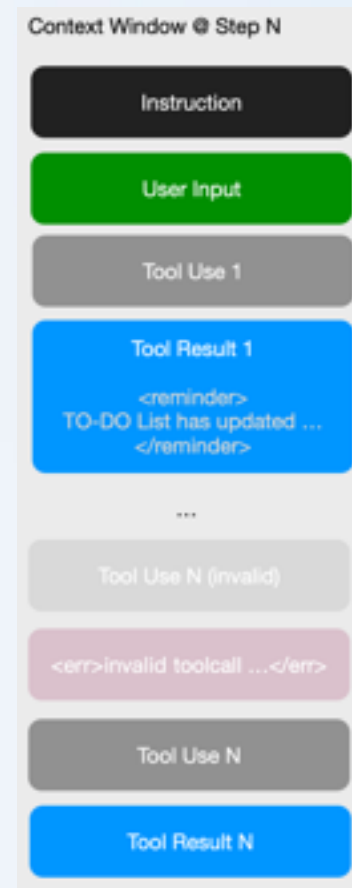
- 引入 TO-DO Tool 约束流程避免模型过度发散，同时在过程中可以被更新以实现纠偏
- 为了防止模型在多轮后遗忘重要信息，利用 Reminder 机制反复强调，让模型注意力更集中

Effective Feedback

- 将 Lint Error 信息在模型每次修改完文件后及时反馈，尽早规避语法错误
- 对于模型幻觉导致的非法 JSON Schema、无效工具调用等进行及时引导纠正

Keep Tools Simple

- 精简工具列表，避免引入多个语义重叠的工具，降低模型决策难度
- 使用 Native Function Call 代替 JSON Schema 描述工具，减少格式幻觉



TRAE Agent 架构演进 – Agent 2.0

优化上下文窗口管理，让多轮会话上下文更加连贯



Short-term Memory

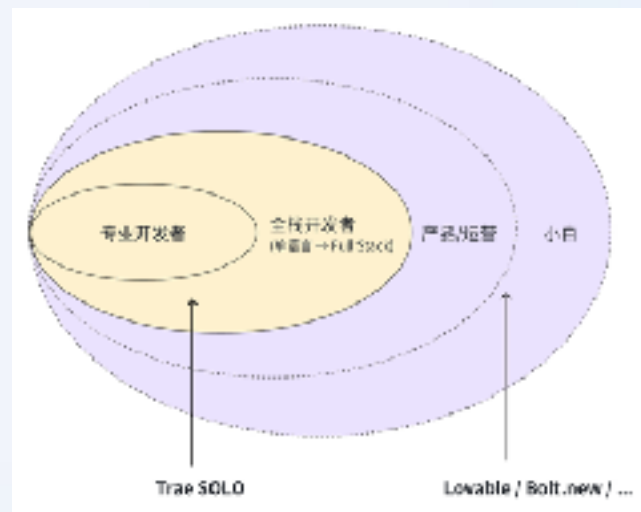
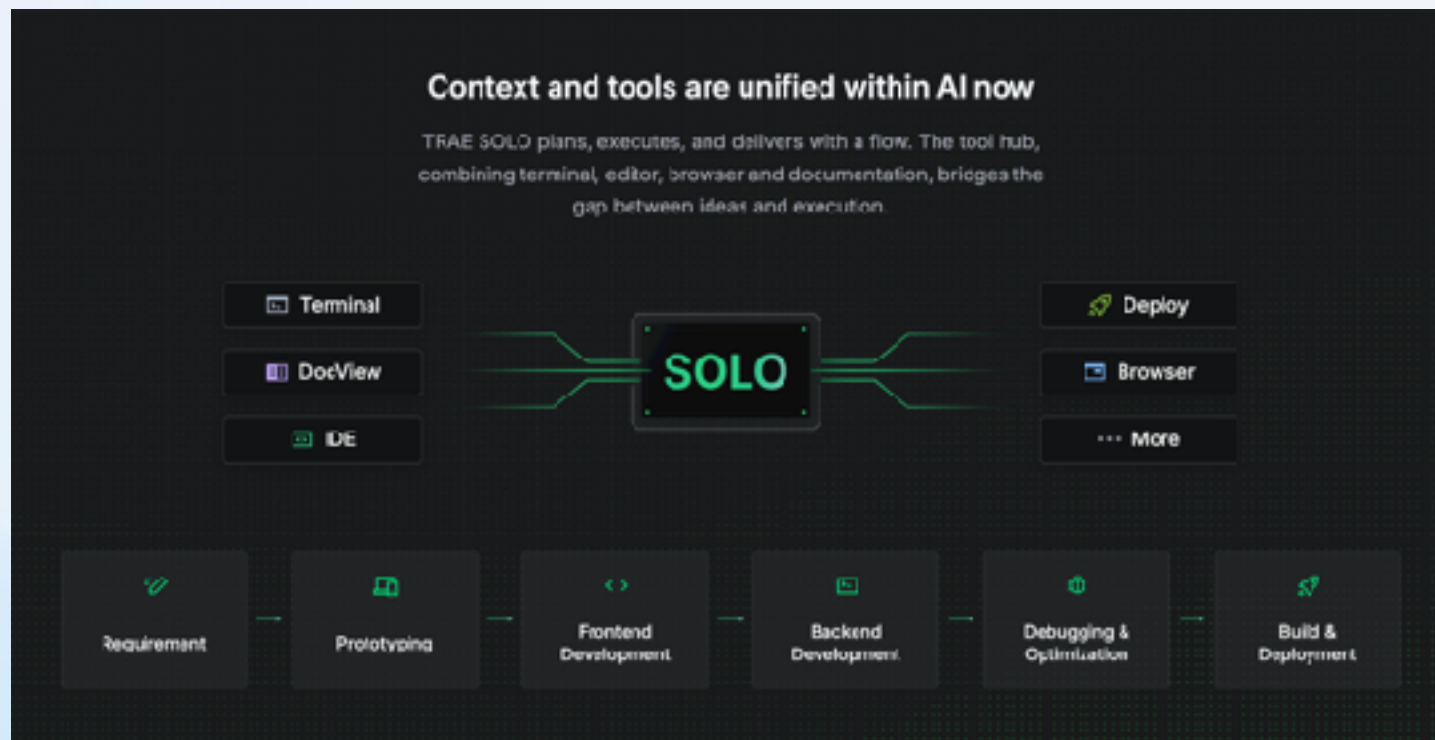
- 关键上下文信息 (#Context)
- 会话历史

Long-term Memory

- 自定义规则 (Rules)
- 记忆碎片 (Momento)
- TO-DO List

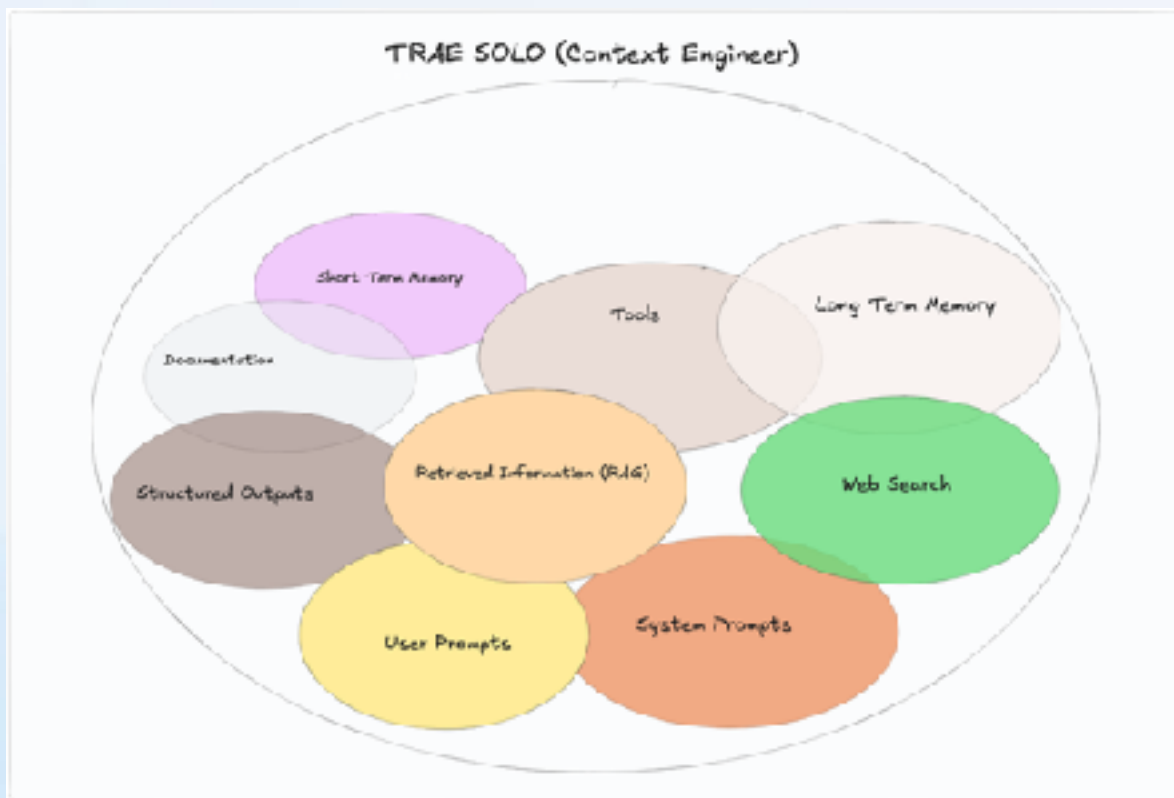
TRAE Agent 架构演进 – SOLO 模式

从 AI 辅助编程 到 AI 自主编程的一次跨越



TRAE Agent 架构演进 – SOLO 模式

更多的上下文以及更长的流程，对 Agent 架构提出了新的挑战

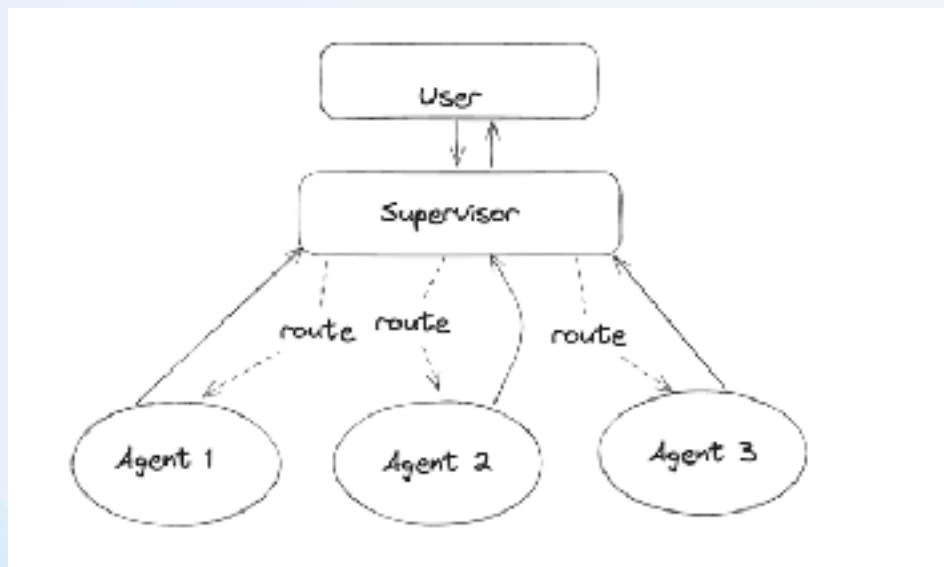


针对复杂长耗时任务面临的问题：

- 完成复杂任务，固定上下文窗口不够用怎么办？
- 任务时间越来越长，如何提高效率减少等待？
- 人如何更好地参与关键决策，确保目标符合预期？
- 当需求和环境发生变化，AI 如何及时感知？

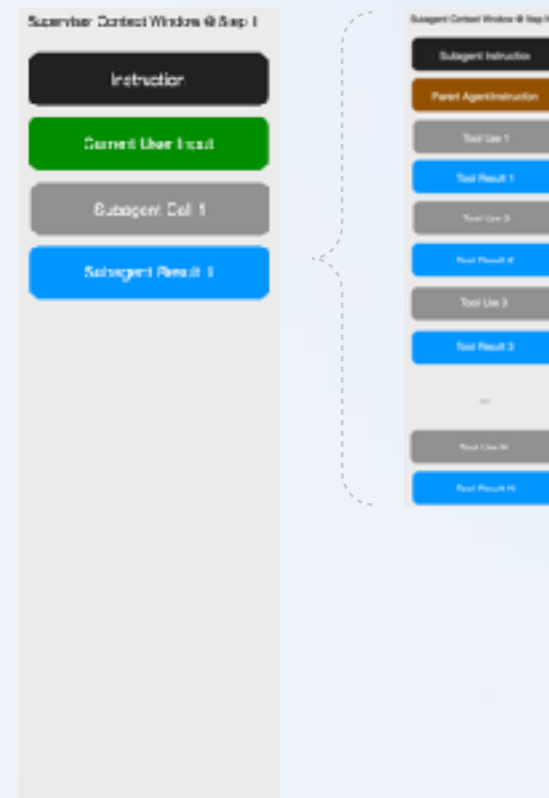
TRA E Agent 架构演进 – 面向复杂长耗时任务

引入 Multi-Agent 架构，将部分特定任务的上下文窗口隔离



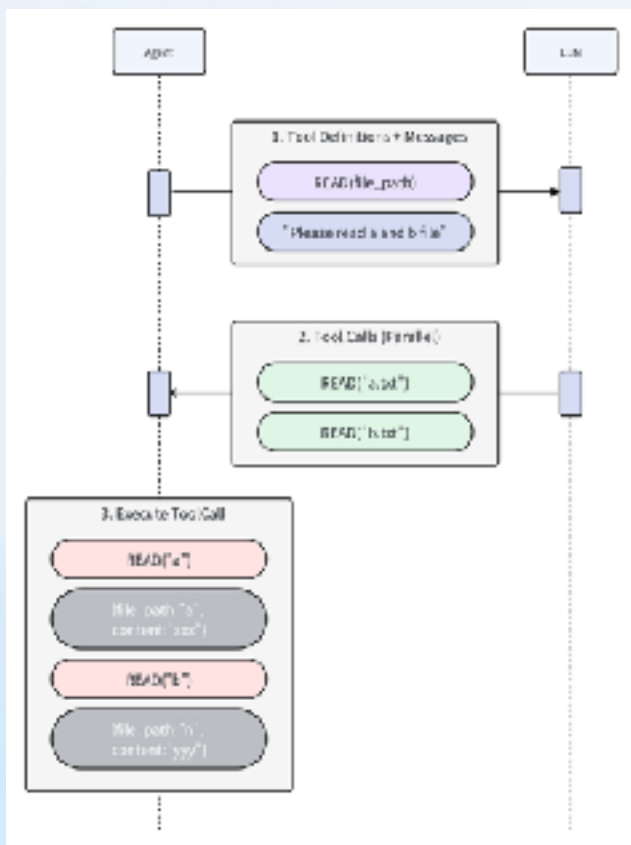
Subagent as a Tool

- Subagent 上下文窗口独立
- Subagent 调用完全交由 LLM 决策
- Subagent 根据主 Agent 指令进入 Agentic Loop



TRAE Agent 架构演进 – 面向复杂长耗时任务

利用模型并行 ToolCall 能力，降低执行耗时



Parallel ToolCall

- 通过 PE 引导 LLM 尽可能在合适场景触发并行 ToolCall
- 并行 ToolCall 可以有效减少 LLM 交互次数及 Token 消耗

Parallel Subagent

- 基于 LLM 并行 ToolCall 能力，进一步实现 Subagent Loop 并行执行
- 在例如搜索等并行安全场景，能极大降低执行耗时



TRA E Agent 架构演进 – 面向复杂长耗时任务

Human-in-the-loop 机制，让人及时参与关键决策

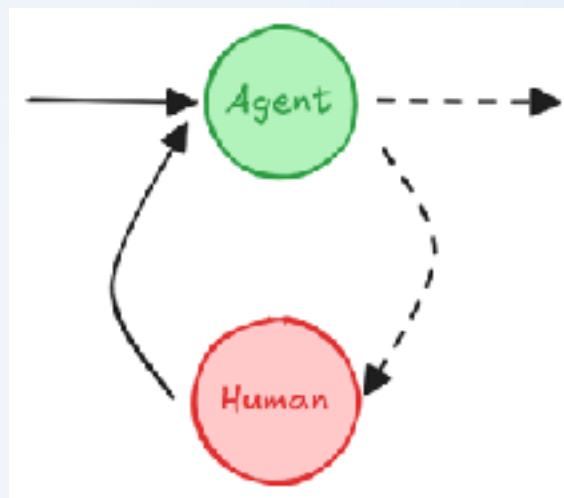


关键路径决策

- SOLO 模式下默认开启 Auto-Run 会自动执行绝大部分操作，但对于一些意图不明确或者高危的场景，模型会将决策权移交给人

修改或补充信息

- Spec-driven 场景，需求文档生成后需要人进行确认或者进一步修改
- 外部集成的三方工具/平台，例如 Supabase 等，需要人介入完善配置



TRA E Agent 架构演进 – 面向复杂长耗时任务

通过 Message Queue，随时感知需求和环境变化

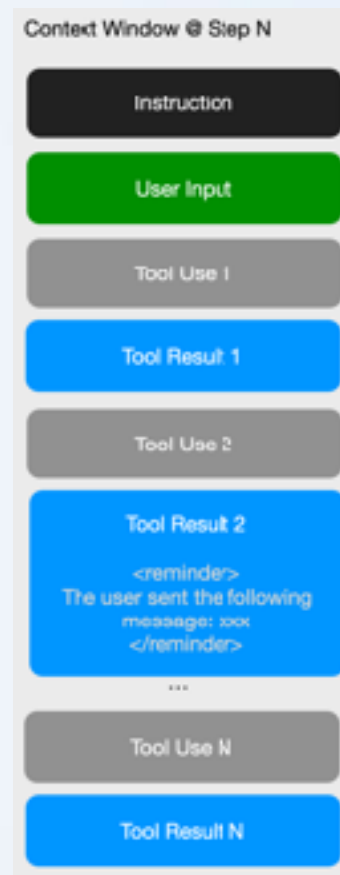


需求追加

- 采用异步消息队列机制，支持随时随地发需求
- 追加的消息会在下一轮模型思考被消费，从而影响模型后续决策

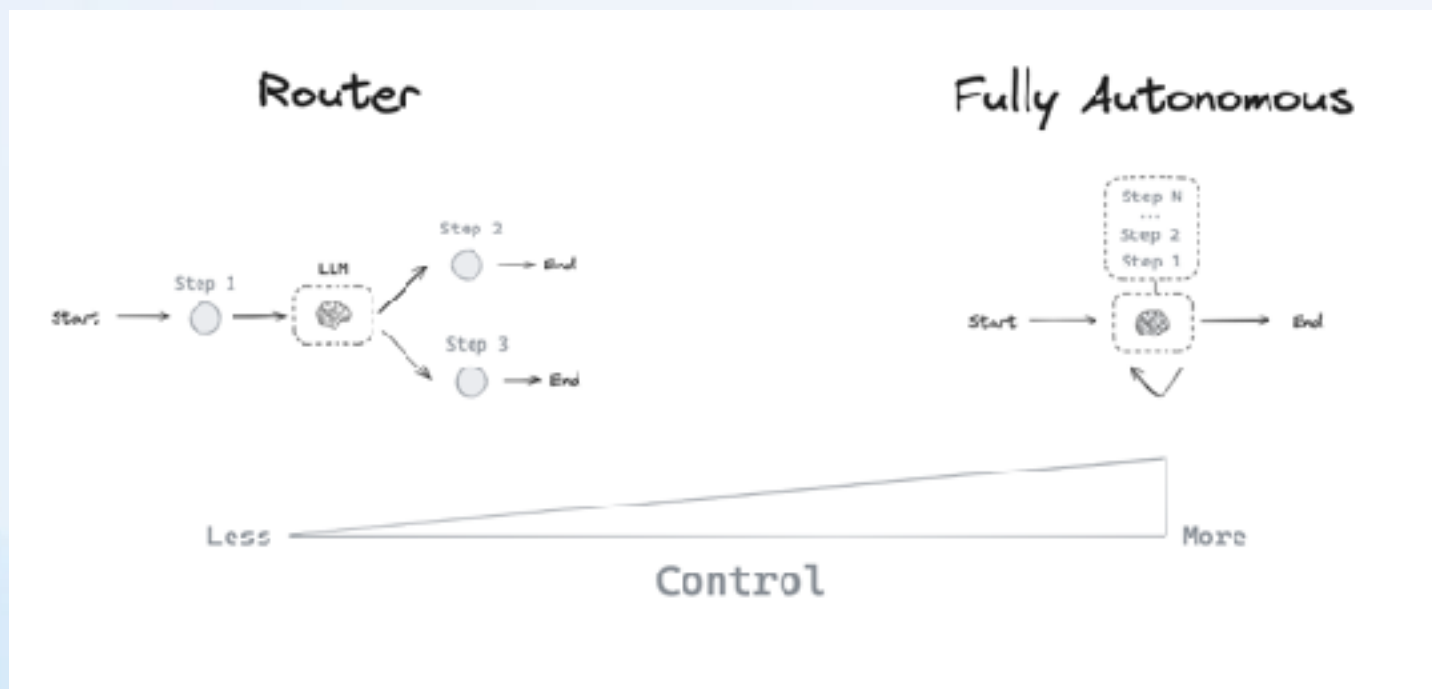
环境感知

- 除了来自人的消息，还会持续消费例如 Lint Error 等 IDE 环境的消息，使得模型能更加及时感知这些信息



TRA E Agent 架构演进 – 持续迭代

Workflow (工程驱动) or Agentic (模型驱动) ?



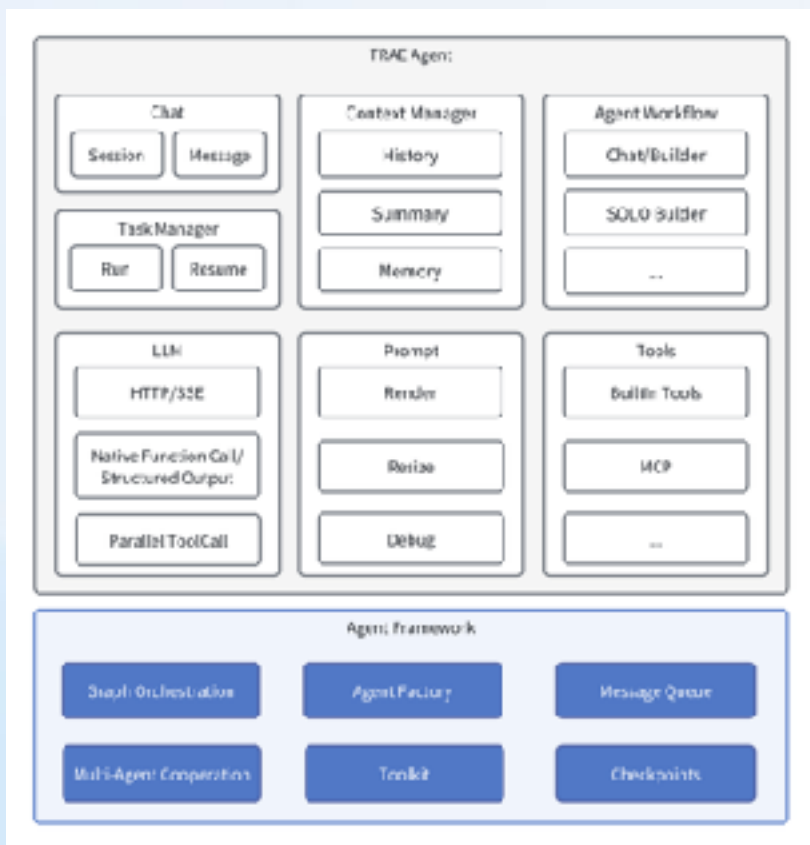
Agentic 和 Workflow 不是非黑即白的开关，而是一个连续的光谱。我们的架构要提供可持续的迭代能力，从而在 LLM 能力足够的地方更自主化，在 LLM 还不太擅长的地方通过固定编排把专家经验固定下来，随着模型的进化可以低成本的替换节点实现逻辑。



知乎

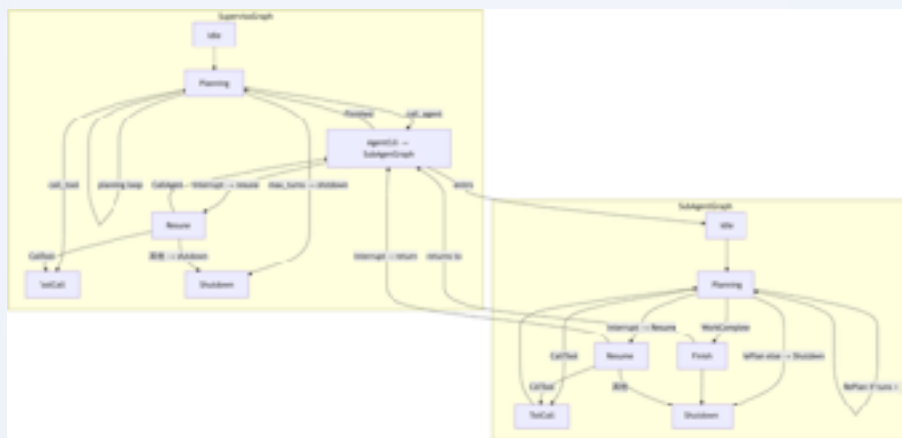
TRAE Agent 架构演进 – 持续迭代

基于自研 Agent Framework，支持 Agent 架构灵活编排



Graph Orchestration

- Graph 编排能力可以方便实现各类 Multi-Agent Workflow



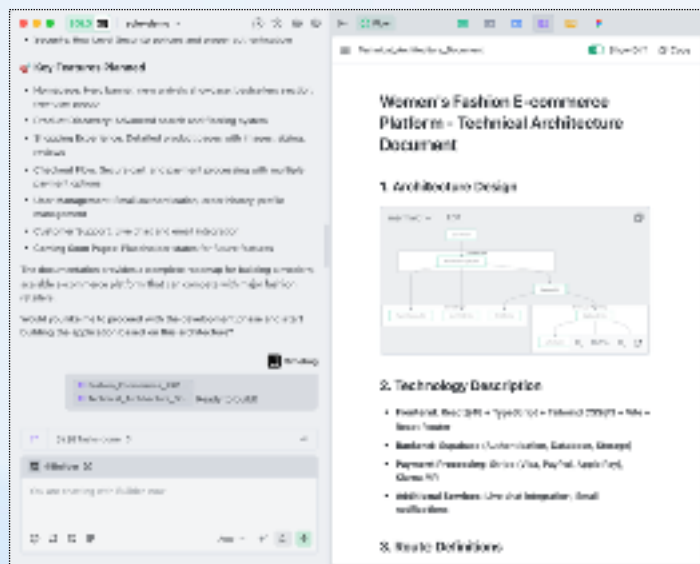
Agent Factory

- Agent 工厂提供快速定义一个 Agent 的能力 (Name/Describe/Prompt/Tools)

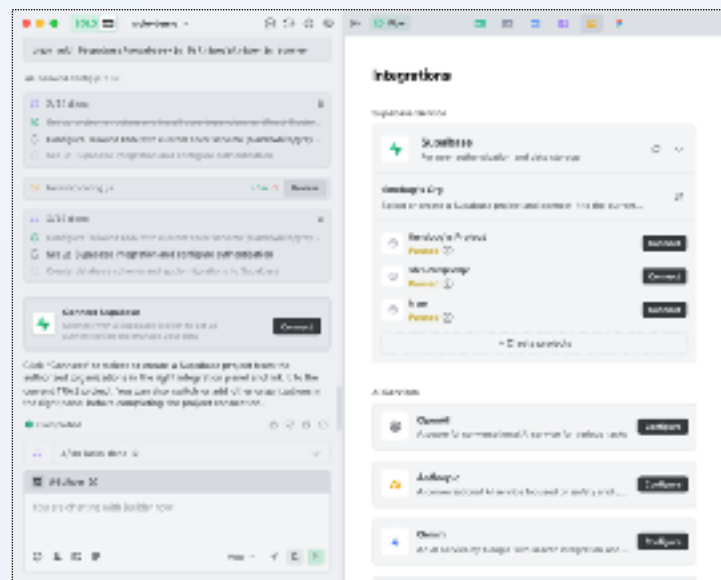
03 案例分享

SOLO Builder：从零开始完成 Web 应用设计、开发和部署

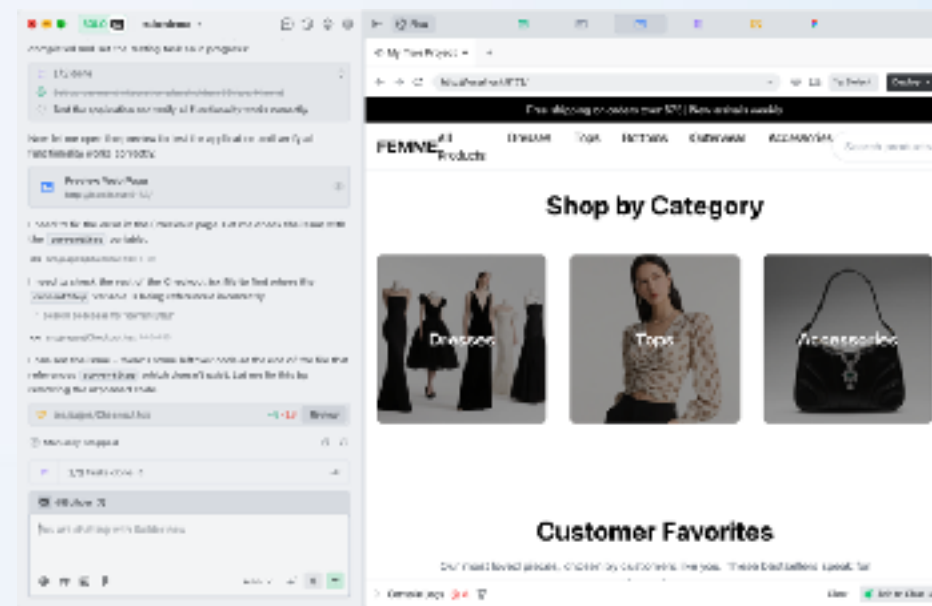
SOLO Builder – Web 应用开发流程



Spec-Driven

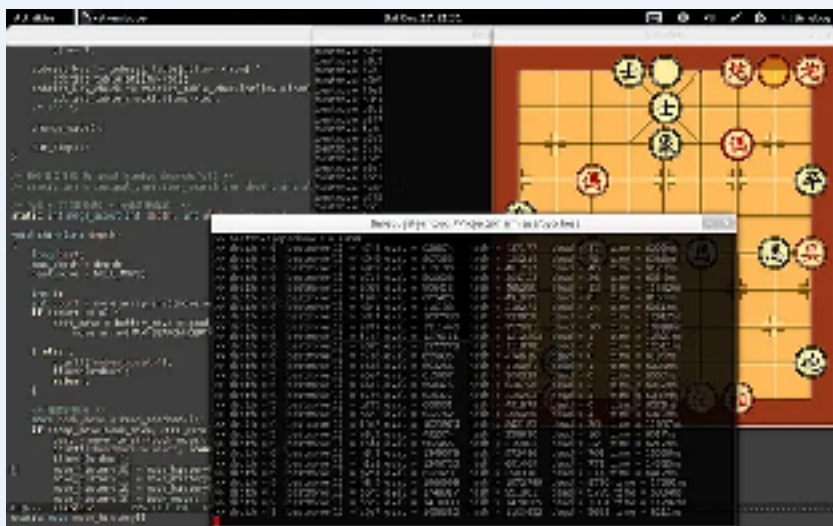


Supabase Integration



Preview and Deploy

SOLO Builder – 真实案例



2012 年手搓版中国象棋人机对战游戏

<https://github.com/timebug/harmless>



2025 年用 SOLO Builder 完成复刻:

- 游戏交互 (AI 生成所有前后端代码)
- 人机算法 (AI 通过学习老的 C 语言代码实现复刻)
- 登录注册/战绩排行榜 (AI 完成 Supabase 集成)
- 在线运行 (AI 完成 Vercel 部署)

04

未来展望

Next?



Coding Agent 未来展望

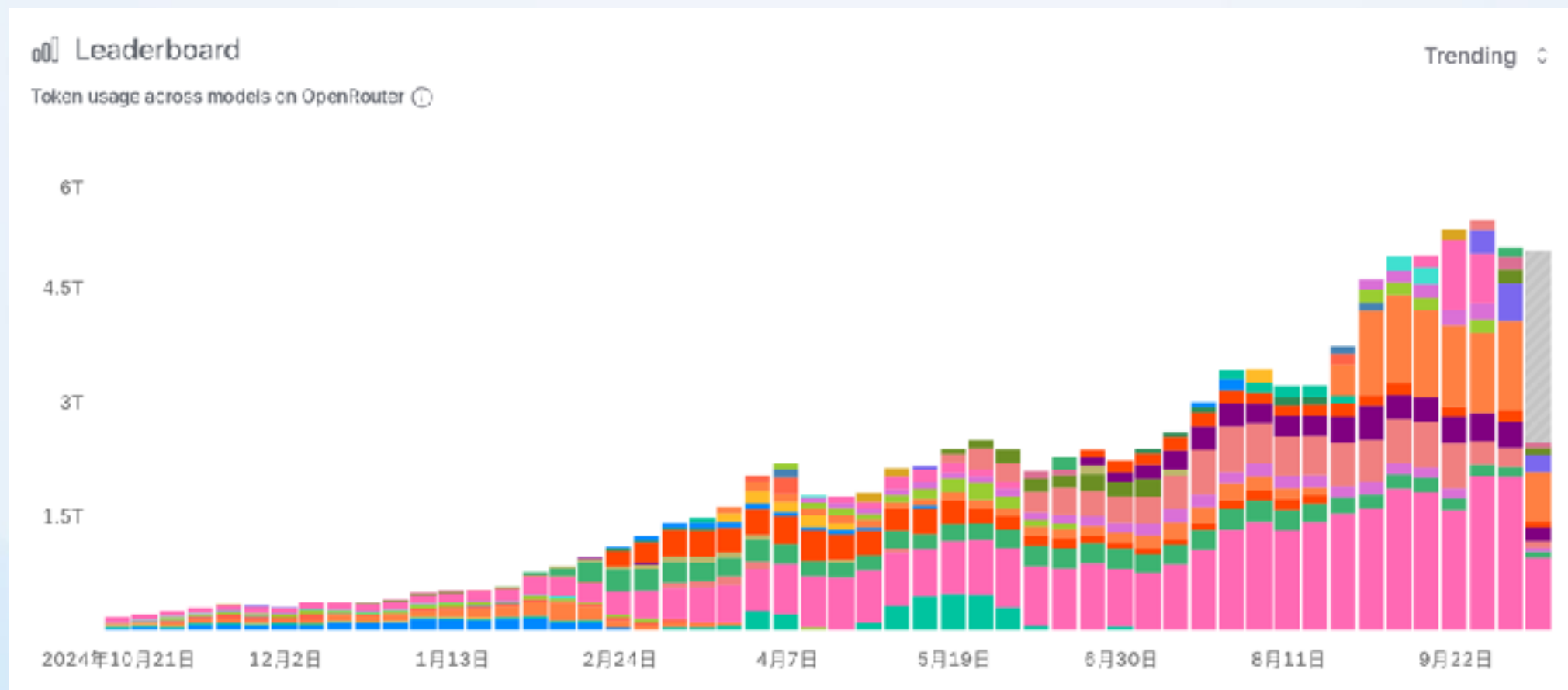
Coding Agent 连接一切

- 连接 IDE/Editor (ACP)
- 连接环境和工具 (Agent Hook/MCP/...)
- 连接其他异构 Agent (A2A-Protocol)

Coding Agent 无处不在

- 在本地和 IDE 协同
- Cli 形态嵌入到 CI/CD 流水线
- 在远程 sandbox 运行完成长耗时异步任务

AI 时代变化很快，全力向前跑



📍 北京

QCon

全球软件开发大会

会议时间：4月10-12日

- 大模型赋能 AI Ops
- 越挫越勇的大前端
- 云上业务架构演进
- 多模态大模型及应用
- 海外 AI 应用创新实践

📍 北京

AICon

全球人工智能开发与应用大会

会议时间：6月27-28日

- 端侧智能
- AI Agent
- 多模态大模型
- 金融+大模型

📍 上海

QCon

全球软件开发大会

会议时间：10月23-25日

- AI Agent
- 大模型训练推理
- 端侧 AI
- 搜推深度融合
- 数智金融

4月

5月

6月

8月

10月

12月

📍 上海

AICon

全球人工智能开发与应用大会

会议时间：5月23-24日

- 通用大模型
- AI Agent
- 垂直领域应用
- 模型可解释性

📍 深圳

AICon

全球人工智能开发与应用大会

会议时间：8月22-23日

- 模型效率与部署
- 多模态大模型
- 大模型安全
- 智能硬件

📍 北京

AICon

全球人工智能开发与应用大会

会议时间：12月19-20日

- 通用大模型
- 智能硬件
- LM Ops
- 具身智能

极客邦科技 2025 年会议规划

促进软件开发及相关领域知识与创新的传播



参会咨询



查看会议

THANKS

大模型正在重新定义软件

Large Language Model Is Redefining The Software

